System i5 und iSeries
Systemmanagement

- Betriebssystem
- Administration und Systemsteuerung
- Datenhaltung
- Datenbank und Integration
- Client Connectivity
- Netzwerk und Serverdienste

Grundwerk mit 36. Ergänzung
Stand September 2013

ITP VERLAG

Grundwerk mit 36. Ergänzung, Stand September 2013

Es wurde jeder Versuch unternommen, korrekte Informationen zu liefern. Der Herausgeber und der Autor garantieren nicht für die Richtigkeit des Buches und übernehmen keine Verantwortung für die Informationen, die dieses Buch beinhaltet oder nicht beinhaltet.

Folgende Begriffe sind Handelsmarken der International Business Machines Corporation in den Vereinigten Staaten und/oder anderen Ländern: IBM, i5, AS/400, OS/400 und OV/400. Alle anderen Markennamen von Unternehmen und Produkten sind das Eigentum der entsprechenden Inhaber.

Diese Publikation darf ohne Genehmigung des Copyright-Inhabers in keiner Form ganz oder teilweise reproduziert werden.

Satz und CD-ROM Erstellung: Popp Media Service,
86161 Augsburg, www.poppmedia.de

© 2005-2013 ITP VERLAG GmbH Kaufering, Deutschland
ISBN 3-930176-58-0

ITP VERLAG GmbH
Kolpingstraße 26
D-86916 Kaufering
☎ **+49 8191 9649-0**
🖷 **+49 8191 70661**
🌐 **www.midrange.de**
service@midrange.de

Sehr geehrte Leserinnen, sehr geehrter Leser,

Sie halten das Basis-Grundwerk des neuen Handbuches „eServer i5 und iSeries Systemmanagement“ in den Händen. Das Werk richtet sich mit den Themen

- Systemgrundlagen/Architektur
- Systembedienung/Steuerung
- Datenhaltung/Datenbank
- Betriebssysteme/Integration
- Serverdienste/Webserver
- Drucken
- Client Connectivity/Netzwerk

sowohl an Einsteiger wie auch an versierte Administratoren. Das vorliegende Werk wird Sie mit dem grundsätzlichen Wissen über Funktionsweise und Aufbau des eServer i5 sowie des Betriebssystems i5/OS versorgen. In regelmäßigen Abständen erarbeiten wir Ergänzungslieferungen, um dieses Werk und Ihr Wissen auf dem Laufenden zu halten.

Das Nachschlagewerk enthält in verständlicher Form und mit zahlreichen Screenshots all das, was Sie täglich wissen müssen, um das System bedienen, mit ihm kommunizieren, es Ihren Ansprüchen anpassen und kontrollieren zu können.

Die Fachautoren des Grundwerks begleiten das System seit Jahren und stellen Ihnen ihr Praxiswissen nun gebündelt zur Verfügung. Die Fachbeiträge aus den jeweiligen Rubriken sind auf dem neuesten Stand und auf die deutsche Version des i5/OS V5R3 (und früher) abgestimmt.

Die erste Ergänzungslieferung mit weiterführenden Texten sowie neuem Material ist für Ende Dezember 2004 vorgesehen; danach werden Sie in regelmäßigen Abständen (4x jährlich) auf dem neuesten Wissensstand gehalten.

Ich möchte mich nun für Ihr Vertrauen bedanken und wünsche Ihnen viel Spaß bei der Lektüre des Grundwerks!

Thomas Seibold
Herausgeber

Fachautoren:

Autor	Firma	e-Mail
Ursula Althoff	IBM Deutschland	ursula-althoff@de.ibm.com
Ronald Böhmer	Epos GmbH	ronald.boehmer@eposgmbh.com
Robert Engel	Robert Engel GmbH	ren@renedv.de
Willy Günther	IBM Deutschland	wiguenth@de.ibm.com
Sabine Jordan	IBM Deutschland	sjordan@de.ibm.com
Heinz J. Kopp	IBM Deutschland	hjkopp@de.ibm.com
Klaus-Peter Luttkus	Epos GmbH	luttkus@eposgmbh.com
Norbert Masannek	IBM Deutschland	norbert_masannek@de.ibm.com
Peter Nimz	IBM Deutschland	peter.nimz@de.ibm.com
Carmen Rasche	Epos GmbH	carmen.Rasche@eposgmbh.com
Edelgard Schittko	IBM Deutschland	ej_schittko@de.ibm.com
Helmut Stein	IBM Deutschland	helmut.stein@de.ibm.com
Hans-Jörg Zeig	ZeBIS Beratung/ Informationssysteme	hjzeig@zebis.de

1 Kapitel

Systemgrundlagen / Architektur

1

Seite 2

Intro

Die allermeisten User haben nur wenig Ahnung davon, wie das Computer-System ausgestattet ist, das sie täglich nutzen. Noch weniger Ahnung darf davon erwartet werden, wie die vielen Server, die da im Netzwerk ihre Dienste tun, funktionieren. Glücklicherweise ist dies auch für die meisten User gar nicht nötig. Es ist schließlich auch nicht nötig zu wissen, wie ein Vergaser oder eine elektronisch gesteuerte Einspritzpumpe funktioniert, um ein Auto zu fahren.

Autor: Michael Wirt, Chefredakteur Midrange MAGAZIN

Was wir, die vielen Autoren dieses Buches, hier in diesem Buch tun wollen, ist hingegen schnell erklärt. Wir wollen Ihnen unser Wissen zur Verfügung stellen. Wir alle sind Spezialisten in einzelnen Fachgebieten und unser Coach, der Herausgeber dieses Buches, Thomas Seibold, will von jedem Einzelnen von uns immer nur das Beste, um ein für Sie wertvolles Werk zu schaffen. Meine Aufgabe ist es, Ihnen die Grundlagen der Systemarchitektur der eServer i5 ein wenig näher zu bringen. Keine Bange, ich schreibe nicht über die technische Tiefen der Speicher, Prozessoren, Kabel, Stecker und den vielen anderen wichtigen Dingen, die da ständig ihre Arbeit tun. Ich halte mich an die Software, an das „BIOS“ bzw. das Betriebssystem in seinen Grundfunktionen. Zunächst einmal die wesentlichen Merkmale der iSeries-Architektur:

Wesentliche Architektur-Merkmale der iSeries

Die Architektur der iSeries ist im Wesentlichen durch folgende fünf Merkmale gekennzeichnet:

1. Technologische Unabhängigkeit
2. Single Level Store
3. Hardware-Integration
4. Software-Integration
5. Objektbasiertes Design

Technologische Unabhängigkeit

Technologische Unabhängigkeit der iSeries wird durch das so genannte Machine-Interface (MI) realisiert. Dies bedeutet, dass sowohl das Betriebssystem OS/400 – die Middleware – als auch alle Kunden-Anwendungen ausschließlich mit MI kommunizieren können sowie isoliert sind von der darunter liegenden Hardware. MI selbst kommuniziert mit der Hardware via Kernel (Systems Licensend Internal Code), der von IBM bereitgestellt wird und der die jeweils aktuelle Hardware unterstützt. Damit sind die Investitionen der Kunden in ihre Anwendungen komplett geschützt, weil technologisch bedingte Hardware-Änderungen kaum Kosten verursachen. Das beste Bei-

spiel hierfür war der reibungslose Übergang von der 48-Bit-CISC-Technologie auf die 64-Bit-RISC-Technologie vor einigen Jahren. Heute stellt uns die IBM mit dem Power5-Prozessor bereits die neunte Generation der 64-Bit-Prozessoren zur Verfügung.

Single Level Store

Single-Level Store der iSeries beschreibt die Tatsache, dass alle Kundendaten und Programme, unabhängig davon, ob diese sich auf der Platte oder im Hauptspeicher befinden, logischer Bestandteil eines einzigen 64-Bit-Adress-Raumes sind. Egal wo sich die Daten befinden, das System beziehungsweise das Storage-Management der iSeries verwaltet automatisch die entsprechenden Zugriffe; wobei die Referenzierung seitens der Benutzer lediglich über den Namen erfolgt. Diese besondere Form der Hauptspeicher- und Plattenverwaltung bietet ideale Voraussetzungen hinsichtlich der Speichervirtualisierung. Ein anderer Vorteil liegt im schnellen Task-Switching – und dies ist wiederum vor allem im kommerziellen Umfeld von Bedeutung. In Windows- oder UNIX-Umgebungen bedeutet Task-Switching, dass jeweils ein neuer Adress-Raum geladen werden muss, bevor die Tasks dann ausgeführt werden können – diese Operation ist vergleichsweise komplex und langsam.

Hardware-Integration

Hardware-Integration bedeutet, dass die iSeries primär entwickelt und optimiert wurde als ein kommerzieller Business-Server für datenintensive Anwendungen. Dabei wurde der Fokus auf Aspekte gerichtet wie ausgewogene Prozessor-Leistung, starke I/O-Kapazitäten, hohe Bandbreiten und schnelles Task-Switching für unterschiedlichste Workloads.

Software-Integration

Software-Integration heißt, dass standardmäßig eine Software-Infrastruktur bereitgestellt wird, die für die Rolle als Business-Server erforderlich ist. Diese Software ist für die Architektur optimiert, von IBM-Labors ausgetestet und vollständig integriert worden. Das beste Beispiel hierfür ist die relationale Datenbank DB2 UDB for iSeries. In derzeit aktuellen i5-Modellen ist die Integration von 5 Betriebsystemen Realität. Da läuft i5/OS gleichzeitig mit LINUX im i5 und gleichzeitig mit LINUX auf dem integrierten xServer (iXS) – und das alles gleichzeitig mit Windows und AIX. Natürlich ist es notwendig, den unterschiedlichen Systemen jeweils eigene Umgebungen zu schaffen. Wir nennen diese integrierten Umgebungen „Logische Partition“. Der Begriff „logisch“ muss so sein, da wir physisch in der Lage sind, einzelne Prozessoren in ihrer Leistung auf mehrere Partitionen zu verteilen. Dabei stehen immerhin bis zu 10 Micro-Partitionen pro Prozessor zur Verfügung. Später in diesem Werk gehen einzelne Spezialisten noch näher auf diese Technologie ein.

Objektbasiertes Design

Objektbasiertes Design bedeutet, dass innerhalb der Architektur der iSeries alles als Objekt betrachtet wird. Die Objekte selbst bestehen aus einem beschreibenden und einem Daten-Teil. Der wesentliche Vorteil dieses Ansatzes besteht darin, dass abhängig vom Objekt-Typ jeweils nur ganz bestimmte Operationen zulässig sind. Damit begründet sich auch die außergewöhnliche Sicherheit der iSeries. Es ist bis heute kein Fall dokumentiert, in dem eine ordnungsgemäß eingerichtete iSeries gehackt wurde – ein Aspekt von unschätzbarem Vorteil in der gegenwärtigen Netzwerk-Ära. Umfragen bei Unternehmen zeigen, dass die Integration unterschiedlichster Technologien aufgrund der damit verbundenen Risiken, Kosten und Aufwände zu einer verstärkten Frustration führt. Die hier beschriebene Architektur der iSeries hilft den Unternehmen, sich auf ihr eigentliches Kern-Geschäft zu konzentrieren.

Grundwerk 10/2004

1

Seite 6

LPAR

Seit mehr als fünf Jahren hat sich Logical Partitioning (LPAR) auf der iSeries als die Schlüsseltechnologie für Server-Konsolidierung entwickelt, und sie hilft iSeries Anwendern, ihre CPU-Auslastung zu maximieren. In gleichem Maße, wie die Leistungsfähigkeit der modernen Multiprozessor-Systeme immer weiter anwächst, steigt die Anforderung, mehrere Workloads unabhängig voneinander zu verarbeiten. Mit LPAR werden Server in mehrere unabhängige virtuelle Systeme aufgeteilt, jeder mit seiner eigenen Hardware, seinem eigenen Betriebssystem und eigener Anwendungsumgebung.

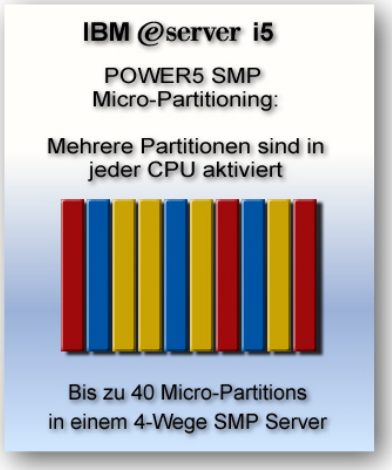
Autor:
Helmut Stein,
IT Consultant im iSeries Technical Support

Virtualization Engine

IBM eServer i5 ist der erste Server auf dem Markt, der die in dreijähriger Entwicklungszeit in IBM Labors entwickelte IBM Virtualization Engine einsetzt. Wichtiger Bestandteil der Virtualisierungstechnik ist der Hypervisor Support, der mehrere Betriebssysteme auf einem Server ermöglicht. Damit wird dynamisches LPAR auf multithreaded CPUs in Verbindung mit Subprocessor Partitioning ermöglicht. Der LPAR Hypervisor wird auf den neuen Systemen als Teil der Firmware ausgeliefert und im Service-Prozessor gespeichert.

Verbessertes Micro-Partitioning

Auf der iSeries ist das so genannte „Micro-Partitioning“, die Aufteilung eines Rechners in virtuelle Maschinen, auch innerhalb einzelner Prozessoren bereits seit Einführung der POWER4-Architektur auf den Modellen i825, i870 und i890 verfügbar. Mit den neuen IBM eServer i5-Modellen fällt die Grenze von 32 logischen Partitionen, es werden nun bis zu 254 logische Partitionen auf einem POWER5-Server möglich sein. Die maximale Anzahl von 10 Partitionen je Prozessor bleibt bestehen, auf dem neuen i5-Modell 570 können somit bis zu 40 logische Partitionen betrieben werden. In zukünftigen Systemen werden maximal 32 Prozessoren in einer i5/OS-Partition unterstützt.



Micropartitioning

Grundwerk 10/2004

1.1
Seite 2

Hardware Management Console

Bis zum OS/400 Release V5R2 musste immer OS/400 in der Primary Partition installiert sein. Das gilt auch in Zukunft auf allen bisher verfügbaren iSeries Modellen. Auf den neuen POWER5-basierenden i5 Servern wird das Management von LPAR von der Primary Partition auf die Hardware Management Console (HMC) verlagert. Sämtliche bereits heute bekannte Funktionen – wie das Erstellen von Partitionen, das dynamische Verschieben von Ressourcen, das Starten von Partitionen usw. – werden von der HMC übernommen.

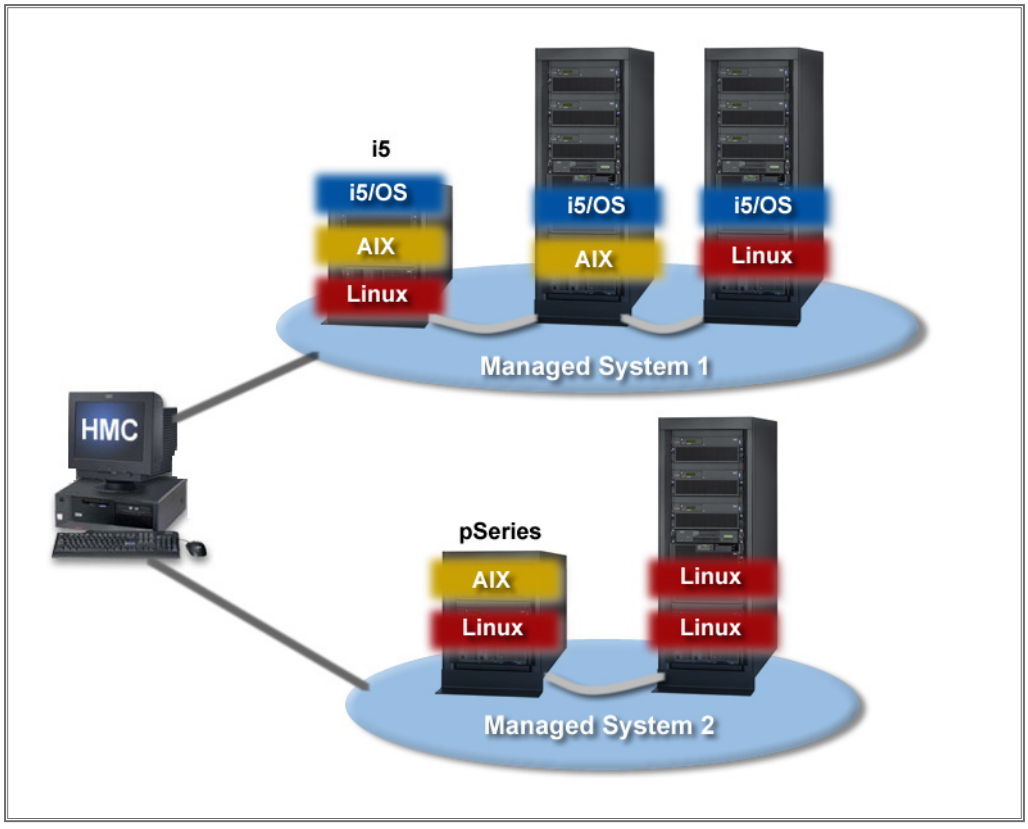
Was ist neu bei LPAR auf dem IBM eServer i5 mit i5/OS?	
LPAR User Interface	Hardware Management Console zum LPAR Management
Maximale Anzahl Partitionen	Bis zu 40 Partitionen, maximal 10 Partitionen je Prozessor
Partition Typ	Keine Primary Partition erforderlich, Partition Profil, System Profil
Erstellen einer Partition	Neue logische Partition und Partition Profil ohne IPL sofort verfügbar
Zusätzliche Betriebssysteme	Linux (neuer Kernel erforderlich), AIX
Minimum und Maximums-Werte ändern	Ein IPL der betroffenen Partition erforderlich
Prozessoren	Ausgeschaltete Partitionen mit dedizierten Prozessoren stellen diese dem Shared Pool zur Verfügung
Hauptspeicher	Wird in 16 MB-Schritten dynamisch verändert
I/O Ressourcen	Keine Bus-Ownership mehr, I/O Ressourcen werden Slot-weise zugeordnet
LPAR Configuration Data	HMC und Service Prozessor haben die LPAR Konfiguration gespeichert
Konsole	HMC, Twinax, Operations-Konsole über LAN oder direkt

Um Teile von Prozessoren unterschiedlichen Partitionen zur Verfügung zu stellen, muss ein Shared Processor Pool erstellt werden. Im Shared Pool wird zwischen dem capped und dem uncapped Modus unterschieden. In einer capped Partition kann nie mehr als die definierte Prozessorleistung verwendet werden. Im Gegensatz dazu können uncapped Partitionen ihre definierte Prozessorleistung überschreiten. Bei Bedarf bekommen diese Partitionen Prozessorleistung bis zum Maximalwert der definierten virtuellen Prozessoren zur Verfügung gestellt, falls diese Ressourcen nicht von anderen Partitionen verwendet werden. Die richtige Zuordnung von virtuellen Prozessoren wird also noch wichtiger als heute schon. Die Zuteilung von Prozessorleis-

tung wird über einen Gewichtungsfaktor geregelt, der entsprechend der Wichtigkeit der Workload in der Partition zu definieren ist.

Der einfachste Weg, um Server oder Partitionen umzukonfigurieren, ist die Verwendung von Profilen. In den Profilen sind die den Partitionen zugeordneten Ressourcen definiert, die bei der Aktivierung dann den jeweiligen Partitionen zugewiesen werden.

Die Hardware Management Console für eServer ist ein System, das Managed Systems verwaltet. Dazu gehören eServer Hardware, logische Partitionen und Capacity on Demand. Es gibt lokale und remote Hardware Management Consolen (HMCs), die sowohl als Desktop als auch im Rack betrieben werden können. Die HMC ist nur für die neuen i5-Modelle erforderlich. Vorhandene iSeries-Modelle können auch bei i5/OS über die bekannten Schnittstellen verwaltet werden.



LPAR Konfiguration mittels Hardware Management Console (HMC)

Um die neuen IBM eServer i5-Modelle 520 und 570 für LPAR zu konfigurieren, ist immer eine HMC erforderlich. Änderungen der LPAR-Konfiguration können durch Skripte von der HMC vorgenommen werden. Heutige Primary Partitions zum Management der LPAR Konfiguration werden nicht mehr für diese Aufgabe benötigt, dies übernimmt die HMC. Ein großer Vorteil der HMC ist es, dass mehrere Server verwaltet werden können und dass das Management auch remote durchgeführt werden kann.

1.1
Seite 4

AIX nativ

Wurden AIX-Anwendungen auf den iSeries Modellen lediglich in der Runtime-Umgebung PASE unterstützt, so kann auf den neuen i5-Modellen AIX 5L nativ in logischen Partitionen laufen. Durch Unterstützung mehrerer Betriebssysteme können Unternehmen mit IBM eServer i5 ihre IT-Infrastruktur vereinfachen, neue Anwendungen implementieren sowie die Unternehmensaktivitäten auf einem einzigen, äußerst flexiblen und widerstandsfähigen Server konsolidieren. Mit dem IBM eServer i5 Server haben die Kunden eine Vielfalt an branchenführenden Anwendungen zur Auswahl, die für viele unterschiedliche OS-Umgebungen konzipiert sind – wie z. B. i5/OS, Windows, Linux und AIX 5L.

Die Integration von Windows

Autor:
Norbert Masannek,
IBM eServer iSeries
Specialist System
Sales Central Region

Wenn man sich heute bei IBM eServer iSeries und AS/400 Kunden umschaut, stellt man folgendes fest: Über 90 Prozent der Kunden haben neben der iSeries auch Windows-Server im Einsatz. Darüber hinaus haben diese Anwender auch auf ihren Clients Windows-Betriebsysteme installiert. Der Zugriff von diesen Clients auf die iSeries erfolgt in den meisten Fällen mit der 5250-Emulation vom iSeries Access.

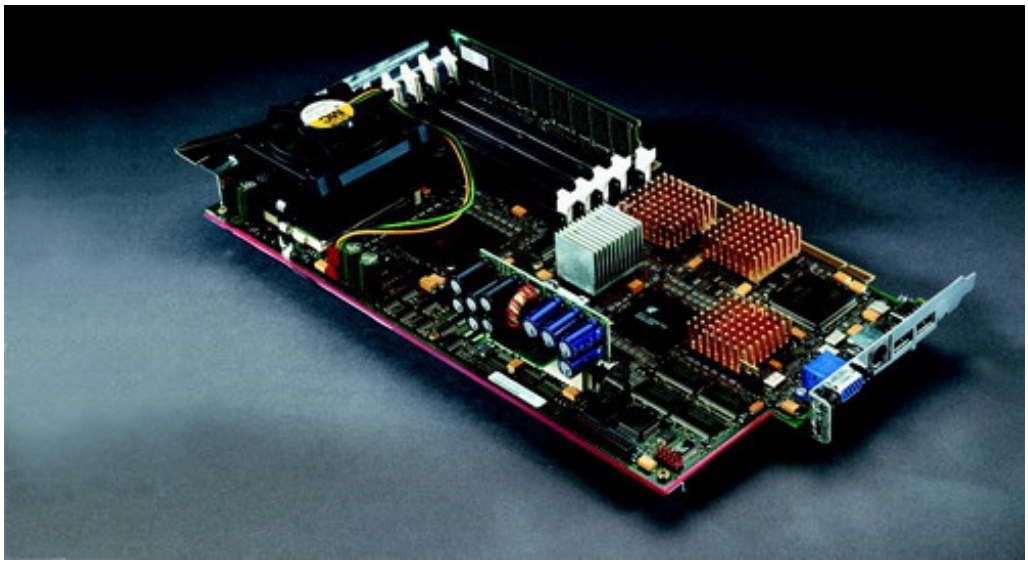
- Es gibt immer mehr Anwendungen, die auf beiden Betriebssystem laufen. Die Anwendung beispielsweise läuft auf dem Windows-Server und nutzt die Datenbank der iSeries.
- Mainframe-, Unix-, Windows- und iSeries-Anwender suchen nach Möglichkeiten, um ihre Server zu konsolidieren. Windows-Betriebssysteme auf der iSeries laufen zu lassen, ist daher ein sehr guter Weg, um Server zu konsolidieren.

Die Strategie, um diesen Wunsch der Windows-Konsolidierung auf der IBM eServer iSeries zu realisieren, sieht so aus: Funktionen werden auf der iSeries zur Verfügung gestellt, mit denen man die Windows-Server managen kann.

Intel-basierende PC-Server gibt es auf der AS/400 seit V3R1. Seit der Ankündigung von OS/400 V4R2 1998 werden auch Windows-Betriebssysteme unterstützt.

Möglichkeiten der Konsolidierung

Es gibt nun zwei Möglichkeiten, um die Konsolidierung durchzuführen: Das erste Angebot ist der Integrated xSeries Server (IXS). IXS ist eine Karte, die in die iSeries eingebaut wird, auf der dann Windows läuft.

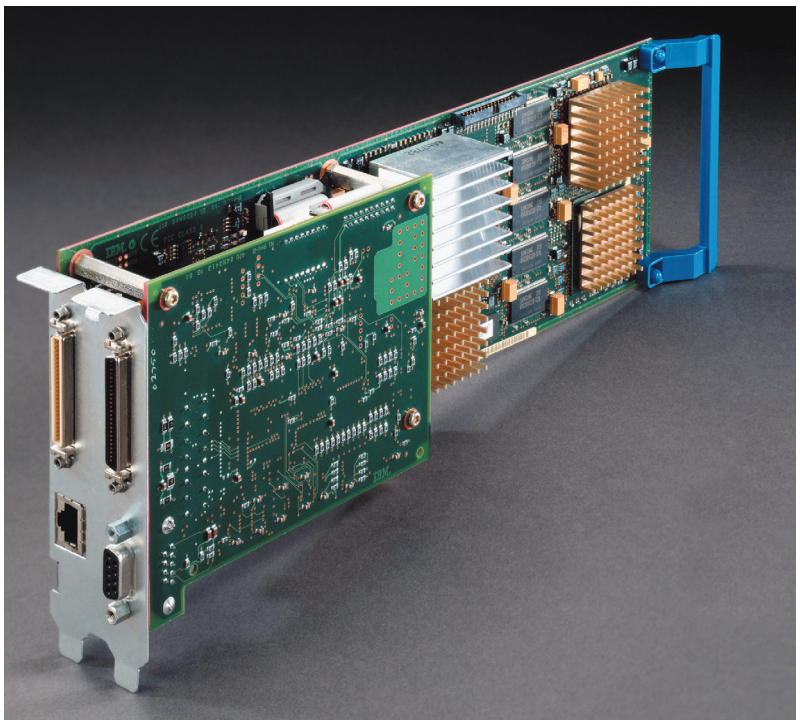


Diese Karte ist wie ein Motherboard zu betrachten, auf dem sich u. a. der Prozessor und die Hauptspeicherkarten befinden.

Die andere Hardware – wie Platten, LAN-Adapter, CD-ROM und Bandlaufwerke – stellt ausschließlich die iSeries zur Verfügung. Der aktuelle IXS verfügt über einen INTEL 2.0 GHz Xeon Prozessor mit 512K L2 Cache und unterstützt bis zu 4 GB Hauptspeicher und 2 TB Plattenplatz. Ferner hat er einen integrierten Ethernet LAN-Adapter und unterstützt darüber hinaus noch bis zu 3 LAN Adapter, die in der iSeries installiert sind. 4 USB Ports runden das Bild ab.

Als zweites Angebot gibt es nun die Möglichkeit, auch große xSeries Server an die iSeries anzuschließen, zum Beispiel das Modell 445 mit bis zu 8 Prozessoren. Dadurch erhöht sich die Skalierbarkeit dieser Integrationslösung sehr deutlich. Man kann nun auch sehr große Workloads konsolidieren. Auch hierbei wird die Hardware der iSeries genutzt – sowie Platten, Bandeinheiten etc.

Der Anschluss dieser xSeries Server erfolgt über den IXA (Integrated xSeries Adapter), der in den xSeries Server eingebaut wird und dann über HSL-Kabel (HSL= High Speed Link) mit der iSeries verbunden wird. Das Ganze stellt sich dann wie eine normale Erweiterungseinheit im HSL Loop dar.



Der Integrated xSeries Adapter

Detaillierte Informationen der angebotenen Modelle finden Sie unter:

http://www-1.ibm.com/servers/eserver/iseries/windowsintegration/.

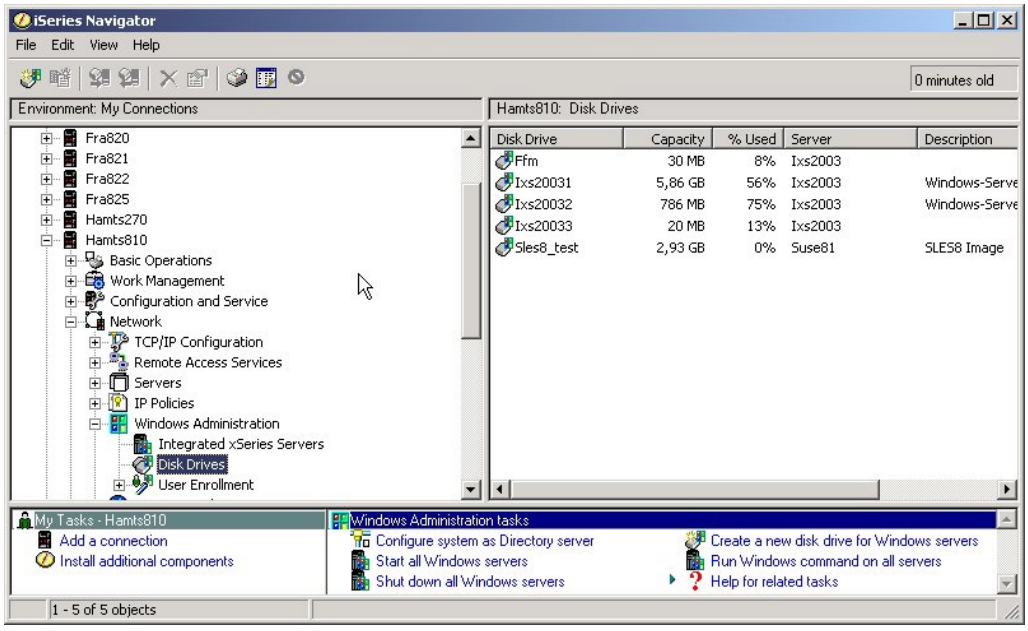
Eine interne virtuelle 1 GB Ethernet-Verbindung zwischen der iSeries und IXS/IXA macht die Lösung unabhängig von einem Ausfall des lokalen Netzes, d.h.: Die Verbindung zwischen den Servern bleibt auch dann bestehen, wenn das LAN einmal ausfallen sollte.

Dass die benötigten Device Driver der iSeries Hardware – wie Platten, Bänder etc. – durch Windows verwendet werden können, wird durch die Windows-Integrationssoftware auf der iSeries bereitgestellt. Hardware und Software werden von IBM getestet. Eine hohe Zuverlässigkeit ist somit gewährleistet. Sowohl auf dem IXS als auch auf dem IXA laufen Windows 2000 Server und Windows 2003 Server.

Automatisches Storage Management

Wo liegen nun die praktischen Vorteile beim Einsatz dieser Lösung? Die IBM eServer iSeries ist weltweit das einzige System mit einem automatischen Storage Management System. iSeries-Kunden beschäftigen keinen Storage-Spezialisten. Ein Schlüsselpunkt bei der Realisierung ist, dass die iSeries als SAN für die Windows-Server dient. Dieses Prinzip bringt die iSeries-Technologie auch auf die Windows-Server und erhöht so ihren Wert.

iSeries-Plattenplatz wird den Windows-Servern zur Verfügung gestellt, indem man einen Storage Space erstellt. Diesem Storage Space ordnet man dann Windows zu. Windows erkennt dieses Objekt dann als ganz normales Laufwerk. Dies kann auch dynamisch geschehen, dass heißt: Man kann den verfügbaren Plattenplatz auf dem Windows-Server erhöhen, ohne ihn zu booten. Die Storage Spaces können bis zu 64 GB groß sein – und man kann maximal 32 Storage Spaces einem IXS/IXA zuordnen. So kommt man auf maximal 2 TB Platten pro IXS/IXA.



Storage Space zuordnen

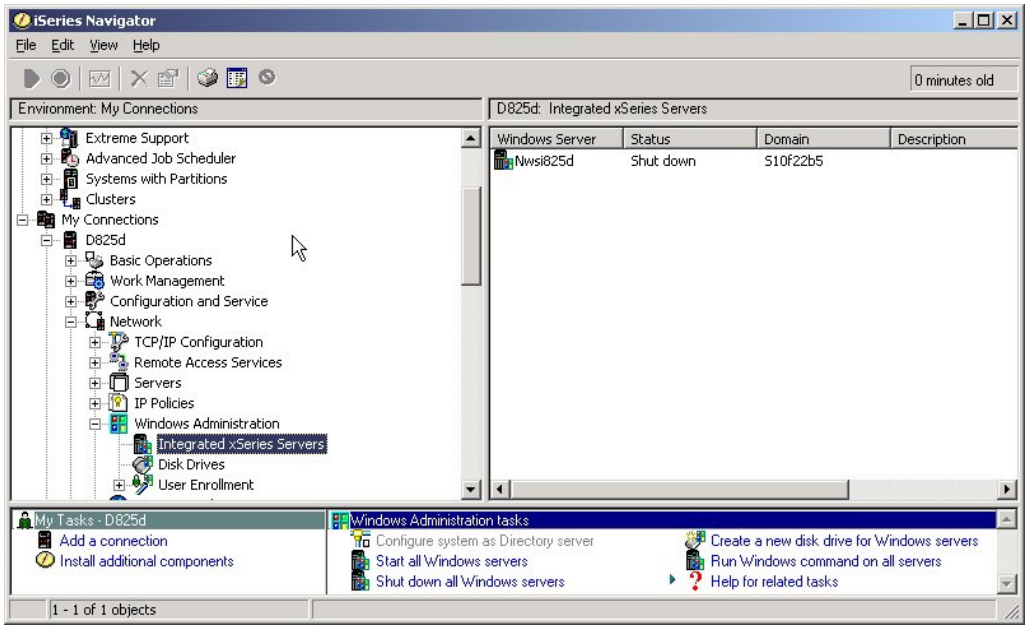
Da die meisten iSeries-Anwender ihre Platten durch RAID-5 oder Spiegelung schützen, ist dieser Schutz auch für Windows auf dem IXS/IXA verfügbar. Da die Storage Spaces – wie jedes andere Objekt auf der iSeries – auch über alle Platten verteilt wird, nutzt Windows beim Plattenzugriff alle Plattenarme der iSeries und hat daher bei vielen I/Os Performancevorteile gegenüber konventionellen PC-Servern.

Durch dieses Prinzip sind folgende Möglichkeiten vorhanden:

- Gemeinsames Backup: Da die Storage Spaces-Objekte auf der iSeries sind, werden sie bei einer Gesamtsicherung der iSeries mitgesichert.
- Hot Spare: Hat man einen zusätzlichen IXS/IXA als Hot Spare zur Verfügung, kann man im Falle eines Ausfalls innerhalb weniger Minuten seinen Windows-Server wieder verfügbar haben, indem man in der Network Server Description die Hardware „Ressource“ ändert.
- Test: Man kann z.B. die Storage Spaces des Produktions-Servers kopieren und diese einem Test-Server zuordnen. Der Produktions-Server wird nun heruntergefahren; die NWSD des Test-Servers wird auf dieser HW Ressource hochgefahren. Nun können hier Service Packs etc. getestet werden und zwar in der „Produktionsumgebung“.

Vorteile in der Administration

In der täglichen Administration ergeben sich u. a. folgende Vorteile: Eine gemeinsame Benutzer-Administration ermöglicht das Anlegen von OS/400-Benutzern oder Gruppenprofilen in eine Windows 2000/2003 Server Domain oder auf einem individuellen Windows-Server. Diese Funktion beinhaltet zusätzlich die Synchronisation von OS/400-Passwörtern mit einer Windows 2000/2003 Domain oder einem einzelnen Windows-Server. Einträge in das Windows 2000 Event Log erscheinen auch in einer Nachrichtenwarteschlange auf der iSeries. Das ermöglicht dem OS/400 Operator, die iSeries zu managen und gleichzeitig auch die Windows 2000 Server auf dem IXS/IXA zu überwachen. Durch Monitore im Management Central kann man auf die Nachrichten aus dieser Nachrichtenwarteschlange auch automatisch reagieren und auf bestimmte Aktionen aufsetzen.



Verwaltung integrierter xSeries Server via iSeries Navigator

Im iSeries Navigator kann man zusätzlich die Statistikdaten der Windows-Server überwachen, wie z.B. CPU-Auslastung, Paging, Anzahl der angemeldeten Benutzer und die aktuelle Plattenauslastung. Auch Windows-Befehle können von hier aufgerufen werden.

Mit Management Central Pervasive ist es möglich, von einem Handy oder einem PDA aus, folgende Funktionen auf dem IXS/IXA durchzuführen: Statusanzeige des Servers, Shutdown und Restart, Absetzen von Windows-Befehlen und die Überwachung von Events. Alle oben angeführten Funktionen sind auch sehr gut remote durchführbar, d.h., dass diese Lösung auch gerade für Firmen mit Niederlassungen optimal geeignet ist.

Abschließend lässt sich sagen, dass das Angebot für viele IBM eServer iSeries-Kunden, die Windows-Server im Einsatz haben und gerade vor der Migration von NT 4.0 auf Windows 2003 stehen, sehr interessant ist.

1.2
Seite 6

Die Integration von LINUX

Autor:
Robert Engel

Heutzutage sind viele Organisationen mit der Herausforderung einer explosiv wachsenden Informationstechnologie-Infrastruktur konfrontiert. All zu oft passieren diese Erweiterungen ad hoc – gerade so wie die geschäftliche Entwicklung dies bei Bedarf fordert. Als Ergebnis davon sind viele Firmen mit einem Wildwuchs von Servern auf Microsoft Windows-Basis sowie damit einhergehenden immensen Herausforderungen und Kosten in Bezug auf die Verwaltung dieser Systeme belastet. Dank der leistungsfähigen Kombination von Skalierbarkeit, Zuverlässigkeit und Verwaltbarkeit von iSeries-Systemen und der Flexibilität von Linux können Unternehmen eine neue Qualität der Vereinfachung nutzen sowie gleichzeitig Arbeitsumgebungen erweitern und Kosten reduzieren.

Mit den eServern i5 können IBM Kunden die leistungsfähigen Funktionen von i5/OS (der neuesten Generation des OS/400) verwenden und haben außerdem den Zusatznutzen einer Vielzahl von Highlights – dazu gehören:

- Kombination der hoch skalierbaren, verfügbaren und verwaltbaren Basis der eServer i5-Architektur mit der Flexibilität von Linux.
- Möglichkeit zur Konsolidierung von mehreren Stand-Alone Servern auf bis zu 254 Linux-Partitionen, die auf einem einzelnen eServer i5 laufen.
- Angebot von virtuellen Diensten für Linux-Partitionen – inklusive Platten, Band, Ethernet-LAN, CD-ROM und DVD.
- Der neue IBM i5 Server ist das neueste Mitglied der iSeries-Familie sowie der erste Server, der mit dem neu entwickelten POWER5-Prozessor ausgestattet ist.

Flexibel in gemeinsam benutzten Ressourcen

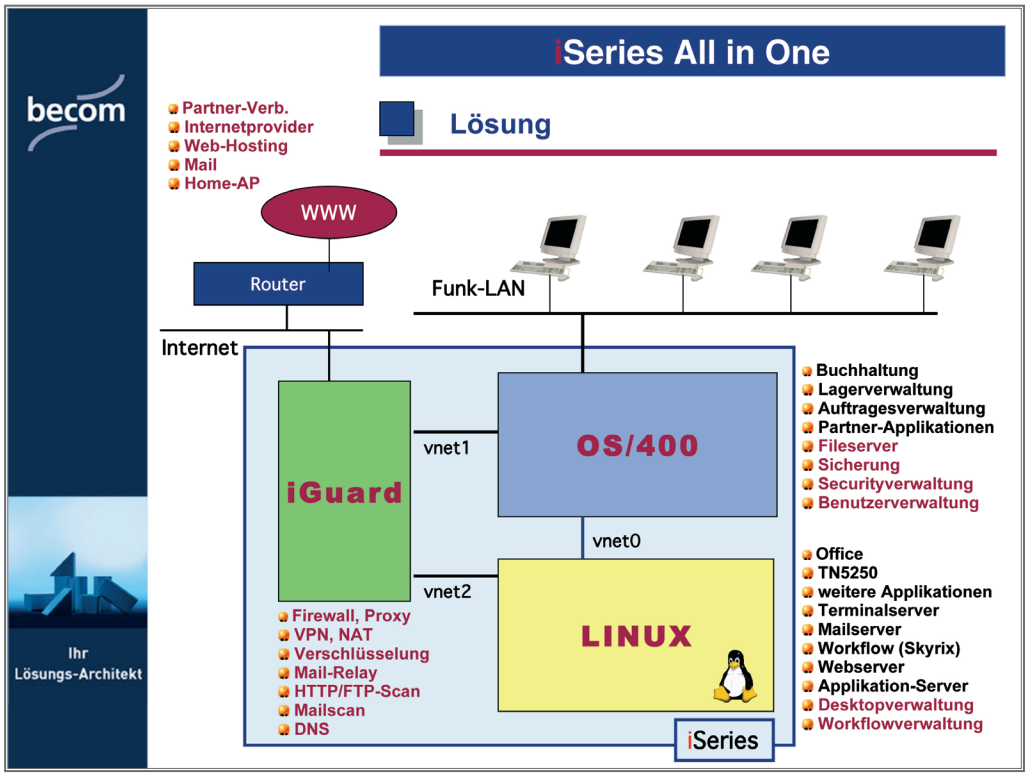
Linux und i5/OS-Partitionen können virtuelle Einheiten wie Platte, Band, Ethernet, CD-ROM und DVD gemeinsam benutzen – alle unter der Steuerung von i5/OS. Dadurch, dass Partitionen virtuelle Einheiten gemeinsam nutzen, helfen die eServer i5 die Hardware-Anforderungen für Linux-Umgebungen zu reduzieren. Aus diesem Grund können Unternehmen schnell und leicht Linux-basierende Anwendungen einsetzen, indem sie bereits vorhandene I/O-Ressourcen, Wissen und Techniken direkt mit einfließen lassen können, die sie eh schon haben. Außerdem ermöglicht logische Partitionierung mit bis zu zehn einzelnen Partitionen auf einem einzelnen Prozessor ein effektiveres Ausnutzen von Prozessor-Ressourcen. Und mit den Möglichkeiten von Capacity on Demand können Kunden dauerhafte oder temporäre Kapazitäten hinzufügen, gerade wenn sie diese brauchen. Zusätzliche Ressourcen müssen nicht permanent gekauft werden. Es wird nur dafür bezahlt, wenn sie tatsächlich benötigt werden. Mehr noch: Kunden können Prozessoren für Linux-Anwendungen hinzufügen, ohne zusätzliche i5/OS-Lizenzge-

Grundwerk 10/2004

bühren bezahlen zu müssen, was noch größere Einsparungen als bisher mit sich bringt.

Linux Workload auf dem gleichen Server

Dank der fortschrittlichen, logischen und dynamischen Partitionierungs-Möglichkeiten der Server i5 können mehrere Linux-Workloads mit anderen e-Business-Lösungen auf einem einzelnen Server kombiniert werden. So können Kunden ihre Anwendungen an einer Stelle verwalten und nutzen.



Der All-in-One-Ansatz der iSeries

Durch den Einsatz von LPAR, der automatischen Ressourcen-Übertragung, Capacity on Demand, virtuellem Speicher, POWER5-Prozessoren und 64 Bit Linux-Distributionen können eServer i5 auch anspruchsvollsten Belastungen standhalten. Außerdem haben Linux-Anwendungen Zugriff auf i5/OS-Daten, Dateien und Anwendungen. So wird es Kunden ermöglicht, die neueste Generation von Linux-basierten e-Business-Anwendungen einzusetzen und die Leistung ihrer bestehenden iSeries-Plattformen zu erweitern. Auf diese Weise können Unternehmen ihre IT-Investitionen maximal ausnutzen, bestehende Ressourcen einsetzen und Wissen für zukünftiges Wachstum verwenden.

Eine starke Grundlage

Die eServer i5-Familie liefert eine zuverlässige und skalierbare Grundlage mit der Flexibilität, mehrere Umgebungen laufen zu lassen und schnell auf neue Belastungen zu reagieren. i5 Server können i5/OS, AIX 5 L, Linux und Mic-

rosoft Windows simultan (über integrierten xSeries Adapter oder integrierten xSeries Server) ausführen. Sowohl Betriebssysteme als auch Anwendungsumgebungen und Sprachen – wie WebSphere, Lotus Domino, Java und hoch effiziente Anwendungen mit 5250 Online-Transaktionsverarbeitung (OLTP) – stehen zur Verfügung. Deshalb bietet die eServer i5-Familie eine stabile Plattform, welche die IT-Infrastruktur vereinfacht, geringe Risiken mit sich bringt sowie Kosten herunterfährt und Produktivität steigert, um den Anforderungen von heute und morgen zu entsprechen. Und dank ihrer integrierten Verwaltungsmöglichkeiten werden Linux Workloads auf eServer i5 einfacher als je zuvor.

Leistung partitionieren

Wenn fortschrittliche LPAR-Techniken ausgenutzt werden, können Kunden mehrere Stand-Alone-Server in bis zu 254 Linux-Partitionen auf einem einzelnen eServer i5 konsolidieren. Automatische Prozessor- und dynamische Speicherressourcen-Verschiebung zwischen einzelnen Partitionen helfen, um sofort auf veränderte Anforderungen reagieren zu können. Außerdem stellt die fortschrittliche virtuelle Ethernet-Technik schnelle (bis zu 1 Gbit/Sekunde) und sehr sichere Verbindungen unter diesen Betriebsumgebungen sowie eine wirkungsvolle serverinterne Kommunikation bereit. Nicht zuletzt können Unternehmen mit eServern i5 ihre konsolidierten Hardware-Ressourcen über mehrere Betriebssystem-Umgebungen hinweg zentral verwalten. Dies hilft, um flexibel zu bleiben und um Kosten für Verwaltung und Support von solch heterogenen Umgebungen zu reduzieren.

Virtualisierung von Speicher-Ressourcen

eServer i5 bieten SAN-ähnliche Funktionen für Linux-Partitionen. Durch Speichervirtualisierung können eServer i5-Kunden, i5/OS, AIX, Linux und Windows-Plattenressourcen von einem einzelnen Managementsystem aus verwalten. eServer i5 helfen Platten über RAID zu schützen und können dem Linux zugewiesenen Plattenplatz hinzufügen, verschieben oder löschen. Außerdem profitieren Linux-Partitionen von den vollen i5/OS-Systemsicherungsfähigkeiten und können eServer i5-Bandeinheiten für Sicherungsoperationen nutzen. Speichervirtualisierung hilft Unternehmen, die Speicher ihrer eServer i5-Umgebung bei niedrigen Gesamtkosten (Total Cost of Ownership) zu nutzen und dabei sowohl Verfügbarkeit als auch Zuverlässigkeit von Linux und i5/OS Workload zu verbessern. Mit dem eServer i5 kann i5/OS alle Plattenlaufwerke für multiple Betriebsumgebungen verwalten – inklusive Linux-Partitionen. Virtuelle Speicherbereiche, die für jede Linux-Partition zur Verfügung stehen, umfassen alle Plattenlaufwerke auf dem eServer i5 und werden zentral verwaltet. Daraus resultieren die verbesserte Leistung und die niedrigeren Kosten.

Grundwerk 10/2004

Integration für leichteren Einsatz

eServer i5 bieten viele Integrationspunkte, die i5/OS-Anwendungen und Daten verwenden, um bekannte Linux-Anwendungen zu unterstützen. Deshalb sind Linux-Anwendungen auf dem eServer i5 in der Lage, auf i5/OS-Daten, Dateien sowie Anwendungen zuzugreifen und sie können Jobs in der i5/OS-Umgebung aufrufen. Solche Lösungen beinhalten Anwendungen, die auf dem WebSphere Application Server und DB2 UDB basieren, genauso wie e-Business-Anwendungen, elektronischer Geldverkehr und WWS-Anwendungen. Wenn die Linux-Anwendungen ausgenutzt werden, um das Leistungsspektrum von i5/OS zu erweitern, können eServer i5-Kunden ihre IT-Investionen optimieren. Außerdem bieten eServer i5 den Kunden die innovative Fähigkeit, diese Anwendungen in einer oder mehreren Partitionen auszuführen, wobei das i5/OS als führendes Betriebssystem in anderen Partitionen läuft.

Unterstützung von IBM

IBM bietet seinen Kunden einen umfassenden technischen Service und Support. Dieser hilft den Kunden bei der Auswahl, Implementierung bis hin zur Nutzung von IT-Lösungen, die auf ihre Geschäftsanforderungen zugeschnitten sind. IBM-Experten können sowohl bei Geschäfts- als auch bei IT-Beratung sowie bei umfassenden Systemverwaltungs-Services helfen. IBM hat eine gewaltige Erfahrung in der Beratung von Kunden und bei der Entwicklung deren e-Business-Infrastrukturen. Um ihre durchgehende Lösung für Linux auf eServer i5 fertig zu stellen, bietet IBM ein Portefeuille von Linux Services an: Berater für die Beurteilung, Gestaltung und Implementierung von Linux Workloads inklusive Daten- und Druckdienste, TCP/IP-Infrastruktur, Web und Sicherheit. Daneben gibt es Angebote für Linux-Installationen, Partitionierungs-Konfiguration und Linux- bzw. i5/OS-Anwendungs-Implementierung. Der Linux-Supportbereich der IBM Global Services ist für den umfassenden, unternehmensweiten „Rund-um-die-Uhr“-Support sowie für die Fehlerbehebung da.

AIX auf eServer i5

Autor:
Willy Günther,
iSeries Technical Sales IBM Deutschland GmbH

Mit dem Announcement vom 4.5.2004 hat die IBM ein Nachfolgesystem für die iSeries in den Markt gestellt: eServer i5 – und aus OS/400 wurde i5/OS. Außer einem anderen Namen und neuer Hardware, die auf dem Power5-Prozessor basiert, wurden auch viele funktionale Erweiterungen implementiert. Eine dieser Erweiterungen ist die Umsetzung des Statement of Direction. Damit wird AIX zukünftig auch auf eServer i5-Systemen verfügbar sein.

Voraussetzungen

AIX ist nur auf den neuen eServer i5-Modellen mit Power5-Prozessor lauffähig. Man benötigt als minimale Konfiguration ein Zwei-Prozessor-System, um AIX 5L V5.2 auf einem eServer i5 zu installieren. Auf bestehenden iSeries-Modellen – zum Beispiel einer 825 mit Power4-Prozessor – kann AIX nicht installiert werden.

Als erste Version wird ein Update für AIX 5L V5.2 verfügbar sein, das die Unterstützung für den Power5-Prozessor enthält. Geplante Verfügbarkeit für diese Version war der 27.8.2004. Zu einem späteren Zeitpunkt ist eine AIX-Version geplant, die weitere Funktionen unterstützt.

Grundlage für die Implementierung von AIX auf eServer i5 ist wie bei Linux die Möglichkeit, durch Logische Partitionierung (LPAR) ein physisches System in verschiedene unabhängige „virtuelle Systeme“ bzw. Partitionen aufzuteilen, auf denen unterschiedliche Betriebssysteme laufen können. Damit lassen sich vorhandene Hardware-Ressourcen sehr flexibel aufteilen.

AIX 5L Version 5.2

Mit AIX 5L Version 5.2 können dedizierte Prozessoren einer AIX-Partition zugeordnet werden. Das bedeutet: Es müssen immer ganze Prozessoren einer AIX-Partition zugewiesen werden.

Deshalb lässt sich diese AIX-Version auch nur auf einem eServer i5-System mit mindestens zwei Prozessoren installieren. Eine AIX 5L V5.2-Partition benötigt als Minimum 128 MB Hauptspeicher und 1 GB Plattenspeicher. Der Hauptspeicher einer AIX-Partition ist in 16 MB-Schritten veränderbar.

Unter AIX 5L Version 5.2 können nur dedizierte Ressourcen von AIX verwendet werden. Das heißt: AIX verwaltet diese Ressourcen – wie z. B. Platten oder Ethernet Adapter – selber. Die entsprechenden Adapter sind dieser Partition zugeordnet.

Die Kommunikation erfolgt über einen in der Partition installierten LAN-Adapter. Ein Lastausgleich zwischen Partitionen ist bei dedizierten Prozessoren nicht möglich. Es besteht aber die Möglichkeit, die dedizierten Prozessoren einer Partition für andere Partitionen zu nutzen, wenn die entsprechende Partition nicht in Betrieb ist.

Angelegt und verwaltet wird eine AIX-Partition durch die Hardware Management Console (HMC). Die Hardware Management Console stellt auch die Konsole für AIX zur Verfügung.

Die Lizenz für AIX 5L Version 5.2 ist nicht im Preis für ein eServer i5-System enthalten und muss separat bestellt werden. Die AIX 5L-Lizenzen orientieren sich an den AIX-Prozessor-Gruppen, nicht an den eServer i5-Prozessor-Gruppen. Die IBM eServer i5 Modelle 520 werden der AIX-Prozessor-Gruppe E5 und die IBM eServer i5 570 Modelle der AIX-Prozessor-Gruppe F5 zugeordnet. Für jeden Prozessor, der einer AIX-Partition zugeordnet wird, benötigt man eine AIX-Lizenz. Für jede AIX-Lizenz muss zusätzlich ein IBM Software-Wartungsvertrag über ein oder drei Jahre abgeschlossen werden.

Zukünftige geplante Erweiterungen

Mehr Flexibilität ist für eine zukünftige AIX-Version geplant, die dann die Funktionen der IBM Virtualization Engine voll nutzen kann. AIX wird dann virtuelle Platten und virtuelles Ethernet unterstützen. Diese Technologie wird schon heute von den meisten iSeries Linux-Installationen benutzt und hat sich in der Praxis gut bewährt.

Diese zukünftige AIX-Version wird auch „Micro Partitioning“ unterstützen. Das bedeutet, einer AIX-Partition kann auch nur ein Teil eines Prozessors zugewiesen werden. Der dynamische Ausgleich von Prozessor-Ressourcen innerhalb eines „Shared Processing Pools“ zwischen Partitionen zum Ausgleich von Workload-Spitzen ist dann ebenfalls möglich.

Einsatzgebiete

AIX 5L auf eServer i5 ist interessant für Kunden, die bereits Unix-Anwendungen betreiben, deren Kern-Plattform aber die iSeries ist. Diese Kunden müssen heute für einzelne Anwendungen einen zusätzlichen Unix-Rechner installieren. Zukünftig können Sie diesen Workload auf eServer i5 integrieren. Erst die Integration ermöglicht es, alle Vorteile der „IBM Virtualization Engine“ voll zu nutzen.

Mit der Möglichkeit, AIX 5L auf einem eServer i5 zu implementieren, eröffnen sich für bestehende iSeries-Kunden neue Möglichkeiten, um moderne und leistungsfähige Unix-Anwendungen auf Ihrer Plattform einzusetzen.

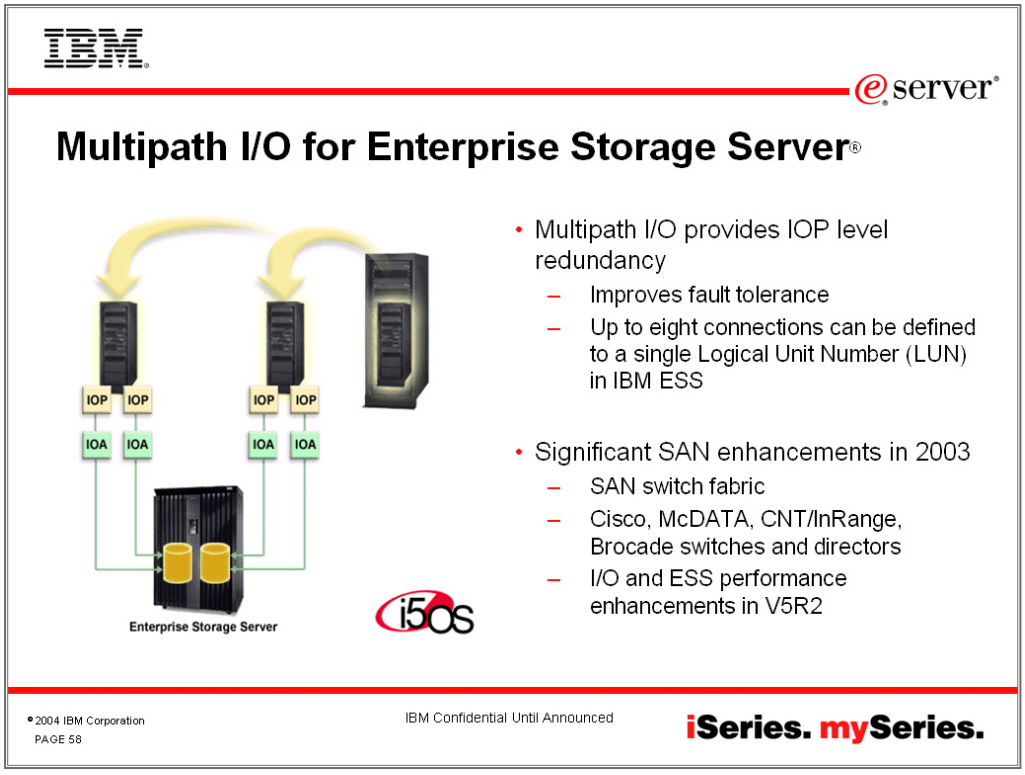
Ein neues Store-Age mit i5/OS

Mit den IBM Ankündigungen vom April 2004 im Storage-Bereich und vom Mai 2004 im iSeries-Bereich gibt es einige Neuerungen, die im Nachfolgenden kurz vorgestellt werden.

Autorin:
Edelgard Schittko, iSeries Technical Sales der IBM Deutschland GmbH

IBM Enterprise Storage Server (ESS) und iSeries/i5

Seit OS/400 V5R1 ist es möglich, einen IBM ESS über Storage Area Network (SAN) an eine iSeries anzuschließen – unter Verwendung von PCI bzw. PCI-X Fibre Channel Disk Controllern (iSeries Feature 2766 für V5R1 bzw. iSeries/i5 Feature 2787 für V5R2 bzw. V5R3). Die Platten der ESS werden dabei als externe Platten für die iSeries verwendet. Die externen Platten werden über die Definition von Logical Units (LUNs) dem OS/400 hinzugefügt, wobei für jeden 2766/2787 maximal 32 verschiedene LUNs definiert werden können. Bisher konnte jede Logical Unit (LUN) nur einmal über das SAN dem OS/400 bekannt gegeben werden. Mit i5/OS V5R3 ist der so genannte Multipath Support möglich, d.h.: Bis zu acht Verbindungen (Pfade) können maximal zu einer Logical Unit definiert werden. Damit wird einerseits eine höhere Ausfallsicherheit erreicht, denn falls doch mal eine Verbindung (Pfad) ausfallen sollte, dann wird ein anderer Pfad verwendet, um die entsprechende Logical Unit zu erreichen. Andererseits kann durch die Verwendung mehrerer Pfade eine gewisse Performance-Verbesserung erreicht werden, weil ein Basic Load Balancing nach dem Round-Robin Algorithm durchgeführt wird.



Multipath I/O für Enterprise Storage Server

Für die Implementierung des Multipath Support ist eine genaue Hardware-Planung notwendig, weil i5/iSeries-seitig mehr Karten und zugehörige Steckplätze benötigt werden.

Beispiel:

Um für eine i5 Modell 570 für 32 LUNs je zwei Verbindungen (Pfade) – maximal acht sind möglich – über den Multipath Support zu ermöglichen, werden zwei PCI-X Fibre Channel Disk Controller 2787 und zwei dedizierte PCI Input/Output Processoren (IOPs) 2844 verwendet. Wobei für jeden 2787 insgesamt 32 LUNs definiert werden – 16 als „Erst"-Verbindung (Pfad) und die anderen 16 als „Zweit"-Verbindung (Pfad). Jede LUN hat also zu beiden 2787 je einen Pfad.

Damit i5/OS V5R3 die verschiedenen Verbindungen unterscheiden kann, bekommen sie automatisch eindeutige Namen – DMPxyz – zugeordnet, obwohl diese Verbindungen alle auf die gleiche LUN verweisen; der Default Name DDxxx für diese LUN wird durch die neuen Bezeichnungen DMPxyz ersetzt.

In Zusammenhang mit den aktuellen Ankündigungen soll hier bzgl. IBM ESS noch auf folgende weitere Neuerungen hingewiesen werden:

ESS-seitig wird mindestens Level 2.2.0 benötigt, bevor ein Upgrade der angeschlossenen iSeries auf i5/OS V5R3 durchgeführt wird bzw. bevor eine i5 angeschlossen werden kann. Details dazu siehe Memorandum to Users:

http://publib.boulder.ibm.com/infocenter/iseries/v5r3/ic2924/info/rzaq9.pdf.

Die neu angekündigte IBM ESS Modell 750 ist sowohl für iSeries als auch für i5 unterstützend, weitere Details siehe:

http://www.storage.ibm.com/disk/ess/ess750/pdf/ess750-datasheet.pdf.

IBM TotalStorage UltriumTape 2U Autoloader 3581

In der Storage-Ankündigung vom 27.04.2004 ist der IBM TotalStorage UltriumTape 2U Autoloader 3581 enthalten, welcher aus iSeries/i5-Sicht sehr interessant ist. Dieses neue Modell 3581 gibt es seit 14.05.2004 und zwar in zwei verschiedenen Varianten: 3581-L28 für LVD SCSI-Anschluss und 3581-F28 für FibreChannel- (2Gbps-) Anschluss. Es gibt jetzt also erstmals ein einzelnes Ultrium2-Laufwerk mit Autoloader und FibreChannel-Anschluss, was sicherlich für viele iSeries/i5-Kunden relevant sein könnte. Außerdem verfügt dieses neue Modell 3581 über insgesamt acht Slots statt sieben wie die

bisherige 3581 – und der über Feature 7004 bestellbare BarCodeReader belegt keinen Slot mehr. Die maximal mögliche Kapazität der neuen 3581 beträgt native 1.6 TB bzw. mit Komprimierung 3.2 TB bei Verwendung aller acht Slots mit Ultrium2-Kassetten. Weitere Details siehe:

http://www.storage.ibm.com/tape/lto/3581/prod_data/pdf/gm13-0614-00.pdf.

IBM TotalStorage UltriumTape 2U Autoloader 3581

Für den Anschluss der neuen 3581-L28 bzw. -F28 an eine iSeries/i5 werden in der iSeries/i5 folgende Adapter benötigt: für 3581-L28 entweder PCI-X Tape Controller 5702 (iSeries) bzw. 5712 (i5) oder PCI-X Tape/DASD Controller 5705 (iSeries) und 5715 (i5); für 3581-F28 der PCI-X Fibre Channel Tape Controller 5704 (iSeries oder i5).

Anschluss-Controller

Auch die neuen 3581-Modelle können und sollten mit dem Lizenzprogramm 5722-BR1 IBM Backup Recovery and Media Services for iSeries (BRMS) verwendet werden. Details zu BRMS und den Erweiterungen in BRMS mit i5/OS V5R3 sind zu finden unter:

http://www-1.ibm.com/servers/eserver/iseries/service/brms

und

http://www-1.ibm.com/servers/eserver/iseries/service/brms/v5r3news.htm.

1.5
Seite 4

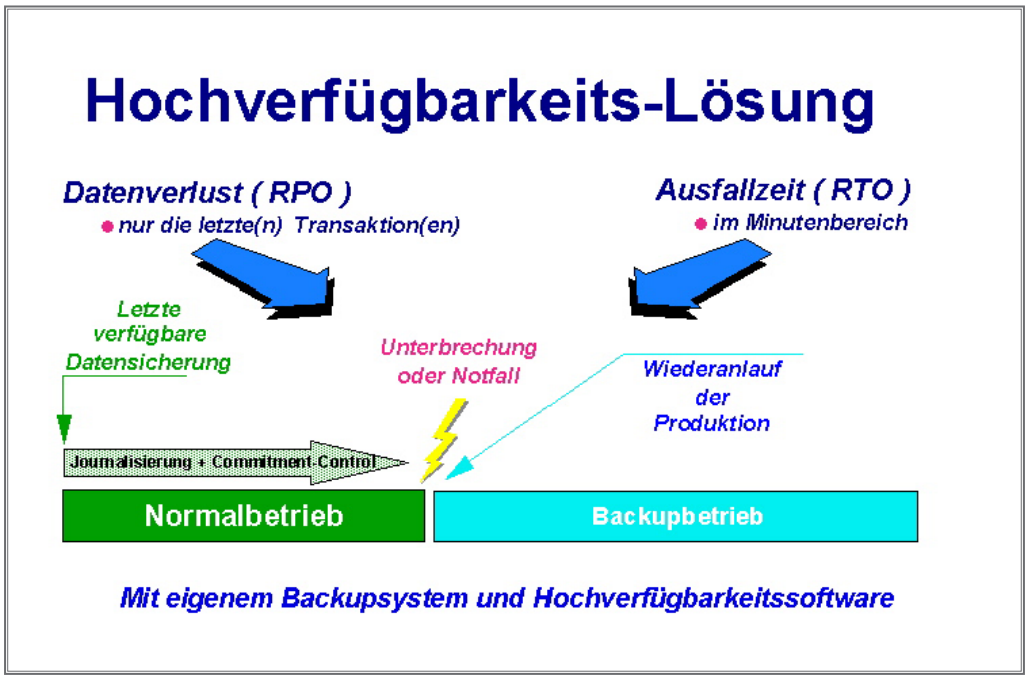
Auswahl-Kriterien für Hochverfügbarkeits-Lösungen

Autor:
Peter Nimz, Senior Consultant der IBM

Größere Investitionsentscheidungen – wie zum Beispiel der Kauf eines Hauses oder eines Autos – basieren in der Regel auf einer Entscheidungsmatrix. Das heißt: Man wird sich im Vorfeld genau überlegen, welche Kriterien erfüllt sein müssen. Dies sollte auch für die Auswahl einer Hochverfügbarkeits-(HV-) Lösung gelten.

Bei der Evaluierung von HV-Lösungen sind zwei Zielsetzungen von wesentlicher Bedeutung: Recovery-Time und Recovery-Point. Diese beiden Aspekte sollen im Folgenden kurz für drei Typen von HV-Lösungen diskutiert werden; es sind die Topologien:

- shared/switched Devices,
- physische Kopien,
- logische Kopien.



Zielsetzung ist die Minimierung von RTO und RPO

Recovery-Time-Ojective (RTO)

Recovery-Time ist die Zeitspanne, die erforderlich ist, um nach einem geplanten oder ungeplanten Ausfall den normalen, operationalen Betrieb wieder aufnehmen zu können. Bezüglich der Daten stellt sich vor allem die Frage nach der Zeitdauer bis der Datenzugriff wieder erfolgen kann – und zwar:

- im Allgemeinen,
- für das Betriebssystem,
- für die Anwendungen beziehungsweise für die Endbenutzer.

Zum Zeitpunkt des Ausfalls kann man sowohl für shared/switched Devices als auch für physische Kopien (synchrone Replikation unterstellt) davon ausgehen, dass im Normalfall die Daten im Allgemeinen zugänglich sind. Der Stand der Daten entspricht dem Zustand, der zuletzt gerade noch geschrieben werden konnte.

Bei logischen Kopien kann es sein, dass anhängig von dem Status der Synchronisation zum Zeitpunkt des Ausfalls noch einige Veränderungen geschrieben werden müssen, bevor die Daten im Allgemeinen zugänglich sind. Alle Lösungen benötigen eine gewisse Zeit für die Ausführung der so genannten Failover-/Switchover-Prozeduren, bis die Daten dann für das Betriebssystem verfügbar sind. In diesem Zusammenhang ist anzumerken, dass sich die Recovery-Time bei Speichersystem-basierenden physischen Kopien erheblich verlängert, wenn ein IPL des Servers erforderlich ist, bevor auf die Daten zugegriffen werden kann.

Unter der Annahme, dass ausreichend hohe Bandbreiten zur Verfügung stehen und die Verzögerungen beim Schreiben von Veränderungen zeitlich gegen Null tendieren, können Anwendungen und Benutzer auf eine logische Kopie unmittelbar nach Durchführung der Failover-/Switchover-Prozeduren zugreifen, ohne dass zusätzliche Wiederherstellungsaktivitäten erforderlich sind.

Recovery-Point-Objective (RPO)

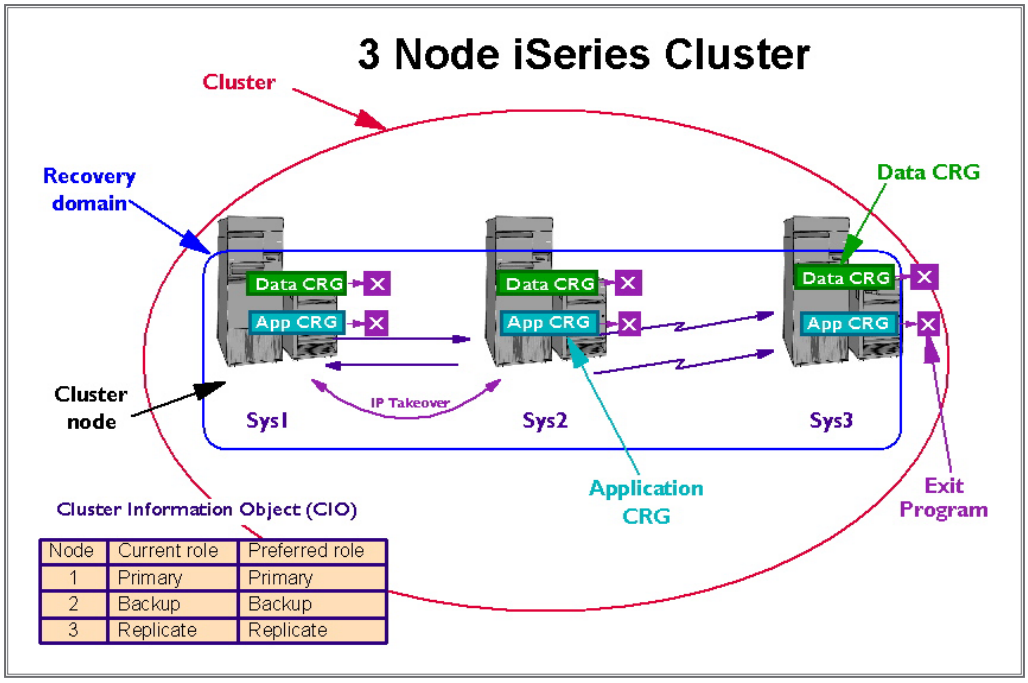
Recovery-Point ist der Punkt im Zeitablauf, auf den wieder zurückgesetzt wird, nachdem der Recovery-Prozess abgeschlossen ist und damit gleichzeitig auch der Punkt, auf dem der operationale Betrieb und die Anwendungen/Endbenutzer nach dem geplanten oder ungeplanten Ausfall wieder aufsetzen. Sowohl bei den shared/switched Devices als auch bei den physischen Kopien (synchrone Replikation unterstellt) entspricht der Recovery-Point exakt dem Zustand dessen, was zuletzt gerade noch auf die Platten geschrieben werden konnte. Veränderungen, die sich noch im Hauptspeicher befanden, sind verloren. Dies bedeutet, dass für kritische Daten unbedingt Journaling eingesetzt werden sollte.

Bei logischen Kopien entspricht der Recovery-Point dessen, was zuletzt gerade noch übertragen und vom Backup-Server akzeptiert wurde. Veränderungen, die nicht mehr übertragen/akzeptiert werden konnten, sind verloren.

Für alle Lösungen gilt, dass die Integrität der Transaktionen und Daten nur dann absolut sichergestellt werden kann, wenn die Anwendungen Commitment-Control verwenden, weil der Recovery-Point dann nur solche Transaktionen und Daten beinhaltet, die vollständig übertragen und akzeptiert worden sind. Ohne die Verwendung von Commitment-Control ist die Wahrscheinlichkeit hoch, dass der Recovery-Point unvollständige Transaktionen und Daten enthält.

iSeries Cluster-Lösung

Die maximale Anforderung an eine HV-Lösung lautet: Die Recovery-Time sollte möglichst gegen Null tendieren – und für den Recovery-Point wird höchstens der Verlust der letzten gerade noch angefangenen Transaktion akzeptiert. Um diesen beiden Ansprüchen möglichst gerecht werden zu können, stellt die iSeries bereits seit 1999 Cluster-Lösungen zur Verfügung.



iSeries-Cluster für Business Continuity

Ein Cluster (siehe Bild) stellt einen Verbund von Systemen dar, die als eine einzige „Computing-Resource“ zu betrachten sind. Das Betriebssystem OS/400 enthält so genannte „Cluster-Resource-Services“, die es erlauben, einen Verbund von bis zu 128 Systemen aufzubauen. Weiterhin definieren diese Services, welche kritischen Ressourcen (Daten, Anwendungen, Devices) gegen einen geplanten oder ungeplanten Ausfall geschützt werden sollen. Kommunikationskomponenten wie „Heart-beating“ sorgen dafür, dass festgestellt werden kann, wenn einzelne Systeme im Verbund nicht mehr erreicht werden können. Bei Ausfällen oder Engpässen können kritische Cluster-Ressourcen automatisch auf ein Backup-System übernommen werden. Jedes System im Verbund kann diverse kritische Ressourcen „hosten“ und für jede dieser Ressourcen können wiederum Backups definiert werden. Ein System kann somit einerseits Backup für eine kritische Ressource sein, während es gleichzeitig andere kritische Ressourcen seinerseits „hostet“.

Mit der Version V5R2 des Betriebssystems OS/400 stehen zwei Technologien zur Verfügung, um die so genannte „Resiliency“ (bezeichnet quasi die Robustheit gegenüber äußeren Einflüssen) der Daten zu gewährleisten: einerseits die

logische Replikation, bei der eine so genannte „Cluster-Resource-Group" (CRG) für die Daten mit dieser logischen Replikation verknüpft wird. Dieser Mechanismus ermöglicht bei geplanten oder ungeplanten Ausfällen einen koordinierten, automatischen Failover beziehungsweise einen vom Benutzer initiierten Switchover auf das Backup-System. Andererseits können so genannte „Switchable Independent Disk Pools" (auch als Independent Auxiliary Storage Pools oder IASPs bezeichnet) mit einer Daten-CRG verbunden werden, um den soeben beschriebenen koordinierten, automatischen Failover beziehungsweise den benutzerinitiierten Switchover durchzuführen.

Die Cluster-Architektur der iSeries beinhaltet darüber hinaus aber auch Mechanismen, die im Fehlerfall dafür sorgen, dass eine Koordination stattfindet zwischen der Resiliency der Daten und der Resiliency der auf diesen Daten basierenden Anwendungen. Hierfür ist es erforderlich, neben der Daten-CRG auch eine Cluster-Resource-Group für Anwendungen zu definieren. Diese Anwendungs-CRG stellt sicher, dass die mit der Anwendung verbundene IP-Adresse beim Fail-/Switchover automatisch auf das neue primäre System übernommen wird. Das Exit-Programm der Anwendungs-CRG sorgt dann dafür, dass ein ordnungsgemäßer Restart für die Anwendungsumgebung erfolgen kann und die Endbenutzer exakt zu dem gewünschten Recovery-Point geführt werden.

Verfügt Ihr Unternehmen über eine HV-Strategie?

Die Wettbewerbsfähigkeit der Unternehmen wird in Zukunft mehr denn je von der Verfügbarkeit ihrer IT-Systeme und Anwendungen abhängig sein. Vor diesem Hintergrund ist es zwingend erforderlich, eine auf die Geschäftsanforderungen abgestimmte Hochverfügbarkeits-Strategie zu implementieren, um optimal auf geplante/ungeplante Ausfälle oder gar für ein Disaster vorbereitet zu sein. Bei der Evaluierung der unterschiedlichen HV-Lösungen ist es wichtig, exakt zu verstehen, was die einzelnen Lösungsansätze hinsichtlich RTO, RPO und Clustering leisten können. Vor der Entscheidung ist es sinnvoll, Referenzen der Lösungsanbieter zu prüfen, um im Vorfeld die Durchführung und den realistischen Zeitaufwand eines kompletten Fail-/Switchover (vom primären System zum Backup-Server und auch wieder zurück!) mit dem eigenen Umfeld abzugleichen. Die IBM und ihre High-Availability-Partner verfügen über eine langjährige Erfahrung in diesem Umfeld – nutzen Sie diese Expertise. Weitere Informationen zu diesem Thema finden Sie auch unter:

http://www.ibm.com/servers/eserver/iseries/ha.

Der neue IBM WebSphere Portal Express Server

Autorin:
Ursula Althoff,
IBM Deutschland
Systems Sales und
Technical Support
iSeries

Im November 2003 kündigte IBM WebSphere Portal Express und Express Plus für Multiplatforms Version 5.02 an, der einen Portal Server für die IBM eServer iSeries einschließt. Dieses IBM WebSphere Portal ist eines der umfassenden Portal-Angebote der Industrie. Es enthält ein breites Spektrum von Portal-Techniken, die Ihnen helfen, B2C-, B2B- und B2E-Portale zu entwickeln und zu betreiben.

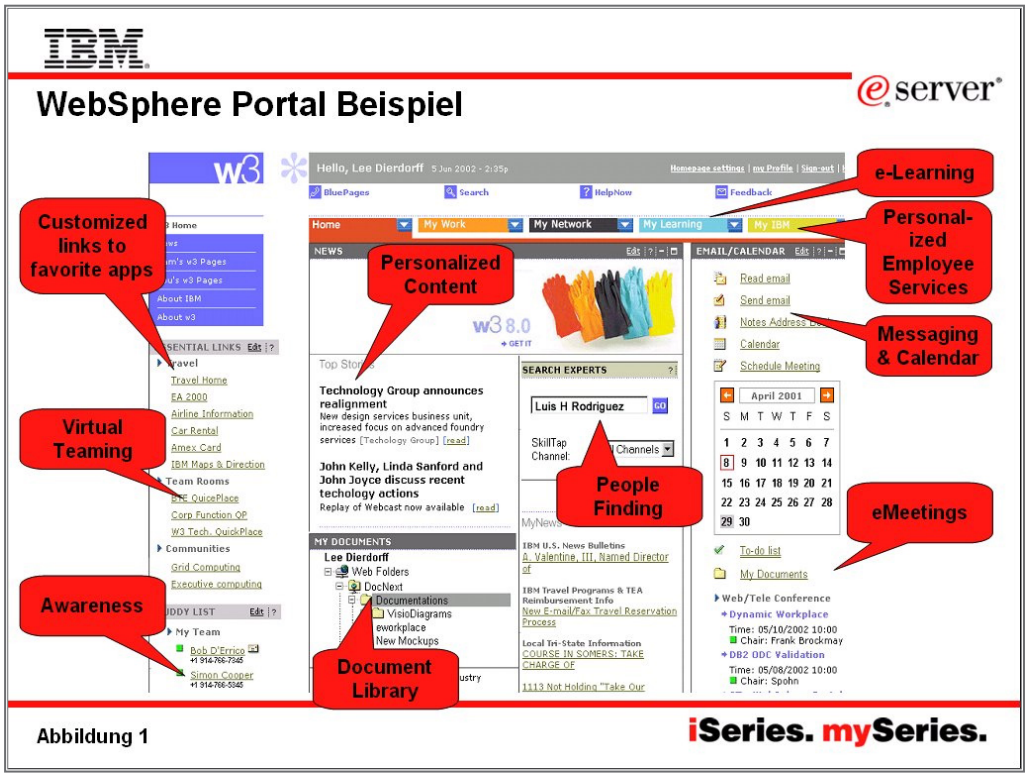
Ein Portal kann am besten als Server definiert werden, der Inhalte aus unterschiedlichen Quellen, die auf einem oder mehreren unterschiedlichen Servern zu finden sind, in eine gemeinsame Benutzerschnittstelle, eine Web-Seite oder einen Satz von Web-Seiten integriert.

Portale stellen somit einen vereinheitlichten, personalisierten Zugriffspunkt auf Informationen und Anwendungen für Benutzer bereit. Die dynamischen Inhalte können von unterschiedlichen Quellen zur Verfügung gestellt werden. Portale tun noch viel mehr: Sie liefern wertvolle Funktionen wie Sicherheit, Suchfunktionen, Single Sign On, Kollaboration und Workflow. Tatsächlich sind Portale die nächste Generation von Desktops, die e-Business-Anwendungen aller Arten über das Netz für alle Arten von Clients bereitstellen.

Vorteile und Einsatz von IBM WebSphere Portalen

Portal-Benutzer können automatisch die dynamische Information bekommen, die sie brauchen. Sie können schnell Geschäftsprozesse über kritische Anwendungen ausführen. Sie können mit Portal-Benutzern innerhalb und außerhalb Ihres Unternehmens kollaborieren. Durch Bereitstellung dieser industrieführenden Portal-Lösungen für Ihr e-Business hilft IBM Ihnen, Angestelltenproduktivität zu steigern, Kosten zu reduzieren sowie die Beziehungen zu Ihren Kunden und Handelspartnern zu stärken. Die folgende Abbildung zeigt beispielhaft, wie ein Portal verschiedene Informationen und Anwendungen einem Benutzer zur Verfügung stellen kann.

Grundwerk 10/2004



Beispielportal mit Informationen aus unterschiedlichen Anwendungen

WebSphere Portal Express und Express Plus bieten IBM eServer iSeries-Kunden einen Weg, sowohl existierende als auch neue Anwendungen in eine robuste, zuverlässige und skalierbare Portal-Umgebung auf iSeries zu integrieren. Hervorzuheben ist die schnelle und einfache Installation des Portal Express oder Express Plus Servers auf dem iSeries Server; sie wird über einen eigenen Installations-Assistenten (Wizzard) unterstützt.

Planung Ihrer Portal Strategie

Setzen Sie WebSphere Portal ein, um einen neueren und produktiveren „Desktop“ zu entwickeln, der eine einzigartige Ansicht von Daten und Anwendungen bietet. Sie können iSeries-Daten, 5250-Anwendungen, die in RPG oder COBOL geschrieben sind, kollaborative Anwendungen wie Lotus Domino, IBM Lotus Instant Messaging and Web Conferencing (Sametime) 6.5.1 for iSeries und IBM Lotus Team Workplace (QuickPlace) 6.5.1 for iSeries sowie bestehende Web-Anwendungen integrieren. Als ersten Schritt können iSeries-Kunden durch Einsatz von iSeries Access for Web bestehende 5250-Anwendungen sehr einfach in Ihr Portal aufnehmen – und neben diesen die oben erwähnten Kollaborations-Anwendungen zur Verfügung stellen. Angestellte, Verkäufer, Lieferanten, Verteiler und Endbenutzer können so von einer Umgebung profitieren, die all die Information und Ressourcen liefert, die sie brauchen, um ihre Arbeit in einer einzelnen, konsolidierten Netzumgebung durchzuführen.

Auf ein Portal wird über einen Web-Browser zugegriffen. Jede „Portal-Seite“ ist aus einem oder mehreren Portlets – Java Code – zusammengesetzt. Diese Portlets liefern Informationen oder Anwendungen aus einer Quelle innerhalb einer IT-Infrastruktur. Portlets können entwickelt werden, um auf spezifische Informationsquellen zuzugreifen – oder sie werden von externen Quellen bereitgestellt. iSeries-Kunden steht eine Vielzahl von Portlets zur Verfügung.

IBM-Produkte, die Web-Zugriff auf bestehende 5250-Anwendungen und Portal-Unterstützung beinhalten:

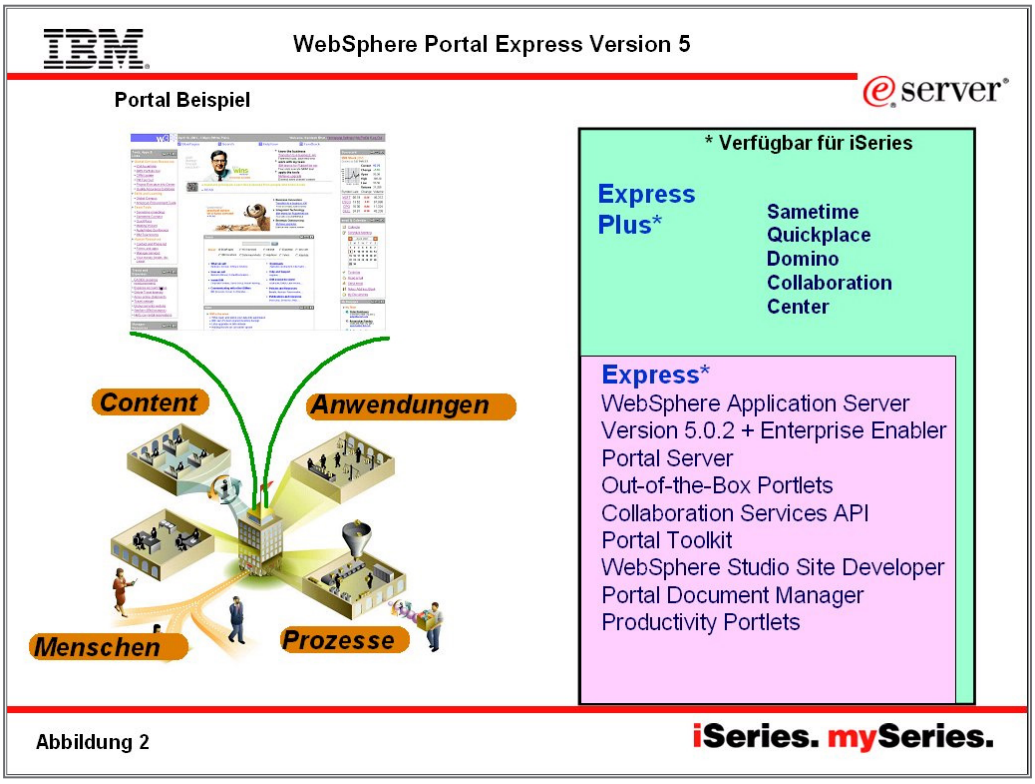
- iSeries Access for Web liefert ein Portlet, über das alle iSeries-Anwendungen – innerhalb einer WebSphere Portal Express Umgebung – integriert werden können.
- Das IBM WebFacing Tool – bereitgestellt in WebSphere Development Studio Client, Advanced Edition for iSeries Version 5.1 – beinhaltet ein iFrame Portlet. 5250-Anwendungen, die mit dem WebFacing Tool als Web Anwendungen bereitgestellt wurden, können somit in die Portal-Umgebung einbezogen werden.
- WebSphere Host Access Transformation Services, Version 5.0 ermöglicht WebSphere Portal Integration.
- Der IBM Host On-Demand Version 8.0 Emulator kann innerhalb einer WebSphere Portal-Umgebung laufen.
- WebSphere Portal Express Version 5.0.2 liefert Portlets, die Zugriff zu Lotus Kollaborations-Anwendungen ermöglichen (siehe oben).

Der WebSphere Portal Katalog

http://www.ibm.com/developerworks/websphere/zones/portal/catalog/

beschreibt die Portlet-Angebote, die für WebSphere Portal-Produkte verfügbar sind. Diese Angebote werden von IBM und den Geschäftspartnern der IBM geschaffen.

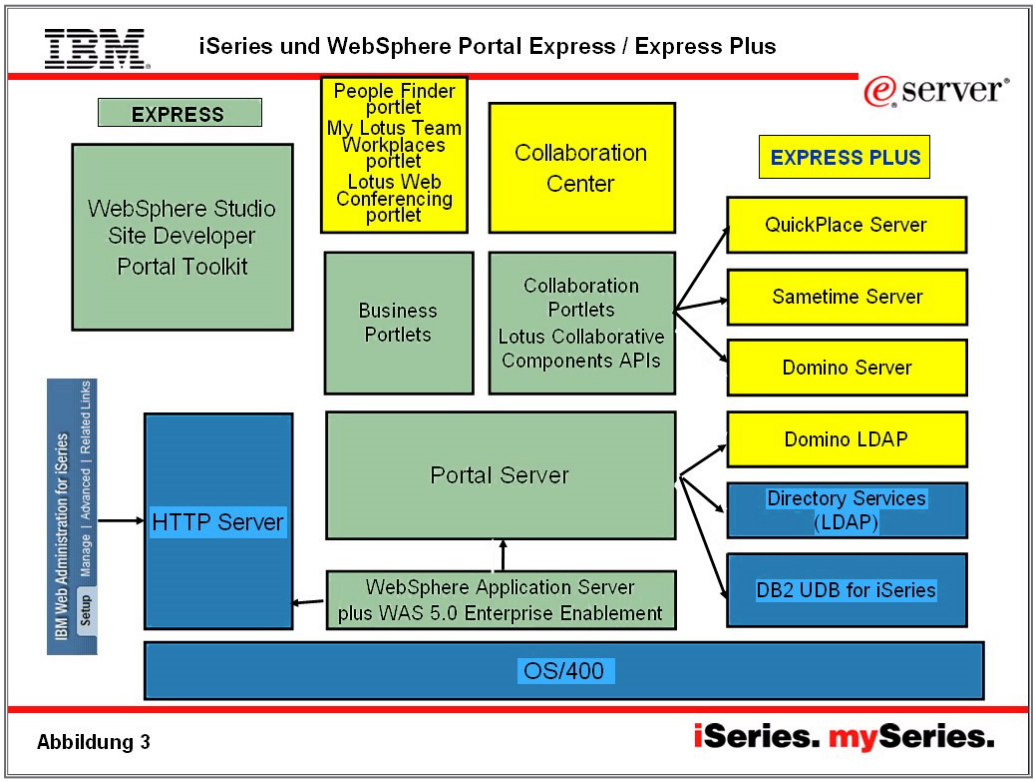
Die folgende Abbildung zeigt die Komponenten, die mit der WebSphere Portal Express Software ausgeliefert werden.



Komponenten, die mit WebSphere Portal Express ausgeliefert werden

Mit WebSphere Portal Server Express Plus for iSeries werden zusätzliche Produkte – wie Lotus Domino, IBM Lotus Instant Messaging and Web Conferencing (Sametime) 6.5.1 for iSeries und IBM Lotus Team Workplace (QuickPlace) 6.5.1 for iSeries – mitgeliefert. Diese Zusatzkomponenten können alle auf dem gleichen iSeries Server – wie auch der Portal-Server selbst – eingesetzt werden. Es entstehen somit keine Server-Farmen, sondern alles ist auf der iSeries integriert.

Die in der folgenden Abbildung aufgezeigte Software-Umgebung für Portal-Lösungen auf iSeries Servern verdeutlicht, welche Komponenten hier zum Einsatz kommen.



Funktionsweise einer Portal-Lösung auf iSeries

Diese Elemente sind die mit dem iSeries-Betriebssystem ausgelieferten Komponenten

Komponenten von WebSphere Portal Express

Zusätzliche Komponenten von WebSphere Portal Express Plus

WebSphere Portal Express Plus in iSeries with Enterprise Edition

WebSphere Portal Express und Express Plus für Multiplatforms Version 5.02 kann über benutzerbasierte Lizenzen (Paket von jeweils 20 Benutzern – bis zu 2.000 Benutzern) erworben werden. Für potentielles Wachstum kann ein Benutzer eine unlimitierte Lizenz – CPU-basiert – erwerben.

Kunden, die eine iSeries 825, 870 oder 890 nach dem 8. September 2003 oder eine i520 kaufen, erhalten mit der Enterprise Edition Software eine Lizenz für 20 Benutzer für WebSphere Portal Express Plus for Multiplatforms Version 5.02. Beim Kauf einer i570 und der Enterprise Edition sind 60 Benutzer-Lizenzen enthalten.

1.7
Seite 6

Weitere Informationen zu diesem Thema finden Sie auf den folgenden Web Seiten:

WebSphere Portal Express for Multiplatforms

http://www.ibm.com/servers/eserver/iseries/software/websphere/portal/express.html

IBM developerWorks – WebSphere Portal Zone – Informationen für Entwickler und Administratoren von WebSphere Portalen

http://www.ibm.com/developerworks/websphere/zones/portal/

WebSphere Portal for Multiplatforms support page

http://www-306.ibm.com/software/genservers/portalexpress/support/

iSeries Directory Services (LDAP)

http://www.ibm.com/servers/eserver/iseries/ldap/

IBM HTTP Server for iSeries

http://www.ibm.com/servers/eserver/iseries/software/http/

Sametime and QuickPlace on iSeries

http://www.ibm.com/support/docview.wss?rs=203&uid=swg21091353

QuickPlace for integration with Sametime on iSeries

http://www.ibm.com/support/docview.wss?rs=203&uid=swg27003786

Die SQL Query Engine

Autorin:
Sabine Jordan, IBM Technical Support iSeries

Mit der Ankündigung vom 4. Mai 2004 hat IBM unter anderem bekannt gegeben, dass es mit dem neuen Release i5/OS weitere Verbesserungen im Bereich SQL geben wird. Aus diesem Anlass sollen in diesem Artikel nochmals die Fakten über die neue SQL Query Engine (SQE) dargestellt werden, die seit OS/400 V5R2 zur Verfügung steht.

IBM und DB2 UDB sind marktführend im Bereich SQL und insbesondere im Bereich der Query Optimierung. Um diese Technologien schneller für DB2 UDB für iSeries zur Verfügung zu stellen, wurde beschlossen, den Query Optimizer und Teile der Runtime Engine mit OS/400 V5R2 neu zu entwickeln. Diese neue Query Engine ist unter dem Namen SQL Query Engine (SQE) bekannt, da sie in erster Linie für SQL Interfaces wie imbedded SQL, ODBC und JDBC entwickelt wurde. Dabei wurden neueste objektorientierte Programmiermethoden eingesetzt, die auch in Zukunft eine Erweiterung der Query Engine wesentlich vereinfachen werden. In mehreren Schritten wird nach und nach der „alte" Query Optimizer durch die SQE abgelöst.

Im Folgenden sollen zuerst einmal die einzelnen Elemente von SQE im Detail erläutert werden. Diese Elemente sind der Query Dispatcher, der Query Optimizer, die Data Access Primitives, der Statistik Manager sowie der Plan Cache.

Der Query Dispatcher

Um einen möglichst reibungslosen Übergang zu SQE zu gewährleisten, wird die SQE schrittweise zur Verfügung gestellt. Im Lauf der Zeit werden immer mehr SQL Statements von SQE abgearbeitet werden. Es gibt also momentan zwei Query Engines: SQE und CQE.

Durch den Query Dispatcher ist das Vorhandensein der SQE für den Programmierer und für den Endanwender transparent. Der Query Dispatcher analysiert jedes SQL Statement und entscheidet anhand von bestimmten Kriterien, ob das Statement von CQE oder von SQE abgearbeitet werden soll. Weder der Endanwender noch der Programmierer müssen sich darum kümmern, dass es zwei Query Engines gibt – und sie können das Verhalten des Query Dispatchers auch nicht beeinflussen.

The Query Dispatcher

- V5R2 – SQE Statements:
  - Read only SQL
  - Single Table (keine Joins)
  - Keine Verwendung von OR oder IN
  - Keine Verwendung von SMP
- Zusätzliche Unterstützung mit V5R2 PTF SI07650:
  - Joins
  - Verwendung von OR oder IN
  - Verwendung von SMP
- Zusätzliche Unterstützung mit i5/OS:
  - Views
  - Unions
  - Subqueries
  - INSERT, UPDATE, DELETE
  - STAR_JOIN

Der Query Dispatcher

Der Query Optimizer

Der neue Query Optimizer muss im Gegensatz zum CQE Optimizer weniger Arbeit verrichten. Die primäre Aufgabe des Optimizers ist es, eine möglichst gute Implementierung eines SQL Statements zu finden, also aus den möglichen Strategien und Algorithmen (wie zum Beispiel Verwendung eines Indexes oder sequentielles Lesen der gesamten Tabelle) diejenigen auszuwählen, die eine möglichst schnelle Ausführung des Statements gewährleisten. Der CQE Optimizer hat zusätzlich die Aufgabe, statistische Informationen aus Tabellen und Indizes abzuleiten und Informationen über die Arbeitsumgebung zu sammeln (wie zum Beispiel Größe des Hauptspeichers oder Zahl der CPUs). Diese zusätzlichen Aufgaben sind einer der Gründe, warum die Code-Basis von CQE im Lauf der Zeit immer schwieriger zu warten und zu verbessern war. IBM hat das Design des SQE Optimizers wesentlich vereinfacht, indem diese Statistikaufgaben und die „Beobachtung" der Query-Umgebung in eine eigene Komponente – den Statistik Manager – ausgelagert wurde und der Optimizer nur noch für seine eigentliche Aufgabe – die Optimierung von Queries – zuständig ist.

Die meisten Änderungen in den Strategien und Algorithmen des SQE Optimizers wurden auf einer Ebene vorgenommen, die für den Anwender oder Programmierer kaum nachvollziehbar sind. Es gibt jedoch einige grundlegende Änderungen, die auch für den „normalen" Anwender sichtbar sind. Nahezu jeder iSeries-Anwender kennt die Situation, dass eine Query unerwartet lange läuft, weil zur Ausführung der Query ein temporärer Index aufgebaut wird. Dieses Phänomen ist mit der Einführung des SQE Optimizers

verschwunden. Die neue Query Engine benötigt für keine ihrer Implementierungen einen temporären Index, weil eine Reihe alternativer Implementierungen zur Verfügung stehen. Außerdem werden zukünftig wesentlich häufiger als bisher Table Scans (sequentielles Lesen der Tabelle) verwendet, weil die Ausführung von Table Scans in der SQE Query Engine

Data Access Primitives

Während der Optimizer quasi das Rezept liefert, um eine Query auszuführen, sind die Data Access Primitives sozusagen der Koch, der die eigentliche Ausführung der Query übernimmt. Auch in diesem Bereich wurden erhebliche Änderungen vorgenommen.

Wie schon erwähnt, wurde die Ausführung von Table Scans verbessert, indem der Algorithmus zum einen an das Single Level Storage-Konzept der iSeries angepasst wurde und zum anderen verfügbarer Hauptspeicher und Plattenressourcen aggressiver genutzt werden als bisher. Ein weiterer Bereich, in dem spezielle Anpassungen vorgenommen wurden, sind Queries, die das Schlüsselwort DISTINCT verwenden. Tests im Labor haben gezeigt, dass Queries mit dem Schlüsselwort DISTINCT bei Verwendung der neuen Data Access Primitives bis zu acht mal schneller laufen als vorher.

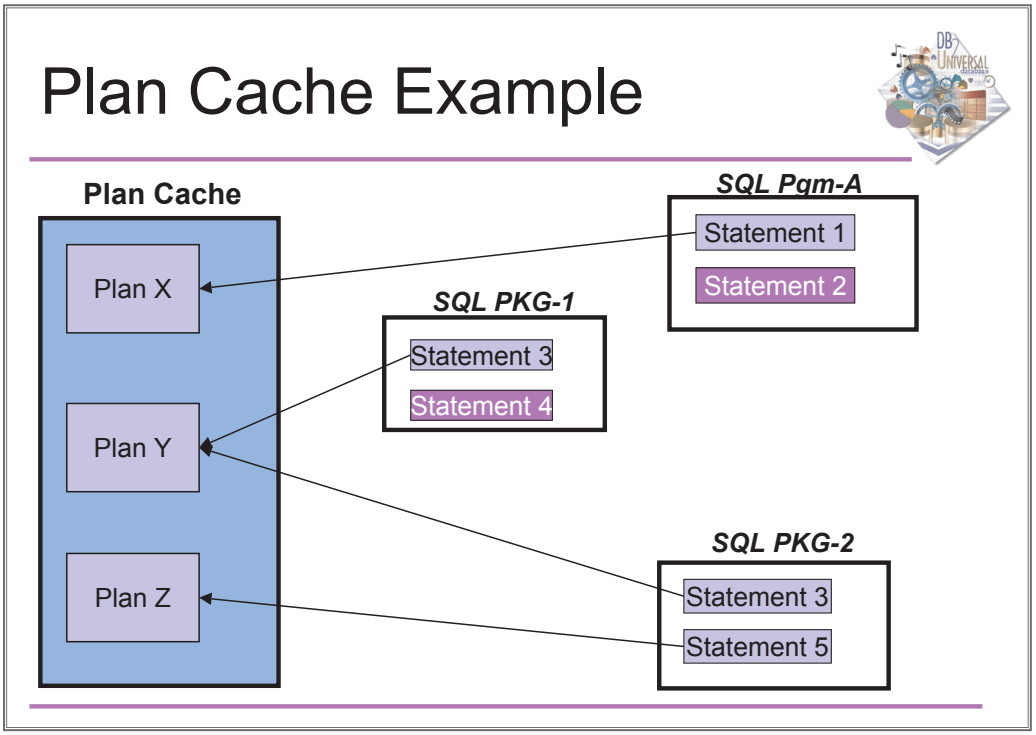
Auch im Bereich von temporären Objekten wurde eine wesentliche Verbesserung erzielt. Komplexe Queries greifen oft auf Daten aus mehreren Tabellen zu – und das kann es notwendig machen, temporäre Result Sets zu erstellen, die im Lauf der Query-Ausführung Zwischenergebnisse speichern (wenn zum Beispiel das Ergebnis einer Query nach Feldern aus unterschiedlichen Tabellen sortiert werden soll). Diese temporären Result Sets wurden von DB2 UDB auf dem gleichen Wege erzeugt, wie dies ein Benutzerprogramm tun würde. Mit den SQE Data Access Primitives wurde hier die Performance durch zwei wesentliche Änderungen erheblich verbessert.

Zum einen verwendet SQE ein internes Objekt, um Zwischenergebnisse zu speichern – statt wie bisher, ein echtes OS/400 Datenbankobjekt zu erzeugen. Außerdem wird dieses temporäre Objekt durch Operationen unterhalb des TIMI (Technology Independant Machine Interface) gefüllt und auf diese Weise verhindert, dass Daten zwischen Komponenten oberhalb und unterhalb der TIMI-„Grenze“ hin- und hergeschoben werden.

Plan Cache

SQE verwendet einen sich selbst verwaltenden Plan Cache, um Zugriffspläne von häufig verwendeten SQL Statements mehrfach verwenden zu können. Durch die Wiederverwendung von Zugriffsplänen kann die erneute Optimierung einer Query eingespart werden – was insgesamt zu einer verbesserten Systemperformance führt.

Der Plan Cache kann von jedem Statement verwendet werden, dessen Optimierung und Ausführung von SQE übernommen wird – egal über welches Interface (interaktives SQL, imbedded SQL, dynamisches SQL, SQL Package) dieses Statement abgesetzt wurde. Das bedeutet auch, dass der Zugriffsplan anders gespeichert werden muss, als dies bisher der Fall war. Wie in folgender Abbildung dargestellt, wird der Zugriffsplan bei Verwendung von SQE in zwei Teile aufgespalten.



Plan Cache Beispiel

Im SQE Plan Cache wird der Optimierungsteil (das „Rezept“) gespeichert. Die Teile des Zugriffsplans, die den SQL Statement-Text und ähnliche Informationen enthalten, werden an der Stelle abgespeichert, an der sie auch bisher abgelegt wurden (also zum Beispiel im SQL Package) – zusammen mit einem virtuellen Link zum zugehörigen Plan im Plan Cache.

Wenn das gleiche SQL Statement immer wieder von verschiedenen Interfaces und Programmen ausgeführt wird, so können diese nun einen gemeinsamen Zugriffsplan verwenden, statt jeweils einen eigenen Plan zu erstellen. Außerdem können im Plan Cache für das gleiche SQL Statement mehrere Zugriffspläne abgelegt werden (zum Beispiel ein Plan für die Verwendung von SMP – Symmetric Multi Processing – und ein Plan ohne die Verwendung von SMP oder ein Plan, der generiert wurde, bevor Indizes zur Verfügung stehen, sowie ein Plan, der nach der Erstellung von Indizes generiert wurde). Durch die zentrale Speicherung der Zugriffspläne können auch hier wieder alle Interfaces sofort von einem neuen Plan profitieren – und müssen nicht erst selbst eine Neu-Optimierung durchführen.

Der SQE Plan Cache verwaltet sich selbst. Es ist also nicht notwendig, dass sich ein Administrator um den Plan Cache kümmert. DB2 reserviert automatisch Platz für den Plan Cache und überwacht auch die Verwendung dieses reservierten Platzes. Bevor der Plan Cache überläuft, löscht DB2 UDB automatisch alte Pläne aus dem Plan Cache. Entscheidungskriterium ist dabei, wie häufig ein bestimmter Plan verwendet wurde und wann der Plan das letzte Mal verwendet wurde. Außerdem wird der Plan Cache bei einem IPL gelöscht. Dies kann zu einer kurzfristigen Verschlechterung der Performance von Queries führen, da bei der erstmaligen Ausführung eines bestimmten SQL Statements eine erneute Optimierung erfolgen muss. Momentan stehen keine Tools zur Verfügung, mit deren Hilfe ein Anwender den Inhalt des Plan Caches ansehen oder verändern kann.

Der SQE Plan Cache ist außerdem die Grundlage für einen zukünftigen selbstlernenden Query Optimizer. Ein selbstlernender Query Optimizer wäre in der Lage, die Informationen im Plan Cache (erwartete Ausführungszeit, erwartete Anzahl von Datensätzen) mit den tatsächlichen Ergebnissen einer Query zu vergleichen. Wenn die tatsächliche Ausführungszeit länger ist als erwartet, dann könnte der Optimizer dies in einer zukünftigen Implementierung feststellen und automatisch versuchen, eine bessere Strategie zur Ausführung des betreffenden Statements zu finden.

Statistik-Manager

Der Statistik-Manager ist zuständig für das Sammeln und Speichern von Statistikdaten. Ja, tatsächlich – DB2 UDB für iSeries verwendet jetzt Statistikdaten, die NICHT auf Indizes beruhen. Bevor Sie jetzt jedoch unruhig werden und überlegen, woher Sie einen Datenbank-Administrator bekommen, der sich um das Anlegen und Pflegen der Statistiken kümmert, machen Sie sich keine Sorgen: Das Sammeln und Pflegen der Statistikdaten läuft auf der iSeries standardmäßig vollkommen automatisch.

Zu den Aufgaben des Statistik Managers gehört es auch, Fragen des Optimizers zu beantworten. Das können Fragen sein wie etwa: „Wie viele Datensätze einer Tabelle erfüllen ein bestimmtes Selektionskriterium?“. Der Statistik-Manager wird versuchen, diese Frage möglichst genau zu beantworten. Dazu kann er zum einen auf vorhandene Statistikdaten (die zusammen mit der dazugehörenden Tabelle gespeichert werden) zugreifen oder er kann die Antwort aus vorhandenen Indizes ableiten. Wenn beides nicht möglich ist, dann gibt der Statistik-Manager Default-Werte als Antwort zurück. Diese Default-Werte können durchaus mit der realen Verteilung der Daten in einer Tabelle übereinstimmen, sie können aber auch erheblich von der Realität abweichen.

Statistikdaten geben immer Auskunft über die Datenverteilung innerhalb einer einzelnen Spalte, wobei für eine Tabelle mehrere dieser so genannten „Column Statistics“ existieren können. Sie enthalten im Einzelnen folgende Werte:

- Column Cardinality: Die Anzahl der verschiedenen Werte in einer Spalte.
- Histogramm: Hier wird die Verteilung der Werte dargestellt. Die unterschiedlichen Werte einer Spalte werden in einzelne Gruppen unterteilt, wobei jede Gruppe in etwa die gleiche Anzahl Sätze enthalten soll.
- Frequent Value List: Dies ist eine Liste der 100 am häufigsten vorkommenden Werte einer Spalte.

Im Gegensatz zu anderen Datenbank-Implementierungen kann DB2 UDB für iSeries automatisch feststellen, welche Statistikdaten benötigt werden – sozusagen „Statistik on demand". Nachdem eine Optimierung einer Query durchgeführt wurde, für die die Fragen des Optimizers nur mit Hilfe von Default-Werten beantwortet werden können, wird eine Anfrage zur Erstellung von Statistikdaten an einen Hintergrundjob (QDBFSTCCOL) geleitet. Dieser Job sammelt die entsprechenden Anfragen und führt sie mit einer sehr niedrigen Priorität aus, um „normalen" Workload auf dem System nicht zu beeinflussen. Wenn die Sammlung der Statistikdaten abgeschlossen ist, werden diese Daten in der zugehörigen Tabelle gespeichert und stehen für nachfolgende Query-Optimierungen zur Verfügung. Die automatische Sammlung von Statistikdaten wird über den Systemwert QDBFSTCCOL gesteuert, der folgende Werte haben kann:

- *ALL: Sowohl Benutzer als auch das System können die Erstellung von Statistikdaten veranlassen (dies ist der Default-Wert).
- *NONE: Es können keine Statistikdaten im Hintergrund gesammelt werden. Das heißt nicht, dass das System Anfragen zur Erstellung von Statistikdaten nicht sammelt – lediglich der Hintergrundjob wird nicht aktiv. Wenn der Systemwert wieder auf *ALL oder *SYSTEM geändert wird, dann werden alle aufgelaufenen Anfragen abgearbeitet.
- *USER: Nur Anwender können Statistikdaten im Hintergrund sammeln lassen.
- *SYSTEM: Nur das System kann Statistikdaten im Hintergrund sammeln lassen.

Über den iSeries Navigator steht ein grafisches Interface zur Verfügung, um gesammelte Statistikdaten anzusehen und diese auch zu verwalten (Löschen von bestehenden Statistiken, Anfordern von neuen Statistiken).

Statistikdaten können so wie Indizes verwendet werden, um Fragen des Optimizers zu beantworten. Es gibt jedoch einige grundlegende Unterschiede, was den Speicherbedarf und die Pflege der Daten angeht. Statistikdaten für eine einzelne Spalte belegen etwa 12 K Speicherplatz, egal wie groß die zugehörige Tabelle ist. Außerdem werden Statistikdaten im Gegensatz zu Indizes nicht sofort aktualisiert, wenn sich die zugehörige Tabelle ändert, so dass durch die Statistikdaten ein wesentlich geringerer System-Overhead erzeugt wird.

Natürlich können durch erhebliche Änderungen in der Tabelle auch Statistikdaten veralten. Dies wird durch den Statistik-Manager immer dann überprüft, wenn er diese Daten zur Beantwortung von Fragen des Optimizers verwendet. Wenn sich die zugehörige Tabelle um mehr als 15 Prozent geändert hat (Zahl der Datensätze ist um 15 Prozent gewachsen oder geschrumpft, mehr als 15 Prozent der Datensätze wurden geändert), dann werden diese Statistikdaten als veraltet betrachtet. Wenn dies der Fall ist, gibt der Statistik Manager wieder eine entsprechende Anfrage an den QDBFSTCCOL-Job, um die entsprechenden Statistikdaten zu aktualisieren. Ganz wichtig ist auch, dass Statistiken lediglich während der Optimierungsphase verwendet werden können. Statistikdaten können NICHT zur Ausführung von Queries verwendet werden. Indizes werden also nicht überflüssig – lediglich Indizes, die bisher nur erstellt wurden, um Statistiken zur Verfügung zu stellen und die NICHT für die Ausführung von Queries verwendet wurden, können gelöscht werden.

Was bedeuten diese Änderungen nun für Ihre reale Systemumgebung? SQE ist zur Erzeugung von optimalen Zugriffsplänen stärker vom Vorhandensein von Statistikdaten abhängig als dies bei CQE der Fall war. Das kann bedeuten, dass es nach einem Release-Wechsel bei der ersten Ausführung eines SQL Statements zu schlechteren Antwortzeiten kommen kann (weil notwendige Statistikdaten noch nicht vorliegen). Da der Statistik-Manager aber feststellt, dass die entsprechenden Statistiken nicht vorliegen, werden diese automatisch gesammelt und bei der nächsten Ausführung des SQL Statements herangezogen. Ab diesem Zeitpunkt sollte die Antwortzeit mindestens so gut sein wie dies unter CQE der Fall war.

Wie auch bisher ist eine gute Indexstrategie die Grundlage für eine gute Query-Performance. Weitere Informationen zur Erstellung von optimalen Indizes finden Sie im Internet unter: http://www.iseries.ibm.com/developer/bi/documents/strategy/strategy.html. Dort befindet sich das Indexing Strategies Whitepaper.

Zusätzlich zu diesen „Architekturänderungen“ gibt es mit i5/OS eine Reihe von Erweiterungen im SQL-Sprachumfang. So werden zum Beispiel die Unicode Zeichensätze UTF-8 und UTF-16 unterstützt. Über die ENCRYPT und DECRYPT SQL-Funktion besteht die Möglichkeit, einzelne Spalten einer Tabelle verschlüsselt abzuspeichern und so zu schützen. Portierungen aus dem Oracle-Umfeld werden durch die Unterstützung von Sequence-Objekten wesentlich vereinfacht. Damit steht neben den Identity Columns eine weitere Möglichkeit zur Verfügung, automatisch Werte für eine Spalte generieren zu lassen.

1.8
Seite 8

IBM i5/OS, die künftige Generation

IBM eServer i5/OS V5R3 bezeichnet das integrierte Betriebssystem, welches auf den bewährten Funktionen von OS/400 V5R2 aufsetzt und diese erweitert.

Autor:
Heinz J. Kopp,
iSeries Software Produktmanagement

Die iSeries Linux/Windows Integration

Die Integration von LINUX und Windows wurde dahingehend verbessert, dass sowohl Linux als auch der integrierte xSeries Server-Adapter jetzt größere virtuelle Plattenbereiche unterstützen. Einem Linux-Server können 64 TB, einem Windows-Server unter Verwendung von Microsoft Clustering Service 48 TB als maximale Plattenkapazität zugeordnet werden. Über die Windows-Administrationsfunktion innerhalb des iSeries Navigator kann Linux gestartet und beendet werden; diese Funktion wird ebenfalls zum Erstellen, Löschen und Verbinden virtueller Plattenlaufwerke für Linux-Server eingesetzt. Benutzer, welche über i5/OS in der Windows-Umgebung registriert wurden, können ihre Windows-Passwörter innerhalb von Windows verwalten. iSeries Access für Linux stellt einen neuen Client innerhalb der iSeries Access Family dar, arbeitet native unter Linux auf einem Intel-, einem Power-basierten PC oder in einer logischen Partition eines iSeries Servers und enthält eine 5250-Emulation – vergleichbar mit dem native Windows 5250-Emulator – sowie einen ODBC Driver.

Performance Management für eServer iSeries (PM iSeries)

Das Performance-Management ist in i5/OS integriert, sammelt automatisch Daten über mehrere Partitionen sowie über unterschiedliche Betriebssysteme und übergibt diese dem eServer Workload Estimator für die Analyse und Ermittlung des erforderlichen Systembedarfs. Während bisher PM iSeries auf Partitionsebene lediglich das Betriebssystem OS/400 berücksichtigte und für eine „Cross Partition View“ in allen Partitionen aktiviert werden musste und zudem die Resultate eine manuelle Aufbereitung erforderten, wird mit V5R3 unter Verwendung der IBM Director Multiplatform Server Option der Virtualization Engine der Einsatz in einer dynamischen Umgebung mit den Betriebssystemen i5/OS, Linux und AIX 5L („multi-OS“, „multipartition“) unterstützt.

Remote Control Panel

Ein Remote Control Panel kann direkt an einer „secondary“ Partition angeschlossen werden. Bei Ausfall z.B. infolge eines temporären Netzproblems wird die Operations Console ohne Eingriff eines Administrators wieder aktiviert.

Grundwerk 10/2004

Um auf eServer i5 Servern Partitionen zu erstellen, zu definieren und zu ändern oder Capacity on Demand zu nutzen, ist eine Hardware Management Console (HMC) erforderlich, welche auch als i5/OS System Console verwendet werden kann. Diese Hardware Management Console erspart die Implementierung einer dedizierten primären Partition, wie es für den Power4 OS/400 Server erforderlich war.

In einer „on Demand" Systemumgebung sind Standards und Globalisierung wesentliche Voraussetzungen für Integration und Offenheit. DB2 UDB (Universal Database) bietet diese Flexibilität mit der Möglichkeit, auf die Datenbank über zahlreiche Client Interfaces zuzugreifen. Erweiterungen für den RPG SQL Pre-Compiler und Performanceverbesserungen helfen sowohl Kunden als auch unabhängigen Lösungsanbietern, diese Interfaces problemlos zu nutzen.

iSeries Web Technical Support

Der iSeries Web Technical Support informiert über eine e-Mail, wenn ein kumulatives oder ein Gruppen PTF verfügbar ist. Software-Probleme können mit Hilfe von benutzerdefinierten Bezeichnungen über eine webbasierte Anwendung gemeldet werden.

Das „Intelligent Communications Trace Analyzer Tool" wird vom Web heruntergeladen und diagnostiziert umfangreiche Kommunikations-, Performance- und Sicherheitsprobleme. WebSphere Application Server Express für iSeries ist im Betriebssystem enthalten und wird mit allen i5/OS-Bestellungen ausgeliefert.

Print Services Facility

Das Print Services Facility für OS/400 unterstützt neue IBM eServer iSeries-Drucker und implementiert Phase II von Unicode durch einen erweiterten Support von TruType, OpenType Fonts und dem Import von standardisierten Image- und Grafikobjekten aus dem Netz oder dem Web in eServer iSeries Dokumente.

Cross-Site Mirroring (XSM)

Das Cross-Site Mirroring (XSM) generiert eine gespiegelte Kopie eines „independent disk pool" auf einem System, welches geografisch getrennt vom Ursprungssystem installiert sein kann. Bei Problemen auf dem Ursprungssystem kann der Backup-Server als Owner dieser Kopie die Aufgabe des primären Servers übernehmen. „Geographic Mirroring" als Subfunktion von XSM ist Bestandteil der Option 41, HA Switchable Resources von OS/400.

Multipath Support

Der Multipath Support für IBM TotalStorage Enterprise Storage Server (ESS) erlaubt die Definition von bis zu acht Verbindungen zum Anschluss von I/O-Prozessoren einer iSeries an eine einzelne logische Unit eines IBM TotalStorage Enterprise Storage Servers, wobei jede Verbindung unabhängig arbeitet und für einen unterbrechungsfreien Zugriff bei Ausfall eines einzelnen Pfades sorgt.

Eine neue Option innerhalb der „save-while-active" Funktion gestattet, eine Sicherungsoperation zu beenden, ohne Transaktionsbegrenzungen zu berücksichtigen, sofern es sich um Anwendungen handelt, welche Commitment Control verwenden. In der Vergangenheit scheiterte eine Sicherung aller Objekte an den Transaktionsbegrenzungen daran, dass die Applikationen die Commitment-Grenzen innerhalb eines definierten Zeitraums nicht erreichten.

Erweiterungen des Integrated File System

Die Erweiterungen des „Integrated File Systems" betreffen Performanceverbesserungen für SAV- und RST-Operationen, „Scan Objects" Parameter für den Einsatz von Virenschutzprogrammen sowie die Verfügbarkeit neuer Parameter, um Objekt-Subsets einfacher aufzunehmen oder auszuschließen.

Elektronisches Proof of Entitlement

Ein elektronisches Proof of Entitlement (ePoE) wird das bekannte PoE in gedruckter Form ersetzen. Mit der Ankündigung von i5/OS wird der Programmservice für OS/400 V5R1 und ausgewählte Programme mit Wirkung vom 30. September 2005 zurückgezogen.

1.9
Seite 4

Virtualization Engine: Virtualisierungstechnologien für eServer-Plattformen

Mit 3 Jahren Entwicklungsarbeit in moderne Software-Technologien hat die IBM nun in Mainframe-Qualität die Verwaltung aller eServer-Systeme vereinfacht. Die IBM Virtualization Engine ist ein Bündel fortschrittlicher Software- und Mainframe-Technologien, die die IT-Infrastrukturen künftig wirtschaftlicher und deren Betrieb einfacher machen sollen. Die neue Virtualization Engine, die in diesem Jahr in IBM eServer- und Storage-Produkten verfügbar gemacht wurde, wird unter anderem das Management heterogener Umgebungen vereinfachen, die aus Servern von IBM und Systemen anderer Hersteller bestehen.

Die Virtualization Engine ist das Ergebnis aus insgesamt 40 Jahren Innovation auf dem Mainframe und dreijähriger Forschungs- und Entwicklungsarbeit, u.a. im Entwicklungszentrum in Böblingen. Sie ist als Lösung für eine Reihe von geschäftsrelevanten Problemen konzipiert. So soll sie unter anderem die Milliardenkosten reduzieren, die durch ungenutzte oder schlecht ausgelastete Computer- und Speichersysteme entstehen. Außerdem wird sie die Integration und gemeinsame Verwaltung aller Teile einer IT-Infrastruktur vereinfachen. In der Virtualization Engine kommen Mainframe-Technologien zum Einsatz, mit denen Kunden nun auch mit anderen Systemen effiziente, sichere und kostengünstige IT-Umgebungen aufbauen können.

„Die Virtualization Engine richtet sich an alle Kunden, die sich nicht mehr auf die einzelnen Systeme konzentrieren wollen, sondern denen der Nutzen der kompletten Betriebsumgebung wichtig ist," erläutert Bill Zeitler, Group Executive und Senior Vice President, IBM Systems and Technology Group. Außer Mainframe-Technologien kommen Basis-Tools für Server-Provisioning und Management von Tivoli sowie offene Grid-Funktionalitäten aus der WebSphere Runtime-Entwicklungsumgebung erstmals auf einer ganzen Reihe von IBM Systemen zum Einsatz.

Wichtige Bestandteile der Virtualization Engine sind fortschrittliche Technologien zur „Mikro-Partitionierung". Wie bisher nur auf dem Mainframe können Kunden nun auch auf UNIX und anderen Systemen bis zu 10 virtuelle Server pro Prozessor betreiben. Ein 4-Prozessor-System kann so in bis zu „40-Wege" partitioniert werden, bei gleichzeitigem Einsatz unterschiedlicher Betriebssysteme und Betriebssystem-Versionen. Teil der Virtualization Engine sind außerdem neue integrierte System-Services, darunter ein stark vereinfachter Ansatz für das Management heterogener Systeme. Dadurch wird die Anzahl von Systemen, die eine einzelne Person verwalten kann, merklich gesteigert. Eine integrierte Version der Tivoli Provisioning-Technologie erlaubt es den Kunden, zusätzliche Server bei Bedarf innerhalb weniger Minuten zu installieren, anstatt wie früher innerhalb von Tagen.

1.10
Seite 2

Die Features der Virtualization Engine im Einzelnen:

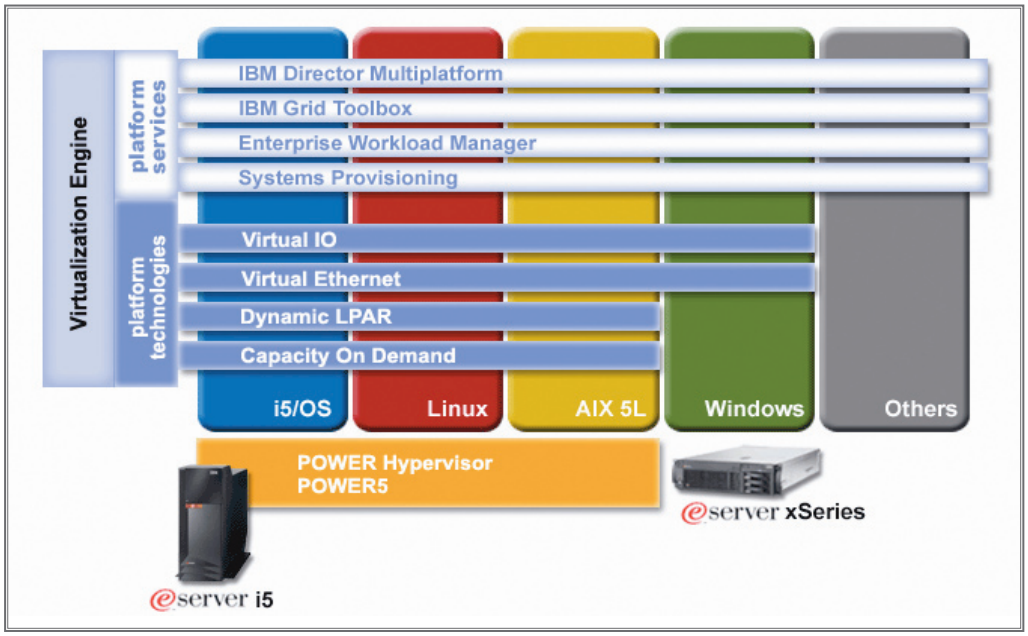
Virtualization Engine Technologien:

- Die hochentwickelte Technologie zur Mikro-Partitionierung für Systeme mit IBM Prozessoren basiert auf Mainframe-Technologie. Sie umfasst virtuelle Netzwerke, Hauptspeicher und LAN und ermöglicht es Kunden, jeden Prozessor in bis zu 10 voll funktionsfähige Server zu partitionieren.

Virtualization Engine Services:

- **IBM Director Multiplattform** ist eine zentrale Plattform für die Kontrolle und Verwaltung der IT-Umgebung; sie managt Systeme von IBM und anderen Herstellern sowie Cluster und Grids, die sich über mehrere Länder erstrecken können. Mit Hilfe der Plattform kann eine einzelne Person verschiedene Umgebungen von einer einzigen Konsole aus verwalten; kostenintensive Trainings für unterschiedliche Systeme entfallen.
- **Enterprise Workload Management** und Provisioning Tools optimieren die Ressourcen, vereinfachen die Verwaltung und erhöhen die Verfügbarkeit aller Systeme im Netzwerk.
- **Grid-Fähigkeit** für verteilte Systeme auf Basis der Open Grid Services Architektur und der WebSphere Technologie.

Mit der Virtualization Engine implementiert IBM erstmals Technologien auf anderen Systemen, die zu der hohen Auslastungsrate von etwa 80 Prozent bei Mainframe-Systemen beigetragen haben - im Vergleich zu nur circa 15 Prozent in UNIX- und Windows-Umgebungen. Damit senkt die Virtualization Engine die Betriebskosten erheblich.



Plattformübergreifende Systemsteuerung und Überwachung mit der Virtualization Engine

IBM Virtualization Engine-Plattformtechnologien

Auf der Basis dieser vollständig integrierten Produkte können iSeries-Server Ihrem Unternehmen dabei helfen, IT-Investitionen durch Rationalisierung der Verwaltung heterogener Partitionen oder Server zu maximieren. Die zum ersten Mal auf iSeries-Servern bereitgestellten IBM Virtualization Engine-Plattformtechnologien können dazu eingesetzt werden, um die unternehmensinterne IT-Infrastruktur ohne Unterbrechung der gewohnten Abläufe zu vereinfachen und dadurch Kräfte für geschäftliche Innovation und geschäftliches Wachstum freizusetzen.

- **IBM eServer On/Off Capacity on Demand**

  IBM bietet eine schnelle, unterbrechungsfreie Methode um zusätzliche, in iSeries-Servern integrierte Prozessorkapazität zu aktivieren. Diese flexiblen Optionen wurden durch die Angebote Reserve Capacity on Demand, Trial Capacity on Demand und Memory on Demand erweitert.

- **Dynamische logische Partitionierung (LPAR)**

  LPAR stellt ein optimiertes Gerüst für die Systemkonsolidierung auf iSeries-Servern bereit. Kunden nutzen LPAR zur Optimierung der IT-Investitionen durch eine rasche Konsolidierung mehrerer Einzelanwendungsserver auf einer kleineren Anzahl von iSeries-Servern. Mit neuen Funktionen wie den Partitionen mit unbegrenzter Kapazitätsnutzung (uncapped partition) kann ungenutzte Prozessorkapazität auf iSeries-Servern automatisch auf der Basis der geschäftlichen Prioritäten zur gemeinsamen Nutzung eingesetzt werden.

- **Simultanes Multithreading (SMT)**

  Durch gleichzeitige Ausführung mehrerer Threads auf demselben Prozessor aktiviert SMT die gleichzeitige Ausführung mehrerer Instruktionsströme, was zu einer Optimierung der Prozessornutzung führt. Durch POWER5-Prozessoren auf iSeries-Servern wird SMT vollständig automatisiert, ohne dass Anwendungsänderungen oder eine Anwendungsoptimierung erforderlich wäre.

- **Unterstützung für virtuelles Ethernet**

  Das iSeries-System stellt Unterstützung für das virtuelle Ethernet-LAN bereit, um Hochgeschwindigkeitsverbindungen zwischen logischen Partitionen und Microsoft Windows-basierten Servern zu ermöglichen. Das auf POWER5 basierende iSeries-System bietet 1GB an Kommunikationspfaden für das virtuelle Ethernet zwischen mehreren Betriebssystemen wie i5/OS, Linux und AIX 5L an. Segmente des virtuellen Ethernet können dynamisch erstellt und gelöscht werden. Der Zugriff auf virtuelle LAN-Segmente kann aufgrund der Anforderungen an Sicherheit oder Trennung des Datenverkehrs eingeschränkt werden.

- **Virtuelle Ein-/Ausgabe**

  Die virtuelle Ein-/Ausgabe ermöglicht die gemeinsame Nutzung physischer Ressourcen wie z. B. Speicher, Adapter und Einheiten zwischen Partitionen. Mehrere Partitionen können die physischen Ein-/Ausgaberessourcen gemeinsam nutzen, und jede Partition kann gleichzeitig sowohl virtuelle als auch physische Ein-/Ausgabeeinheiten einsetzen. Die virtuelle Ein-/Ausgabe ermöglicht Administratoren, neue Partitionen zu erstellen, ohne dem System physische E/A-Adapter hinzufügen zu müssen.

Ausblick: IBM Virtualization Engine-Plattformservices

Mithilfe der Unterstützung durch POWER Hypervisor ergänzen IBM Virtualization Engine-Plattformservices die IBM Virtualization Engine-Plattformtechnologien und tragen zur Vereinfachung der IT-Infrastruktur bei.

- **IBM Director Multiplatform**

  Selbst ein komplexes, leistungsstarkes System benötigt einfache Verwaltungstools, um effektiv zu sein. Aus diesem Grund stellt IBM Director Multiplatform eine allgemeine webbasierte Konsole für die Systemüberwachung Ihrer on demand Betriebsumgebung bereit, die es Ihrem Unternehmen ermöglicht, heterogene Systeme und Ressourcen intuitiv zu verwalten.

- **IBM Grid Toolbox V3 for Multiplatforms**

  Durch die Verbindung von Systemen und Ressourcen über eine einzige Verwaltungsschnittstelle kann IBM Grid Toolbox V3 for Multiplatforms Unternehmen dabei unterstützen, sichere und robuste Infrastrukturen unter Verwendung offener Standards zu erstellen.

- **Systemeinrichtung**

  Die Systemeinrichtungsfunktionen auf iSeries-Servern, konzipiert, um die Ressourcennutzung zu maximieren sowie Rechenzentren zu rationalisieren, unterstützen die Implementierung virtualisierter IT-Ressourcen sowie ihre Nutzung für unterschiedliche Zielsetzungen, die von den sich ständig ändernden Geschäftsanforderungen vorgegeben werden.

- **IBM Enterprise Workload Manager (EWLM)**

  EWLM dient zur Überwachung von Datenverarbeitungsressourcen, Antwortzeiten und Transaktionsfluss, damit Unternehmen das Service-Level-Management und die Kapazitätsplanung verbessern können. Außerdem lassen sich mithilfe der Selbstverwaltungsfunktionalität Probleme erkennen und das Workload-Management für heterogene, vielschichtige Workloads optimieren.

xSeries-Lösungen für die Windows-Integration auf iSeries

Der integrated xSeries Server für iSeries ist ein Windows Server auf einer Karte in einem iSeries-Server. Mit dem Integrated xSeries Adapter kann ein n-Wege xSeries-Server direkt mit einem iSeries-Server verbunden werden. Das System wird dadurch leistungsfähiger und flexibler. Beide Produkte nutzen die Speicher- und Managementressourcen der iSeries und integrieren iSeries- und xSeries-Operationen für weniger Kosten und Komplexität

Innovative Integration

Viele Unternehmen betreiben heute heterogene Umgebungen mit Intelbasierten Servern, auf denen Windows-Anwendungen ausgeführt werden, und iSeries- Servern mit Basisanwendungen. Dieser Server-Mix kann erforderlich sein, um die Anwendungen zu implementieren, die das Unternehmen benötigt. Es kann daraus allerdings eine problematische Umgebung entstehen, in der viele einzelne Intel-Server unterstützt werden müssen, was die Kosten in die Höhe treibt und das Management erschwert. Die iSeries bietet zwei Lösungen, die eine leistungsstarke, flexible und extrem rentable Alternative zu intelbasierten Serverfarmen darstellen – Integrated xSeries Server und Integrated xSeries Adapter.

Beide Produkte liefern nahtlos integrierte, einfach zu verwaltende Lösungen für die Windows Server-Implementierung und stellen eine kostengünstige und effiziente Alternative zur Ausführung von Windows auf mehreren Standalone-Servern auf Intel-Basis dar. Der Unterschied zwischen den beiden Lösungen ist die Art und Weise, wie die iSeries/x-Series-Hardwareintegration erreicht wird. Die einzelnen Lösungen wurden für verschiedene Anforderungen konzipiert.

Windows besser im Griff

Der Integrated xSeries Server ist ein Intel-basierter Server auf einer PCI Karte (Peripheral Component Interconnect), die in einem iSeries-Server installiert werden kann. Es steht ein leistungsstarker Intel Xeon-Prozessor mit 1,6 GHz und Hyperthreading-Technologie zur Verfügung. Ein einziger physischer Prozessor wird damit zu zwei logischen Prozessoren. Darüber hinaus ist ein integrierter Ethernet-Adapter mit 10/100 Mb/s enthalten. Der Integrated xSeries Server nutzt die Systemmanagement-, Kommunikations- und Speicherressourcen der iSeries, funktioniert jedoch ansonsten wie ein Windows-Standalone-Server. In einem einzigen iSeries-Server können insgesamt 48 einzelne Integrated xSeries Server installiert werden.

Vereinfachte Verwaltung

Dadurch entsteht auf einer einzigen, einfach zu verwaltenden Plattform eine ganze Serverfarm – die von der Zuverlässigkeit, Flexibilität und dem einfachen Management der iSeries profitiert. Der Integrated xSeries Adapter ist

Grundwerk 10/2004

eine PCI-basierte Schnittstellenkarte, die in ausgewählten Modellen der xSeries installiert wird und einen High-Speed-Link von einem Gigabyte pro Sekunde (GB/s) zu einem iSeries- Server bietet. Das erlaubt zentralisierte Speicherprozesse, integrierte Operationen und zentrales Systemmanagement. Über Integrated xSeries Adapter können insgesamt 60 n-Wege-xSeries-Server direkt an einen einzigen iSeries-Server angeschlossen werden. Dadurch können eine höhere Intel-Prozessorleistung implementiert und die zukunftsweisende IBM X-Architecture bestmöglich genutzt werden. Jeder xSeries-Server verfügt über eigene Prozessoren, eigenen Speicher und eigene Integrated xSeries Adapter und nutzt, wie auch der Integrated xSeries Server, die Plattenspeicher-, Bandspeicher-, DVD- und Systemmanagementressourcen der iSeries.

Beide Lösungen sind für die lokale Konsolidierung von Intel-Servern und Speicher sowie das zentrale Management von iSeries- und xSeries-Servern in verteilten Umgebungen konzipiert. Mit diesen Lösungen erreichen Sie die Flexibilität, die Sie benötigen, um eine konsolidierte Infrastruktur zu konfigurieren, die Ihren speziellen Anforderungen am besten gerecht wird. Wenn Sie viele kleinere Windows-Server möglichst effizient implementieren wollen, verwenden Sie am besten einen einzigen iSeries-Server und installieren darin viele Integrated xSeries Server. Wenn Sie dagegen geschäftskritische Verarbeitungsleistung und Skalierbarkeit benötigen, könnten Sie einen leistungsstarken 2-Wege-, 4-Wege- oder 8-Wege-xSeries- Server über Integrated xSeries Adapter mit Ihrem iSeries-Server verbinden.

Was auch immer Sie planen, Sie können sich auf echte Integration verlassen, die Ihnen jeweils nur das Beste der iSeries- und xSeries-Server bietet, und sich nach Belieben den für Ihr Unternehmen am besten geeigneten Anwendungsmix zusammenstellen. Dabei betreiben Sie die gesamte Implementierung als eine einzige Infrastruktur, die entsprechend einfach aufgebaut ist.

Integrierte Operationen

Zentrales, grafisches Servermanagement:

iSeries Navigator bietet eine grafische Benutzerschnittstelle für das Management von iSeries- und Windows-Servern. Administratoren können Server einfach starten und stoppen, OS/400-Benutzer an einer Windows- Domäne anmelden und Speichermanagement-Tasks durchführen, z. B. einem Windows-Server neue virtuelle Festplatten hinzufügen. Mit iSeries Navigator for Wireless können Administratoren den Serverstatus abfragen, Server starten und stoppen und Windows-Befehle für Integrated xSeries Server von einem webfähigen Mobiltelefon, einem Personal Digital Assistant (PDA) oder einem Web-Browser aus ausführen.

Synchronisierte Sicherheit:

Mit iSeries-Lösungen für die Windows-Integration können Benutzer-IDs und Kennwörter für OS/400 und Windows 2000 Server integriert und zentralisiert werden. Wird ein Benutzer der OS/400-Umgebung hinzugefügt, kann er automatisch auch einer Windows 2000 Server-Umgebung mit den entsprechenden Berechtigungen hinzugefügt werden. Wenn der Benutzer sein OS/400-Kennwort ändert, kann darüber hinaus das Windows- Kennwort automatisch synchronisiert werden. Durch diese Integration können Kosteneinsparungen bei der Benutzerverwaltung erzielt werden, da das Jonglieren mit verschiedenen Benutzer-IDs, Kennwörtern und Berechtigungen in den verschiedenen IT-Umgebungen ein Ende hat.

Konsolidierter Backup:

iSeries Integrated xSeries-Lösungen bieten konsolidierten Backup für OS/400 und Windows 2000 Server, mit dem Unternehmen ihre Hardware- und IT-Supportressourcen optimieren können. In einer typischen Windows- Serverfarm können Daten auf mehreren Servern verstreut sein – auf allen Servern werden Backupprozesse ausgeführt und für das Management sind mehrere Bandspeicherlaufwerke erforderlich. Beim Einsatz der iSeries/xSeries-Integration werden Speicher- und Backupressourcen der iSeries- und Windows-Server gemeinsam genutzt und konsolidiert. Eine Bandspeichereinheit kann für die gesamte Infrastruktur automatisierte Backups durchführen. So einfach ist das.

Zuverlässige, sichere und schnelle Datenübertragung:

Mit der iSeries können OS/400 und integrierte Windows-Server über High-Speed-Virtual-Ethernets Daten übertragen. Diese Netzwerke können für Windows-zu-Windows- und Windows-zu-OS/400-Übertragungen verwendet werden. Da es keine Kabel, Anschlüsse, Knoten oder Router gibt, ergeben sich auch weniger potenzielle Fehlerquellen. Der Datenaustausch auf dem Netzwerk fließt innerhalb der iSeries, nicht über externe Netzwerke. Mit Virtual Ethernets werden die Datenübertragung zuverlässiger und sicherer und die Netzwerkübertragung zwischen den Servern schneller (1 Gb/s). Darüber hinaus wird der Datenaustausch auf externen Netzwerken reduziert.

Außergewöhnliches Speichermanagement

Einer der wichtigsten Vorteile der iSeries ist die einmalige Speicherarchitektur – alle Festplatten können als einziger Pool von RAID-5- geschütztem Speicher (Redundant Array of Independent Disks) verwaltet werden. Das heißt, dass sich Benutzer und Systemadministratoren nicht um die exakte physische Adresse von Daten kümmern müssen. Alles wird von der iSeries dynamisch, transparent und einfach verwaltet. Die moderne Speicherarchitektur der iSeries ist darüber hinaus flexibler als herkömmliche Windows-

Serverimplementierungen. Darin werden jedem einzelnen Server üblicherweise dedizierte Festplattenlaufwerke zugeordnet und die einzelnen Serverkapazitäten getrennt verwaltet. Mit der integrierten iSeries-Lösung kann ein Pool von Plattenlaufwerken erstellt werden, auf den OS/400 und die Windows-Server zugreifen, und die iSeries kann die Daten automatisch auf alle Plattenlaufwerke des Systems verteilen. Diese Architektur kann zu einer enormen Performancesteigerung und einer effizienteren Nutzung von Hardwareressourcen beitragen.

Über den Integrated xSeries Server und den Integrated xSeries Adapter wird die außergewöhnliche SAN-verwandte Managementfunktionalität der iSeries für Windows transparent zur Verfügung gestellt. Windows wird Plattenspeicher zugeordnet, indem aus dem iSeries-Pool von Plattenspeicherressourcen ein Speicherobjekt (z. B. virtueller Plattenspeicherplatz) erstellt wird. Es können bis zu 32 Speicherbereiche erstellt und mit den einzelnen Integrated xSeries Servern oder xSeries-Servern, die über Integrated xSeries Adapter angeschlossen sind, verbunden werden. Die einzelnen Speicherbereiche können zwischen 1 MB und 64 GB umfassen, bis zu maximal 2 TB pro Windows-Server. Darüber hinaus können mehrere Speicherbereiche mit dem Windows-Dienstprogramm für die Plattenspeicherverwaltung für umfangreiche Speichervolumen zusammengeschlossen werden. Mit den iSeries-Funktionen für das Speichermanagement können dem Windows-Server Speicherbereiche dynamisch hinzugefügt werden.

Flexible und zuverlässige Serverimplementierung

Die iSeries/xSeries-Integration und die Speichervirtualisierung bieten Kunden innovative Möglichkeiten, mit denen sie ihre Windows-Serverumgebungen zuverlässiger gestalten und deren Wiederherstellung vereinfachen können. Falls ein Windows-Server ausfällt, können Sie die Speicherbereiche des Servers schnell und einfach auf einen anderen 'Hot-Spare'-xSeries-Server übertragen, ohne die iSeries erneut starten zu müssen. Es ist durchaus möglich, mit weniger Intel-Servern eine höhere Verfügbarkeit zu erzielen. Darüber hinaus wird die Flexibilität gesteigert, da zum Schutz mehrerer Produktionsserver ein 'Ersatzserver' verwendet werden kann. Zur Steigerung der Verfügbarkeit von Windows-Anwendungen durch automatische Failoverfunktionalität unterstützt die iSeries/xSeries-Integration Microsoft Cluster Services in Windows 2000 Advanced Server.

In einer typischen Windows-Serverimplementierung kann es mehrere Serverkonfigurationen geben – Produktionsserver, Entwicklungsserver und/oder Testserver jeweils mit den zugehörigen Software- und Einheitentreibern. Die Durchführung von Tests und die Implementierung von Änderungen auf mehreren Servern kann auf Grund der vielen verschiedenen Konfigurationen, die beibehalten werden müssen, problematisch sein. Ein Testserver muss möglicherweise für alle verschiedenen Konfigurationen zur Verfügung gestellt

werden, oder er muss komplett neu aufgebaut werden, um einem Produktionsserver zu entsprechen, falls Änderungen implementiert werden müssen. Auf Grund der iSeries/xSeries-Integration kann ein einziger xSeries-Server problemlos mehrere Test- und Implementierungsumgebungen unterstützen. Dieser Server kann jedem anderen Server 'zum Verwechseln ähnlich sehen', da die Server-Hardware vollkommen unabhängig von den Speicherbereichen sein kann, auf denen eine bestimmte Windows-Serverkonfiguration und Daten installiert sind. xSeries-Server haben kein physisches Boot-Laufwerk – alle Windows-'Laufwerke' sind nur Speicherobjekte im iSeries-Speicherpool. Die Speichervirtualisierung bietet enorme Flexibilität bei der Implementierung neuer Intel-Server, da für den Betrieb der einzelnen Windows-Server nur Daten eines Speicherbereichs erforderlich sind. Durch das Kopieren von Speicherbereichen können neue Server einfach implementiert und Änderungen einfach getestet werden. Im Falle von Softwarefehlern oder beschädigten Daten ist durch das Zurückschreiben von Speicherbereichen eine schnelle Wiederherstellung kompletter Windows-Server möglich.

Vorteile der iSeries/xSeries-Integration:

- **Bewährte Anwendungslösungen:**

  Der Integrated xSeries Server ist eine xSeries ServerProven-Plattform, auf der Softwareanbieter ihre Anwendungen testen und damit die Vorlaufzeit bis zur Produktion verkürzen können. Der Integrated xSeries Adapter ist ein Server-Proven-Hardwareadapter. Viele der ergänzenden Anwendungen für die xSeries entsprechen ebenfalls dem ServerProven-Standard.

- **Unterstützung für ergänzende Anwendungen:**

  Einige Softwareanbieter stellen Anwendungen zur Verfügung, die auf Plattformen mit mehreren Servern aufbauen. Bei diesen ergänzenden Anwendungen müssen OS/400- und Windows Server-Umgebungen die Gesamtlösung unterstützen. Mit dem Integrated xSeries Server und dem über den Integrated xSeries Adapter angeschlossenen xSeries-Server ist diese Anforderung schnell erfüllt, denn es wird eine integrierte IBM Lösung zur Verfügung gestellt.

- **Einfaches Management von Windows-Anwendungen:**

  Durch die Unterstützung für Citrix MetaFrame2 und Microsoft Windows Server (mit integrierter Terminal-Serverunterstützung) können Windows-Benutzeranwendungen auf den Desktop übertragen werden. Da Windows-Desktopanwendungen auf dem Server ausgeführt und nicht auf jedem einzelnen Desktop installiert werden, kann das Management extrem vereinfacht werden. Änderungen an der Anwendungssoftware müssen nur einmal auf Serverebene vorgenommen werden, danach werden sie automatisch und transparent auf jedem Desktop des Unternehmens implementiert.

Integrated xSeries Server

- Intel Xeon-Prozessor mit 1,6 GHz und Hyperthreading-Technologie
- 512 KB On-Chip-L2-Cache
- Bis zu 4 GB ECC DDR200-Speicher
- Bis zu 2 TB Plattenspeicher
- Integrierter Ethernet-Adapter mit 10/100 Mb/s sowie bis zu drei zusätzliche, Hot-Plug-fähige Netzwerkadapter, die sich aus Token-Ring-Adaptern mit 10/100/1000 Mb/s und 100/16/4 Mb/s zusammensetzen können
- 4 USB-Ports (Universal Serial Bus-Ports)
- Einheitentreiber zur gemeinsamen Nutzung von Plattenspeicher-, Bandspeicher-, DVD-und CD-ROM Ressourcen der iSeries
- Unterstützt auf den iSeries-Modellen 270, 800, 810, 820, 825, 830, 840, 870 und 890 mit OS/400 Version 5 Release 2 (V5R2) oder höher
- Unterstützung für Windows 2000 Server oder Windows 2000 Advanced Server
- Auf einem einzigen iSeries-Modell 270 können bis zu drei, auf Modell 800 bis zu vier, auf Modell 810 13, auf Modell 820 12, auf Modell 825 36, auf Modell 830 28, auf Modell 840 32 und auf den Modellen 870 und 890 bis zu 48 Integrated xSeries Server installiert werden.

Integrated xSeries Adapter

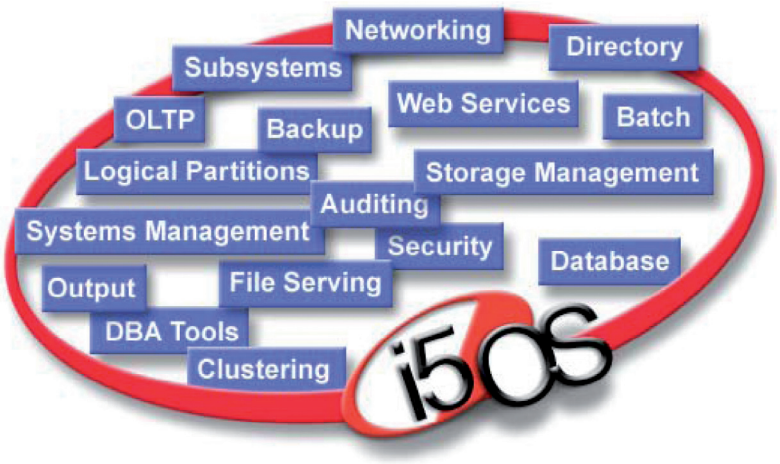
- 64-Bit-PCI-Adaper mit 66 MHz. Installiert in ausgewählten xSeries-Servern, u. a. den Modellen 235, 255, 360 und 440
- High-Speed-Link-Kabel für die Verbindung mit einem iSeries-Server
- Unterstützt auf den iSeries-Modellen 270, 820, 830 840 und 890 mit OS/400 Version 5 Release 1 (V5R1) und auf den Modellen 800, 810, 825, 870 und 890 mit OS/400 V5R2
- Unterstützung für Windows 2000 Server oder Windows 2000 Advanced Server
- Auf einem iSeries-Modell 270 können über Integrated xSeries Adapter bis zu zwei xSeries-Server direkt angeschlossen werden, auf Modell 800 drei, auf Modell 810 sieben, auf Modell 820 acht, auf Modell 825 18, auf Modell 830 16, auf Modell 840 32 und auf den Modellen 870 und 890 bis zu 60.

IBM Announcement Januar 2006

Aus iSeries wird System i5

Es hat schon etwas von einer Gewohnheit, auch wenn es bis zur offiziellen Verkündung durch IBM wie immer „strictly confidential“ gehalten wird, so erwarten die iSeries-Anhänger es schon beinahe zu Beginn eines jeden Jahres: das Announcement der Neuerungen, welche es im Umfeld der i5 sowohl auf Ebene der Hardware als auch im Bereich der mittlerweile verschiedenen Betriebssysteme auf dieser System-Familie gibt.

Mit dem diesjährigen Frühjahrs-Announcement, das gleichzeitig die wohl bedeutungsvollste Ankündigung der IBM im Bereich der i5 für das Jahr 2006 sein dürfte, verfolgt IBM nicht das Ziel der Ankündigung einer neuen Produktlinie, sondern vielmehr die Festigung und den Ausbau der i5 in der bisher bekannten Form.



Eines der Highlights des diesjährigen Announcements ist die Einführung einer weiteren, neuen Prozessor-Generation, welche die Serie der POWER5-Prozessoren ablöst, die in den bisherigen i5-Modellen zum Einsatz gekommen ist: POWER5+. Damit verfügt die i5, wie ihr Schwestermodell die p5, jetzt auch über die neueste Generation der Chips aus dem Hause IBM. Dabei bietet der neue Prozessor neben einer gesteigerten Leistung auch Optimierungen beim Einsatz der verschiedenen Betriebssysteme, die native auf der i5 betrieben werden können.

Das besondere an diesem neuen Chip ist die Integration von zwei Prozessoren auf einem einzelnen Chip, der außerdem ab einem bestimmten Modell über einen vergrößerten Cache verfügt. Das macht sich dann deutlich bei der Performance bemerkbar. Je nach Modell kommt diese neue Prozessor-Generation mit unterschiedlichen Taktraten von 1.9 bis 2.2 GHz zum Einsatz. Bereits das Einstiegsmodell in Form des Systems i5 520 verfügt dabei nun mit einer maximaler Leistung von bis zu 7100 CPW über durchaus beachtliche Leistungswerte. Zum Einsatz kommt in diesem Modell ein POWER5+-Prozessor mit einer 1.9 GHz Taktung. Traumhaft erscheinen schon beinahe die maximalen Leistungen des Highend-Modells in Form der 595, deren Endausbau-Stufe eine Prozessor-Leistung von bis zu 184.000 CPW erreicht.

Die bisherige Leistung eines POWER5-Prozessors konnte mit der neuen Prozessor-Generation von 3000 CPW pro Prozessor auf jetzt 3800 CPW auf POWER5+ gesteigert werden.

Neben der Anordnung mehrerer Prozessoren auf einem Chip spielt gerade auch die Erweiterung des L3 Cache eine wesentliche Rolle. Bis auf das Einstiegsmodell der 520 verfügen alle Maschinen über diesen L3 Cache, der sich besonders für Anwendungen im Java- und WebSphere-Umfeld spürbar bemerkbar macht. Mit der neuen Chip-Generation stehen nun bis zu 72 MByte Cache zur Verfügung (im Vergleich dazu waren es beim Vorgänger – dem POWER5 – maximal 32 MByte Cache). Ebenfalls erweitert wurde der Bereich des DDR2-Speichers, der von bisher maximal 266 MHz auf nun maximal 533 MHz erweitert wurde.

Abweichend zu den bisherigen Announcements „fehlen“ neue Modellbezeichnungen. Der Grund liegt darin, dass IBM keine Modell-Neueinführung mit dieser Ankündigung verbindet, sondern eine Art „Evolution“ oder Weiterentwicklung der bisherigen Modelle durchführt.

IBM stattet die bereits auf dem Markt etablierten Modelle 520, 550, 570 und 595 in der angekündigten Form mit der neuesten Technologie aus.

Wer jetzt im Glauben ist, IBM habe mit der unveränderten Fortführung der Modellbezeichnungen der Systeme i5 dieses Mal auf Neubenennungen verzichtet, der sei eines Besseren belehrt: Zukünftig wird dieser Bereich der IBM-Systeme nicht mehr die Bezeichnung „iSeries“ oder „i5“ tragen, sondern er wird im Zuge der Vereinheitlichung der Benennungen für die verschiedenen Server-Familien der IBM in „IBM System i5“ umgetauft (siehe Produktpalette IBM System i5-Modelle).

IBM System i5	520	550	570	595
Processor **Max CPW**	1, 1/2-way 1,9 POWER5+ 7100	1/4-way 1,9 POWER5+ 14.000	2/16-way 2,2 POWER5+ 58.500	8/64-way 1,9 POWER5+ 184.000
Key Changes	**Accelerator CoD** DDR2 Memory Optional IOP/ HSL	DDR2 Memory Optional IOP	**Simpler Structure** & Licensing DDR2 Memory Optional IOP	Expanded I/O Optional IOP
i5/OS Release	V5R3 & V5R4	V5R3 & V5R4	V5R3 & V5R4	V5R3 & V5R4
Value/Express	✔	Editions		
Standard	✔	✔	✔	✔
Enterprise	✔	✔	✔	✔
Solution	✔	✔		
Domino (Workplace)		✔		
HA	✔	✔	✔	✔
CBU			✔	✔

Produktpalette IBM System i5-Modelle

Auch weiterhin bietet IBM die neuen Maschinen in Form spezieller Pakete an – optimiert auf den jeweiligen Bedarf der Kunden. So bilden die Value- und Express-Versionen die Einstiegsbasis in die Welt der i5, deren weitere Ausbaustufen in Form der Enterprise Version und speziellen Lösungs-Bundles im Hochverfügbarkeits-Umfeld enden und auch für die Realisierung komplexester Anforderungen geeignet sind (siehe System i5 520 – Value und Express).

For Core Business Applications		Web-enabled, Java & Groupware and Core Business Applications
1200/60 P10	Accelerator	1-way POWER5+ 1,9 GHz with L3 Cache 3800/60 P10
600/30 P5		1-way POWER5+ 1,9 GHz 3100/30 P5

System i5 520 – Value und Express

Leider hat eine Besonderheit der i5 auch bei den neuen Modellen weiter Bestand: Das ist die interaktive Leistung, welche für die Ausführung klassischer 5250-Anwendungen nach wie vor benötigt wird.

Neu im Zusammenhang der Leistungsdifferenzierung innerhalb der einzelnen Modellgruppen ist die Aufteilung in Maschinen in „Web Enabled Systems“ und Maschinen, die für die Ausführung konventioneller Anwendungen zum Einsatz kommen. Hier unterscheiden sich die einzelnen Modelle nicht mehr nur – wie bisher – durch die interaktive Leistung, sondern auch in der maximalen Leistung im Batch-Bereich.

Die unterschiedlichen Batch-Leistungen stehen ausnahmslos bei allen Modellen zur Verfügung und ermöglichen so die Auswahl des optimalen Systems für die jeweiligen Anforderungen. Unter anderem können Kunden auch im Bereich der 520 in der Standard-Edition zukünftig zwischen verschiedenen Modellen wählen, die entweder mit einem oder mit zwei Prozessoren verfügbar sind.

Interessant ist in diesem Zusammenhang auch der so genannte „Accelerator“, mit dem die CoD- (Capacity on Demand-) Technik nun auch für Ein-Wege Maschinen zur Verfügung steht. Die Option, zusätzliche Leistung, die im Inneren der Maschine „schlummert“, bei Bedarf freizuschalten, war bisher nur den Mehrprozessor-Maschinen vorbehalten. Der Accelerator ist dabei mit CoD der Mehrprozessor-Maschinen durchaus vergleichbar. Dabei ist diese Leistungsfreischaltung auf den Batch-Bereich beschränkt und wirkt sich nicht auf die interaktive Performance der Maschine aus. Der Accelerator kann, wie CoD auf den Mehrprozessor-Maschinen auch, optional in Form eines zusätzlichen Features erworben werden und steht sowohl bei den Express-Versionen als auch bei den Value-Editionen zur Verfügung. Anders als bei CoD wird für die Aktivierung des Accelerators keine HMC benötigt, sondern die Aktivierung erfolgt einfach über die Eingabe eines Software-Schlüssels.

IBM setzt mit der Modellpolitik der i5-Systeme damit konsequent die Abdeckung eines breites Marktsegments fort. Dabei steht der Mittelstand klar im Fokus der Positionierung der Systeme. Alleine im Express-Bereich, in dem nur Ein-Wege-Prozessoren angeboten werden, können Kunden mittlerweile aus 7 verschiedene Versionen wählen. Im Gegensatz zu den bisherigen Ankündigungen, ist mit den neuen POWER5+-Prozessoren nicht die Notwendigkeit der Installation des neuesten i5 Releases verbunden. Als Mindest-Betriebssystemversion benötigen die neuen Prozessoren die Version V5R3 des i5/OS. Damit ist bei LPAR-Maschinen der Paralleleinsatz der beiden Release-Versionen V5R3 und V5R4 möglich. Das macht den Einsatz auch in solchen Situa-

tionen attraktiv, in denen lediglich die Hardware abgelöst werden soll, während die Version des Betriebssystems zunächst jedoch unverändert weiter betrieben werden kann. Damit entstehen nicht – wie in der Vergangenheit – regelrechte Stress-Situationen, weil Anwendungs-Software unter Umständen noch gar nicht für ein neues Betriebssystem-Release freigegeben war.

Ein weiteres Highlight dieses Frühjahr-Anouncements ist sicher die Ankündigung des neuen i5/OS Release V5R4. Zwar bietet IBM Erweiterungen der Funktionen in den verschiedenen Bereichen der i5 nicht nur in Form neuer Versionen oder Release-Varianten an, sondern das Unternehmen ist seit einigen Jahren dazu übergegangen, neue Funktionen sowohl für das Basis-Betriebssystem als auch für die verschiedenen Lizenzprogramme in Form von PTFs zur Verfügung zu stellen. Dennoch: Ein neues Release stellt auch heute noch eine Besonderheit dar. Mit im Fokus der neuen Modelle des i5-Systems steht die Unterstützung von Java-basierten Anwendungen. Diese Java-Anwendungen können nun – dank der Verfügbarkeit der POWER5+-Prozessoren – auch auf den Einstiegsmodellen der System i5 ausgeführt werden. Kunden können im Bereich des Einstiegsmodells i5 520 in der Standard-Edition zwischen zwei verschiedene Paketierungen wählen. Bei beiden kommt jeweils der POWER5+-Prozessor zum Einsatz.

Auch zukünftig wird IBM die Systeme mit unterschiedlichen Editionen anbieten. Neben der Value-, Express- und Standard-Edition können Kunden auch bei den neuen Modellen wieder die Enterprise-Edition wählen (siehe System i5 520 – Enterprise Edition).

For Core Business Applications		
1200 CPW with full 5250 P10	2800 CPW with full 5250 P10	3800/7100 CPW with full 5250 1/2-way POWER5+ 1,9 GHz with L3 Cache P20
		For e-business and Java Applications

System i5 520 – Enterprise Edition

Die „Königsklasse“ in Form der Enterprise-Edition bietet neben der Wachstumsmöglichkeit auch eine uneingeschränkte 5250-Leistung.

Neben den Einstiegsmodellen 520 richtet sich das Angebot im Bereich der Modellgruppe 550 an den gehobeneren Mittelstand bzw. an solche Unternehmen, deren Anforderungen an die Systemleistung über die Leistungsfähigkeit der Modelle 520 hinaus geht.

Die Modellgruppe 550 unterscheidet sich im Wesentlichen zu den Vorgängermodellen 550 durch die Leistungssteigerung auf Prozessor-Ebene, die in der Endausbau-Stufe bis zu 14.000 CPW bietet (siehe System i5 550).

For e-business and Java Applications	
1-way POWER5+ 1,9 GHz with L3 Cache 3800 CPW P10	1/2-way POWER5+ 1,9 GHz with L3 Cache 3800/7100 CPW P20

i5 520 – Standard Edition

3800/14.000 CPW 1/4-way POWER5+ 1,9 GHz with L3 Cache P20

System i5 550

	i5/OS License*	Enterprise Enablement**
570 2/4-way	1	1
4/8-way	1	1
8/16-way	1	1

* With Standard and Enterprise Edition
** With Enterprise Edition

System i5 570

26.700/184.000 CPW 8/16-way up to 64-way POWER5 1,9 GHz with L3 Cache P50/P60

System i5 595

Auch im diesem Bereich der 550er Modelle werden die bisherigen Editionen (Standard, Enterprise, Domino/Workplace, Solution) weiter angeboten und um eine High Availability- (HA-) Edition ergänzt.

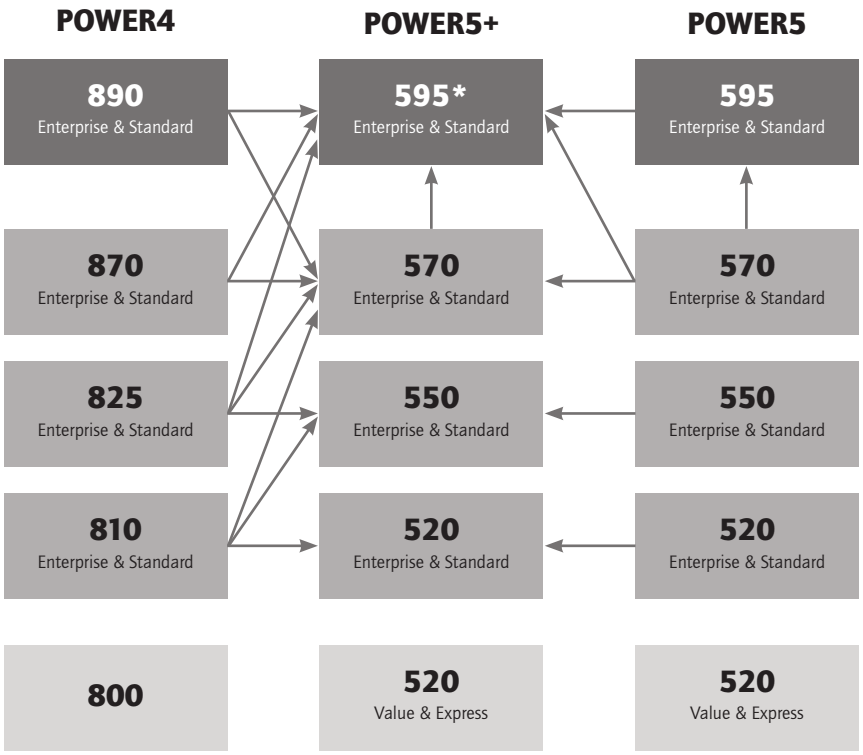
Einen weiteren Leistungssprung bieten die System i5-Modelle 570. Unter anderem lösen auch bei diesen Modellen die neuen POWER5+-Prozessoren den Vorgänger POWER5 ab. Im Gegensatz zu den übrigen Systemen i5 sind die Prozessoren, die in den Modellen 570 zum Einsatz kommen, mit 2,2GHz höher getaktet als die POWER5+-Prozessoren der 520 bzw. der 550, deren Taktrate mit 1.9 GHz niedriger ausgelegt ist. Nicht mehr angeboten wird in dem Segment der 570-Modelle die Ein- bzw. Zwei-Wege Maschine.

Die Entscheidung zu einem Modell 570 dürfte in Zukunft zumindest in Bezug auf die Software-Kosten leichter fallen, denn die Mindestmenge der erforderlichen i5/OS-Lizenzen in der Endausbaus-Stufe wurde endlich reduziert. Waren bisher bei mehr als vier Prozessoren auch vier i5/OS-Lizenzen erforderlich, so wurden mit diesem Announcement die Lizenzierung im i5/OS-Bereich dahingehend geändert, dass jetzt auch bei vier oder mehr Prozessoren nur eine i5/OS-Lizenz (und wie bisher auch das Enterprise-Enablement) erforderlich ist. Damit sinken in diesem Fall die Software-Kosten erheblich (siehe System i5 570)!

Eine Besonderheit stellt in dieser Frühjahrsankündigung die Modellreihe der 595 Systeme dar. Hier werden nach wie vor die POWER5- und nicht die neueren POWER5+-Prozessoren verwendet. Laut Aussagen der IBM sind die Leistungsreserven dieser Highend-Maschinen derart groß, dass in diesem Fall auf den Einsatz der noch leistungsfähigeren POWER5+-Prozessoren – zumindest derzeit noch – verzichtet wurde (siehe System i5 595).

Eine Neuerung im Bereich der Modelle 595 ist das so genannte „Optimum Care“, das als Bestandteil der Standard- und Enterprise-Editionen angeboten wird. Dabei handelt es sich um eine spezielle Betreuung im Projektmanagement während der Planungsphase und während der Installation dieser Maschinen. Diese Leistung wird direkt durch IBM, und nicht durch IBM-Partner erbracht. Damit ist „Optimum Care“ eine Erweiterung der bisherigen Voucher, die es auch für die übrigen System i5-Modelle gibt.

Unverändert sind die Upgrade-Möglichkeiten innerhalb der POWER4- und POWER5-Welt. Dabei ist auch zu beachten, dass auch mit den POWER5+-Prozessoren Upgrades zwischen den verschieden System i5-Modellgruppen nur beschränkt möglich sein werden (siehe Prozessor-Upgrades).



* The new 595 1,9 GHz is a POWER5 processor

Prozessor-Upgrades

Das Announcement wird nicht nur durch neue Hardware bestimmt, auch wenn weitere Details im Bereich der Hardware, hier sind zum Beispiel die Unterstützung von iSCSI und RAID6 zu nennen, in der Aufzählung der Hardware News fehlen. Gerade die Erweiterungen im neuen Release des i5/OS in Form des V5R4 haben es sprichwörtlich in sich.

Zum einen bietet dieses neue Release eine 32-bit basierte Java Virtual Machine (JVM), die speziell auch für die kleineren Maschinen von Bedeutung ist. Mit Einsatz dieser JVM, die im Gegensatz zu der 64 bit Version einen wesentlich geringeren Ressourcenbedarf in Bezug auf Speicher aufweist, können nun auch auf kleineren System i5-Modellen Java-Anwendungen mit einer wesentlich verbesserten Leistungscharakteristik ausgeführt werden. In Verbindung mit den L3 Cache-Prozessoren bietet diese JVM eine verbesserte Basis für den Einsatz moderner Technologien. Dabei können wir in Zukunft zwischen dem Einsatz der 32-bit JVM und der 64-bit Variante wählen, die beide auf den Maschinen verfügbar sind.

Mit dem „On demand Performance Center“ wird Administratoren ein neues Werkzeug innerhalb des iSeries Navigators zur Verwaltung und Überwachung der Datenbank angeboten. Mit Hilfe dieses optionalen Tools kann so zum Beispiel die Verwaltung von Indizes durchgeführt oder auch nicht genutzte Indizes ermittelt werden. Außerdem lassen sich in Zukunft mit zusätzlichen Einstellmöglichkeiten die Ausführung von bestimmten Diensten oder Funktionen, denken wir einmal an SQL und Querys, durch Limitierung der maximalen Ressourcezuordnung für diese Funktionen einschränken. Das kann unter Umständen durchaus sehr positive Auswirkungen auf die Gesamtleistung der Maschine zur Folge haben, wie sich der eine oder andere Leser, der vielleicht auch schon seine Erfahrungen mit der Ausführung von komplexen SQL- oder Query-Anweisungen gemacht hat, vorstellen kann.

Auch das Zusammenspiel zwischen Java und DB2/UDB wurde mit dem neuen Release – neben einigen weiteren Optimierungen an der Datenbank – weiter verbessert.

Unruhe hat die letztjährige Abkündigung der Twinax-Adapter ausgelöst. Doch damit verfolgt IBM lediglich weiter die Strategie der Ablösung der in die Jahre gekommenen 5250-Anwendungen. Sicher lassen sich diese auch über andere Kommunikationswege, wie TCP/IP problemlos ausführen. Doch die Rufe der Anwender nach „modernen“ Oberflächen werden immer lauter. Auch mit der aktuellen Ankündigung wird das Ziel der IBM, klassische 5250-Anwender sanft in die Welt der grafischen Oberflächen zu überführen, weiter verfolgt.

Mit der Version V5R4 des i5/OS bietet IBM einige Erweiterungen bzw. Veränderungen in diesem Bereich an. Die Rede ist von dem neuen „WebFacing Deployment Tool“, das aus der Zusammenführung der beiden Bereiche „WebFacing“ und „HATS“ entstanden ist. Damit stellt IBM in V5R4 ein Runtime Environment zur Verfügung, mit der WebFacing-Anwendungen und HATS-Lösungen zusammen eingesetzt werden können. Damit entfallen die in der Vergangenheit sehr lästigen Fehlersituationen, die immer dann aufgetreten sind, wenn eine WebFacing-Anwendung eine 5250-Funktion aufgerufen hat, welche nicht mit WebFacing-Werkzeugen bearbeitet worden ist. Mit dem in V5R4 neuen Runtime Environment wird eine solche 5250-Funktion automatisch mit HATS-Funktionen in HTML umgesetzt und kann zusammen mit WebFacing-Anwendungen eingesetzt werden.

Einen weiteren Anreiz, den Einsatz der grafischen Oberflächen in Form von WebFacing oder HATS in Erwägung zu ziehen, ist die ausnahmslose Ausführung dieser Anwendungen im Batch-Bereich der POWER5+-Maschinen. Bis zur Einführung der POWER5-Prozessoren wurde auf den älteren Maschinen – sowohl für die Ausführung von WebFacing als auch für HATS – interaktive Leistung benötigt. Seit POWER5 werden alle WebFacing- Funktionen im Batch-Bereich der Maschinen ausgeführt, während HATS weiterhin interaktive Leistung benötigt. Mit V5R4 und POWER5+ ist endlich auch die letzte Barriere in Form der Ausführung von HATS im Batch-Bereich der Maschinen gefallen. Das bedeutet unter Umständen spürbare Kosteneinsparungen bei Neuanschaffungen, wenn auf die interaktive Leistung vollständig verzichtet werden kann.

Beim Einsatz von WebFacing und HATS wird lediglich die Darstellung der Anwendungen verändert, während die Anwendungslogik gleich bleiben kann. Doch Stillstand bedeutet bekanntlich Rückschritt. Deshalb werden auch RPG-Fans ihre Freude an dem neuen Release V5R4 haben. IBM hat auch wieder in der aktuellen i5/OS-Variante einige wesentliche Erweiterungen im RPG-Bereich realisiert. Unter anderem kann jetzt Free-Format SQL in RPG-Code eingebunden werden. Damit bieten sich natürlich einige neue Optionen im Zusammenhang mit dem Einsatz von RPG an. Weiterhin wurde die Unterstützung der WebServices mit RPG optimiert. Dazu bietet die neue Version des WebSphere Development Studio Client spezielle Funktionen und Assistenten, die es erlauben, RPG-Anwendungen in Verbindung mit Web-Anwendungen einsetzen zu können. Der Hintergrund für die Bereitstellung dieser erweiterten Funktionen ist der Wunsch vieler Kunden, bestehende und bewährte RPG-Programme in eine SOA-Architektur einbinden und nutzen zu können.

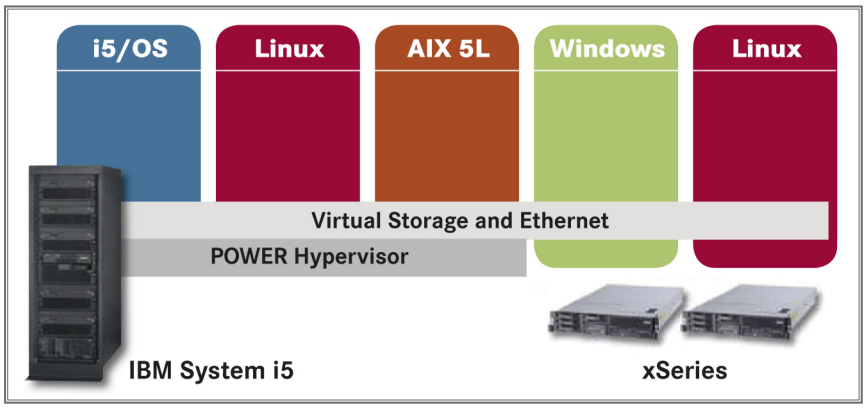
Damit sind wir aber noch nicht am Ende der Neuerungen im Bereich der „klassischen Anwendungsentwicklung“, denn auch das Umfeld der CL-Programmierung wurde mit dem i5/OS Release V5R4 bedacht. Neu ist in diesem Zusammenhang die Unterstützung von Subroutinen in CL-Code. Damit verbunden ist eine bessere Strukturierung des Quellcodes solcher CL-Programme. Ebenfalls neu ist die Unterstützung von Pointern, mit denen unter anderem der Direktaufruf von APIs aus einem CL-Programm heraus ermöglicht wird. Bisher konnten solche APIs nur über einen Zwischenschritt (zum Beispiel mit Hilfe eines aus CL aufgerufenen C-Programms) verwendet werden. Durch die Unterstützung der Pointer entfällt dieser lästige Zusatzschritt.

Die Neuerungen im Bereich der Anwendungsentwicklung reichen aber auch in die Welt der Java- und .NET-Entwickler hinein. Mit der neuen Version des WebSphere Development Studio Client werden auch in diesen Bereichen Neuerungen angeboten. Interessant ist auch die native Unterstützung von PHP mit dem Release V5R4 des i5/OS.

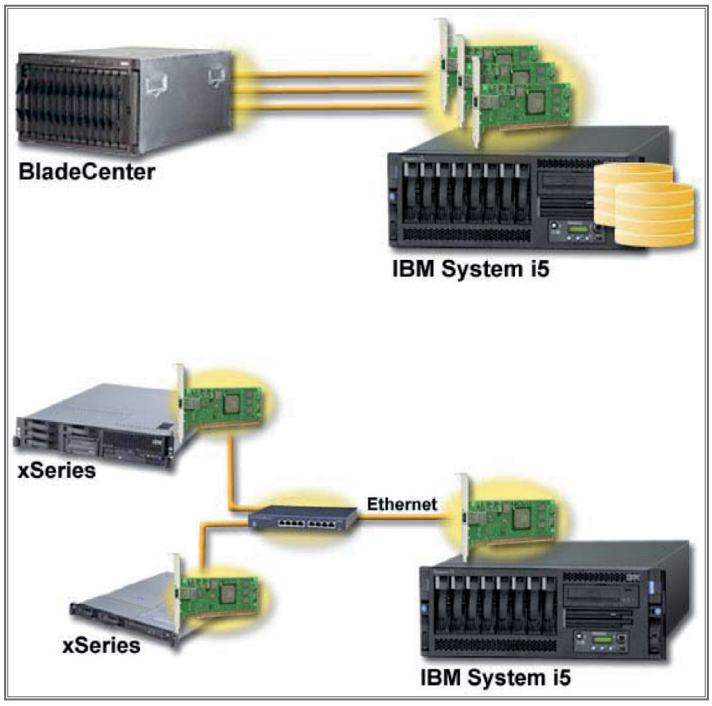
Viren haben es von Hause aus schwer, direkt auf der i5 Schaden anrichten zu können. Leider lässt sich mit dem Einsatz von i5/OS (oder auch den übrigen auf der i5 verfügbaren Betriebssystemen) nicht vermeiden, dass Viren dort angelegt oder auch weiter verteilt werden können. Aus diesem Grund hat IBM bereits in den vorhergehenden Release-Versionen des i5/OS spezielle Virenscanner für die Installation auf der i5 angeboten, deren Einsatz mit der Version V5R4 nun auch auf LINUX und AIX-Partitionen erweitert wurde.

Auch der Bereich der Kommunikation wurde mit V5R4 nicht „vergessen". Mit dem so genannten Enterprise Extender bietet IBM eine neue Unterstützung im Bereich des SNA Supports, der neben eklatanten Performance-Steigerungen auch die bisherigen SNA-Dienste, welche in den älteren Release Versionen des i5/OS verfügbar waren, unterstützt. Damit bietet sich eine Ablösung der in die Jahre gekommenen AnyNet-Lösungen an (siehe Neue Virtualization Engine). Eine Art Absichtserklärung verbirgt sich derzeit noch in der Ankündigung der Unterstützung von iSCSI, das im Laufe dieses Jahres für die i5-Modelle verfügbar sein wird. Damit ist dann in Zukunft neben der Anbindung bestimmter xSeries-Modelle auch die Anbindung an Blade Center möglich (siehe Integrierte xSeries, BladeCenter via iSCSI).

Der Bereich der Virtualisierung hat in den letzten Jahren stark an Bedeutung gewonnen. Auch in V5R4 setzt sich die Entwicklung mit der Verfügbarkeit der Virtuellen Bandlaufwerke weiter fort. Mit Hilfe dieser Technologie werden wir in Zukunft Datensicherungen auf Platte durchführen können. Damit kann das Datensicherungsfenster extrem verkleinert werden. Eine längst überfällige Neuerung ist die Erweiterung der Datensicherung im Bereich der Spoolfiles.



Neue Virtualization Engine



Integrierte xSeries, Bladecenter via iSCSI

Die bereits hohe Verfügbarkeit der Systeme i5 wird durch neue Verfahren im Bereich der Aktualisierung der Firmware weiter verbessert. In Zukunft werden die IPLs, die in Verbindung mit der Installation von PTFs erforderlich waren, sehr stark minimiert. Beim Einspielen von PTFs unter V5R4 werden die meisten Aktionen während der Laufzeit der Maschine durchgeführt und benötigen keinen IPL mehr. Damit entfällt der in vielen Fällen für das Anlegen der PTF erforderliche Neustart des Systems.

IBM stellt das System i5 in den Mittelpunkt zur Realisierung der Anforderungen der Kunden. Dabei ist weniger der technologische Aspekt von Bedeutung, sondern vielmehr die Gesamt-Lösungen, welche mit dem IBM System i5 realisierbar sind. Außerdem wurden bereits die Einstiegsmodelle in Form der 520 mit mehr Leistung ausgestattet, um auch moderne Anwendungen – wie Java-basierte Lösungen oder Groupware – besser und performanter ausführen zu können. Kunden profitieren bei den neuen Maschinen zudem von einem attraktiveren Preismodell. Damit verbessert sich das Preis-/Leistungsverhältnis gerade bei den kleineren Maschinen zum Teil erheblich. Vorbei sind auch die Zeiten der hohen Prozessor-Gruppen, welche den Preis für die Lizenzen des Betriebssystems i5/OS und der zugehörigen Lizenzprogramme

bestimmen. So bietet beispielsweise die Prozessor-Gruppe P05 mittlerweile eine maximale CPW-Leistung von 30/3100. Verbesserungen im Pricing der Software wirken sich nicht nur bei den geänderten Prozessor-Gruppen-Einteilungen aus, sondern werden auch durch einen wesentlich günstigeren Preis des Basisbetriebssystems bestimmt.

Mit der neuen Generation der Systeme i5 hat IBM auf die Erfahrungen der letzten Jahre reagiert. Wurden bisher Lösungen auf Basis von Java, WebSphere, Domino & Co. angeboten, dann musste man beim Einsatz dieser modernen Technologien auf der i5 feststellen, dass die Ausführungsgeschwindigkeit in vielen Fällen zu Wünschen übrig lies. Mit der Einführung des L3 Cache, der bis auf die kleinste i5 520 in allen Maschinen integriert ist, trägt IBM nun auch den Anforderungen durch die modernen Funktionen Rechnung, deren Performance-Verhalten wesentlich durch den Cache-Einsatz beeinflusst werden kann.

Man darf gespannt sein, ob IBM ihr Ziel „Neukundengewinnung“ mit den neuen Prozessoren und den damit verbundenen Möglichkeiten und Verbesserungen erreichen kann. Für Bestandskunden bieten die Neuerungen in den Bereichen Hardware und Software sicher viele Vorteile.

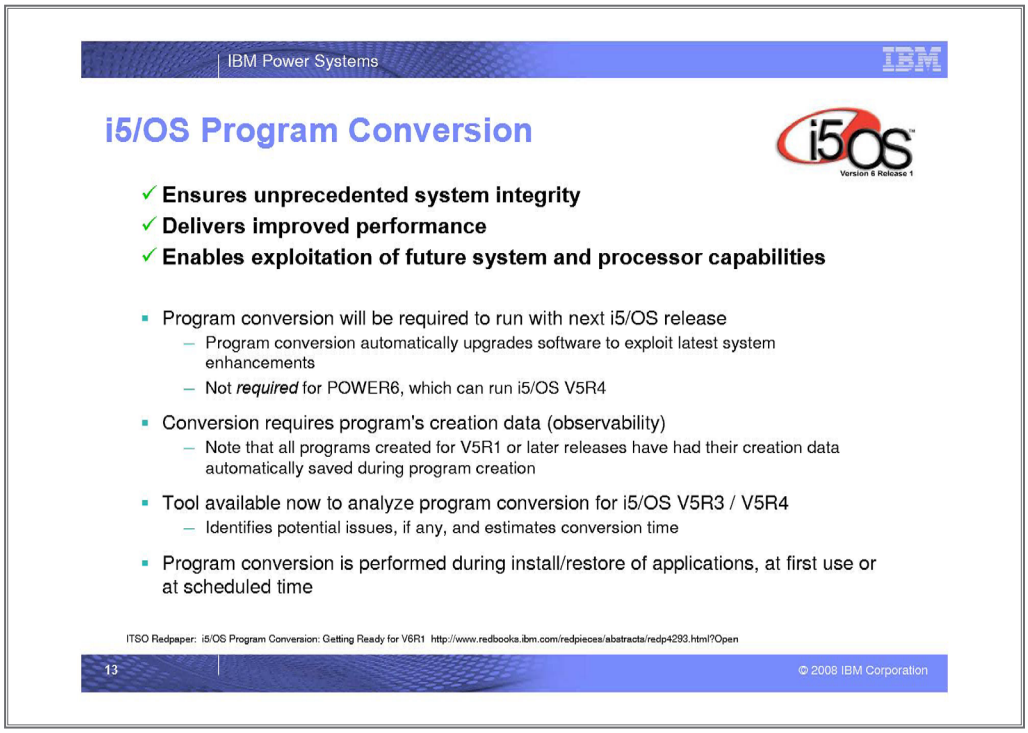
1.12
Seite 14

Aus i5/OS wird i

Die IBM hat Ende Januar 2008 eine neue Version des Betriebssystems i5/OS – i5/OS Version 6 Release 1 – angekündigt; nähere Details wurden dazu im Rahmen eines Presse-Briefings vorgestellt. Diese neue Version von i5/OS enthält zahlreiche Neuerungen („Major Enhancements"), die bahnbrechend sind und sich auf viele Bereiche erstrecken. Verfügbar ist das neue Release i5/OS V6R1 seit dem 21. März 2008.

Der Einsatz von i5/OS V6R1 bedingt für die Applikationen eine Konvertierung der bestehenden Programmobjekte, sofern diese Programme das so genannte MI (Machine-Interface-) Interface verwenden. Das sind also alle Programme die das „Integrated Language Environment" (ILE) oder das „Old Program Model" (OPM) verwenden. Programme, die ausschließlich Java oder „Portable Application Solutions Environment" (PASE) verwenden, sind hiervon nicht betroffen. Eine solche Konvertierung hat es in der Historie von System i bereits zweimal gegeben: einmal 1988 beim Wechsel von System /38 auf die AS/400 und das zweite Mal 1995 beim Wechsel von der 48-bit CISC- auf die 64-bit RISC-Architektur. Im Vergleich zum letzten Wechsel ist der für i5/OS V6R1 nötige Aufwand für die Konvertierung eines einzelnen Programmobjekts wesentlich geringer und dauert pro Objekt – je nach Komplexität der Anwendung, Anzahl der verwendeten Module und des jeweiligen Prozessors – in der Regel wenige Sekunden.

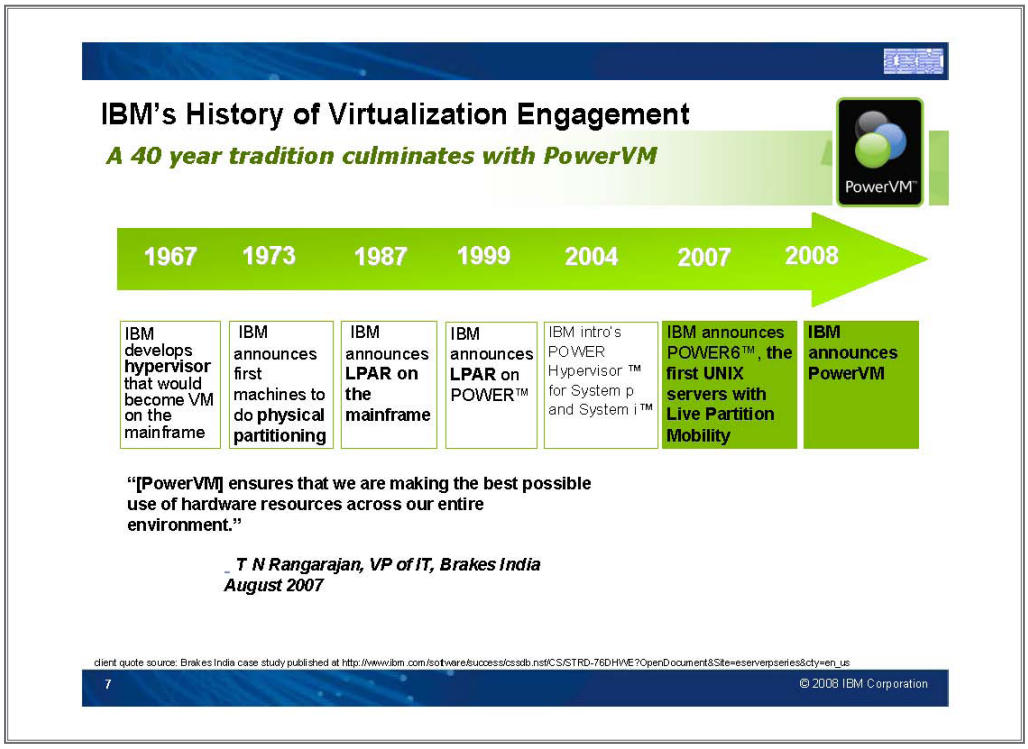
1.13
Seite 2



Die IBM stellt für i5/OS V5R3 und V5R4 ein Analyse-Tool zur Verfügung, um im Vorfeld überprüfen zu können, ob die Programmobjekte die so genannte „Observability“ aufweisen und damit problemlos konvertiert werden können. Weitere Informationen hierzu finden sich einem Redbook, das unter folgendem Link im PDF-Format bezogen werden kann: www.redbooks.ibm.com/redpieces/abstracts/redp4293.html?Open.

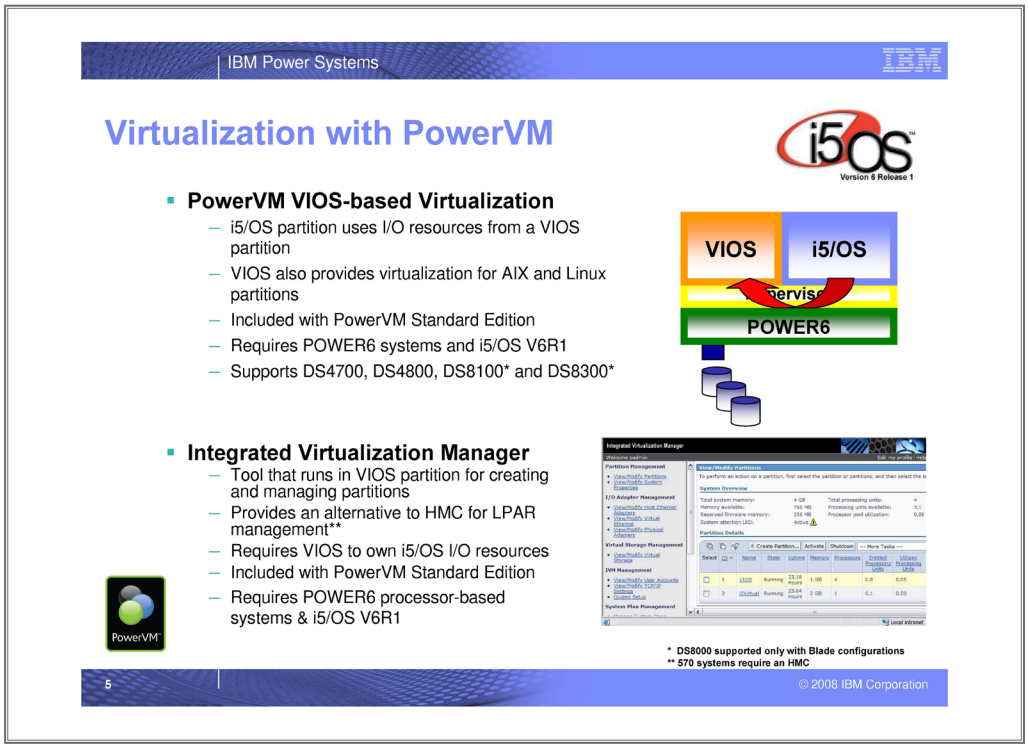
PowerVM – High-End-Virtualisierung

Durch den Einsatz von Virtualisierungstechnologien können multiple Server-Funktionen auf wenigen Servern betrieben werden. So können auf einem einzigen Server beispielsweise mehrere Betriebssysteme sowie verschiedene Geschäftsanwendungen gleichzeitig laufen. Die Verwendung von Virtualisierungskapazitäten kann IT-Kosten und Geschäftsrisiken reduzieren, Effizienz und Flexibilität erhöhen, Installation und Management der IT-Infrastruktur vereinfachen, die Ausfallsicherheit erhöhen und neue Innovationen ermöglichen.

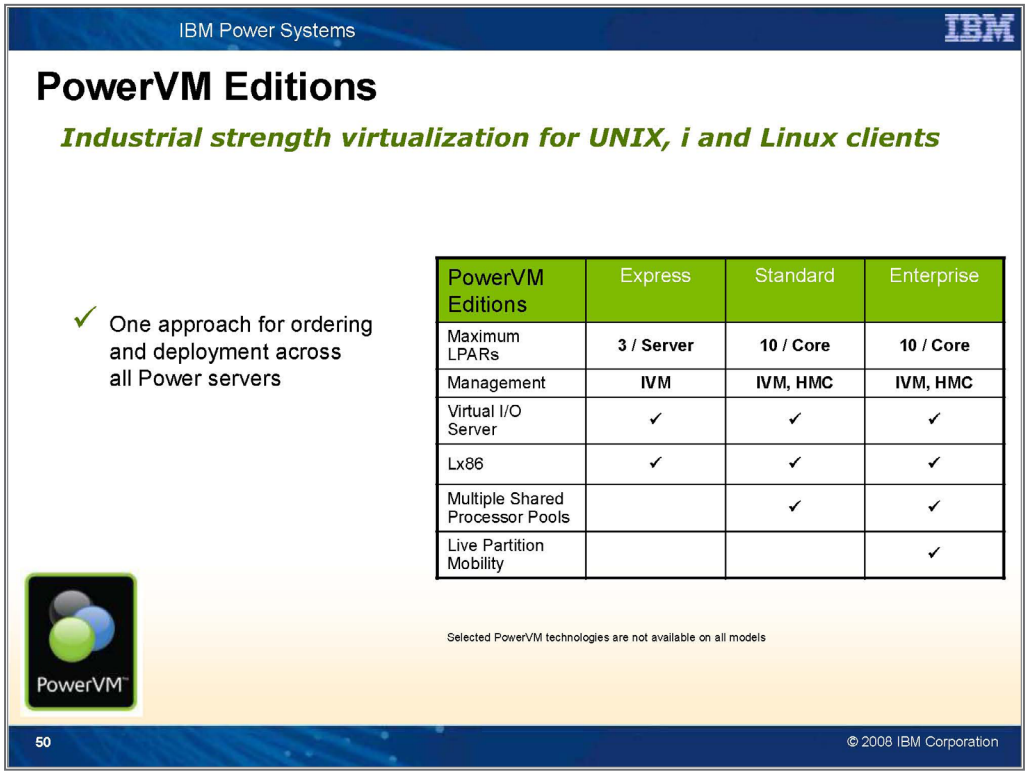


1.13
Seite 4

Basierend auf diesen Anforderungen hat IBM parallel zu i5/OS V6R1 eine neue Virtualisierungsplattform vorgestellt: PowerVM. Diese PowerVM-Plattform – vormals Advanced Power Virtualization (APV) – ist ab sofort in den Editionen Express, Standard und Enterprise verfügbar. Sie soll Kunden dabei unterstützen, ihre IT-Kosten effektiver zu steuern, eine höhere Energieeffizienzrate zu erreichen und die Ressourcenauslastung zu erhöhen. Somit bietet PowerVM umfangreiche Virtualisierungslösungen für die weitverbreiteten Betriebssysteme AIX (das UNIX-Betriebssystem von IBM), Linux und i für System i.



Nachfolgend sehen Sie eine Übersicht über die verschiedenen PowerVM-Editionen:



Security

Der sensible Bereich Security steht in der Prioritätenliste der Kunden und der IBM sehr weit oben. Neu in i5/OS V6R1 sind die beiden folgenden Optionen: zum einen die Möglichkeit, Daten in so genannten „Auxiliary Storage Pools" verschlüsseln zu können, und zum anderen, Datensicherungen verschlüsselt auf ein Bandlaufwerk oder virtuelles Tape abspeichern zu können. Dadurch sind sensible Daten mit der höchstmöglichen Sicherheit geschützt.

Im Umfeld von „Intrusion Detection and Prevention" (Schutz vor unerlaubten Zugriffen) wurden bereits mit i5/OS V5R4 einige zusätzliche Möglichkeiten in das Betriebssystem eingeführt; als Beispiel sei der Intrusion Monitor genannt. Dieser zeichnet entsprechende Zugriffsverletzungen für den Administrator zur weiteren Bearbeitung in Audit-Files auf. Mit i5/OS V6R1 wird dies nun dahingehend erweitert, dass eine Benachrichtigung solcher Zugriffsverletzungen in Real-Time – beispielsweise per e-Mail oder Pager – erfolgen kann. Darüber hinaus gibt es nun erstmalig eine Technologie, die es erlaubt, im Falle eines Systemangriffes („Denial of Service Attacks") Filter zu aktivieren, die dann dafür sorgen, dass solche Angriffe das System möglichst wenig belasten. Dieser Prevention-Mechanismus mit seiner Filter-Funktion basiert auf systeminternen Zeitintervallen und Statistiken; er sorgt beim Überschreiten der Schwellwerte dafür, dass diese Angriffe erkannt wer-

den und dass das System daraufhin automatisch bestmöglich geschützt werden kann. Diese Technologie ist vollkommen neu, steht zum Patent an und wird erstmalig mit i5/OS V6R1 eingeführt.

Der dritte wichtige Punkt behandelt das Thema „digitale Signatur von i5/OS-Betriebssystemobjekten". Diese Signatur sorgt für eine höchstmögliche Integrität innerhalb des Systems und verhindert, dass Systemobjekte in irgendeiner Form manipuliert beziehungsweise modifiziert werden können.

Virtualisierung und Logical Partitioning
Im Zusammenhang mit der Virtualisierung bietet i5/OS V6R1 ebenfalls etliche Erweiterungen. So ist es erstmalig möglich, dass eine i5/OS V6R1-Partition der Host für eine weitere i5/OS V6R1-Partition sein kann. Diese Funktionalität stand in der Vergangenheit bereits für AIX-, Linux- oder Windows-Umgebungen zur Verfügung. Die i5/OS-Hosting-Partition ist für das Management der Ressourcen (Platten, Adapter etc.) zuständig und stellt sie dann den anderen Client-Partitionen virtuell zur Verfügung. Nunmehr ist diese Hosting-Möglichkeit von i5/OS V6R1 also auch für andere i5/OS V6R1-Partitionen gegeben. Dies erlaubt es den Kunden, sehr einfach Test- und/oder Entwicklungsumgebungen aufbauen zu können. Die Client-Partition benötigt jetzt keine eigenen Ressourcen mehr, sondern verwendet die Ressourcen der Hosting-Partition. Dadurch reduzieren sich die Kosten und das Management für diese Partition erheblich.

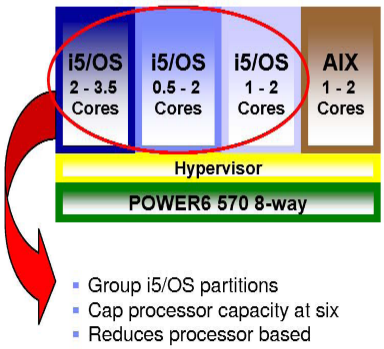
Erweiterungen gibt es auch im Bereich von Logical Partitioning. So ist es mit i5/OS V6R1 möglich, mehrere i5/OS V6R1 Micro-Partitions logisch in einem Pool zusammenzufassen und für diesen Pool eine Obergrenze festzulegen (siehe nachfolgende Abbildung). Diese Obergrenze definiert die maximale Anzahl an Prozessoren, die gleichzeitig in diesem Pool aktiv sein dürfen. Diese Art der Limitierung erlaubt es den Kunden, die Anzahl der erforderlichen Software-Lizenzen sehr einfach zu kontrollieren. Somit ist es auch ein Beitrag zur Kostenreduzierung.

IBM Power Systems

Logical Partitioning Enhancement

- **Shared Processor Pools**
  - Group micro-partitions into separate pools and set limits on the amount of processor capacity each pool can consume
  - Supports i5/OS V5R4 and V6R1, AIX, and Linux partitions on POWER6 processor-based systems
- Can help limit required software licenses

Shared Processor Pool Example



- Group i5/OS partitions
- Cap processor capacity at six
- Reduces processor based software licenses from eight to six

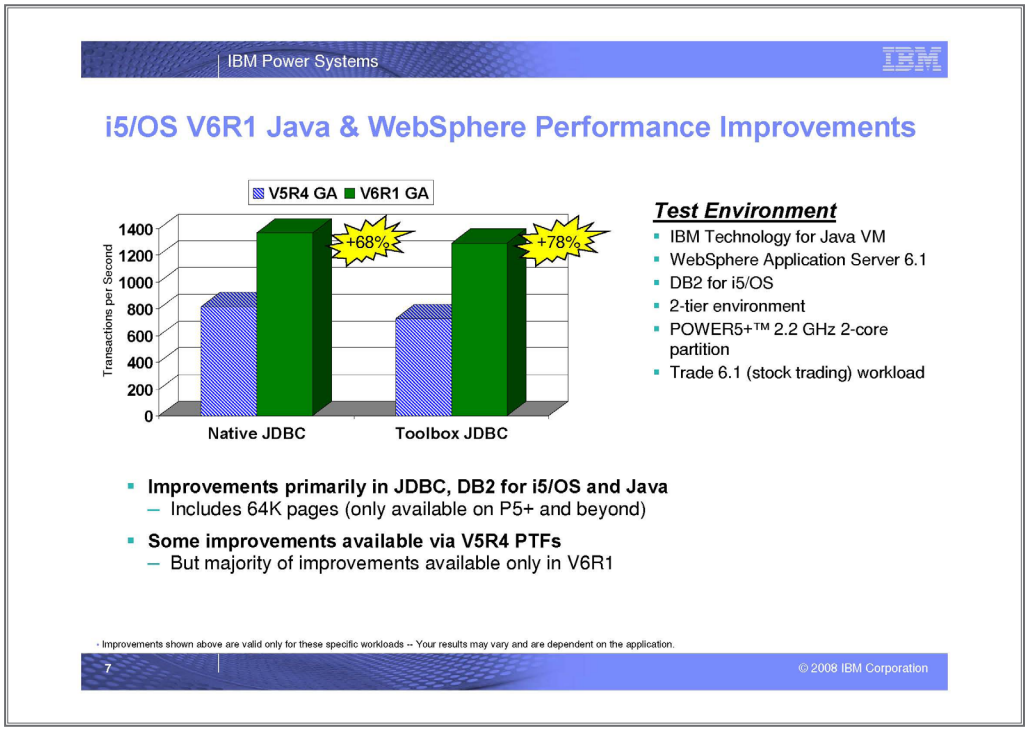
6

© 2008 IBM Corporation

1.13
Seite 8

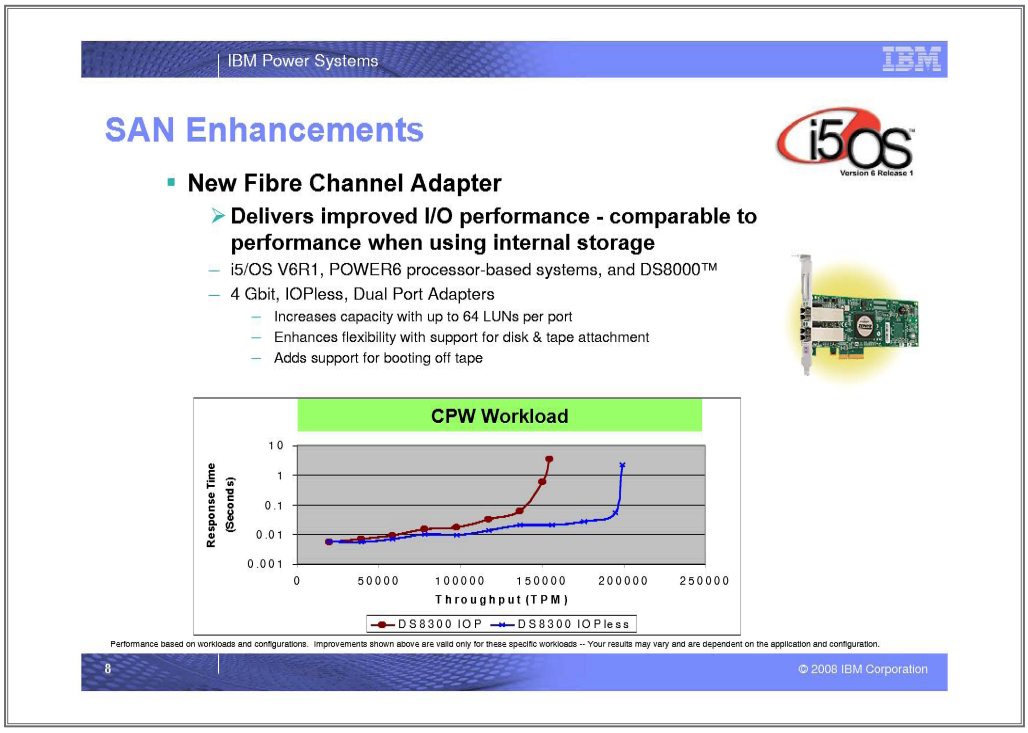
Bessere Java-Performance

Das neue Betriebssystem i5/OS V6R1 bringt auch eine dramatisch verbesserte Java-Performance mit sich. So wurde beispielsweise in einem IBM-internen Benchmark auf einem 2.2 Ghz POWER5+-Prozessor ein „Trade 6.1" Workload (basierend auf Java und WebSphere Application Server 6.1) unter i5/OS V5R4 und i5/OS V6R1 verglichen. Die Leistungssteigerungen bewegten sich in der Größenordnung von etwa 70 bis 80 Prozent. Auch wenn diese Performance-Steigerungen nur für diese spezielle Benchmark-Umgebung gelten, ist doch erkennbar, dass Kunden mit Java-basierenden Anwendungen erheblich vom neuen i5/OS V6R1 profitieren werden.



Neuer Fibre Channel Adapter

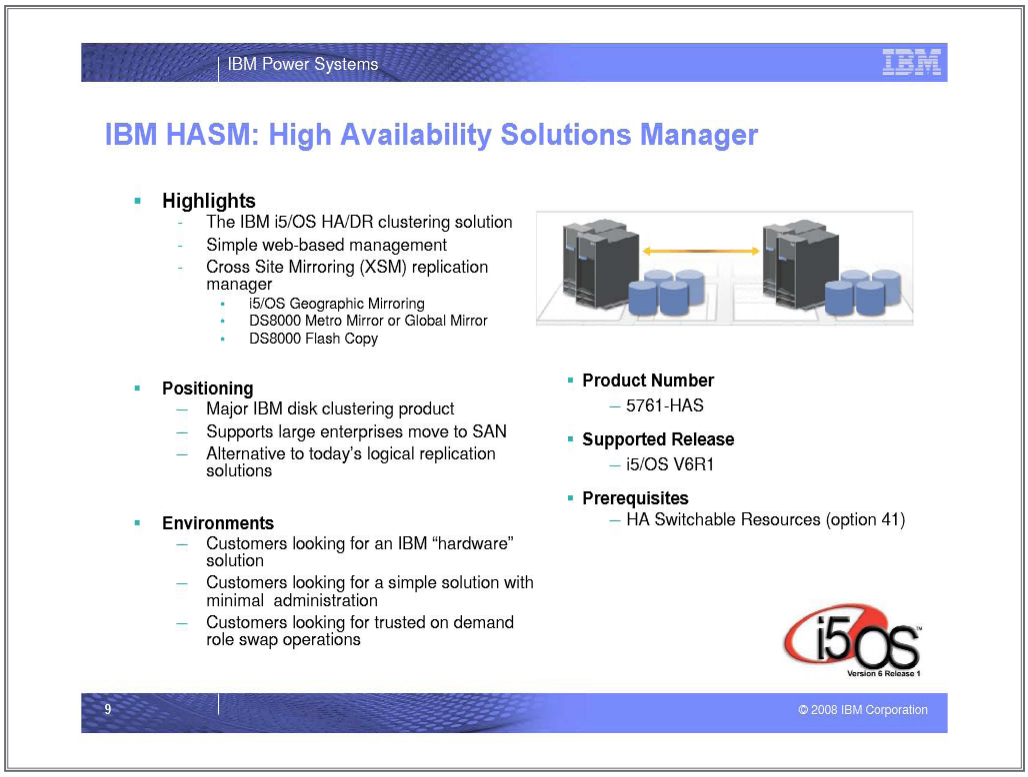
Im Zusammenhang mit i5/OS V6R1 wurde auch ein neuer Fibre Channel Adapter angekündigt. Dieser neue FC-Adapter bietet größeren Kunden mit einer SAN-Umgebung und einem IBM TotalStorage DS8000 eine erheblich bessere Performance im Vergleich zum bisherigen FC-Adapter. Der neue FC-Adapter ist nicht mehr wie der Vorgänger auf einen so genannten Input-Output-Processor (IOP) angewiesen und belegt dadurch nur noch einen PCI-Slot, mit der Folge, dass sich dadurch auch die Kosten erheblich reduzieren. Der neue FC-Adapter wird nur von i5/OS V6R1 in Verbindung mit System i POWER-6-Hardware und IBM TotalStorage DS8100 beziehungsweise DS8300 unterstützt.



1.13
Seite 10

Hochverfügbarkeit inklusive

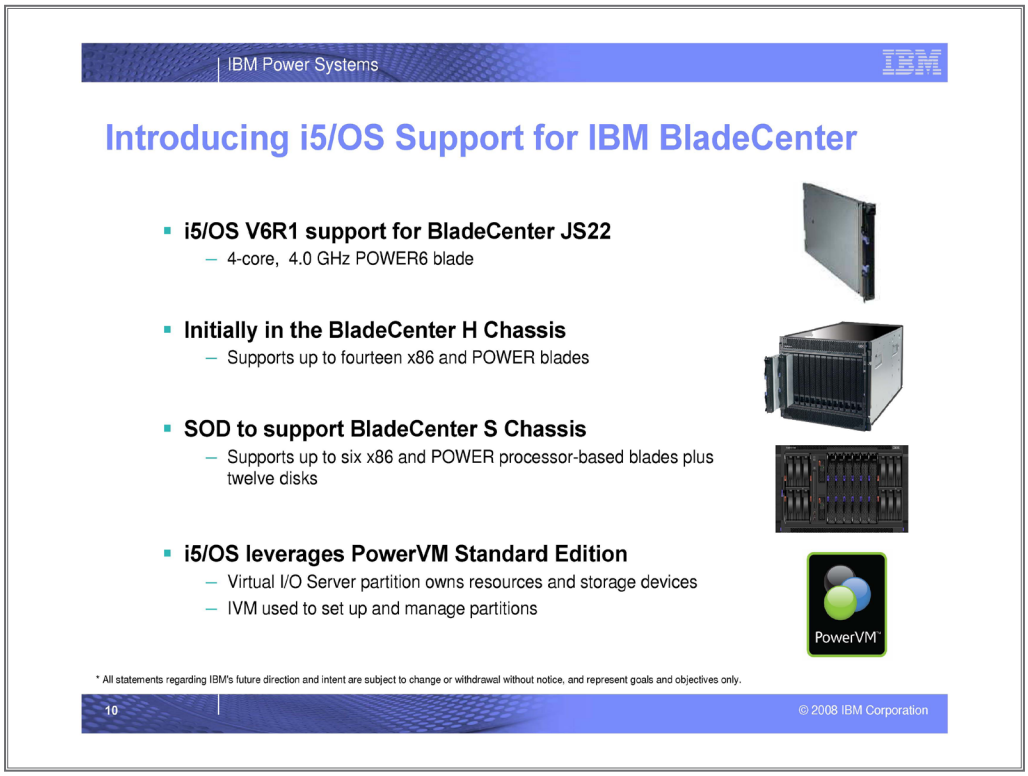
Das Thema Hochverfügbarkeit steht bei vielen Kunden in der Prioritätenliste sehr weit oben. Vor diesem Hintergrund hat die IBM mit i5/OS V6R1 ein neues Lizenzprogramm, den „High Availability Solutions Manager“ (HASM) angekündigt. Diese Cluster-Lösung hilft Kunden, den Einfluss von geplanten und ungeplanten Ausfällen zu minimieren. HASM unterstützt die Implementierung und das Management von Cluster-Umgebungen, die technologisch auf sogenannten „Independent Auxiliary Storage Pools“ (IASP) basieren und in Verbindung mit i5/OS Geographic Mirroring oder DS8000 Metro Mirror oder Global Mirror betrieben werden. Das Management der Lösung geschieht über ein webbasiertes Frontend und gestaltet sich dadurch recht einfach. Anhängig von der Implementierung beziehungsweise von den Anforderungen ist diese Cluster-Lösung sowohl für kleinere Kunden mit einfachen Umgebungen als auch für größere mit komplexeren IT-Strukturen vorgesehen.



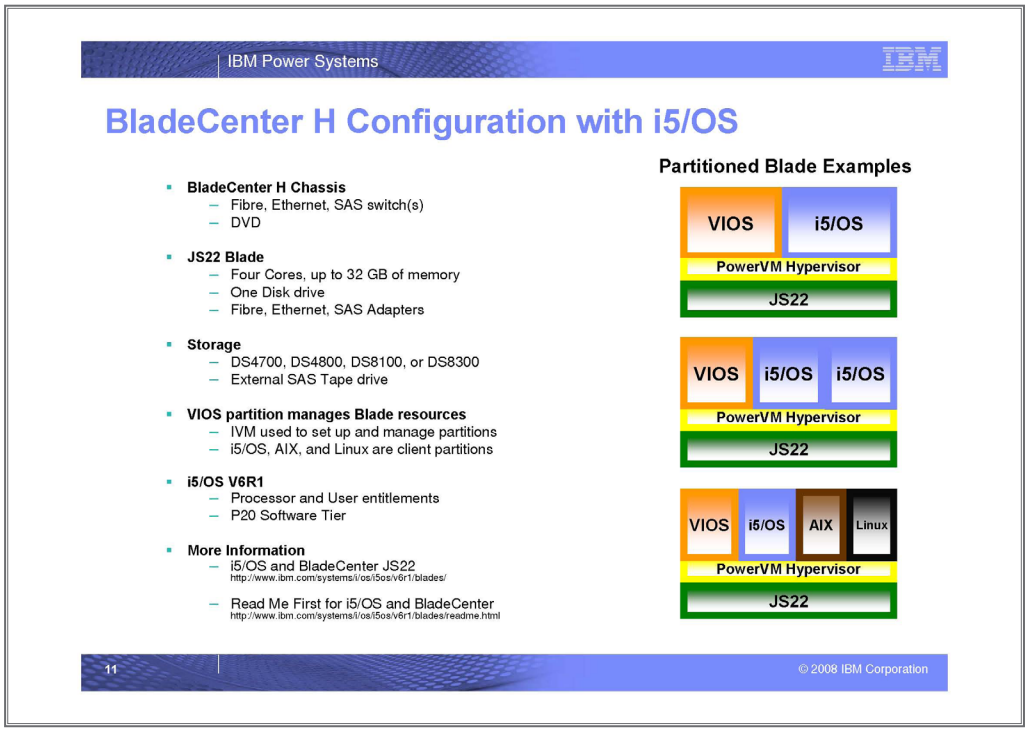
Außerdem wird die Hochverfügbarkeitslösung iCluster als offizielles Lizenzprogramm Einzug in das IBM-Portfolio nehmen. Dieses Produkt iCluster stammt von DataMirror, einem Unternehmen, das 2007 von IBM übernommen worden ist. Zusammen mit dem HASM wird dem Kunden nun ein großes Portfolio an Hochverfügbarkeits-Lösungen aus Hand der IBM angeboten.

i5/OS im BladeCenter

Neu ist auch die Möglichkeit, in einem BladeCenter H-Chassis i5/OS V6R1 als Client auf einem POWER-6-basierten JS 22 Bladeserver laufen zu lassen. Der JS 22 Bladeserver besitzt einen POWER-6-Prozessor mit 4 Cores und hat eine Taktrate von 4 Ghz. Auf der JS 22 muss ein virtueller I/O-Server (VIOS) betrieben werden, der für das Ressourcen-Management verantwortlich ist und mit der Virtualisierungsplattform PowerVM mitgeliefert wird. Das i5/OS V6R1 wird als Client auf demselben JS 22 Bladeserver installiert und nutzt dann die virtuell zur Verfügung gestellten Ressourcen. Die IBM plant zu einem späteren Zeitpunkt auch das BladeCenter S-Chassis in Verbindung mit dem JS 22 Bladeserver durch i5/OS V6R1 zu unterstützen.



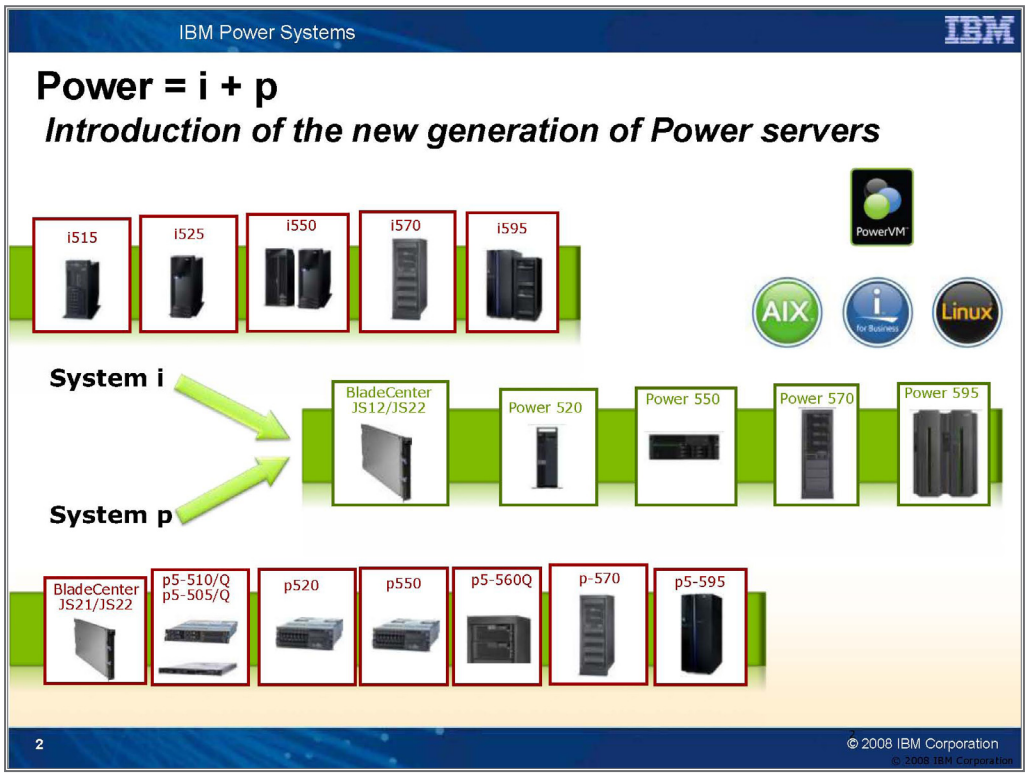
1.13
Seite 12


IBM Power Systems
IBM
BladeCenter H Configuration with i5/OS
BladeCenter H Chassis
Fibre, Ethernet, SAS switch(s)
DVD
JS22 Blade
Four Cores, up to 32 GB of memory
One Disk drive
Fibre, Ethernet, SAS Adapters
Storage
DS4700, DS4800, DS8100, or DS8300
External SAS Tape drive
VIOS partition manages Blade resources
IVM used to set up and manage partitions
i5/OS, AIX, and Linux are client partitions
i5/OS V6R1
Processor and User entitlements
P20 Software Tier
More Information
i5/OS and BladeCenter JS22
http://www.ibm.com/systems/i/os/i5os/v6r1/blades/
Read Me First for i5/OS and BladeCenter
http://www.ibm.com/systems/i/os/i5os/v6r1/blades/readme.html
Partitioned Blade Examples
VIOS
i5/OS
PowerVM Hypervisor
JS22
VIOS
i5/OS
i5/OS
PowerVM Hypervisor
JS22
VIOS
i5/OS
AIX
Linux
PowerVM Hypervisor
JS22
11
© 2008 IBM Corporation

IBM Announcement April 2008

Aus System i5 werden IBM Power Systems

IBM hat im April 2008 die neue IBM Power Systems Produktlinie vorgestellt, welche den System i- und System p-Kunden eine vereinheitlichte, gemeinsame Serverreihe mit vereinfachter Preisstruktur, erweiterter Anwendungsauswahl und verbesserten Energie- und Managementkosten bietet.



Übersicht über die neuen Power Systems-Server

Die neuen IBM Power Systems-Server enthalten POWER6-Prozessoren und unterstützen auf einem einzigen Server gleichzeitig sowohl UNIX- und Linux- als auch i-Applikationen. Durch den Umstieg auf die neue Generation der Power Systems-Server können System i-Kunden ihre geschäftskritischen Anwendungen zu leistungsfähigeren und energieeffizienteren Lösungen hin migrieren – inklusive extrem kosteneffizienter Blade-Systeme – und dabei dieselben i-Applikationen und Betriebssysteme nutzen, auf die sie schon seit rund 20 Jahren vertrauen.

Kunden werden auch Zugang zur POWER6 EnergyScale-Technologie erhalten, die erweiterte Energieverbrauchs-Kontrollmöglichkeiten bietet, sowie zur PowerVM-Virtualisierungstechnologie, die es Kunden ermöglicht, bis zu 160 virtuelle Partitionen in einem einzigen System anzulegen und so die Serverauslastung deutlich zu verbessern. Zusätzlich wird das sichere und integrierte Betriebssystem von IBM, i5/OS, ab sofort schlicht „i“ heißen. Die

1.13.1
Seite 2

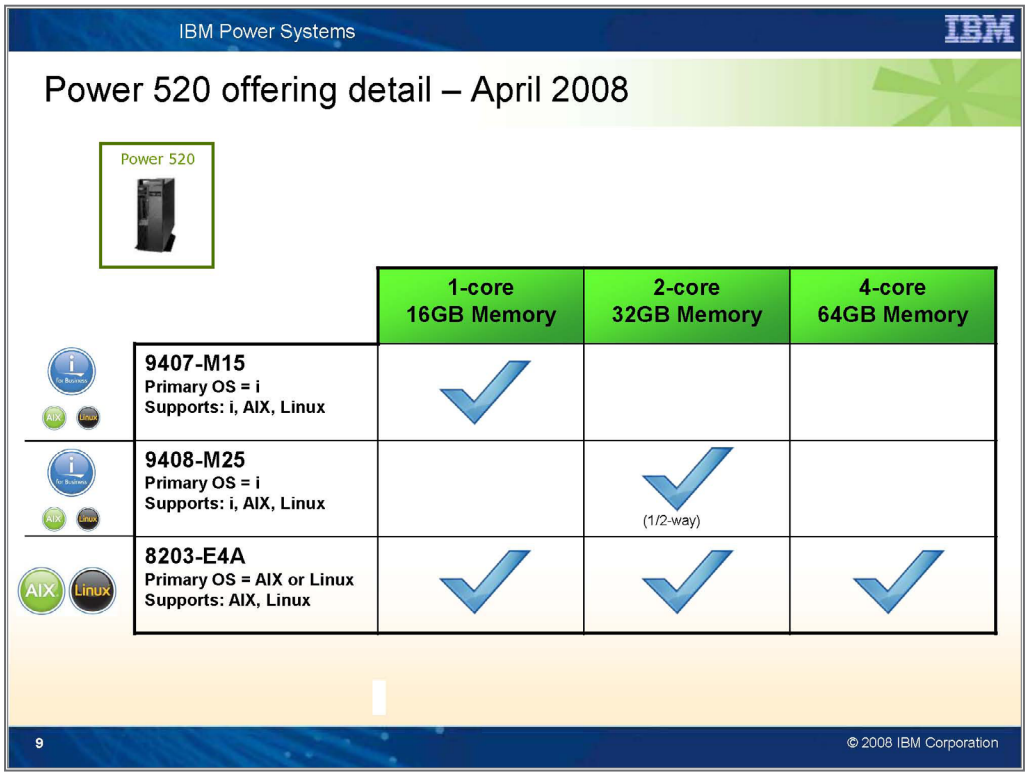
Power-Server mit i-Betriebssystem werden als integrierte Angebote verfügbar sein, die leicht zu konfigurieren, zu installieren und auszurollen sind. Unter dem Namen „i Edition“ werden sie neben integrierten Linux-Editionen und AIX-Editionen für UNIX-Kunden vertrieben werden. Kunden können auch „a la carte“ wählen und i, AIX und Linux auf einem einzigen Power-Server miteinander kombinieren, um die Flexibilität, Systemauslastung und Systemnutzung bestehender Komponenten zu optimieren.

Die Zusammenlegung der System i- und System p-Produktlinien erleichtert IBM Geschäftspartnern den Vertrieb und die Betreuung verschiedener Lösungskombinationen auf unterschiedlichen Betriebssystemen auf der Power Server-Plattform. IBM erkennt zudem die Investition der Geschäftspartner in die Zertifizierung von Vertriebs- und IT-Personal für den Verkauf von System i und System p an. Diese Zertifikate erlauben es Geschäftspartnern beider Plattformen, die neue gemeinschaftliche Plattform mit entsprechenden Betriebssystemen je nach Qualifikation und Authorisierung zu vertreiben.

Die neue Power-Plattform bietet ISVs eine gemeinsame Plattform für i-, UNIX- und Linuxlösungen und erleichtert es den ISVs, die für den Kunden passende Lösung zu finden. Die neue Power-Plattform eröffnet ISV-Partnern von IBM neue Möglichkeiten, um einen größeren Kundenkreis zu adressieren und neue Produkte zu vertreiben, besonders wenn sie ein Lösungsportfolio haben, das auf mehreren Betriebssystemen läuft.

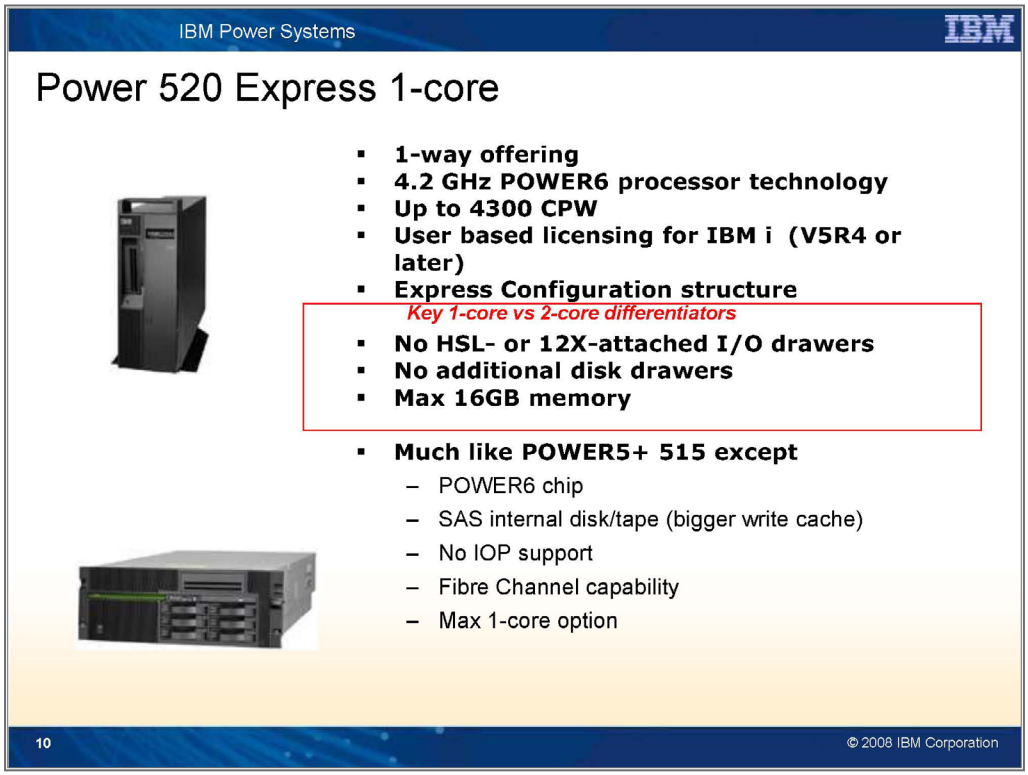
Die ersten Modelle dieser neuen Produktlinie wurden speziell für kleine und mittelständische Unternehmen entwickelt. IBM i Edition Express für Blade Center S, Power 520 Express und Power 550 Express bieten Mehrwert und Leistung in einem Komplettpaket, das an das Budget und die begrenzten IT-Systemsmanagementkapazitäten des Mittelstands angepasst ist:

IBM Power 520 Express
eignet sich besonders für Unternehmen, die über verteilte Anwendungen und Datenbanken verfügen. Das Modell ist als AIX-, Linux- und i-Edition mit drei Prozessorvarianten verfügbar.

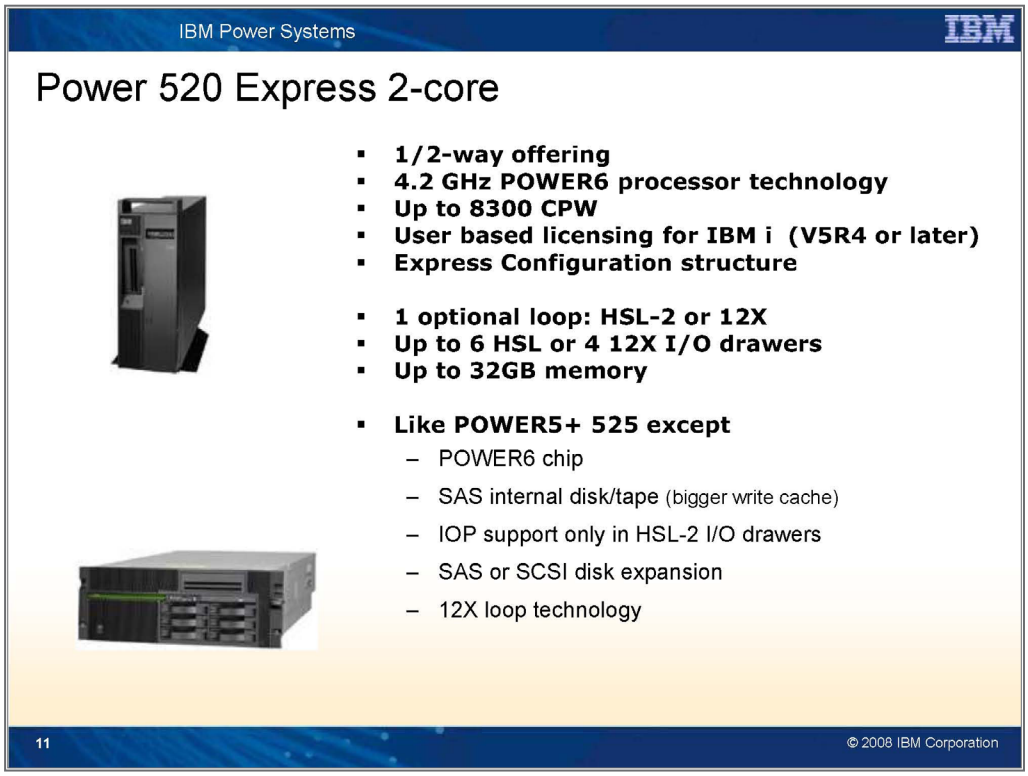


Das Power System 520 ist in drei verschiedenen Ausführungen erhältlich

Die neuen Systeme verfügen über einen 4,2 Ghz-Power6-Prozessor und – im Vergleich zu den bisherigen Maschinen – über einen erweiterten Ausstattungsumfang.

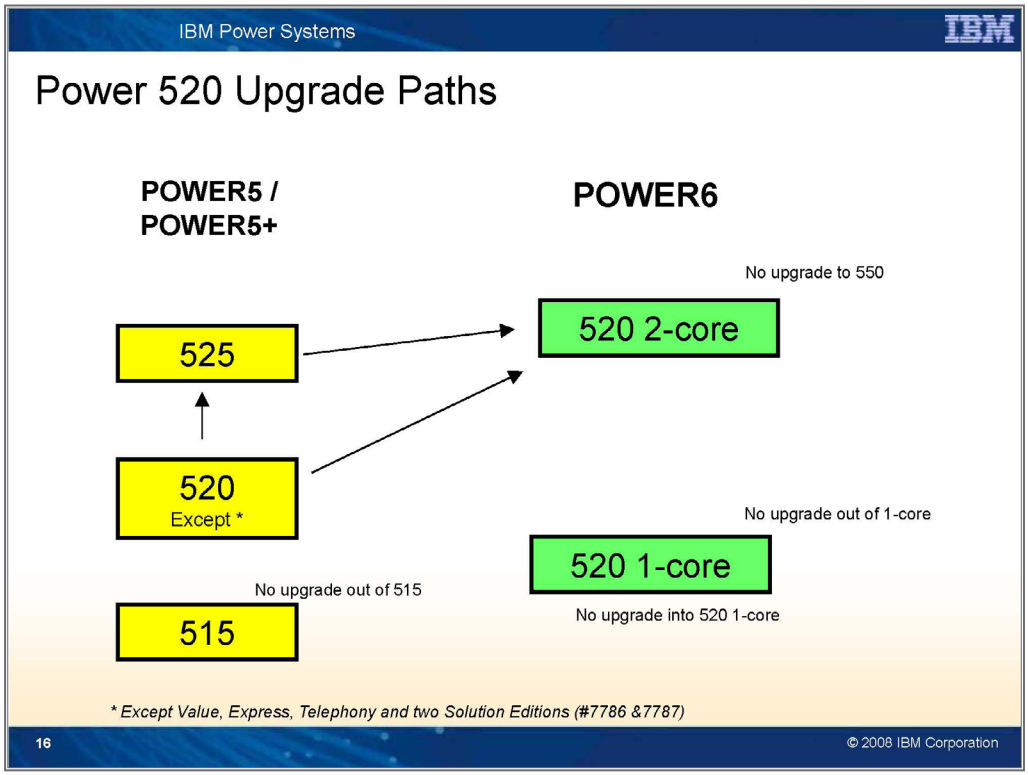


1.13.1
Seite 4



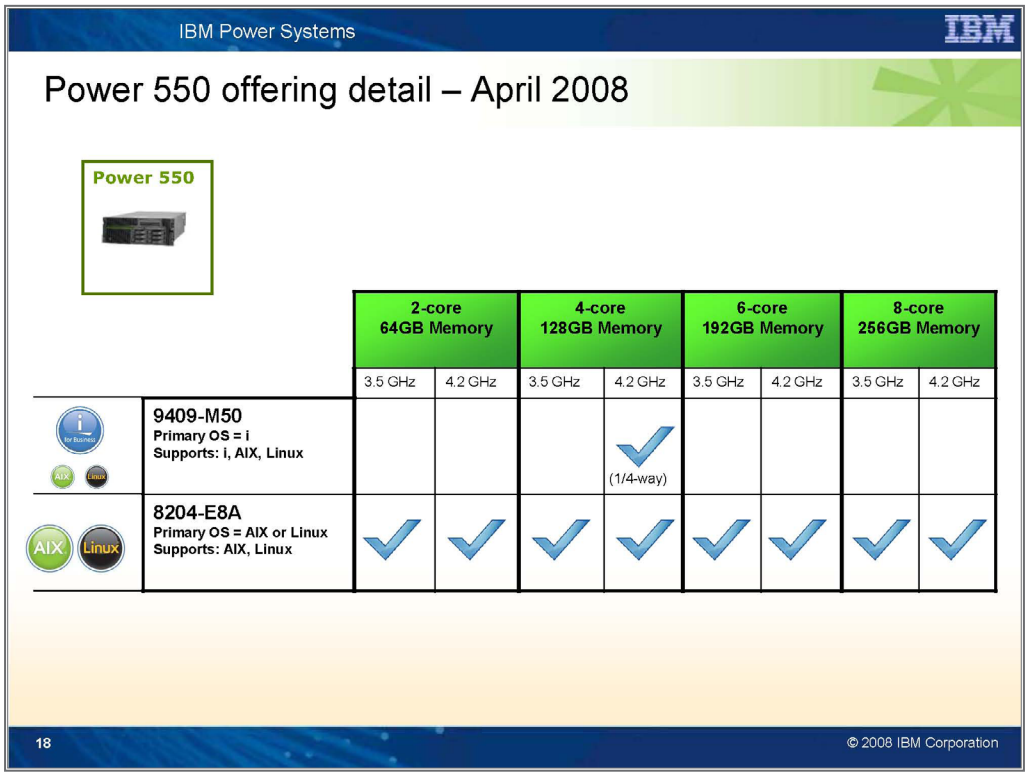
Upgrade von Power5/Power5+ Maschinen

Die nachfolgende Grafik zeigt eine Übersicht über die Upgrademöglichkeiten.

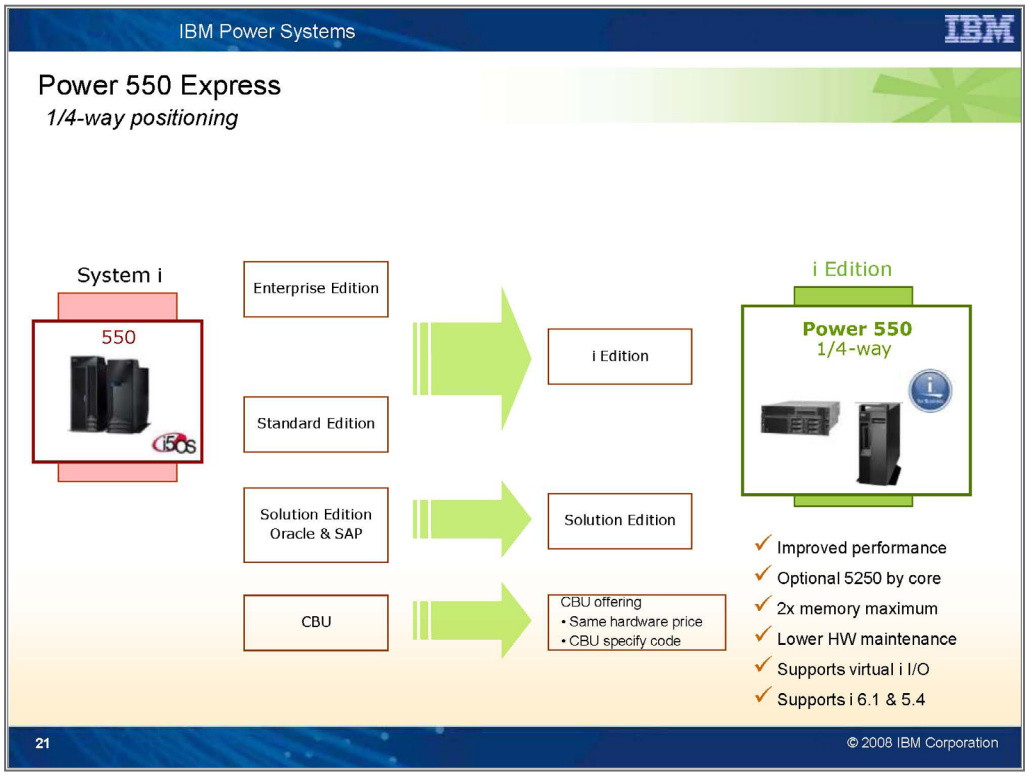


IBM Power 550 Express

bietet viel Leistung in einem mittelgroßen Datenbankserver mit hoher Kapazität und kontinuierlicher Applikationsverfügbarkeit. Das Modell ist mit den Betriebssystemen AIX, Linux und i erhältlich.

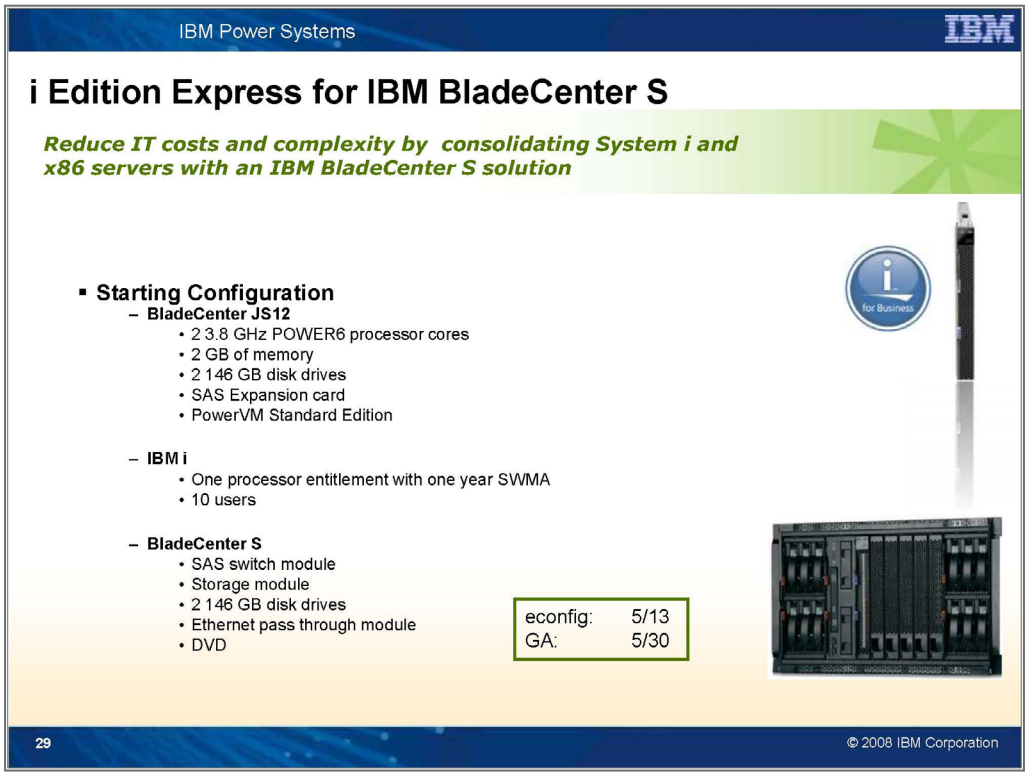


1.13.1
Seite 6

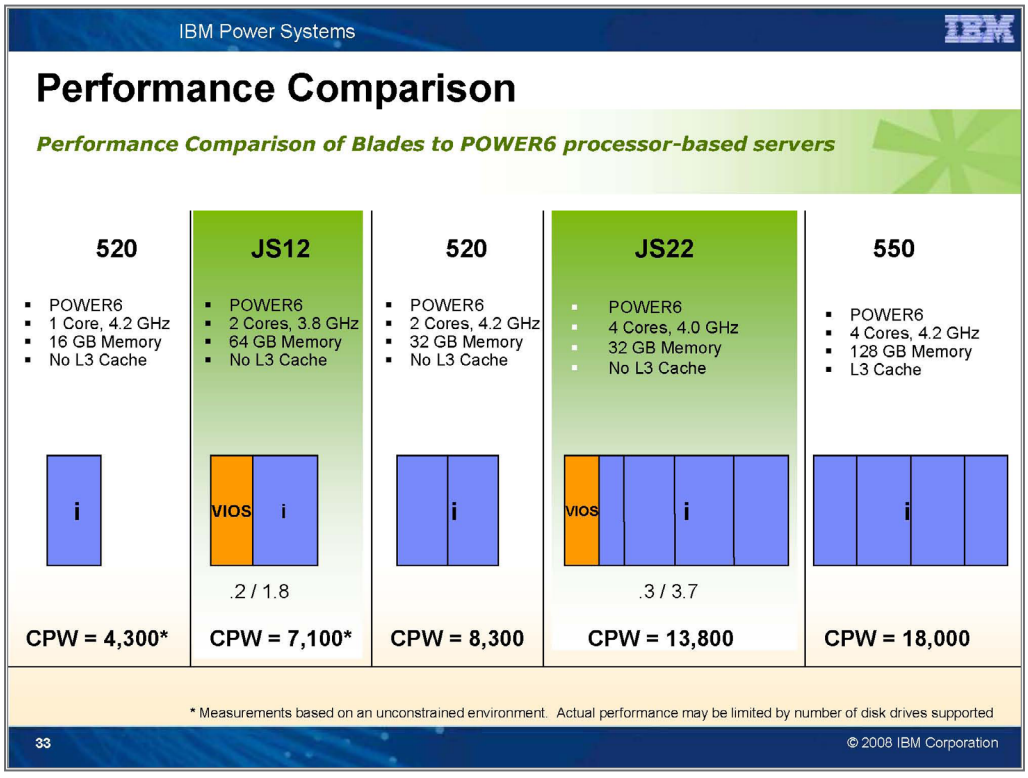


IBM i Edition für BladeCenter S

bietet einen attraktiven Upgradepfad für Kunden von AS/400, der iSeries sowie System i515, 520 und 525. Diese Option ermöglicht nicht nur Kostenersparnis, sondern auch eine vereinfachte Integration von i-Applikationen mit vorhandenen x86-Servern. Die Einstiegspreise für dieses Modell sind bis zu 50 Prozent günstiger als Non-Blade-Konfigurationen mit gleicher Rechenkapazität. Diese Value Proposition eignet sich besonders für System i-Kunden, Geschäftspartner und ISVs. Bei dreijähriger eingeschlossener Wartung können Kunden durch die attraktiven Leasingpreise für dieses Modell oft deutlich weniger aufwenden als heute. Kunden können durch die Migrationsmöglichkeit von bestehenden x86-Servern auf x86-Blades im gleichen BladeCenter S-Chassis einen weiteren Mehrwert erzielen, die Infrastruktur vereinfachen, die Integration erhöhen und auf eine dreijährige Wartung und abgestimmte Leasingangebote zurückgreifen.

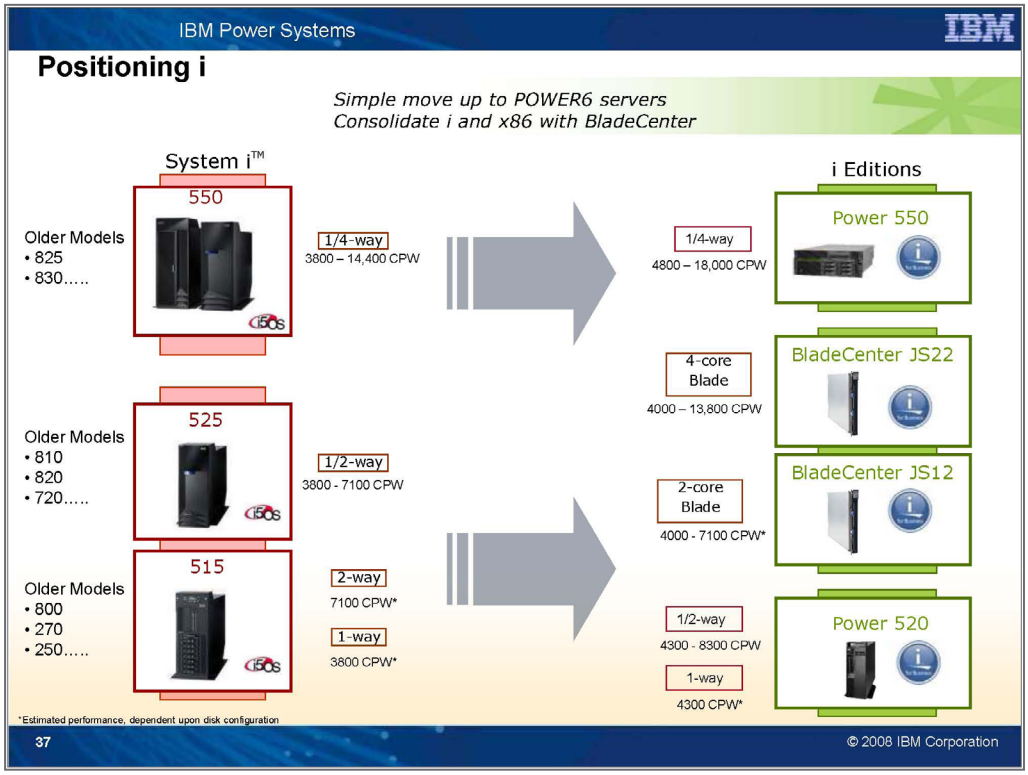


Die nachfolgende Abbildung zeigt einen Leistungsvergleich aller neu angekündigten Maschinen.



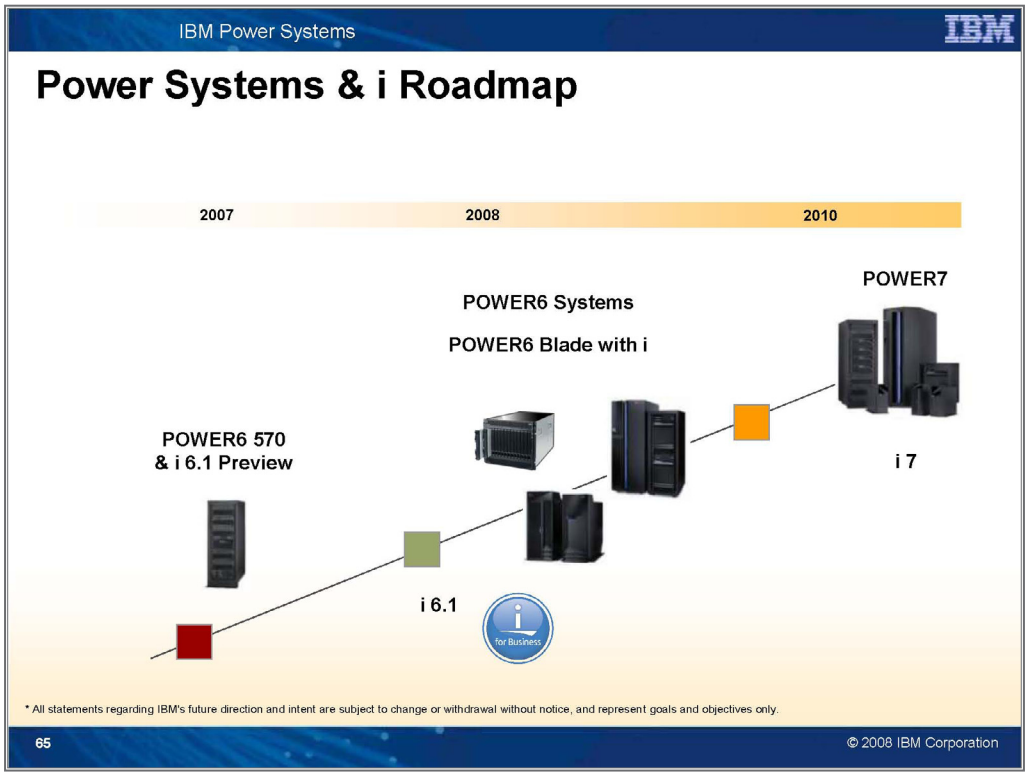
1.13.1
Seite 8

Diese Abbildung zeigt einen CPW-Leistungsvergleich zwischen den bisherigen und den neuen Maschinen.



Ausblick in die Zukunft

Das Announcement vom Frühjahr 2008 stellt sicherlich einen Meilenstein in der Geschichte der erfolgreichen Mittelstandsserver von IBM dar – die technische Entwicklung geht jedoch kontinuierlich weiter. Die nachfolgende Abbildung zeigt eine mögliche Roadmap bis 2010. Wir dürfen also auf weitere bahnbrechende Leistungsdaten gespannt sein.


IBM Power Systems
IBM
Power Systems & i Roadmap
2007
2008
2010
POWER7
POWER6 Systems
POWER6 Blade with i
POWER6 570
& i 6.1 Preview
i 7
i 6.1
for Business
* All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.
65
© 2008 IBM Corporation

1.13.1
Seite 10

IBM Announcement April 2009

Technische Aspekte der Power Systems Ankündigungen vom 28. April 2009

Die gegenwärtigen weltwirtschaftlichen Probleme zwingen die Unternehmen mehr als je zuvor intelligente Lösungen zu finden, um den wachsenden Anforderungen hinsichtlich Kostendruck, Service-Verbesserungen und Risiko-Management besser begegnen zu können. Mit Dynamic Infrastructure stellt IBM den Unternehmen sein gesamtes Portfolio bestehend aus Hardware-, Software- und Service-Lösungen zur Verfügung, um auf diese Herausforderungen eine adäquate Antwort zu finden. In diesem Kontext sind auch die IBM Power Systems Ankündigungen vom 28. April 2009 zu sehen. Die neuen Hardware- und Software-Angebote sollen den Unternehmen helfen, ihre Ressourcen intelligenter zu nutzen, die Betriebs- und Energie-Kosten zu reduzieren und gleichzeitig eine dynamische Infrastruktur für die Herausforderungen des 21. Jahrunderts zu implementieren.

Im Folgenden werden jetzt die wesentlichen technischen Aspekte dieser Ankündigungen aus Sicht von IBM i dargestellt; eine Einbeziehung von AIX und Linux an dieser Stelle würde den Rahmen dieses Beitrages überschreiten.

POWER6+ Technologie

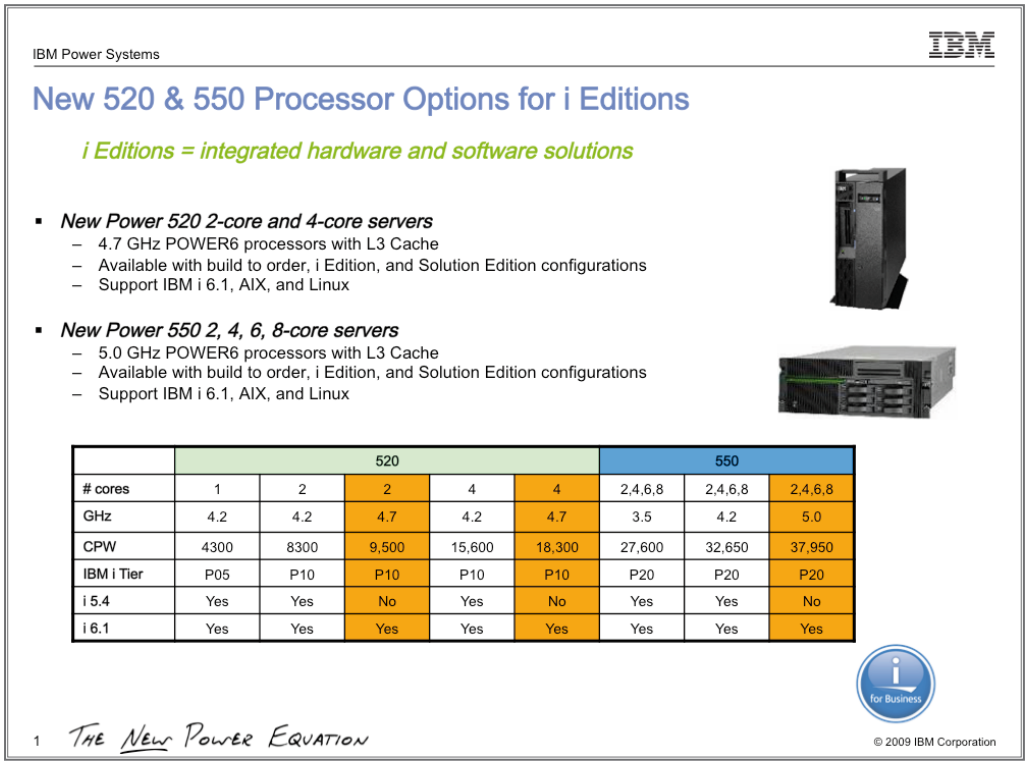
Mit POWER6+ steht ein neuer, leistungsstärkerer Mikro-Prozessor zur Verfügung. Neben den Performance-Verbesserungen unterstützt dieser Prozessor acht neue Memory-Keys, um virtualisierte Umgebungen noch sicherer und robuster betreiben zu können. POWER6+ hat jetzt insgesamt 16 Memory-Keys (8 Kernel, 7 User und 1 Hypervisor), während POWER6 lediglich 8 Memory-Keys (7 Kernel und 1 User) unterstützte. Die neuen POWER6+ Prozessoren kommen mit 4,2 GHz in den neuen BladeCentern JS23/JS43, den neuen 4,7 GHz Modellen der Power 520 und den neuen 5,0 GHz Modellen der Power 550 zum Einsatz.

Aber auch in den bereits im Oktober 2008 angekündigten 3,5 GHz Modellen der Power 560, den 4,4 GHz und 5,0 GHz Modellen der 16-Core Power 570 und den 4,2 GHz Modellen der 32-Core Power 570 war der POWER6+ Prozessor bereits enthalten. Da es sich bei POWER6+ nicht um eine neue Fertigungs-Technologie handelt, sondern eher um ein Design-Refresh, wurde dies in der Vergangenheit nicht explizit publiziert. Mit zunehmendem Einsatz von Live Partition Mobility (zurzeit nur für AIX und Linux verfügbar), also der Möglichkeit im laufenden Betrieb eine Partition von einem System auf ein anderes System zu transferieren, gewinnt die Unterscheidung zwischen POWER6 und POWER6+ aber an Bedeutung. POWER6+ stellt einen so genannten Kompatibilitäts-Modus zur Verfügung, um den Transfer der Partitionen zu ermöglichen, wenn die darunter liegende Hardware unterschiedlich ist.

Da das Betriebssystem IBM i aufgrund seiner Architektur von Anfang an ein sehr ausgefeiltes Memory-Key-Management beinhaltete, profitieren IBM i-Benutzer und -Applikationen von POWER6+ in erster Linie von den höheren Taktraten (Power 550) und dem zusätzlichen L3-Cache in den neuen BladeCentern JS23/JS43 und den neuen Power 520 Modellen.

Power 520 Express und Power 550 Express

Für die kleinen und mittleren Unternehmen wird die Produkt-Palette der Power 520 Express um zwei neue Modelle erweitert. Hierbei handelt es sich um eine 2-Core und 4-Core Variante mit POWER6+ Technologie, einer Taktrate von jeweils 4,2 GHz und zusätzlich 32 MB L3-Cache pro POWER6+ Chip (2-Cores). Die Performance-Steigerungen auf Basis von CPW (Commercial Processing Workload) betragen im Vergleich zu den bisherigen Modellen mit 4,2 GHz (ohne L3-Cache) bei dem 2-Core Modell etwa 14 Prozent und bei der 4-Core Variante circa 20 Prozent. Hinsichtlich der Produkt-Struktur bezüglich der i-Editionen ergeben sich keinerlei Änderungen; es können zusätzlich einfach die Modelle mit 4,7 GHz ausgewählt werden. Analog gilt für die Upgrade-Pfade von POWER5/5+ Servern (9406-520/9406-525) in Richtung Power 520 (8203-E4A) mit POWER6+, dass die Struktur auch hier unverändert bleibt und lediglich um die 4,7 GHz Option erweitert wird. Im Gegensatz zu den bisherigen Power 520 Modellen setzen die neuen Server mit POWER6+ zwingend IBM i 6.1 voraus. Da die neuen Modelle mit POWER6+ (siehe Bild unten) weiterhin in der Prozessor-Klasse P10 verbleiben, hat sich für die Unternehmen auch das Preis-/Leistungsverhältnis entsprechend verbessert.



	520					550		
# cores	1	2	2	4	4	2,4,6,8	2,4,6,8	2,4,6,8
GHz	4.2	4.2	4.7	4.2	4.7	3.5	4.2	5.0
CPW	4300	8300	9,500	15,600	18,300	27,600	32,650	37,950
IBM i Tier	P05	P10	P10	P10	P10	P20	P20	P20
i 5.4	Yes	Yes	No	Yes	No	Yes	Yes	No
i 6.1	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Neue Prozessor-Optionen für Power 520 und Power 550

1.14
Seite 4

Ähnlich wie die Power 520 Express wird auch die Power 550 Express als leistungsstarker Datenbank- und/oder Applikations-Server für mittlere Unternehmen um Modelle mit POWER6+ erweitert. Dabei handelt es sich um 2-, 4-, 6- oder 8-Core Varianten mit 5,0 GHz Taktrate. Die Performance-Verbesserungen (auf Basis von CPW) im Vergleich zu den bisherigen Modellen mit 4,2 GHz bewegen sich in der Größenordnung von circa 14 Prozent bis etwa 20 Prozent. Auch hier bleiben die Strukturen hinsichtlich der i-Editionen und der Upgrade-Pfade unverändert; es kommt lediglich die 5,0 GHz Variante hinzu. Die neuen Power 550 Express Modelle mit POWER6+ verbleiben in der Prozessor-Gruppe P20 und setzen ebenfalls zwingend IBM i 6.1 voraus.

Sowohl Power 520 als auch Power 550 bieten jetzt die Möglichkeit, an Stelle der bisherigen sechs 3,5 Zoll SAS-Platten im Central Electronic Complex (CEC) zukünftig acht 2,5 Zoll Small Form Factor (SFF) SAS-Platten einzusetzen: dies können entweder 2,5 Zoll HDD SAS Platten mit 69 GB oder die neu angekündigten 69 GB 2,5 Zoll Solid State Disk (SSD) sein.

Weitere Informationen zu Power 520 und Power 550 finden sich unter folgender URL :

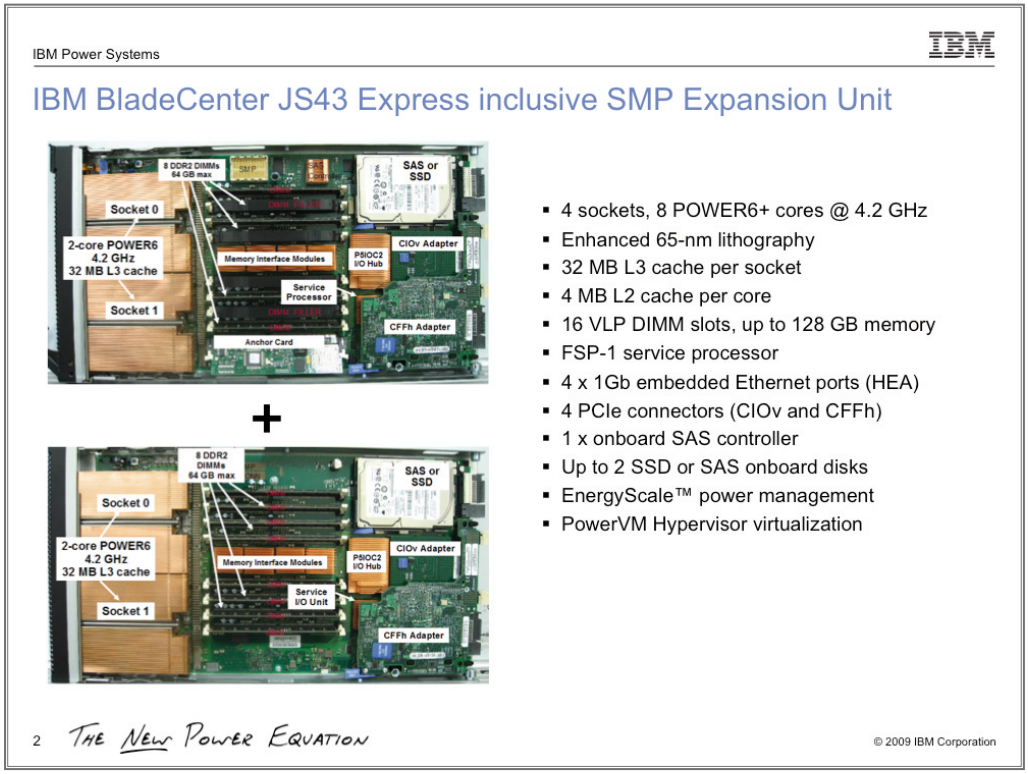
http://www-03.ibm.com/systems/power/hardware/

BladeCenter JS23 Express und JS 43 Express

Die BladeCenter Produkt-Familie, bekannt als einfach zu nutzende und integrierte Plattform mit einem hohen Maß an Flexibilität und Skalierbarkeit, wird um die neuen BladeCenter JS23 Express und JS43 Express mit POWER6+ Prozessor Technologie erweitert.

Bei dem BladeCenter JS23 handelt es sich ein Single-Wide 4-Core 64-bit Blade, mit 4,2 GHz POWER6+ Prozessoren, 32 MB L3-Cache pro Prozessor-Chip (2-Cores) und 8 Double-Data-Rate (DDR2) Memory DIMM-Steckplätzen; insgesamt stehen somit 64 GB Hauptspeicher zur Verfügung. Wahlweise kann zwischen einer 73 GB/146 GB/300 GB SFF SAS-Platte oder einer neuen 69 GB Solid State Disk (SSD) ausgewählt werden.

Technologisch und kaufmännisch kann durch das Hinzufügen einer so genannten SMP Expansion Unit (siehe Bild unten) ein BladeCenter JS23 Express in ein Double-Wide 8-Core 64-bit BladeCenter JS43 Express konvertiert werden – diese Flexibilität und Skalierbarkeit ist zurzeit einzigartig im Markt. Die SMP Expansion Unit stellt eine zweite 4-Core 4,2 GHz Prozessor-Karte, 8 weitere DIMM-Steckplätze und eine zweite SSF SAS- oder SSD-Platte zur Verfügung. Damit ist das BladeCenter JS43 Express mit seinen 8-Cores und insgesamt 128 GB Hauptspeicher hervorragend geeignet, Performance- und Hauptspeicher-intensive Workloads wie beispielsweise Virtualisierung, Infrastruktur-Konsolidierung, Datenbank- und/oder Transaktions-basierende Applikationen zu betreiben.



BladeCenter JS43 Express mit SMP Expansion Unit

Das BladeCenter JS23 mit einer Leistung von 14400 CPW (siehe Tabelle) und das BladeCenter JS 43 mit immerhin 24050 CPW (vor einigen Jahren noch dem High-End Bereich vorbehalten) werden in Verbindung mit IBM i 6.1 und VIOS 2.1.1 in den BladeCenter H und S Chassis unterstützt. Hinsichtlich der IBM i Lizensierung haben sich keinerlei Änderungen ergeben; das bedeutet: die IBM i Lizenzen pro Prozessor-Core verbleiben weiterhin in der günstigen Prozessor-Gruppe P10 und dazu kommt dann die entsprechende Anzahl an IBM i User-Lizenzen in Blöcken von jeweils 10 Benutzern. PowerVM Standard Edition wird für die beiden BladeCenter JS23/JS43 kostenlos mitgeliefert.

IBM Power Systems

IBM

IBM BladeCenter JS12, JS22, JS23 and JS43 Express

The broadest and most flexible range of POWER6 blades in the world!

NEW Q2 09

	BladeCenter JS12 Express	BladeCenter JS22 Express	BladeCenter JS23 Express	BladeCenter JS43 Express
Architecture	3.8 GHz POWER6 SCM 2-core (1 Socket x 2 Cores) Single Wide	4.0 GHz POWER6 SCM 4-core (2 Socket x 2 Cores) Single Wide	4.2GHz POWER6 DCM 4 core (2 Socket x 2 cores) Single Wide	4.2GHz POWER6 DCM 8-core (4 Socket x 2 cores) Double Wide
Memory	4GB to 64 GB DDR2 (Chipkill)	4GB to 32 GB DDR2 (Chipkill)	4GB to 64GB DDR2 (Chipkill)	8GB to 128GB DDR2 (Chipkill)
DASD / Bays	0-2 SAS disk (73 or 146 GB)	0-1 SAS disk (73 or 146 GB)	0-1 SAS disk (73,146 or 300GB) or 0-1 Solid State (69GB)	0- 2 SAS disk (73 ,146 or 300GB) or 0-2 Solid State (69GB)
Daughter Cards	Legacy, SFF or High speed DC	Legacy, SFF, or High speed DC	1 PCI-E CIOv Expansion Card 1 PCI-E CFFh High Speed Expansion Card	2 PCI-E CIOv Expansion Card 2 PCI-E CFFh High Speed Expansion Card
Integrated Features	Keyboard, Video and Mouse Dual Port 1Gb Ethernet SAS Controller USB	Keyboard, Video and Mouse Dual Port 1Gb Ethernet SAS Controller USB	Keyboard, Video and Mouse Dual Port 1Gb Ethernet SAS Controller USB	Keyboard, Video and Mouse Quad Port 1Gb Ethernet SAS Controller USB
Fibre Support	Yes (via BladeCenter)	Yes (via BladeCenter)	Yes (via BladeCenter)	Yes (via BladeCenter)
Redundant Power	Yes (via BladeCenter)	Yes (via BladeCenter)	Yes (via BladeCenter)	Yes (via BladeCenter)
Redundant Cooling	Yes (via BladeCenter)	Yes (via BladeCenter)	Yes (via BladeCenter)	Yes (via BladeCenter)
Service Processor	FSP1 (IPMI, SOL)	FSP1 (IPMI, SOL)	FSP1 (IPMI, SOL)	FSP1 (IPMI, SOL)
Virtualization	PowerVM Standard Edition built-in	PowerVM Standard Edition built-in	PowerVM Standard Edition built-in	PowerVM Standard Edition built-in
Systems Management	IBM Director and CSM IBM EnergyScale Technology	IBM Director and CSM IBM EnergyScale Technology	IBM Director and CSM IBM EnergyScale Technology	IBM Director and CSM IBM EnergyScale Technology
OS Support	AIX, i, Linux	AIX, i, Linux	AIX, i, Linux	AIX, i, Linux
rPerf	14.37	30.26	4W = 36.28	8W = 68.20
CPW	7100	13800	14,400	24,050
BladeCenter Chassis Support	BCE, BCH, BCHT, BCT, BCS	BCH, BCHT, BCS	BCH, BCHT, BCS	BCH, BCHT, BCS

Übersicht der BladeCenter Produkt-Familie

Generell steht jetzt mit IBM i 6.1, und speziellen PTFs, VIOS 2.1.1 und Firmware eFW 3.4.2 für alle BladeCenter JS12/JS22/JS23/JS43 (im BladeCenter S oder H Chassis) und auch für alle POWER6/6+ basierenden Server der Power Systems Produktfamilie der lang erwartete Virtual Tape Support zur Verfügung. Damit kann jetzt eine IBM i 6.1 Client Partition mit Save/Restore-Befehlen oder via BMRS direkt auf ein SAS LTO-4 Tape (TS2240), welches von einem VIOS-Server verwaltet wird, zugreifen. Diese neue Möglichkeit vereinfacht die Sicherung und Wiederherstellung von Daten ganz wesentlich.

Neu ist auch die Unterstützung des SAS Raid Controller Switch Moduls in Verbindung mit POWER6/6+ basierenden Blades im BladeCenter S Chassis ; denn dies war bisher nur für x86-basierende Blades verfügbar. Der Raid-Kontroller selbst unterstützt unter anderem Raid 0, 1, 5 & 10, zwei Storage-Module mit maximal 12 SAS-Platten und die Anbindung der SAS LTO-4 (TS2240) Bandeinheit. Es müssen aus Gründen der Redundanz grundsätzlich zwei dieser SAS Raid Controller Switch Module installiert werden. Für den Betrieb sind IBM i 6.1 und VIOS 2.1.1 erforderlich; weiterhin wird aus Performance-Gründen empfohlen, für IBM i Client Partitionen separate Raid-Sets zu implementieren. Zu beachten ist, dass in Verbindung mit dem SAS Raid Controller Switch Modul kein IBM DS3200 Storage Subsystem unterstützt wird und IBM i 6.1 auch nicht pre-installed bestellt werden kann, da der IBM im Vorfeld ja nicht bekannt ist, wie die Raid-Sets seitens des Kunden aufgebaut werden sollen.

Die i Edition Express for BladeCenter S als besonders preisgünstige Einstiegs-Variante in die BladeCenter Produktlinie partizipiert ebenfalls von den beschriebenen funktionalen Erweiterung wie Virtual Tape Support und den erweiterten Sicherheits-Möglichkeiten durch das SAS Raid Controller Switch Modul.

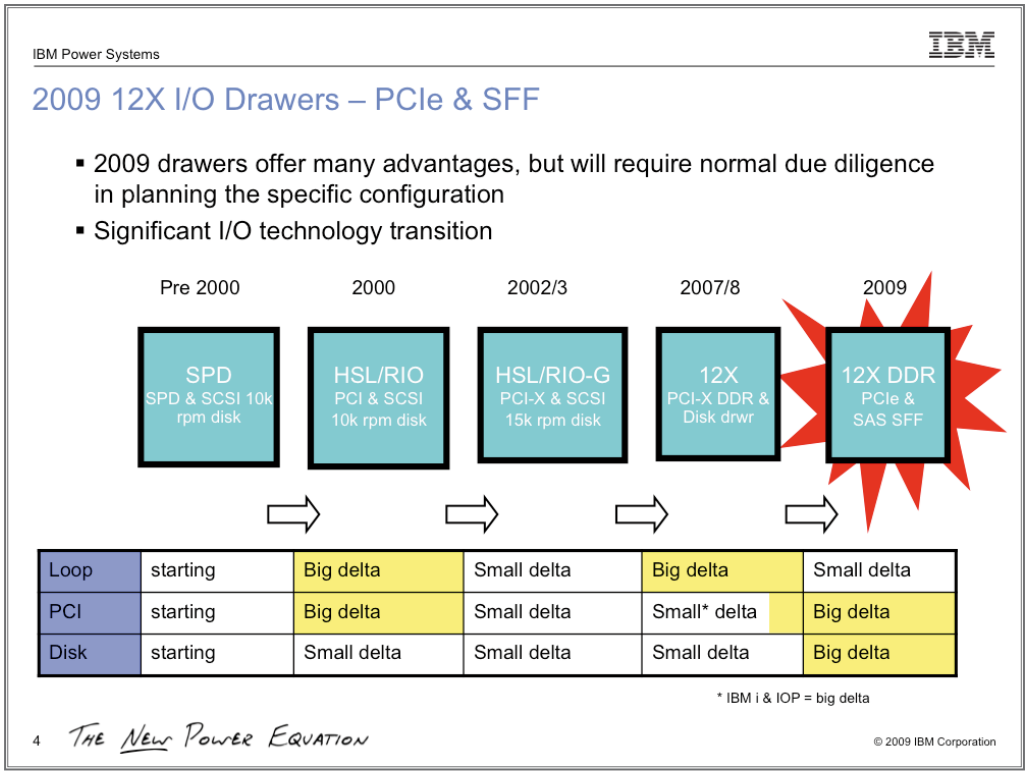
Mehr Informationen zu den BladeCentern finden sich unter folgender URL :

http://www-03.ibm.com/systems/power/hardware/blades/index.html

1.14
Seite 8

I/O-Erweiterungen für POWER6/6+ und POWER5/5+

Mit den Ankündigungen vom 28. April 2009 hat IBM auch eine neue Ära in der I/O-Struktur eingeleitet; denn mit Wirkung zum 28. August 2009 wird die letzte verbliebene 146 GB SCSI-Platte und damit verbunden auch diverse SCSI I/O-Adapter und HSL-basierende I/O-Tower (beispielsweise auch #0595) vom Vertrieb zurück gezogen.



Beginn einer neuen Ära in der I/O-Struktur

Die Darstellung macht deutlich, dass zukünftig 12X Double-Data-Rate (DDR) I/O-Drawer mit PCIe-Adaptern und SAS-Platten eingesetzt werden müssen. Dazu kommen dann die neuen energieeffizienten Solid State Disks (SSD) und ein PCI-X 1,5 GB Cache Raid SAS Adapter, der sowohl von POWER6/6+ basierenden Systemen wie auch von den älteren Servern mit POWER5/5+ unterstützt wird. IBM gibt in diesem Zusammenhang auch den Planungshinweis, dass mit POWER7 keine HSL-basierenden I/O-Tower und damit auch keine Adapter (beispielsweise Twinax, Token Ring, SNA WAN/LAN, ältere Tape Adapter etc.) mehr unterstützt werden, die einen so genannten Input-Output-Prozessor (IOP) voraussetzen.

Weitere Planungsinformationen finden sich unter der folgenden URL:

http://www-947.ibm.com/systems/support/i/planning/upgrade/

PCIe I/O-Drawer und Adapter

Es werden zwei neue PCIe 12X I/O-Drawer (4 EIA) angekündigt, die ausschließlich Steckplätze für PCIe 8x I/O-Adapter und SFF SAS-Platten besitzen und eine Größe von 24-Zoll beziehungsweise 19-Zoll besitzen.

Der 24-Zoll PCIe 12X I/O-Drawer (#5903) ist speziell für die Power 595 gedacht, hat ausschließlich 20 PCIe 8x I/O Adapter-Steckplätze und kann bis zu 26 hot-swap SFF SAS-Platten (73 GB/146 GB für AIX/Linux und 69 GB für IBM i 6.1) aufnehmen. Die 26 SSF SAS-Platten können durch einen Hardware-Schalter in folgende Gruppen aufgeteilt werden: eine Gruppe mit 26 Platten, zwei Gruppen mit jeweils 13 Platten oder vier Gruppen mit jeweils sechs beziehungsweise sieben Platten; wobei IBM i 6.1 lediglich die Variante mit zwei Gruppen mit je 13 Platten unterstützt. Diese Plattengruppen werden dann von PCIe SAS-Kontrollern gesteuert, die sich in demselben I/O-Drawer befinden müssen. Diesen 24-Zoll I/O-Drawer gibt es aus Kostengründen oder weil lediglich PCIe Steckplätze benötigt werden auch noch als Drawer (#5973) ohne Steckplätze für die SSF SAS-Platten; also nur mit den 20 PCIe Steckplätzen. Aus Gründen der Performance und Redundanz können beide 24-Zoll I/O-Drawer logisch in zwei Hälften aufgeteilt werden, wobei dann jede Hälfte eine eigene 12X DDR Anbindung besitzt. Es werden pro 12X Loop maximal nur ein #5903 oder #5973 PCIe 12X I/O-Drawer unterstützt.

Der 19-Zoll PCIe 12X I/O-Drawer (#5902) besitzt Steckplätze für 10 PCIe 8x I/O-Adapter, 18 SSF SAS hot-swap Platten (73 GB/146 GB für AIX/Linux und 69 GB für IBM i 6.1) und wird von Power 520/550/560/570 unterstützt. In Verbindung hiermit gibt es ein Statement of Direction (SOD) seitens der IBM, dass es später in 2009 auch eine Variante ohne SFF SAS-Platten geben wird. Auch dieser 19-Zoll I/O-Drawer kann via Mode-Switch in vier Gruppen aufgeteilt werden, wobei IBM i 6.1 lediglich den Modus mit zwei Gruppen mit je 9 SAS-Platten unterstützt. Pro 12X Loop können maximal zwei #5902 I/O-Drawer eingebunden werden. Generell gilt, dass weder der 24-Zoll noch der 19-Zoll PCIe 12X I/O-Drawer mit 12X PCI-X DDR I/O-Drawern älterer Bauart (beispielsweise #5796) in der gleichen 12X Loop betrieben werden können.

1.14
Seite 10

Für diese neuen PCIe 12X I/O-Drawer stehen auch zwei neue SAS Platten-Kontroller zur Verfügung:

- PCIe Dual-x4 SAS Adapter (#5901)
- PCIe 380 MB Cache Dual –x4 3 Gb SAS Raid Adapter (#5903)

Bei dem #5901 PCIe SAS Adapter handelt es sich um einen Dual-Port Adapter zur Steuerung der SAS-Platten in den PCIe 12X I/O-Drawern und dieser wird unter anderem auch von IBM i 6.1 unterstützt. Da dieser Adapter keinen Write-Cache besitzt, sollte mit Verwendung dieses Adapters in einer i-Umgebung eine sorgfältige Analyse der I/O-Workload einhergehen.

Der #5903 PCIe 380 MB Cache SAS Raid Adapter wird zurzeit nur von AIX und Linux unterstützt; für IBM i 6.1 gibt es ein SOD für das zweite Halbjahr 2009. Dieser Dual-Port SAS Raid Controller besitzt einen im laufenden Betrieb austauschbaren, batterie-gepufferten 380 MB Write-Cache und muss aus Gründen der Redundanz immer paarweise eingesetzt werden.

Da es sich bei beiden SAS Adaptern um Dual-Port Varianten handelt, kann der jeweils andere Port dazu verwendet werden, eine weitere Gruppe von SAS-Platten in den PCIe 12X I/O-Drawern zu steuern oder aber um SAS-Platten in dem bereits bekannten EXP 12S I/O-Drawer (#5886) zu betreiben. Im Gegensatz zu dem #5901 PCIe SAS Adapter kann der #5903 PCIe 380 MB Cache SAS Raid Adapter zusätzlich auch die neuen 69 GB Solid State Disk (SSD) in einem EXP 12S I/O-Drawer oder im CEC der Power 560/570 steuern.

PCI-X 1, 5 GB Cache Raid SAS Adapter

Mit dem PCI-X 1,5 GB Cache Raid SAS Adapter steht in 2009 der leistungsstärkste SAS Platten-Kontroller (circa 35 Prozent mehr Durchsatz im Vergleich zu dem bekannten SCSI 1,5 GB Cache Adapter) für Power Systems Server zur Verfügung. Dieser Adapter kann SAS – und/oder SSD-Platten betreiben, besitzt einen 1,5 GB Write Cache, der wiederum durch einen gleich großen Auxiliary Write-Cache geschützt ist (deshalb werden auch zwei PCI-Steckplätze belegt) und hat einen 1,6 GB Read-Cache. Weiterhin verfügt dieser Adapter über drei SAS-Ports, um SAS-Platten in angehängten EXP 12S I/O-Drawern und /oder im CEC von Power 520/550/560/570 zu steuern. Maximal unterstützt dieser Adapter für POWER6/6+ basierende Systeme 60 SAS-Platten in Verbindung mit kaskadierenden EXP 12S I/O-Drawern; abhängig von der I/O-Workload sind jedoch 24-30 SAS-Platten in der Praxis eine realistische Größenordnung. In Verbindung mit SSD-Platten können aufgrund der enormen Leistungsfähigkeit dieser Platten in einem EXP 12S I/O-Drawer aber nur 8 SSD-Platten pro Adapter angeschlossen werden.

Der PCI-X 1,5 GB Cache Raid SAS Adapter benötigt im Minimum IBM i 5.4.5 (eventuell benötigen aber andere HW-Komponenten oder Funktionalitäten ihrerseits IBM i 6.1) und kann in die PCI-X Slots der CECs von Power 520/550/560/570 eingebaut werden – also in 8203-E4A, 8204-E8A, 8234-EMA und 9117-MMA. Nicht unterstützt sind die bisherigen Modelle 9406-MMA, 9407-M15, 9408-M25 und 9409-M50; denn diese müssen vorher in die Modelle 9117-MMA, 8203-E4A beziehungsweise 8204-E8A konvertiert werden. Darüber hinaus kann dieser Adapter in folgenden I/O-Drawern (in Klammern jeweils die maximale Anzahl) betrieben werden:

– 12X I/O-Drawer : #5796 (2), #5797/5798 (8)
– HSL I/O-Drawer : #0595/5095 (2), #5094/5096 (1), #5294/5296 (2), #5790 (2)

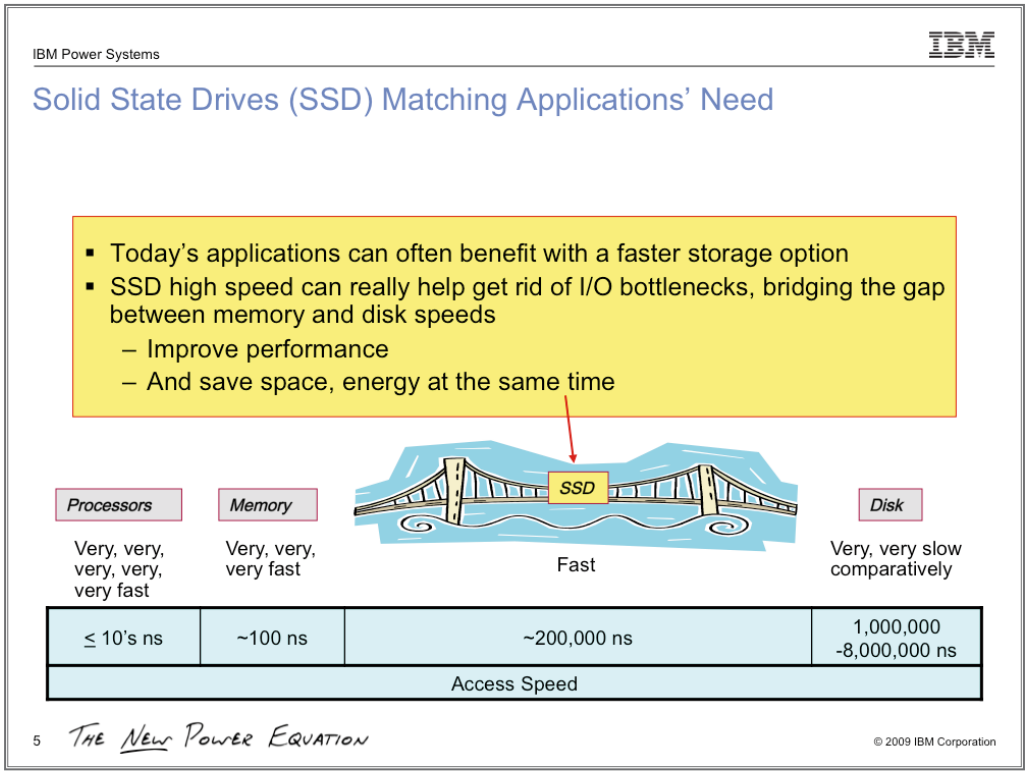
Mit Wirkung vom 17.Juli 2009 steht dieser PCI-X 1,5 GB Cache Raid SAS Adapter auch für die POWER5/5+ basierenden Systeme zur Verfügung, wobei sich die SAS-Platten (140 GB, 283 GB, 428 GB) dann in den EXP 12S I/O-Drawern befinden. Wichtig zu beachten ist in diesem Zusammenhang jedoch, dass dieser Adapter nicht im CEC der POWER5/5+ Systeme erlaubt ist und keine IBM i Load Source Funktionalität mit SAS-Platten, kein SAS-Tape, keine SSD-Platten und auch keine kaskadierenden EXP 12S I/O-Drawer (somit maximal 36 SAS-Platten) unterstützt.

19. Ergänzung 6/2009

1.14
Seite 12

Solid State Disk

Vergleicht man die Zugriffszeiten (siehe Bild) von Prozessor und Hauptspeicher einerseits und den der herkömmlichen Platten andererseits, so wird deutlich, dass hier dramatische Unterschiede bestehen.

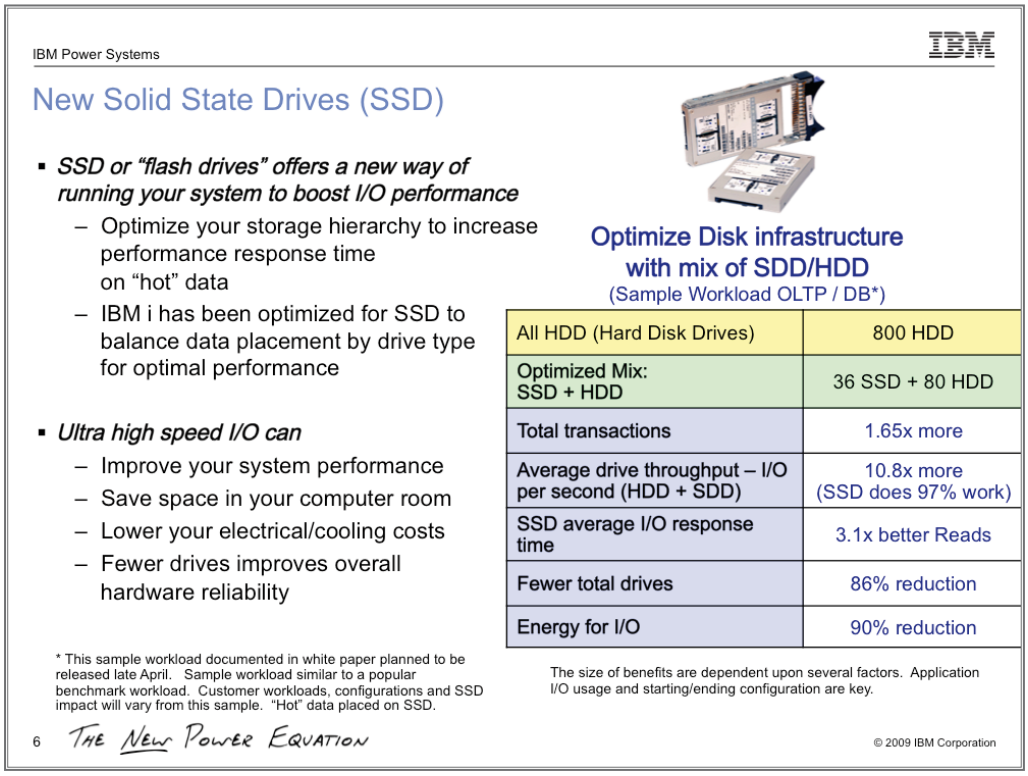


Positionierung von SSD-Platten hinsichtlich der Zugriffszeiten

Diese Lücke wird jetzt durch die neuen Solid State Disk (SSD) geschlossen. Es handelt sich hier um eine völlig neue Platten-Technologie mit einer Zugriffszeit von etwa 200 Mikro-Sekunden, die keine beweglichen Komponenten mehr besitzt, sehr zuverlässig ist, circa 50 Prozent weniger Energie benötigt und bis zu 28.000 I/O-Operationen pro Sekunde erlaubt.

In typischen Kunden-Umgebungen sind die Platten aus Performance-Gründen häufig nur zu etwa 30 bis 50 Prozent ausgelastet; während SSD-Platten aufgrund ihrer Architektur fast mit 100 Prozent Auslastung betrieben werden können. Schaut man sich die Plattenbelegung genauer an, dann stellt man fest, dass einige Daten sehr häufig (Hot-Data) angesprochen werden, andere Daten dagegen aber relativ selten (Cold-Data). Somit liegt es nahe, die Hot-Data auf die SSD-Platten und die Cold-Data auf die herkömmlichen Platten zu platzieren; wobei in dieser Konstellation für die Cold-Data in der Regel dann sogar Platten mit höherer Auslastung und/oder Kapazität verwendet werden können.

IBM i besitzt zurzeit als einziges Betriebssystem im Markt standardmäßig einen so genannten IBM i Load Balancer, eine Trace and Balance Funktionalität, die ein Monitoring der Daten ermöglicht, um die Hot- und Cold-Data zu bestimmen und diese dann automatisch auf die SSD- und HDD-Platten zu verteilen.



Optimierte I/O-Infrastruktur mit SSD-Platten

Das Beispiel-Szenario im Bild zeigt deutlich, welche dramatischen Leistungsverbesserungen (hinsichtlich I/O-Performance, Durchsatz, Energie-Bedarf etc.) durch den Einsatz von SSD-Platten erzielt werden können.

Grundsätzlich kann man sagen, der Einsatz von SSD-Platten bietet sich vor allem für I/O-intensive Workloads an, die einen hohen Anteil an Lese-Zugriffen haben und bei denen vor allem „Random" (nicht so sehr sequentiell) auf die Daten zugegriffen wird; da aber jede Kundensituation unterschiedlich ist, wird eine detaillierte Analyse der jeweiligen I/O-Workload empfohlen.

Die SSD-Platten bieten eine Kapazität von 69 GB und stehen in drei verschiedenen Einbau-Varianten zur Verfügung: für BladeCenter JS23/JS43, als SFF mit 2,5-Zoll und in der herkömmlichen Größe von 3,5-Zoll.

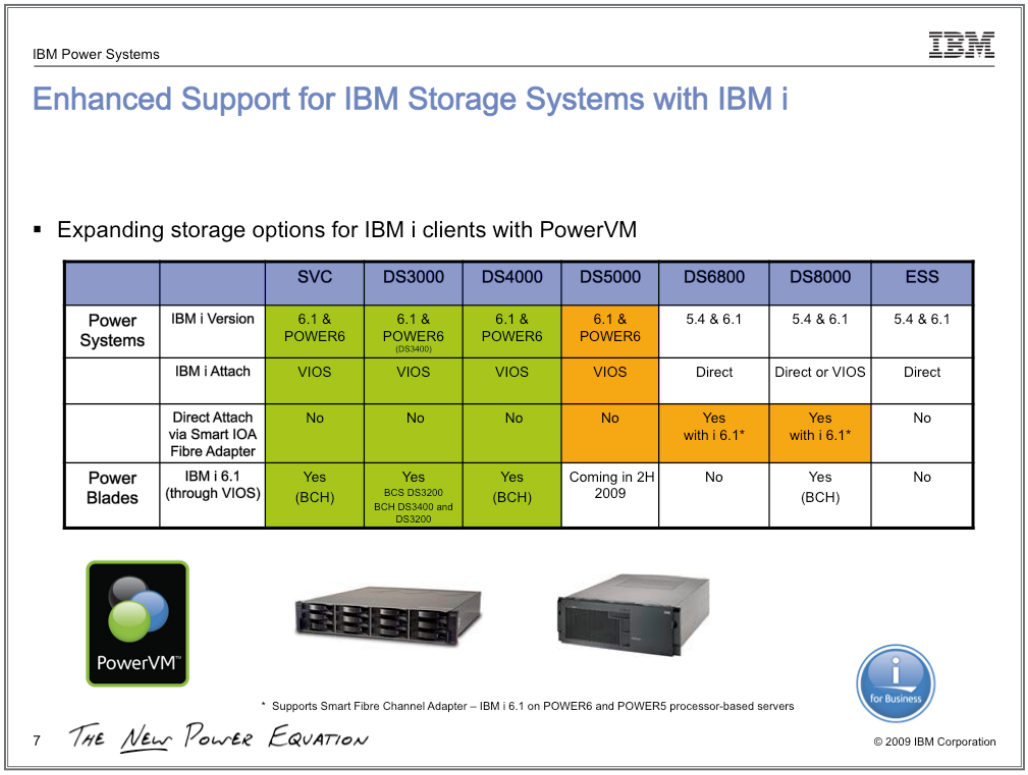
Die 3,5-Zoll SSD-Platte kann in einen EXP 12S I/O-Drawer oder den CEC einer Power560/570 eingebaut werden. Bei einer EXP 12S werden maximal acht SSD Platten und in den CECs der Power 560/570 bis zu sechs SSD-Platten unterstützt. Gesteuert werden die 3,5-Zoll SSD-Platten in der EXP 12S via PCI-X 1,5 GB Cache Raid SAS Adapter oder von einem Paar bestehend aus zwei PCIe 380 MB Cache SAS Raid Adaptern. Im CEC der Power 560/570 erfolgt die Steuerung durch den imbedded Controller (möglichst mit der 175 MB Cache Erweiterung #5673), einem Paar aus zwei PCIe 380 MB Cache SAS Raid Adaptern oder einem PCI-X 1,5 GB Cache Raid SAS Adapter.

In den CECs der Power 520/550 kommen die 2,5-Zoll SSF SSD-Platten zum Einsatz und die Kontrolle erfolgt hier via imbedded Controller (möglichst mit 175 MB Cache) oder einem PCI-X 1,5 GB Cache Raid SAS Adapter.

Die SSD-Platten werden im Minimum von IBM i 5.4.5 unterstützt; aber auch hier ist zu prüfen, ob nicht zusätzlich HW-Komponenten oder Funktionalitäten verwendet werden, die ihrerseits IBM i 6.1 benötigen.

IBM Systems Storage Erweiterungen

Bezüglich der Unterstützung von IBM Systems Storage durch IBM i haben sich mit diesen Ankündigungen ebenfalls zahlreiche Erweiterungen (siehe Bild) gegeben.



		SVC	DS3000	DS4000	DS5000	DS6800	DS8000	ESS
Power Systems	IBM i Version	6.1 & POWER6	6.1 & POWER6 (DS3400)	6.1 & POWER6	6.1 & POWER6	5.4 & 6.1	5.4 & 6.1	5.4 & 6.1
	IBM i Attach	VIOS	VIOS	VIOS	VIOS	Direct	Direct or VIOS	Direct
	Direct Attach via Smart IOA Fibre Adapter	No	No	No	No	Yes with i 6.1*	Yes with i 6.1*	No
Power Blades	IBM i 6.1 (through VIOS)	Yes (BCH)	Yes BCS DS3200 BCH DS3400 and DS3200	Yes (BCH)	Coming in 2H 2009	No	Yes (BCH)	No

Unterstützung von IBM Systems Storage durch IBM i

So können jetzt an POWER6/6+ basierende Power Server in Verbindung mit IBM i 6.1 und VIOS auch IBM Systems Storage DS5100 und DS5300 angebunden werden.

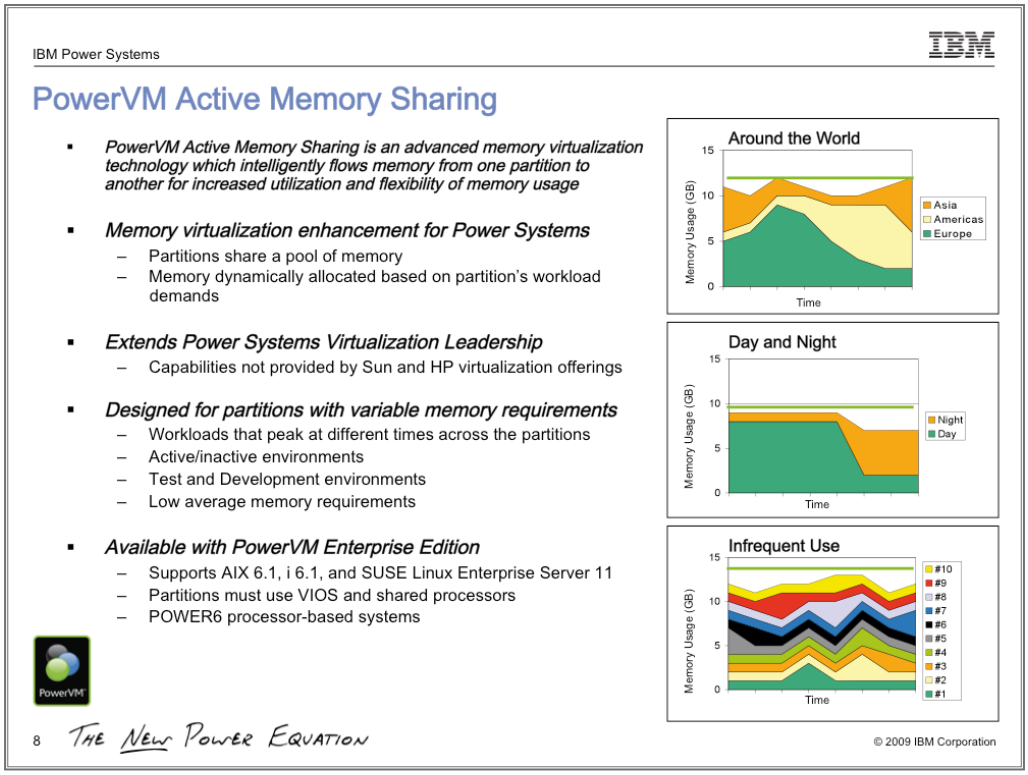
Weiterhin können die bereits bekannten 4 Gb Dual-Port PCI-X Fibre Channel Adapter (#5749) oder der 4 Gb Dual-Port PCIe Fibre Channel Adapter (#5774) in Verbindung mit IBM i 6.1 auf POWER6/6+ basierenden Servern dazu verwendet werden, IBM Systems Storage DS6800 anzubinden.

Bei POWER5/5+ basierenden Servern ist es jetzt auch möglich, IBM Systems Storage DS8100/DS8300 oder IBM Systems Storage DS6800 Subsysteme auf Basis von IBM i 6.1 und dem bekannten Smart Fibre Channel Adapter (#5749) zu betreiben. Es werden in dieser speziellen Konstellation jedoch keine Band-Laufwerke oder -Bibliotheken und auch keine SAN Load Source Funktionalität im Zusammenhang mit diesem Adapter unterstützt.

1.14
Seite 16

PowerVM Active Memory Sharing

PowerVM Active Memory Sharing (AMS) ist eine im Markt zurzeit einzigartige Hauptspeicher-Virtualisierungs-Technologie, die es erlaubt, Hauptspeicher auf intelligente Art und Weise von einer Partition in eine andere dynamisch zu verschieben; mit der Zielsetzung, Kosten zu sparen und den Hauptspeicher flexibler und effizienter zu nutzen.



PowerVM Active Memory Sharing

Basierend auf einem Hauptspeicher-Pool und abhängig von den Workload-Anforderungen in den Partitionen erfolgt jeweils eine dynamische Zuordnung des Hauptspeichers. Entwickelt wurde AMS vor allem für Partitionen mit variablen Hauptspeicher-Anforderungen, für Workloads, deren Spitzenbelastung zu unterschiedlichen Zeiten erfolgt, und für aktive/inaktive Umgebungen. Damit eignet sich AMS beispielsweise für Partitionen mit unterschiedlichen Zeit-Zonen, Tag- und Nacht-Workloads, Test- und Entwicklungs-Umgebungen oder wenn Hauptspeicher-Ressourcen sehr unregelmäßig benötigt werden.

IBM i 6.1 unterstützt PowerVM Active Memory Sharing auf POWER6/6+ basierenden Power Systems Servern und Power Blades. Voraussetzung ist, dass alle I/O-Adapter mit Hilfe von VIOS virtualisiert sind und shared Prozessoren verwendet werden; weiterhin sind VIOS 2.1.1, eFW 3.4.2 und HMC V7 3.4.2 erforderlich.

Dynamische Infrastruktur des 21. Jahrhunderts

Ross Mauri, General Manager, IBM Power Systems fasst die Ankündigungen vom 28. April 2009 kurz und knapp wie folgt zusammen:

„With today's announcements, we're helping our clients take the right steps today to ensure a more productive tomorrow. We're introducing leading Power Systems technology, products and services, with better performance, efficiency and reliability than ever before. Our goal is to help companies gain business insights and work smarter towards a more dynamic infrastructure."

Autor

Der Autor Peter Nimz ist seit über 30 Jahren in der IBM Deutschland GmbH in unterschiedlichsten Funktionen tätig und heute als Produkt-Manager für IBM Power Systems zuständig.

1.14
Seite 18

2 Kapitel

Systembedienung / Steuerung

2

Seite 2

Offen kommunizieren

Die traditionelle iSeries-Umgebung hat sich in den letzten Jahren stark verändert. Früher fanden wir eine iSeries (oder wie wir damals noch sagten: AS/400) relativ isoliert und sicher hinter den Glaswänden der Computer-Räume. Heute ist die Umgebung offener. Nicht selten finden wir neben der iSeries verschiedene Server mit den unterschiedlichsten Betriebssystemen – und alle Systeme kommunizieren miteinander.

Grundwerk 10/2004

2
Seite 4

iSeries-Basiswissen

In einer gemischten IT-Infrastruktur müssen Windows-Administratoren in Urlaubszeiten ihre iSeries-Kollegen vertreten und umgekehrt. Viele iSeries-Systeme werden heute aber noch mit traditionellen Methoden administriert und geschützt. Die Sicherheitskonzepte beschränken sich auf den Datenzugriff der lokalen Benutzer, und den Windows-Kollegen werden nur wenige Befehle an die Hand gegeben, um die Urlaubsvertretung zu überstehen. Durch die zunehmend komplexeren IT-Strukturen des Unternehmens muss aber auch der Zugriff auf eine iSeries neu überdacht werden. Und auch den Urlaubsvertretungen sollte es möglich sein, nicht nur Befehle abzutippen, sondern zu verstehen, warum sie einen Job starten oder einen Druckauftrag umleiten müssen.

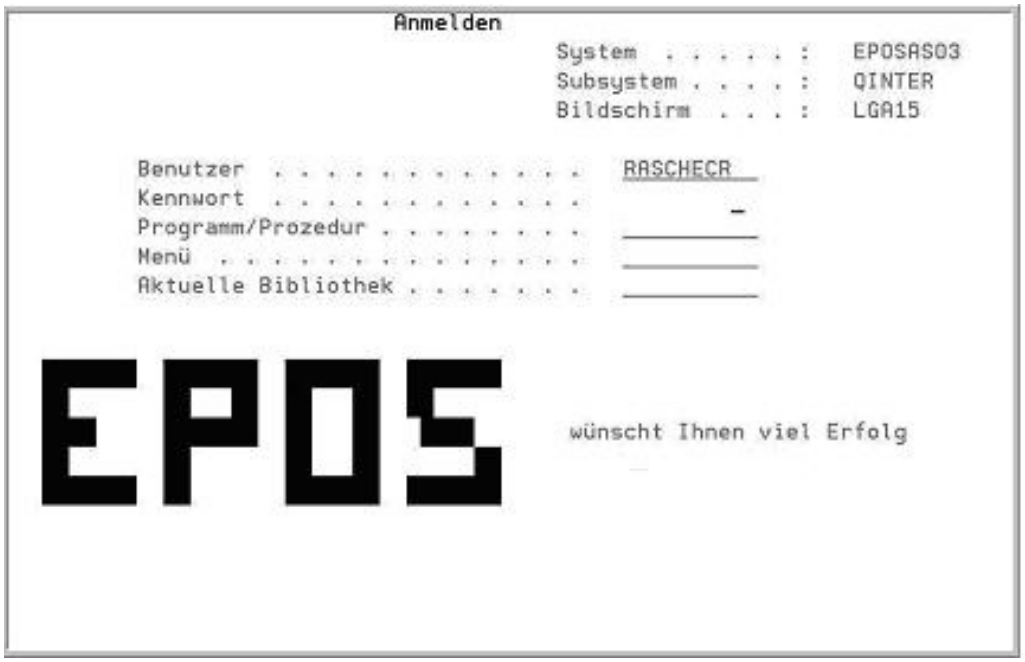
Aber beginnen wir der Reihe nach:

Grundwerk 10/2004

2.1
Seite 2

Anmeldung

Jeder Anwender, der mit einer iSeries arbeiten will, muss sich im System identifizieren. Die Anmeldung erfolgt über den Anmeldebildschirm. Dieser Bildschirm wird Signon-Bildschirm genannt.



Signon-Bildschirm

In dem Anmeldebildschirm finden Sie die folgenden Informationen:

System:

Jedes iSeries-System hat einen definierten Namen. Der Systemname dient der Identifizierung des Systems im Netzwerk und wird vom Administrator vergeben. Üblicherweise hat der Systemname einen Bezug zum Namen Ihres Unternehmens.

Subsystem:

Subsysteme sind logische Bereiche des Arbeitsspeichers. Normalerweise werden alle interaktiven Anmeldungen im Subsystem QINTER ausgeführt.

Bildschirm:

Im System iSeries kann Kommunikation auch zwischen Bildschirmen erfolgen. Dafür werden eindeutige Bildschirmnamen benötigt.

Benutzer:

Jeder Benutzer muss sich mit seinem Benutzernamen im System identifizieren. Für jeden Benutzer ist ein Benutzerprofil hinterlegt, das u. a. die Zugriffsberechtigungen enthält.

2.1.1
Seite 2

Kennwort:

Jedes Benutzerprofil hat ein Kennwort, das nur dem Benutzer bekannt sein sollte. Darum wird das Kennwort bei der Eingabe nicht angezeigt.

Der Benutzerprofilname und das Kennwort sind als Mindesteingaben beim Anmelden im System erforderlich.

Programm:

Hier können Sie den Namen eines Startprogramms eintragen, das ausgeführt wird, wenn Sie sich anmelden. Der Eintrag ist in den meisten Fällen nicht erforderlich, da die Programmausführung über das Benutzerprofil automatisiert werden kann.

Menü:

Das hier eingetragene Menü wird nach Ihrer Anmeldung angezeigt. Das Betriebssystem i5/OS der iSeries ist menügesteuert, d.h. im Grunde genommen brauchen Sie keinen Betriebssystembefehl kennen. Sie können die iSeries über die entsprechenden Menüpunkte steuern.

Aktuelle Bibliothek:

Alle Programme, Dateien und andere Objekte werden organisatorisch in Bibliotheken zusammengefasst und verwaltet. Aus der Sicht des Benutzers sind Bibliotheken Verzeichnisse für andere Objekte. Aus Systemsicht ist eine Bibliothek eine Tabelle mit verschiedenen Objekteinträgen. Die aktuelle Bibliothek (Current Library) ist die private Bibliothek eines Benutzers.

Struktur der Systembefehle

Für jeden Anwender, der nicht nur gelegentlich mit dem System arbeitet, ist das Erlernen der wichtigsten Systembefehle eine lohnende Investition, die zu einer effektiveren Arbeitsweise führt.

Jeder Betriebssystembefehl, der sogenannte CL- (Control Language-) Befehl, dient dem Aufruf einer oder mehrerer Systemfunktionen. CL-Befehle sind stets nach dem gleichen Muster aufgebaut. Sie setzen sich aus der Angabe der auszuführenden Tätigkeit (= dem Verb) zusammen sowie den zu übergebenden Parametern, die den Gegenstand der Tätigkeit näher bezeichnen. Somit hat jeder CL-Befehl den Aufbau: **Verb** + **Objekt**.

```
DSP – Anzeigen          WRK – Arbeiten mit
EDT – Editieren         CHG – Ändern
ADD – Hinzufügen        RMV – Entfernen
DLT – Löschen    STR – Starten
END – Beenden    CPY – Kopieren
SND – Senden
```

Das Objekt bezeichnet den Gegenstand, mit dem gearbeitet werden soll:

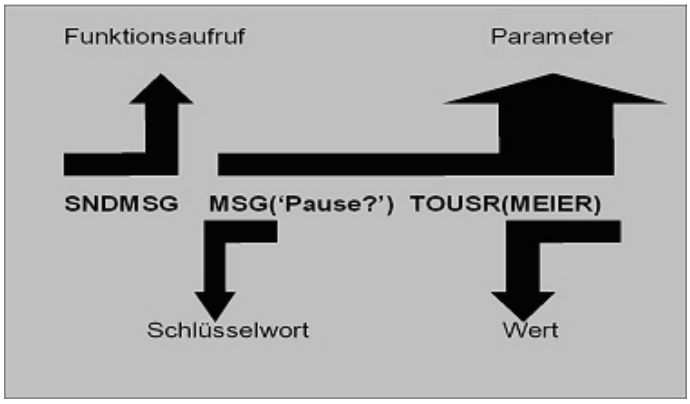
```
ACTJOB – Aktive Jobs              MSG – Nachricht
PGM – Programm                    USRPRF – Benutzerprofil
PRTF – Druckdatei                 SPLF – Spoolfiles
OUTQ – Ausgabewarteschlangen      JOBD – Jobbeschreibungen
```

Die Bezeichnung der Verben und Objekte sind aus dem Englischen hergeleitete Abkürzungen, die immer aus drei Buchstaben bestehen. So steht beispielsweise der CL-Befehl *SNDMSG* für den englischen Langtext „Send Message“. Dem Funktionsaufruf folgen die Schlüsselwörter, die den Funktionsumfang näher spezifizieren. Im oberen Beispiel folgen dem CL-Befehl *SNDMSG* zwei Parameter:

```
SNDMSG MSG('  ')        TOUSR(  )
```

Die Schlüsselwörter verlangen Parameterwerte, die in Klammern zugeordnet werden. Sind alle Parameter eines Befehls bekannt, können die Parameter auch als Positionsparameter übergeben werden. Der Unterschied zwischen Positions- und Schlüsselwortparameter besteht in der Einhaltung der Parameterreihenfolge. Schlüsselwortparameter können in beliebiger Reihenfolge nach dem Funktionsaufruf angegeben werden, während Positionsparameter die Einhaltung der festgelegten Reihenfolge verlangen.

2.1.1.1
Seite 2



Struktur eines Systembefehls

Menüs

2.1.2
Seite 1

Wurde die Anmeldung korrekt und erfolgreich durchgeführt, erscheint ein Anfangsmenü. Welches Menü erscheint, ist abhängig von den Einstellungen Ihres Benutzerprofils. Wenn keine spezifischen Änderungen vorgenommen wurden, erscheint das abgebildete iSeries-Hauptmenü:

```
MAIN                              OS/400-Hauptmenü
                                                        System:   EPOSAS03
Auswahlmöglichkeiten:

     1. Benutzeraufgaben
     2. Büroaufgaben
     3. Allgemeine Systemaufgaben
     4. Dateien, Bibliotheken und Ordner
     5. Programmierung
     6. Datenfernverarbeitung
     7. System definieren oder ändern
     8. Problembehandlung
     9. Menü anzeigen
    10. Unterstützende Informationen - Auswahlmöglichkeiten
    11. Client Access/400-Aufgaben

    90. Abmelden

Auswahl oder Befehl
===> _____________________________________________________________________

F3=Verlassen   F4=Bedienerführung   F9=Auffinden   F12=Abbrechen
F13=Unterstützende Informationen    F23=Anfangsmenü festlegen
(C) COPYRIGHT IBM CORP. 1980, 2002.
```

iSeries-Hauptmenü

Alle iSeries-Bildschirme bestehen gemäß der SAA-Benutzer-Schnittstellendefinition, einer von IBM erstellten Richtlinie zur einheitlichen Gestaltung von Benutzerschnittstellen, aus folgenden Bestandteilen:

Jeder Bildschirm hat eine Überschrift. Der Name des Menüs wird links außen am Bildschirm dargestellt. Der Auswahlbereich informiert über das Funktionsangebot des Menüs. In der Befehlszeile kann ein beliebiger Systembefehl eingegeben werden. Der Funktionsbereich im unteren Bildschirmdrittel informiert über die Funktionstastenbelegung. Sind mehr Funktionstasten aktivierbar, als angezeigt werden können, ermöglicht die F24-Taste ein Blättern in diesem Bereich. Auch heute stelle ich immer wieder fest, dass viele Anwender Probleme haben, auf einer PC-Tastatur die Tasten F11 bis F24 zu finden. Dabei ist die Umrechnung ganz einfach:

Denken Sie einfach an die Uhrzeit! 14 Uhr bedeutet: Es ist 2 Uhr. Wenn Sie also die Taste F14 suchen, benutzen Sie die Tastenkombination „shift F2“. Die Funktionstaste F16 (= 4 Uhr) können Sie durch die Tastenkombination „shift F4“ nutzen.

2.1.2
Seite 2

Aufgrund der komplexen Menüstruktur ist der Aufruf von Systemfunktionen mit Hilfe der Menüs häufig sehr zeitraubend und umständlich. Eine schnellere Möglichkeit des Menüaufrufs ist der Direktaufruf mittels FAST PATH: GO [Menüname]. Voraussetzung hierfür ist, dass Ihnen der jeweilige Menüname bekannt ist.

iSeries-Menüs selbst erstellen

Die iSeries verfügt über ein benutzerfreundliches Menüsystem, dessen Struktur das Auffinden sowie die Auswahl der Betriebssystemfunktionen erleichtert.

Hier ein kurzer Überblick wichtiger iSeries-Menüs:

AS/400 Operational Assistant Menü	GO ASSIST
Hardware und Netzwerkkonfiguration und Verwaltung	GO CFG
TCP/IP konfigurieren	GO CFGTCP
Bereinigungsaufgaben festlegen	GO CLEANUP
System definieren und ändern	GO DEFINE
Plattenspeicher überprüfen	GO DISKTASKS
Gerätestatus verwalten	GO DEVICESTS
Mit Dateien arbeiten	GO FILE
Dateien senden oder empfangen	GO FILETFR
Mit dem Integrated File System arbeiten	GO FILESYS
Mit Hardware-Ressourcen arbeiten	GO HARDWARE
Jobverwaltung	GO JOB
Bibliotheken verwalten	GO LIBRARY
Lizenzprogramme verwalten	GO LICPGM
iSeries Hauptmenü	GO MAIN
Befehle suche	GO MAJOR
Nachrichten verwalten	GO MESSAGE
Netzwerkkonfiguration	GO NETCFG
Netzwerkverwaltung	GO NETWORK
Das System ein- und ausschalten	GO POWER
Drucker verwalten	GO PRINTER
Daten zurücksichern	GO RESTORE
Daten sichern	GO SAVE
Sicherheits-Tools nutzen	GO SECTOOLS
Systemsicherheit verwalten	GO SECURITY
Systemstatus verwalten	GO STATUS
System-Menü	GO SYSTEM
Benutzeraufgaben	GO USER

Um die iSeries-Menüstruktur zu erweitern, steht den Benutzern mit SDA (Screen Design Aid) ein Werkzeug zur Verfügung, das es erlaubt individuelle Menüs zu erstellen.

2.1.2.1

Seite 2

Quellendateien

Zunächst müssen wir aber einige Vorbereitungen treffen. Die iSeries benötigt zur Erstellung eines Menüs eine Quellendatei, in der die später erstellten Sourcen gespeichert werden. Die Quellendatei erstellen Sie mit dem Befehl CRTSRCPF (= Create Source Physical File).

```
                    Phys. Quellendatei erstellen (CRTSRCPF)

 Auswahl eingeben und Eingabetaste drücken.

 Datei  . . . . . . . . . . . . .   qmenusrc__     Name
   Bibliothek . . . . . . . . . .     eps001____   Name, *CURLIB
 Datensatzlänge . . . . . . . . .   92____         Zahl
 Teildatei, falls erforderlich  .   *NONE_____     Name, *NONE, *FILE
 Text 'Beschreibung'  . . . . . .   Quellendatei für Menüs_____________________
_____

                                                                        Ende
 F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F10=Zusätzl. Parameter
 F12=Abbrechen   F13=Verwendung der Anzeige   F24=Weitere Tasten
```

Der Befehl CRTSRCPF

Der Parameter „Datei" (FILE)

Dieser Parameter gibt den Namen und die Bibliothek der zu erstellenden physischen Quellendatei an. Falls Sie keine Bibliothek angegeben, wird die physische Quellendatei in der aktuellen Jobbibliothek gespeichert. Quellendateien werden aber nicht nur für die Menüerstellung benötigt, sondern auch für CL- und RPG-Programme, für Datenbank-, Bildschirm- und Druckdatei-Beschreibungen. Da Sie Bibliotheken nicht hierarchisch strukturieren können, empfiehlt es sich, die Sourcen der verschiedenen Objekte über die Quellendateien zu organisieren. Die IBM schlägt die folgenden Namenskonventionen vor:

1.1
Seite 2

Quellendateiname	Verwendung
QMENUSRC	Menüs
QRPGSRC	RPG-Programme
QRPGLESRC	ILE RPG-Programme
QCLSRC	CL-Programme
QCLLESRC	ILE CL-Programme
QDDSSRC	Datenbank-, Druck- und Bildschirmdateien
QSQLSESS	SQL-Statements

Sie müssen sich an die Namenskonventionen der IBM nicht halten, sondern können beliebige Namen verwenden. Der Nachteil, den die Veränderung der Namen mit sich bringt, ist, dass die Erstellungsbefehle die Standardwerte annehmen. Sie können die Default-Einstellungen natürlich ändern, aber warum sollte man sich diese Mühe machen? Außerdem ist es für neue Mitarbeiter oder Vertretungen sehr viel einfacher, sich in ein System einzuarbeiten, dass die Default-Vorgaben verwendet.

Der Parameter „Satzlänge“

Der Parameter bestimmt die Länge der in der Quellendatei gespeicherten Sätze. Quellendateien sind Objekte der Art *FILE. Sie unterscheiden sich von den Datenbanken nur durch einen festen, einheitlichen Satzaufbau. Das Satzformat einer Quellendatei enthält drei Felder:

Feldbedeutung	Datentyp	Länge
Folgenummer	Numerisch	6 Stellen
Quellentext	Character	80 Stellen
Datum	Numerisch	6 Stellen

Für unser Menü beträgt die Satzlänge 92 Byte. Die Quellenfolgenummer und das Datum umfassen jeweils 6 Byte, die Quellenanweisung belegt 80 Byte.

Der Parameter „Teildatei“

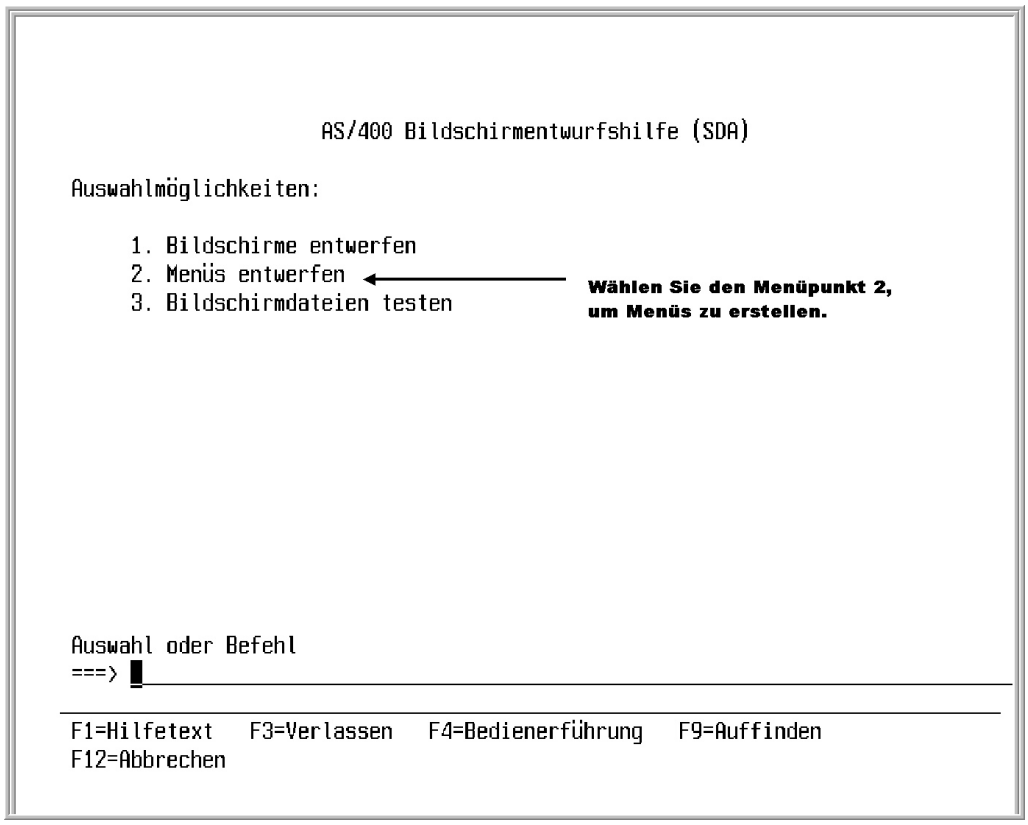
Die hier angegebene Quellenteildatei wird bei der Dateierstellung in der Quellendatei gespeichert. Da wir unser Menü mit SDA erstellen, brauchen wir zu diesem Zeitpunkt noch keine Teildatei zu benennen.

Der Parameter „Text“

Hier hinterlegen wir einen aussagekräftigen Text, der die Verwendung oder den Inhalt des Menüs kurz beschreibt.

SDA – Screen Design Aid

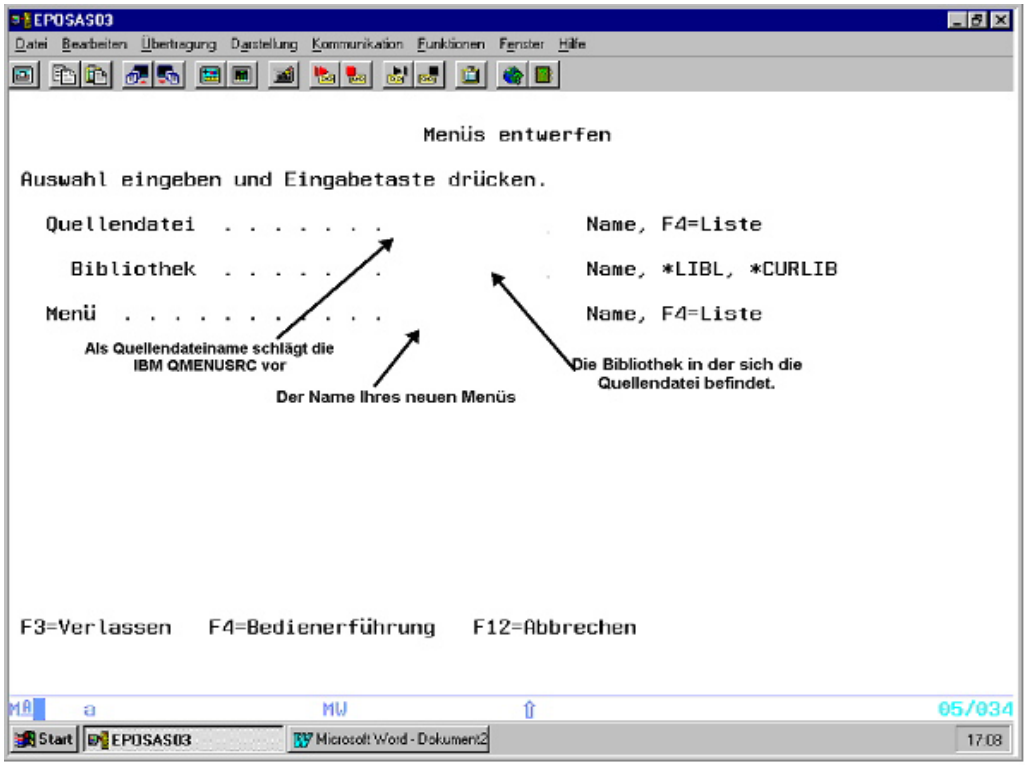
Rufen Sie jetzt mit dem Befehl STRSDA (Start Screen Design Aid) die Bildschirmentwurfshilfe der iSeries auf.



Die Bildschirmentwurfshilfe – SDA

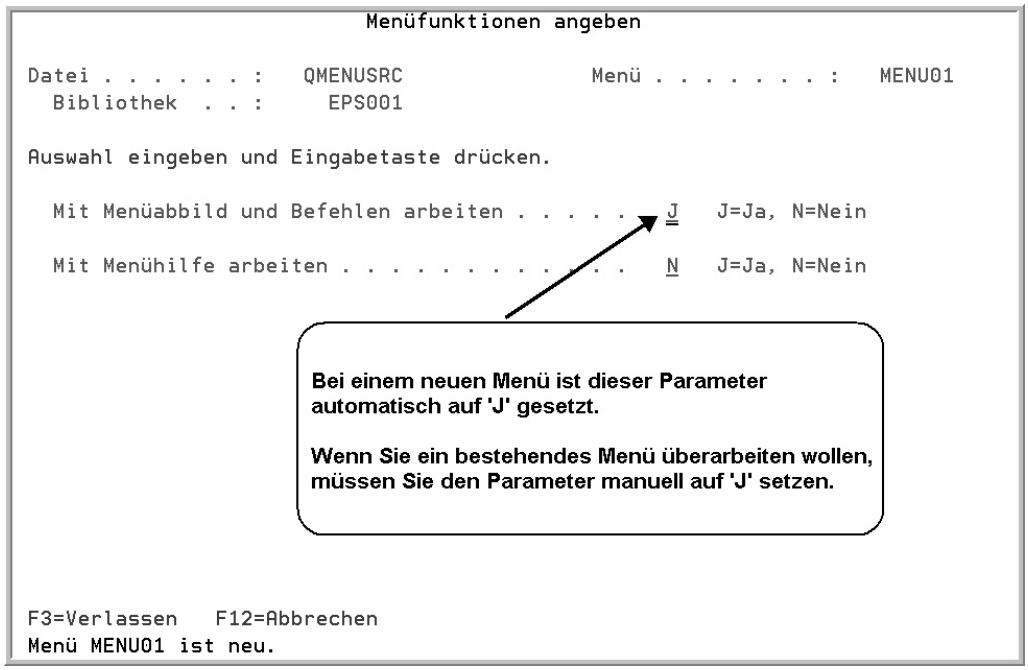
Da wir Menüs erstellen wollen, werden wir uns im Folgenden ausschließlich mit dem „Menüpunkt 2“ befassen. Tragen Sie in dem folgenden Bildschirm den Namen Ihrer Quellendatei, die Bibliothek und einen aussagekräftigen Namen für das Menü ein.

2.1.2.1.2
Seite 2



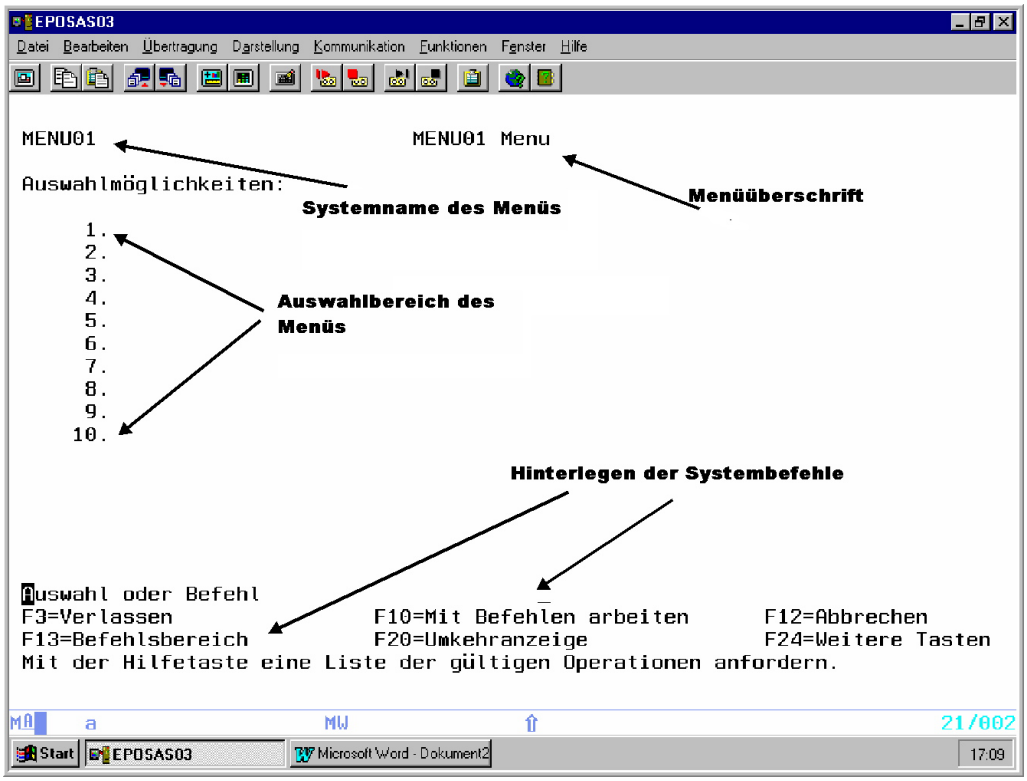
Menüs entwerfen

Wenn Sie Ihre Eingaben jetzt mit der Datenfreigabetaste bestätigen, sehen Sie den folgenden Bildschirm:



Menüfunktionen angeben

Nach der erneuten Freigabe, gelangen wir zur Arbeitsoberfläche für die Menüerstellung:



Arbeitsoberfläche für die Menüerstellung

Den Systemnamen des Menüs generiert SDA anhand Ihrer Eingaben. Unter diesem Namen kann das Menü nach der Fertigstellung mit dem Fast Path aufgerufen werden.

Die Menüüberschrift wird ebenfalls aus den Angaben generiert. Sie kann allerdings verändert und Ihren individuellen Anforderungen angepasst werden. Der Menüauswahlbereich hat standardmäßig 10 Menüpunkte. Sie können den Auswahlbereich aber individuell anpassen. Das iSeries-Menü kann maximal 99 Menüpunkte enthalten.

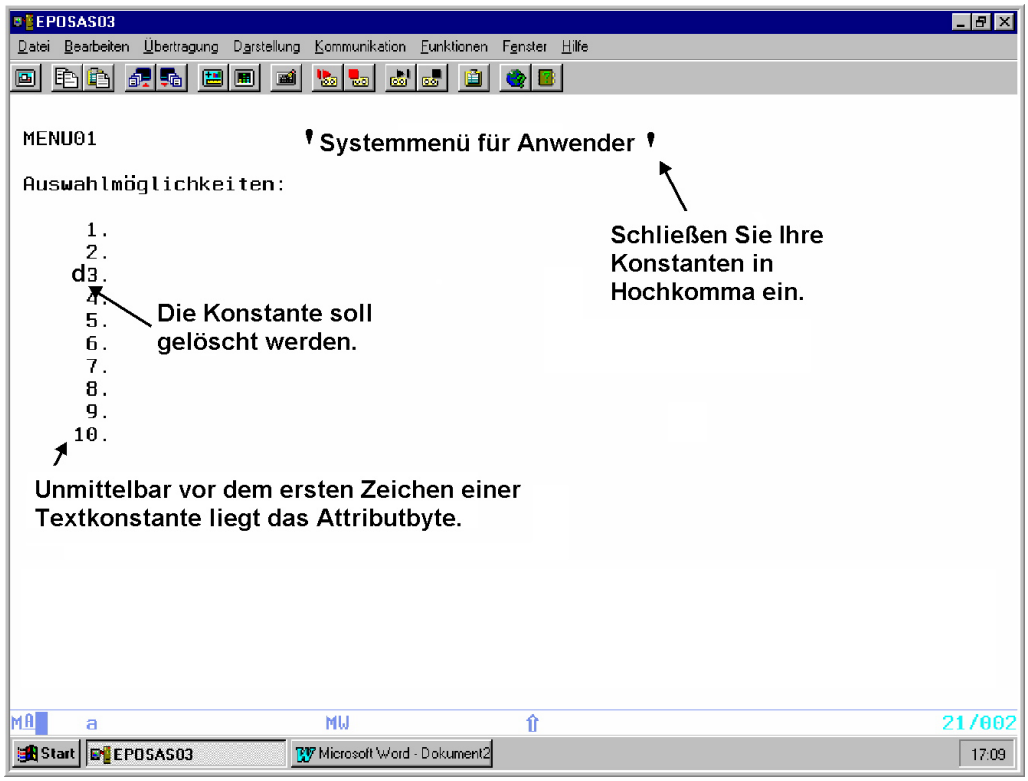
Wir wollen im Folgenden ein Menü aufbauen, das dem Benutzer erlaubt die eigenen Druckausgaben zu bearbeiten und eine beliebige Nachricht an den Systemoperator zu senden. Mit dem Menüpunkt 90 soll der Benutzer sich abmelden.

2.1.2.1.2
Seite 4

Textattribute setzen

Zunächst müssen die Textelemente editiert werden. Geben Sie für den Menüpunkt 1 den Text „Mit eigenen Druckausgaben arbeiten“ und für den Punkt 2 „Hilfe anfordern“ ein. Schließen Sie die Texte in Hochkomma ein, dadurch versteht SDA die einzelnen Wörter als zusammengehörige Konstante und Sie erleichtern sich die spätere Bearbeitung der Konstanten.

Löschen Sie anschließend die nicht genutzten Menüpunkte 3 bis 10. Verwenden Sie beim Löschen des vorhandenen Textes aber nicht die Löschtaste (=Entf). Sie können die bestehenden Konstanten entweder mit Leerzeichen überschreiben oder über die Attribut-Bytes manipulieren. Jede Konstante und jedes Feld verfügt im SDA über Attribut-Bytes. Das Attribut-Byte liegt normalerweise unmittelbar vor dem ersten Zeichen. Im Grunde genommen werden Ihre Hochkommas zu Attribut-Bytes. Um eine Konstante zu löschen, geben Sie einfach den Buchstaben „D“ im Attribut-Byte ein. Sie werden sehen, die Konstante wird gelöscht.



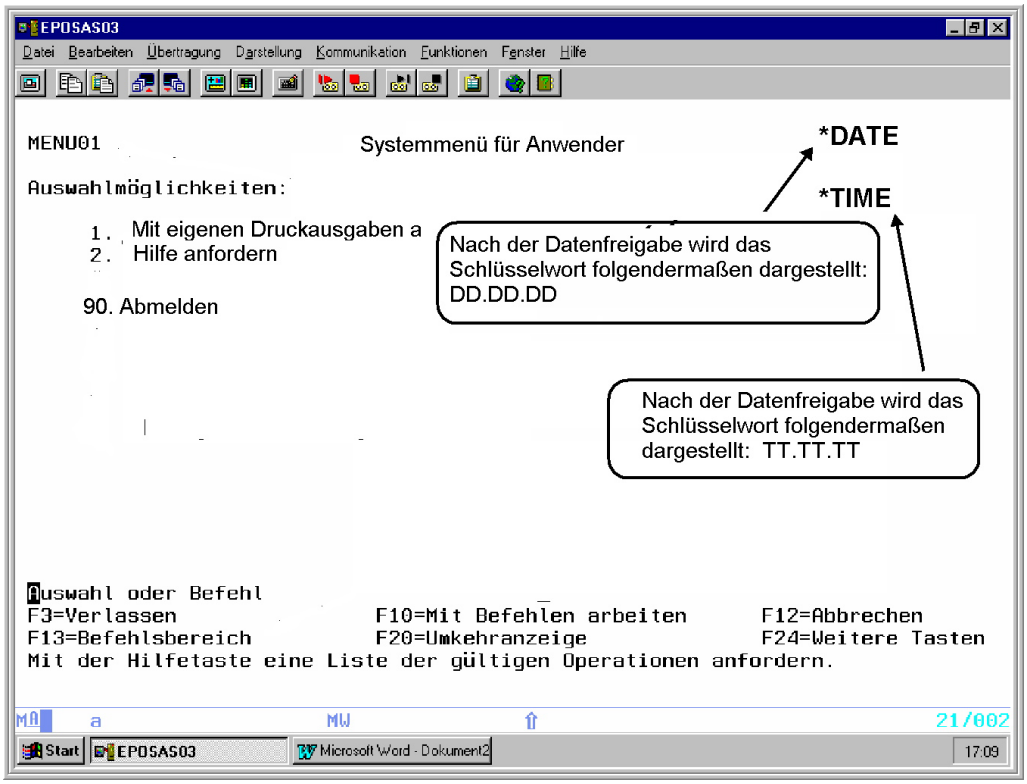
Konstanten editieren

Falls Sie Text in bestehende Konstanten einfügen wollen, dürfen Sie nicht die Einfügetaste (Einfg.) verwenden. Sie würden die Attribut-Bytes derart verschieben, dass SDA Fehler meldet und gegebenenfalls Teile des Bildschirms vollständig entfernt. Entfernen Sie in diesem Fall lieber den bestehenden Text und schreiben Sie ihn anschließend neu.

2.1.2.1.3
Seite 2

Für die Darstellung des Systemnamens, des Benutzernamens, des Datums und der Uhrzeit stellt SDA Ihnen entsprechende SDA-Schlüsselwörter zur Verfügung:

Funktion	Schlüsselwort
*DATE	Systemdatum
*TIME	Systemzeit
*USER	Benutzername
*SYSNAME	Systemname



Systemfunktionen nutzen

Einige Textkonstanten des Bildschirms sind bereits mit Attributen vorbelegt, dazu gehört z.B. die Menüüberschrift, die in Weiß dargestellt wird. Wenn Sie weitere Feldattribute verwenden wollen, positionieren Sie den Cursor vor das erste Zeichen der Textkonstanten – auf das Attribut-Byte – und geben das Zeichen '*' ein. Im darauf folgenden Bildschirm können Sie die definierten Feldattribute verändern oder neue definieren.

Im oberen Drittel des Bildschirms werden die Feldbeschreibungen der zu modifizierenden Konstante mit seiner Bildschirmposition und der Länge angezeigt. Die Angaben sind konstant. Hier eine kleine Anmerkung zu der Bildschirmposition Zeile 1, Spalte 1: Diese Position ist für SDA reserviert, d.h.: Sie können keine Konstante an dieser Position definieren.

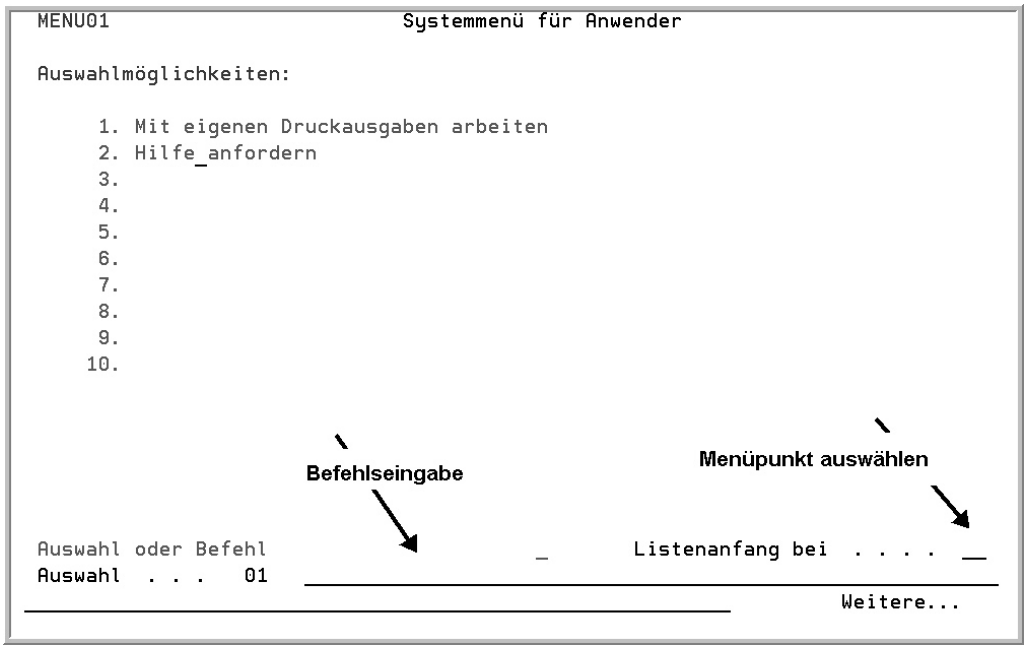
Im mittleren Teil des Bildschirms können Sie zwischen Intensivanzeige (= Darstellung des Feldes in Weiß), Umkehranzeige (= Invers-Darstellung des Feldes), Blinken und Unterstreichen wählen. Im unteren Bildschirmdrittel bestimmen Sie die Farbdarstellung Ihrer Konstanten.

Und noch eine Anmerkung: Falls Ihre Attribute später nicht korrekt angezeigt werden, liegt dies nicht an Ihren Einstellungen im Menü sondern wahrscheinlich an den emulierten 5250-Sitzungen. Die iSeries generiert für Ihre 5250-Sitzungen Bildschirmbeschreibungen. In den Beschreibungen wird eventuell ein Terminaltyp verwendet, der bestimmte Attribute nicht darstellen kann. Sie müssten somit Ihre Device-Beschreibungen manuell überarbeiten oder auf bestimmte Feldattribute einfach verzichten.

2.1.2.1.3
Seite 4

Systembefehle hinterlegen

Bisher haben wir nur die Bildschirmoberfläche gestaltet. Es besteht noch keine Verbindung zu den aufzurufenden CL-Befehlen. Mit der Funktionstaste F13 „Befehlsbereich“ oder der Taste F10 „Mit Befehlen arbeiten“ können Sie die entsprechenden Befehle zuordnen.



Systembefehle hinterlegen

Im unteren Bildschirmbereich sehen Sie die Eingabezeile für die Befehlseingabe. Für unseren ersten Menüpunkt müssen wir nur den Befehl WRKSPLF (=Work With Spoolfiles) eingeben. Sie können – wie auf der Systembefehlszeile – mit der Funktionstaste F4 die Bedienerhilfe für den Befehl anfordern. Für unseren zweiten Menüpunkt verwenden wir den Befehl SNDMSG (=Send Message). Da die Nachrichten an den Operator gesendet werden sollen, müssen Sie den Parameter TOUSR entsprechend abändern. Ein Problem scheint der Nachrichtentext zu sein, den Sie nicht konstant hinterlegen können. Der Nachrichtentext muss vom Benutzer eingegeben werden, wenn er das Menü ausführt. Das Problem lösen Sie durch die selektive Bedienerführung.

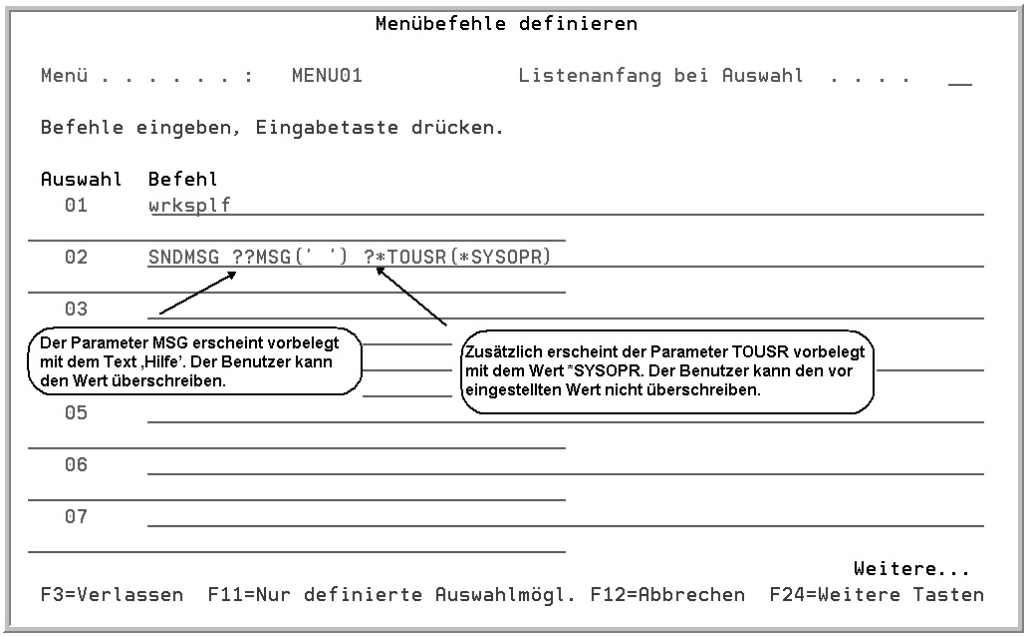
1. Ergänzung 12/2004

2.1.2.1.4
Seite 2

Selektive Bedienerführung

Selektive Bedienerführung ist eine Technik, die es ermöglicht, die Bedienerführungsanzeige aus einen Menü heraus zu aktivieren. Ein Menü kann also eine Anweisung enthalten, die die Bedienerführungsanzeige anzeigt und auf die Eingabe des Benutzers wartet. Sobald der Benutzer seine Eingabe bestätigt, wird der Befehl ausgeführt.

Beispiele für selektive Bedienerführung:

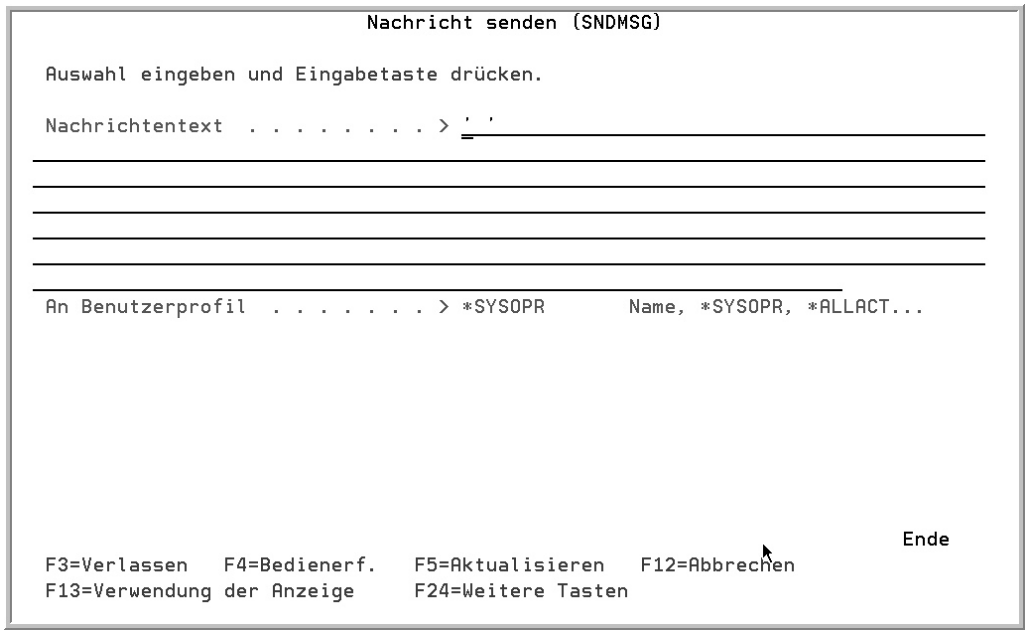


Selektive Bedienerführung

Wir können also in unserem Menü den Parameter MSG als Eingabefeld darstellen und zeigen den Parameter TOUSR als Ausgabefeld zur Information des Benutzers an. Wenn der Anwender das Menü später nutzt, wird er den folgenden Bildschirm erhalten.

2.1.2.1.5

Seite 2



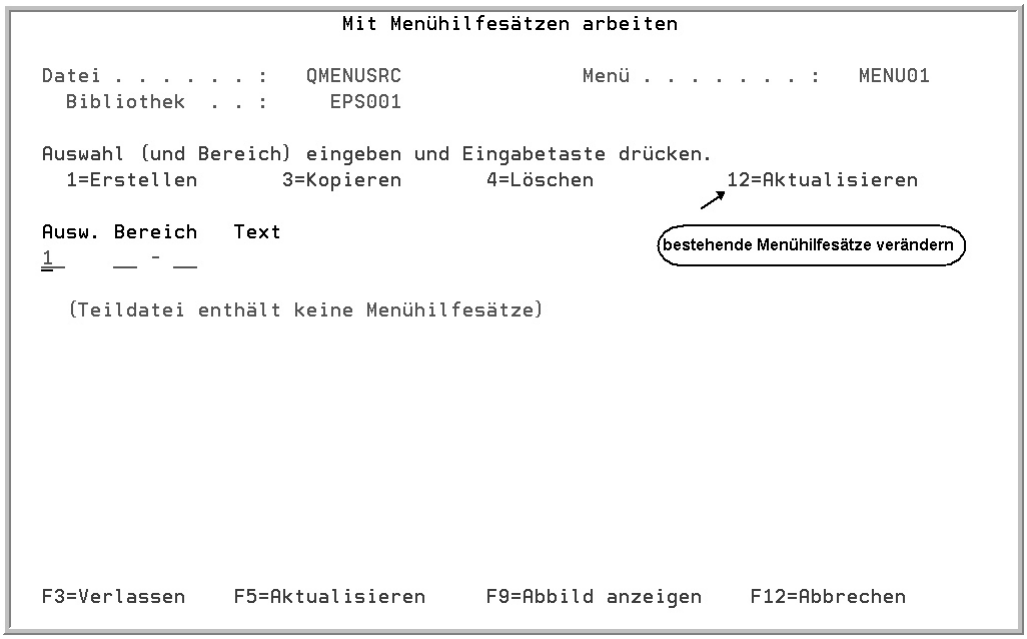
Wirkungsweise der selektiven Bedienerführung

Zum Schluss müssen wir nur noch den Befehl SIGNOFF für den Menüpunkt 90 hinterlegen. Damit wir nicht durch 90 Menüpunkte blättern müssen, legen wir den Listenanfang bei 90 fest und können anschließend den Befehl SIGNOFF eingeben.

Die Text- und Befehlseingabe ist jetzt abgeschlossen. Wenn wir das Menü mit der Funktionstaste F3 verlassen, erscheint wieder der Bildschirm „Menüfunktionen angeben“.

Hilfetexte hinterlegen

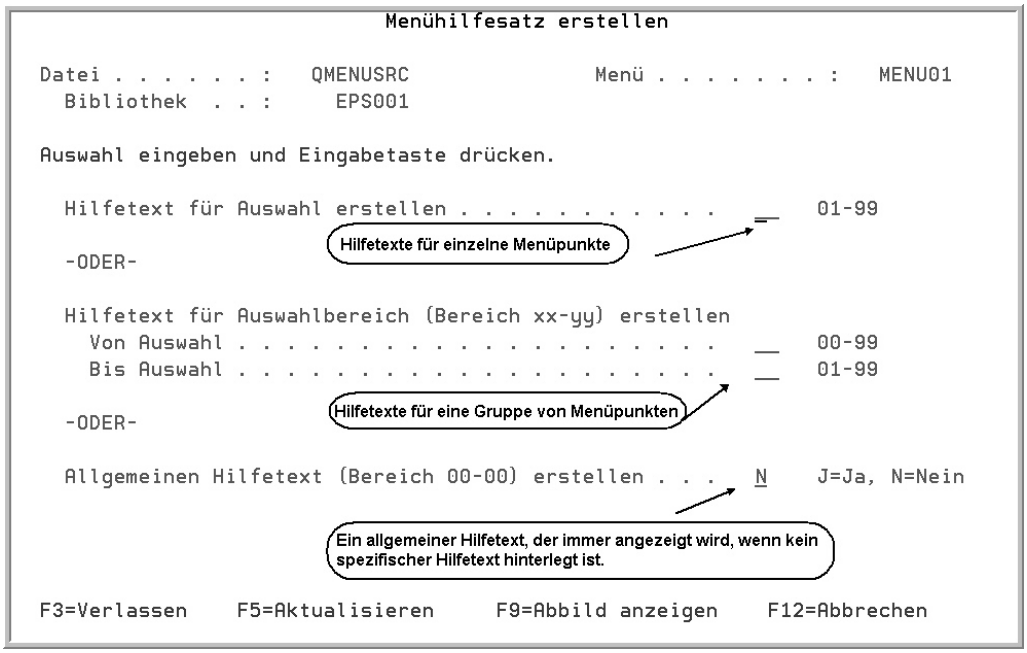
In einem selbst erstellten Menü besteht die Möglichkeit, Hilfetexte für den Anwender zu hinterlegen, die dieser dann bei Bedarf anfordern kann. Wenn Sie jetzt Ihre Hilfetexte hinterlegen wollen, müssen Sie den Parameter „Mit Menühilfe arbeiten“ in diesem Bildschirm auf „J“ setzen und gelangen daraufhin in den Bildschirm „Mit Menühilfesätzen arbeiten“.



Menühilfesätze erstellen und bearbeiten

Nachdem Sie die Auswahl 1 (=Erstellen) eingegeben haben, erscheint ein weiterer Bildschirm, in dem Sie bestimmen, für welche Bereiche der Hilfetext gültig sein soll. Als mögliche Gültigkeitsbereiche können Sie zwischen den folgenden Werten wählen:

2.1.2.1.6
Seite 2

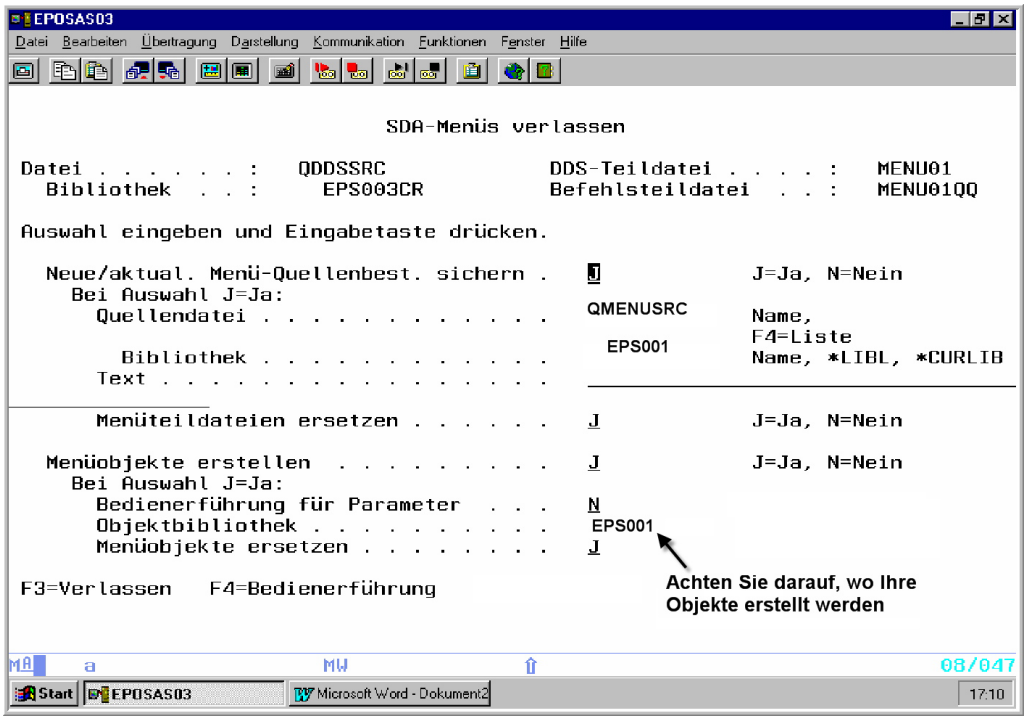


Menühilfesatz erstellen

Sobald Sie einen einzelnen Menüpunkt, einen Auswahlbereich oder den allgemeinen Hilfetext mit „J“ bestätigen, erhalten Sie einen weiteren Arbeitsbildschirm, in dem Sie Ihren Hilfetext hinterlegen können. Die Bearbeitung der Feldattribute Ihres konstanten Hilfetextes, das Löschen und Einfügen weiterer Konstanten erfolgt analog der bisherigen Bearbeitung konstanter Texte.

Die Menüobjekte erstellen

Unser Menü ist jetzt fertig – fast fertig, denn wir müssen noch das ausführbare Menüobjekt erstellen.

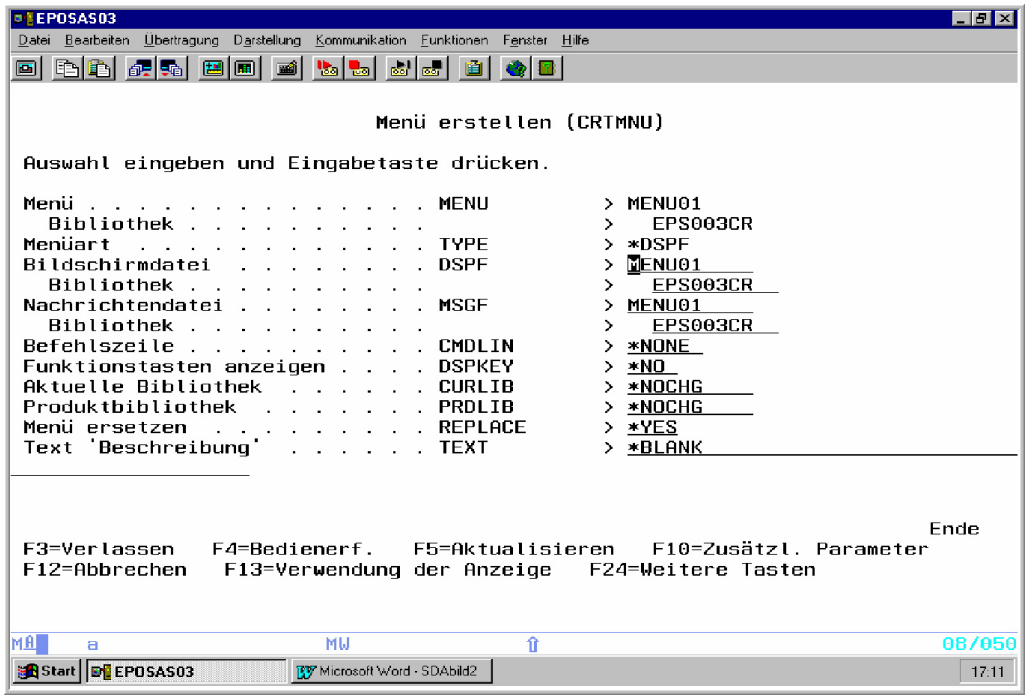


Menüobjekt erstellen

Das System sichert nach der Freigabe den zuvor erstellten Quelltext in zwei Quellenteildateien, die in Ihrer Quellendatei erstellt werden. Die erste Quellenteildatei enthält die Bildschirmbeschreibung. In einer zweiten Teildatei werden zur jeweiligen Auswahlnummer die auszuführenden Befehle hinterlegt. Außerdem generiert SDA das ausführbare Objekt. Korrekterweise erhalten Sie nicht ein, sondern drei Objekte, die alle benötigt werden, damit das Menü genutzt werden kann: *DSPF mit dem Menüabbild, *MSGF mit den Nachrichten und *MENU mit den Befehls- und Verwaltungsinformationen.

Wenn Sie den Parameter „Bedienerführung für Parameter“ auf „J“ setzen, können Sie weitere Umwandlungsparameter festgelegen.

2.1.2.1.7
Seite 2



Die Bedienerführung für Parameter

Der Parameter CMDLIN (Befehlszeile):

*NONE Das Menü erhält keine Befehlszeile.

*SHORT Die Befehlszeile ist einzeilig.

*LONG Die Befehlszeile wird zweizeilig dargestellt.

Der Parameter DSPKEY (Funktionstasten anzeigen):

*NO In dem Menü werden die gültigen Funktionstasten nicht dargestellt.

*YES In dem Menü werden die gültigen Funktionstasten angezeigt.

Der Parameter REPLACE (MENÜ ersetzen):

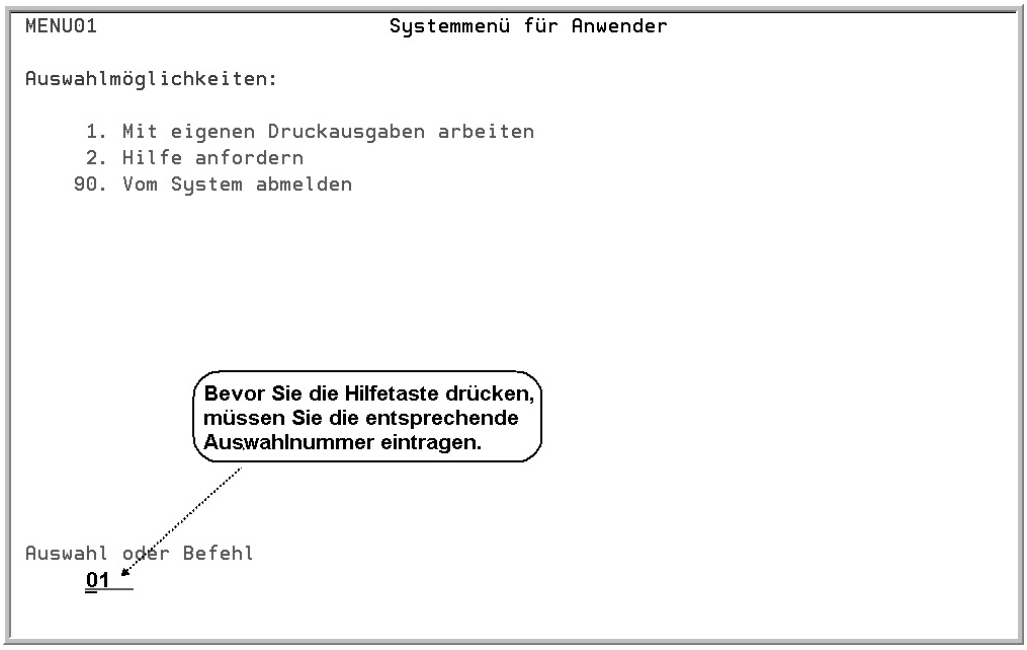
*NO Ein bestehendes Menüobjekt wird nicht ersetzt.

*YES Ein bestehendes Menüobjekt wird ersetzt.

Der Parameter TEXT (TEXT „Beschreibung“):

Ein frei wählbarer Text, der die Menüfunktion oder -verwendung beschreibt.

Zur Überprüfung, ob auch alles funktioniert, rufen Sie Ihr Menü mit dem Fast Path: GO <<menüname>> auf. Achten Sie hierbei auf die korrekte Adressierung des Menüobjektes (vgl. Kapitel 2.4.2.2 und 2.4.2.3). Falls alles korrekt ist, wird Ihr Menü angezeigt.



Das ausführbare Menü

Wenn Sie die punktuelle Hilfefunktion testen wollen, müssen Sie wissen, dass die Hilfefunktionen in den selbst erstellten Menüs nicht Cursor-gesteuert sind, d.h., der Anwender kann nicht den Cursor auf die einzelnen Menüpunkte positionieren und die Funktionstaste F1 nutzen. Stattdessen tragen Sie die entsprechende Auswahlnummer auf die Befehlszeile des Menüs ein und nutzen die Funktionstaste F1. In allen anderen Fällen erscheint stets der allgemeine Hilfetext.

2.1.2.1.7
Seite 4

Hilfefunktion

Mit der Steuersprache CL (=Control Language) können Sie die meisten iSeries-Funktionen steuern, indem Sie die Befehle über die Befehlszeile eingeben, in Programme einbauen oder die Befehle im iSeries Navigator verwenden. Das Menüsystem der iSeries und die Befehle der Steuersprache mögen anfangs vielleicht etwas ungewohnt sein, sie folgen aber einer einfachen Syntax, und OS/400 verfügt über viele Funktionen, die Sie bei der erfolgreichen Verwendung der Menüs und Befehle unterstützen.

Neue Benutzer sollten zunächst, die mit der iSeries ausgelieferte Basisliteratur lesen. Diese Bücher enthalten Informationen zu den Themen:

- Software-Installation
- Ausführung allgemeiner Bediener- und Administrator-Aufgaben
- Einführung in die Informationssuche
- Maßnahmen bei Fehlern

Während wir vor einigen Jahren noch bei jedem Release-Wechsel Unmengen an Handbücher ausgeliefert bekamen, folgte dann die Informationssuche mit der AS/400 Online-Bibliothek auf CD-ROM. Heute allerdings hat auch die IBM das Internet als Informationsquelle entdeckt und das Information Center zum Ausgangspunkt der Informationssuche erklärt. Viele der IBM-Handbücher wurden mittlerweile von der AS/400 Online-Bibliothek in das Information Center verlagert.

Wenn Sie allerdings bei der Ausführung einer Aufgabe auf dem System Hilfe benötigen, haben Sie wahrscheinlich nicht immer die Möglichkeit, das Information Center zu nutzen oder einen Experten um Rat zu fragen. Aus diesem Grund verfügt das System über ein umfangreiches Hilfesystem zu Anzeigen, Befehlen und Nachrichten.

1. Ergänzung 12/2004

2.1.4
Seite 2

iSeries Online-Hilfesystem

Stellen Sie sich vor, Sie verwenden eine unbekannte Anzeige, mit der Sie bisher nicht gearbeitet haben. Sie wissen nicht, welche Eingaben erforderlich sind, und keiner der Kollegen hat Zeit. Kennen Sie diese Situation? In diesen Fällen nutzen Sie die Online-Hilfe. Sie ist die am schnellsten verfügbare Informationsquelle zum System iSeries. Hilfetexte werden auf einer iSeries in der Regel mit der Funktionstaste F1 angefordert. Es sind allerdings – je nach verwendeter Tastaturbelegung – auch andere Belegungen denkbar. Die Auswahl der Hilfefunktionen erfolgt Cursor-gesteuert, d.h.: Der angezeigte Hilfetext ist von der Stellung des Positionsanzeigers abhängig. Für alle Bildschirmanzeigen stehen die folgenden Informationen zur Verfügung:

- Wofür wird die Anzeige verwendet?
- Wie wird die Anzeige verwendet?
- Wie werden die Eingabefelder und/oder Parameter verwendet?

Falls sich der Positionsanzeiger in einem Bereich befindet, für den spezifische Informationen verfügbar sind, werden die Informationen für diesen Bereich angezeigt. Wenn der Cursor z.B. auf einer Menüauswahl, auf ein Eingabe- oder ein Ausgabefeld positioniert wird, öffnet sich ein kontextbezogenes Bildschirmfenster. Positionieren Sie dagegen den Cursor auf den Funktionstastenbereich eines Bildschirms, erhalten Sie eine Erläuterung der in diesem Bildschirm aktiven Funktionstasten. Wird danach die Funktionstaste „F2“ gedrückt, erscheint ein Hilfetext für die gesamte Bildschirmanzeige.

```
                          Mit Objekten arbeiten
.....................................................................
:                    Mit Objekten arbeiten - Hilfetext              :
:  _                                                                :
:     In der Anzeige "Mit Objekten arbeiten" (WRKOBJ) ist eine Liste der   :
:       Objekte aufgeführt, für die der Benutzer über die  Berechtigung    :
:     zur Benutzung verfügt.  Mit dieser Liste können allgemeine Aufgaben  :
:     für die  Objektberechtigung durchgeführt werden.                     :
:                                                                          :
:     Die folgende Objektliste basiert auf dem für den Parameter OBJ       :
:     (Objekt) eingegebenen Wert. Wird beispielsweise T* und Bibliothek    :
:     QGPL eingegeben, wird eine Liste der Objekte in der Bibliothek QGPL  :
:     angezeigt, die mit T beginnen.                                       :
:                                                                          :
:     Die Auswahlnummer neben einem bestimmten Objekt eingeben und die     :
:     Eingabetaste drücken. Es können mehrere Aufgaben nacheinander        :
:     ausgeführt werden, indem Nummern gleichzeitig neben mehrere          :
:     Subsystembeschreibungen eingegeben werden.                           :
:                                                                          :
:                                                          Weitere ...     :
:  F3=Hilfetext verlassen    F10=Zum Anfang   F12=Abbrechen                :
:  F13=Unterstützende Info   F14=Hilfetext drucken                         :
:                                                                          :
:.........................................................................:
```

iSeries-Bedienerhilfebildschirm

2.1.4.1
Seite 2

In einigen Hilfebildschirmen sind bestimmte Wörter oder Ausdrücke gelb hervorgehoben. Hinter diesen Hypertext-Marken verstecken sich weitere Hilfetexte. Positionieren Sie den Cursor auf einer dieser Textmarken und betätigen Sie die Eingabetaste. Es erscheint der verborgene Hilfetext.

Wenn die Taste „F13“ („Unterstützende Informationen“) aktiv ist, kann darüber das Menü „Unterstützende Informationen – Auswahlmöglichkeiten“ aufgerufen werden. Mit den Informationen dieses Menüs können Sie auf die verschiedenen auf dem System verfügbaren Arten von Informationen zugreifen.

```
Hilfetext          Wo kann nach Informationen gesucht werden?

    Für das System IBM AS/400* stehen Informationen in unterschiedlicher
    Form zur Verfügung. Den Positionsanzeiger unter ein Stichwort
    stellen und die Eingabetaste drücken:
     o  Online-Bücher (AS/400 Softcopy-Bibliothek)
           Was ist eine Softcopy?
           Wie werden Softcopy-Bücher verwendet?
           Aufbau der Bibliothek (Softcopy-Bücherfächer)
     o  Gedruckte AS/400-Handbücher (AS/400-Systembibliothek)
           Erste Schritte (Getting Started)
           Bücher für das Lizenzprogramm OS/400
           Weitere Dokumentationen ("Redbooks" und zugehörige
          Informationen)
     o  Online-Informationen
           Online-Informationen im Internet
           Online-Hilfetext
           Online-Hilfsprogramme und Beispiele (Bibliothek QUSRTOOL)
           Online-Ausbildung
           Systemprobleme beheben
                                                              Weitere ...
F3=Verlassen   F10=Zum Anfang   F11=Suchindex   F12=Abbrechen
F13=Unterstützende Info         F14=Hilfetext drucken
```

Das Menü unterstützende Informationen

Wenn Sie den Hilfetext für das aktuelle Stichwort drucken wollen, steht Ihnen diese Funktion generell über die Funktionstaste „F14“ im erweiterten Hilfetext zur Verfügung.

Wussten Sie, dass Sie Internetadressen, die in der Onlinehilfe genannt werden, direkt aus Ihrem Green Screen heraus aufrufen können? Voraussetzung ist, dass Sie mit Client Access eine 5250-Terminalemulation geöffnet haben und Ihr PC über eine Verbindung zum Internet verfügt. Außerdem müssen Sie folgende Client Access Einstellungen wählen:

Online-Information im Internet

Ein Doppelklick auf die URL öffnet Ihnen jetzt die entsprechende Internetseite.

2.1.4.1

Seite 4

Control Language-Befehle suchen

Was aber können Sie tun, wenn Ihnen der zu verwendende Befehl gar nicht bekannt ist oder wenn Sie den Befehl zwar kennen, aber die einzugebenden Parameter unklar sind?

Falls Ihnen Teile des Befehlsnamens bekannt sind, ist es möglich generisch nach dem Befehl zu suchen. Geben Sie einfach die Anfangsbuchstaben des Befehls ein. Für die fehlenden Zeichen verwenden Sie ein '*' als Wildcard. Leider ist es nur möglich, das Ende eines CL-Befehls zu markieren.

```
Auswahl oder Befehl
===> dspusr*
F3=Verlassen   F4=Bedienerführung   F9=Auffinden   F12=Abbrechen
F13=Unterstützende Informationen    F23=Anfangsmenü festlegen
```

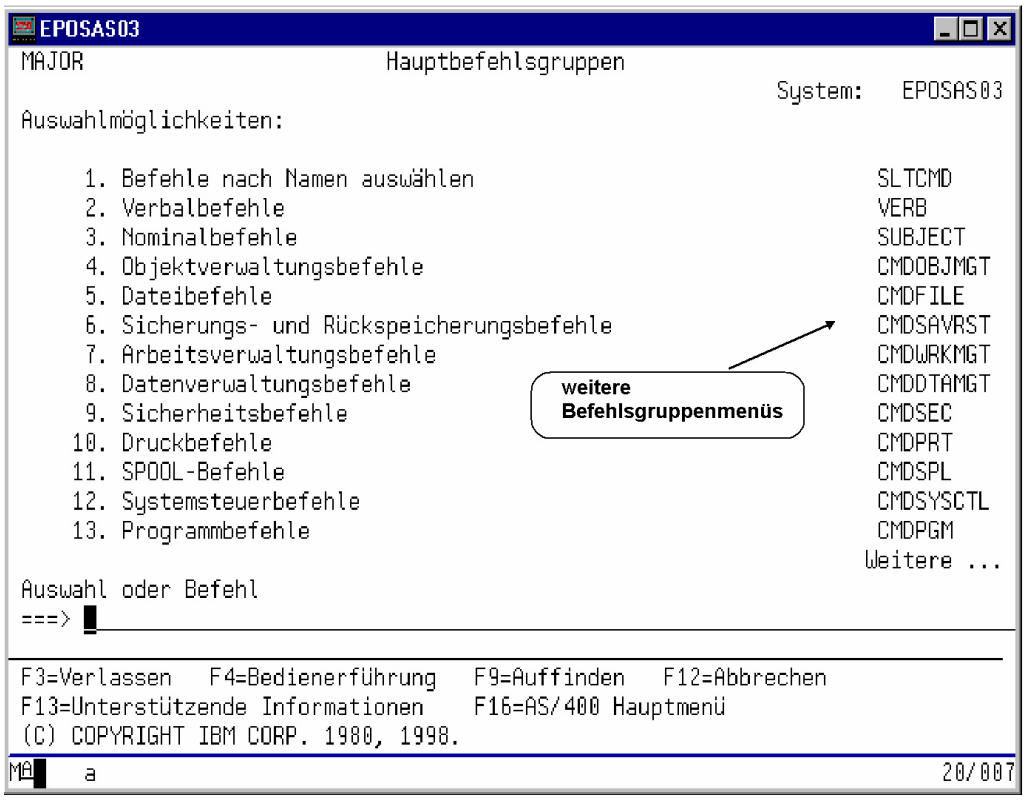
Generische Befehlssuche

Als Ergebnis erhalten Sie alle Befehle, die mit der Zeichenfolge DSPUSR beginnen.

Um eine Übersicht aller zur Verfügung stehenden CL-Befehle zu erhalten, positionieren den Cursor in einer leeren Befehlszeile und drücken die Funktionstaste „F4". Alternativ können Sie das Menü MAJOR auch mit dem Fast Path << GO MAJOR>> aufrufen. Das Menü MAJOR zeigt alle Hauptbefehlsgruppen an. Es ist das erste Menü einer Reihe von Befehlsmenüs, die die Befehle in zusammengehörigen Gruppen zusammenfassen, d.h.: alle Dateibefehle, alle Druckbefehle usw.

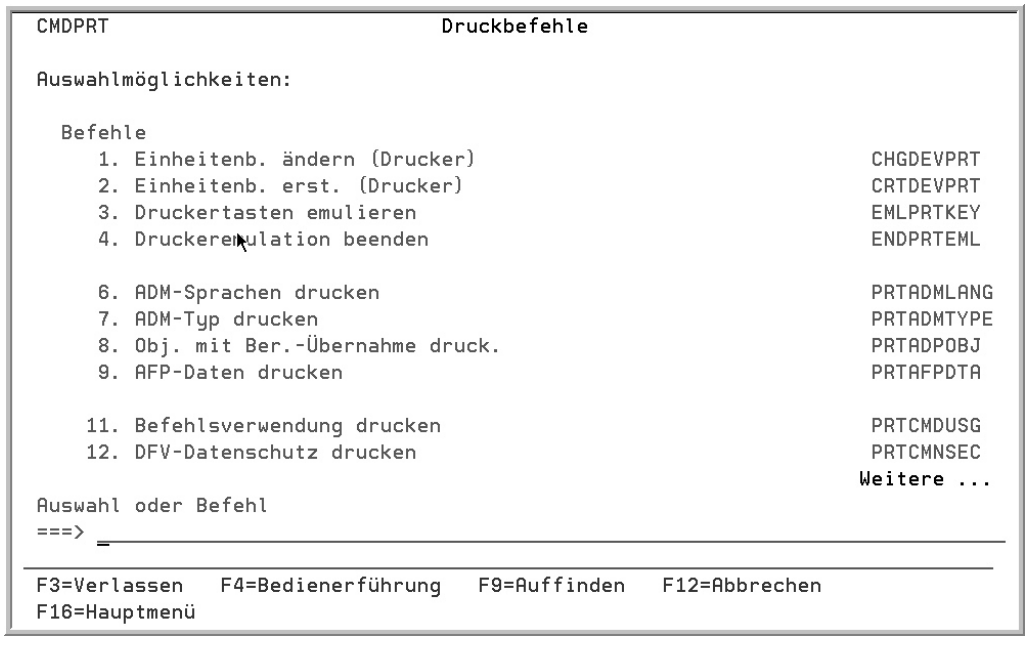
1. Ergänzung 12/2004

2.1.4.2
Seite 2



Hauptbefehlsgruppen

In dem Menü MAJOR können Sie nach Befehlsgruppen suchen. Sobald Sie eine der Gruppen ausgewählt haben, erhalten Sie einen weiteren Bildschirm, der alle verfügbaren Befehle dieser Befehlsgruppe anzeigt.



Die Befehlsgruppe CMDPRT

Wird in einem Befehlsmenü ein Befehl ausgewählt, wird die Bedienerführung für den Befehl angezeigt. Die Bedienerführung (F4) können Sie immer dann verwenden, wenn Sie den Befehl, aber nicht seine Parameter kennen. Sie können sich mit der Funktionstaste F4 die Standardparameter eines Befehls anzeigen lassen. Wenn Sie nicht wissen, welche Parameter einem CL-Befehl zugeordnet sind, können Sie den Befehl ohne Parameter eingeben und anschließend die Bedienerführung (=F4) nutzen. OS/400 zeigt Ihnen die verfügbaren Parameter für den Befehl an.

```
                        Bibliothek erstellen (CRTLIB)

 Auswahl eingeben und Eingabetaste drücken.

 Bibliothek . . . . . .  _________      Name
 Bibliotheksart . . . . . .  *PROD      *PROD, *TEST
 ASP-Nummer . . . . . .  1              1-32, *ASPDEV
 ASP-Einheit  . . . . . .  *ASP         Name, *ASP, *ASPGRPPRI...
 Text 'Beschreibung'  . . . . . .  *BLANK
_____

                                                                      Ende
 F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F10=Zusätzl. Parameter
 F12=Abbrechen   F13=Verwendung der Anzeige   F24=Weitere Tasten
```

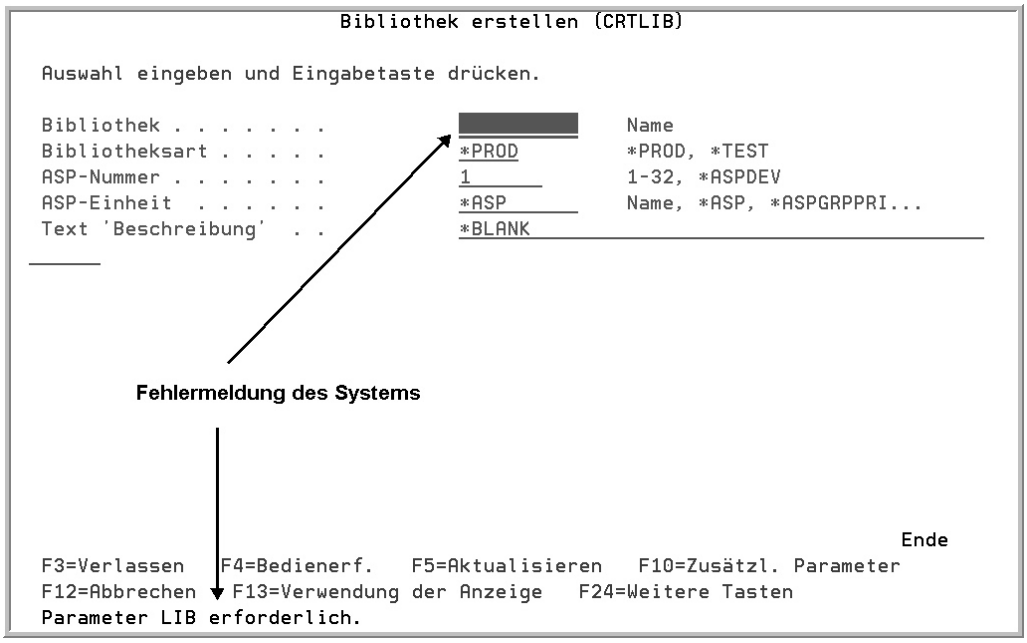
Die Bedienerführung des Befehls CRTLIB

Wenn Sie den Cursor jetzt auf ein Eingabefeld positionieren, können Sie:

- die F4-Taste verwenden, um sich eine Liste der möglichen Parameterwerte anzeigen zu lassen,
- die F1-Taste verwenden, um sich eine Erläuterung des Parameters und jedes Parameterwertes anzeigen zu lassen.

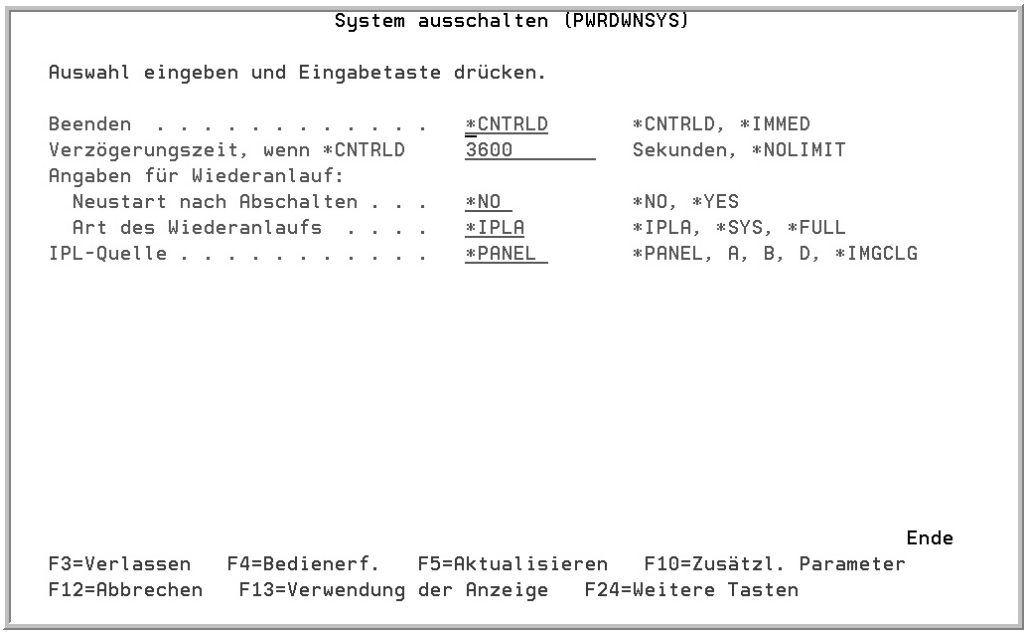
Die meisten Befehle können ausgeführt werden, indem lediglich der Befehlsname und ein oder zwei Parameter angegeben werden. Für die übrigen Parameter werden vom System zur Verfügung gestellte Standardwerte verwendet. Für den Befehl CRTLIB (Bibliothek erstellen) müssen Sie zum Beispiel mindestens den Namen der Bibliothek angeben, die Sie erstellen wollen. Wenn Sie den Befehl ohne Parameter eingeben und sofort freigeben, wird der Befehl ausgeführt und Sie erhalten vom System eine Fehlernachricht, in der Sie aufgefordert werden, Eingaben in allen erforderlichen Feldern vorzunehmen.

2.1.4.2
Seite 4



Ausführung des Befehls CRTLIB ohne Parametereingabe

Solange ein Befehl verbindliche Parametereingaben ihrerseits erwartet, werden Sie zwischen der Bedienerführung und der unmittelbaren Freigabe des Befehls keine wesentlichen Unterschiede feststellen. Den Unterschied lernen Sie aber schnell kennen, wenn Sie die Parameter des Befehls PWRDWNSYS (=Power Down System) betrachten. Aber Vorsicht, benutzen Sie unbedingt die Bedienerführung (F4), um die Parameter anzuzeigen.



Bedienerführung des Befehls PWRDWNSYS

Alle Parameter verfügen über Default-Einstellungen, d.h.: Der Befehl benötigt keine Parametereingaben für die Verarbeitung. Er wird unmittelbar ausgeführt, sobald Sie die Datenfreigabetaste drücken! Dieser kleine Unterschied zwischen Bedienerführung und Datenfreigabe hat dazu geführt, dass ich in meinen Schulungen den CL-Befehl PWRDWNSYS für alle Schulungsteilnehmer sperre, denn die Pausen – da „die iSeries gerade mal wieder runter gefahren wurde“ – kamen leider immer zum falschen Zeitpunkt und waren, wie Sie sich vorstellen können, auch niemals gewollt. Wir können in diesen Fällen einfach Pause machen, aber in Ihrem Umfeld wäre das sicherlich ein größeres Problem. Daher ist es ratsam, sich die Verwendung der Funktionstaste F4 von Anfang an anzueignen.

Falls Ihnen nicht ein Befehl sondern die Bedeutung einer iSeries-Nachricht unklar ist oder Sie nicht wissen, wie das Problem behoben werden kann, können Sie den Cursor auf die Meldung in der Statuszeile positionieren und anschließend die Hilfetaste verwenden. Sie erhalten einen Nachrichtentext, der die aktuelle Nachricht erläutert. Die Anzeige enthält außerdem Vorschläge zur Fehlerbeseitigung. Wenn neben der Nachricht in der Statuszeile ein +- Zeichen steht, bedeutet dies, dass noch weitere Nachrichten vorhanden sind, in denen geblättert werden kann. Auch für diese Nachrichten können Sie die Hilfetaste verwenden.

2.1.4.2
Seite 6

Das iSeries Information Center

Das IBM Information Center ist die strategische Lösung der IBM zur Bereitstellung von Informationen. Dadurch gibt es einen einzigen Informationspool für alle Produkte der IBM. Das Information Center enthält Links zu Handbüchern, Redbooks und dem Technical Studio. Zudem werden wichtige Themen angesprochen wie Clustering, Java, TCP/IP, Web-Serving und logische Partitionen. Vom Information Center aus können Sie nach Themen oder einem bestimmten Stichwort suchen oder eine Auswahl aus einer Themenliste treffen. Um diesen Informationsbestand zu nutzen, rufen Sie die folgende Internet-Seite auf:

http://www.as400.ibm.com/infocenter

Damit diese Idee auch für die iSeries realisiert werden konnte, wurden die Handbücher der früheren Online Library ebenfalls in das Information Center versetzt. Seit dem Release V5R1 wird die Online Library nicht mehr aktualisiert. Sie wird durch die Kategorie „iSeries ergänzende Literatur“ ersetzt. Weitere Informationen erhalten Sie im Information Center unter „Obsolete iSeries-Handbücher“.

Das Information Center ist auch auf CD-ROM verfügbar, dadurch können Sie die Informationen ohne Internet-Zugang nutzen. Sie können die Informationen vollständig oder teilweise auf die iSeries kopieren. Die Installationsfunktion konfiguriert dabei einen HTTP-Server und Suchindizes, damit Sie aus Ihrem Intranet mit einem Java- oder JavaScript-fähigen Browser auf das Information Center zugreifen können. Es ist aber auch möglich, die CD-ROM auf dem lokalen PC zu installieren. Das V5R3 iSeries Information Center wird im Internet regelmäßig aktualisiert. Sie können sich per e-Mail über diese Aktualisierungen benachrichtigen lassen. Diese Funktion können Sie auf der Seite „Subscribe to the Information Center Updates“ abonnieren. Falls Sie eine lokal installierte Version verwenden, können Sie die Aktualisierungen aus dem Internet herunterladen. Wählen Sie hierfür auf Ihrem Computer die Option: „START>Programme>IBM iSeries Information Center V5R3> Konfigurationstools>Aktualisierungen aus dem Internet herunterladen“.

2.1.4.3
Seite 2

Das iSeries Information Center

Nachdem Sie Ihr Land und den Release-Stand gewählt haben, gelangen Sie zur Homepage des Information Center. Da sich im letzten Release-Wechsel, einige Funktionen verändert haben, möchte ich darauf hinweisen, dass ich mich auf den folgenden Seiten ausschließlich auf das Release V5R3 beschränken werde.

Information Center – Homepage

Für das Information Center können Sie als Browser den Netscape Navigator 6.2 oder den Microsoft Internet Explorer 5.5 oder höher verwenden. Optimal arbeitet das Information Center mit den Netscape Navigator 7.0 bzw. dem Microsoft Internet Explorer 6.0 oder höher. Falls ein anderer Browser oder ein früherer Release-Stand verwendet wird, werden die Seiten möglicherweise anders formatiert und einige Funktionen des Centers könnten fehlerhaft sein. Ist eine Aktualisierung Ihres Browsers erforderlich, können Sie den Netscape Navigator unter

http://browsers.netscape.com/browsers/main.tmpl aktualisieren

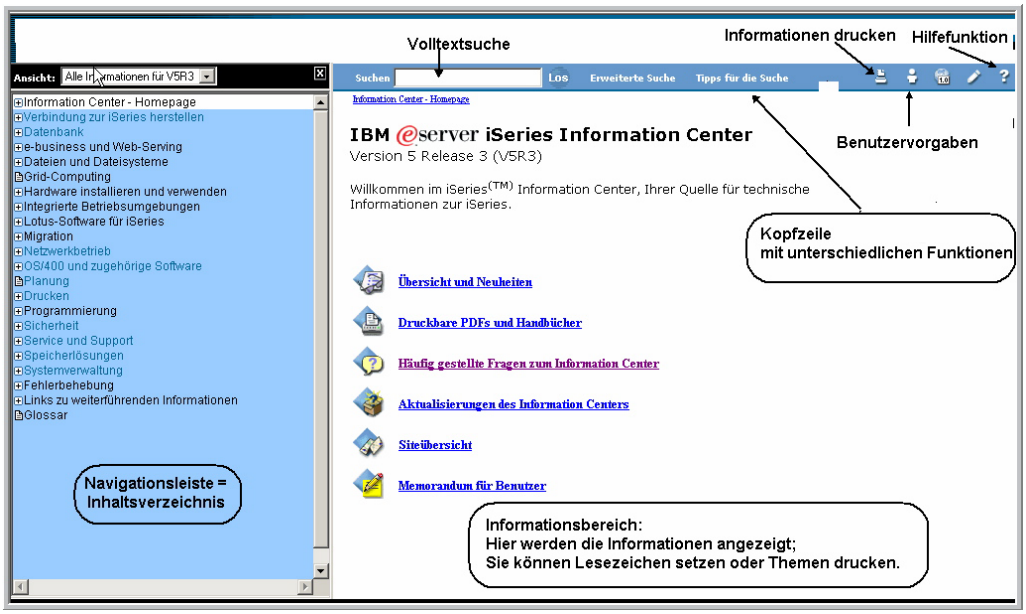
und für den Microsoft Internet Explorer unter

http://www.microsoft.com/downloads/.

Außerdem müssen Sie in Ihrem Browser Java, JavaScript, Cookies und Frames aktivieren.

2.1.4.3
Seite 4

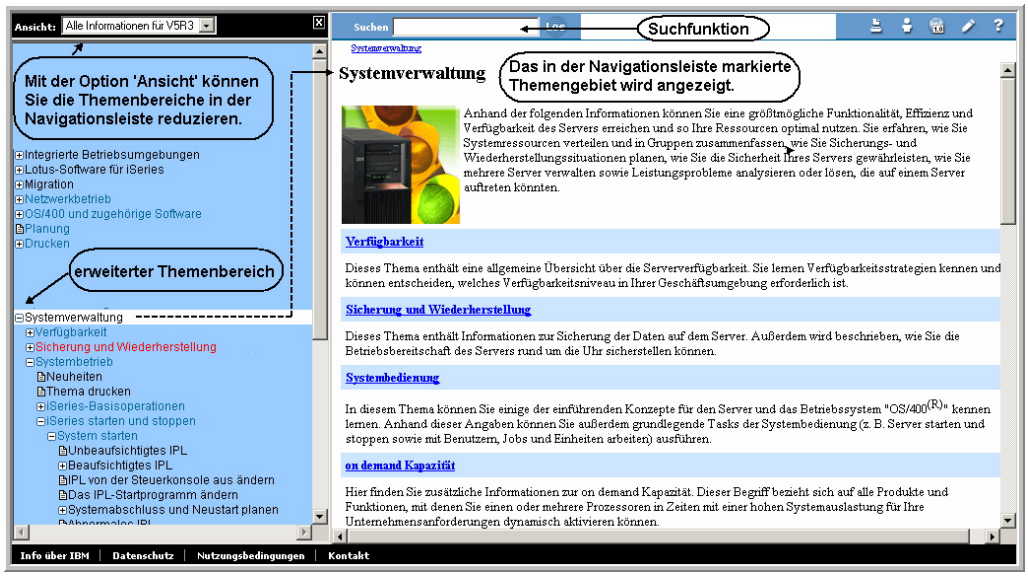
Im iSeries Information Center navigieren



Erläuterung der verschiedenen Frames

Die Navigationsleiste enthält eine Liste aller verfügbaren Themen. Jedes Hauptthema kann aus mehreren Unterthemen bestehen. Die untergeordneten Themen können Sie anzeigen, indem Sie mit der Maus auf das Plus-Zeichen klicken. Falls Sie ein Thema anzeigen wollen, klicken Sie einfach auf ein Thema, das daraufhin im Informationsbereich angezeigt wird. Soll eine bestimmte Stelle der angezeigten Informationen mit einem Lesezeichen gekennzeichnet werden, klicken Sie mit der rechten Maus auf eine beliebige Stelle im angezeigten Artikel, abhängig vom verwendeten Browser erscheint entweder ein Popup-Menü oder ein Direktaufruf. Im Internet Explorer klicken Sie auf „Zu Favoriten hinzufügen“; bei der Verwendung von Netscape Navigator klicken Sie auf „Lesezeichen hinzufügen“.

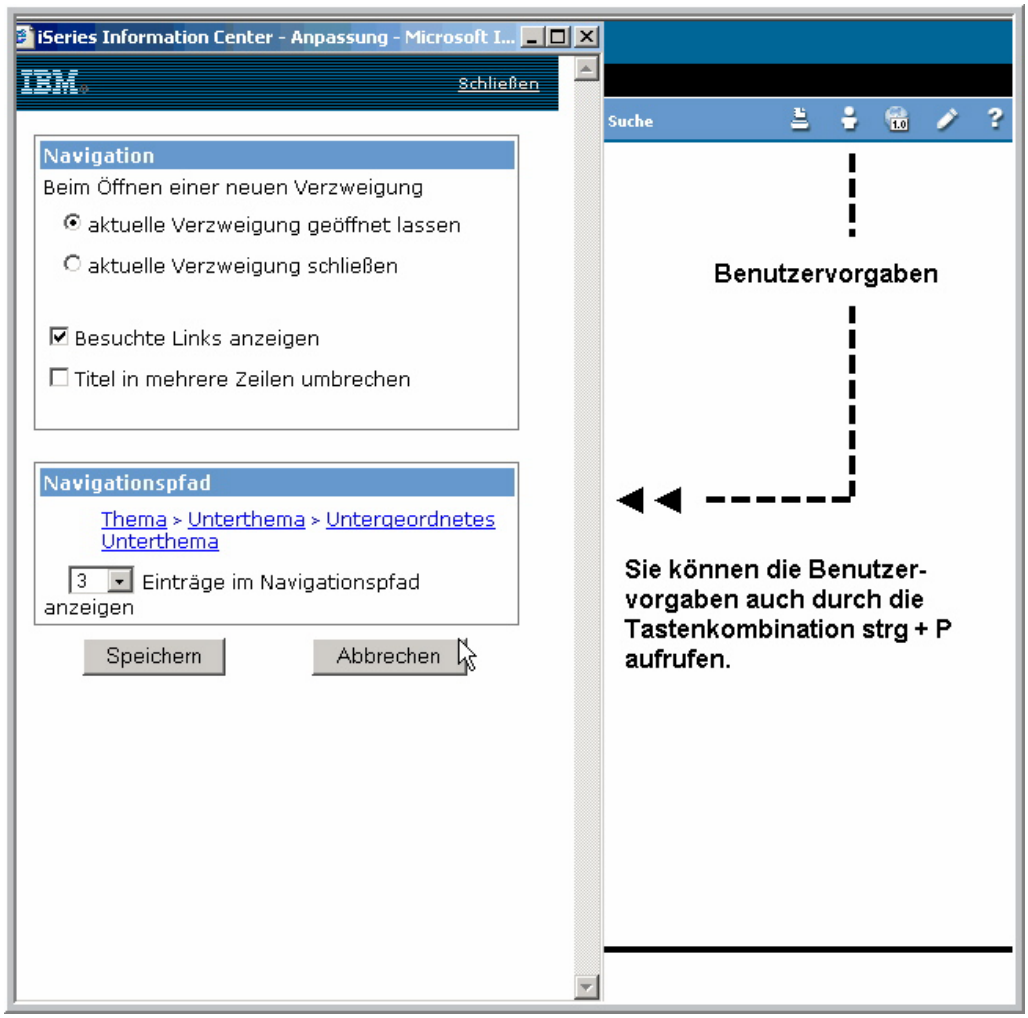
2.1.4.3.1
Seite 2



Navigieren im iSeries Information Center

Um die Themengebiete zu reduzieren, können Sie in der Drop-Down-Liste der Option „Ansicht“ ein spezifisches Themengebiet auswählen. Anschließend werden nur die Themengebiete dieser Ansicht angezeigt. Unabhängig von der verwendeten Ansicht, wird die Suchfunktion allerdings alle Themengebiete durchsuchen.

Sie können die Darstellung der Navigationsleiste individuell anpassen, indem Sie die Option „Vorgaben“ auswählen.



Benutzervorgaben

Beim Öffnen einer neuen Verzweigung

Hier wählen Sie, ob die aktuell angezeigten Unterthemen geöffnet bleiben oder bei der Öffnung einer neuen Verzweigung geschlossen werden.

Besuchte Links anzeigen

Ist diese Option aktiviert, werden alle zuvor ausgewählten Themen in einer anderen Farbe angezeigt; sonst werden alle Themenbereiche identisch dargestellt.

Titel in mehrere Zeilen umbrechen

Lange Thementitel werden abgeschnitten, wenn Sie nicht in einer Zeile angezeigt werden können.

Navigationspfad anzeigen

Mit dieser Option bestimmen Sie, ob der Navigationspfad am Anfang jeder Seite angezeigt wird oder nicht.

2.1.4.3.1

Seite 4

Informationen suchen

Die Suchfunktion im Information Center befindet sich in der Kopfzeile des Information Centers. Es handelt sich hierbei um eine Volltextsuche. Für die Suche nach Oberbegriffen geben Sie Stichwörter ein. Die folgende Angabe sucht beispielsweise nach den Stichwörtern „embedded/SQL“. Das Suchergebnis enthält alle Artikel, in denen einer oder beide Begriffe vorkommen.

Informationen suchen

Wenn Sie nach einzelnen Begriffen suchen, ist die Ergebnisliste häufig sehr komplex. Sie können die Ergebnisse konkretisieren, indem Sie nach einem genauen Ausdruck suchen. Bei der Suche nach einer exakten Übereinstimmung setzen Sie die Wörter in Anführungszeichen.

Die Ergebnisliste enthält jetzt nur Artikel, in denen der gesuchte Ausdruck vorkommt, d.h.: Bei unserem Beispiel werden jetzt alle Artikel gefunden, die den Ausdruck „embedded SQL“ enthalten.

Ausdrücke suchen

Die Suchfunktion ignoriert Interpunktionszeichen sowie extrem kurze oder geläufige Wörter wie z.B. der, ein oder in. Außerdem ignoriert die Suchfunktion die Groß- und Kleinschreibung.

Die Suchergebnisse können Sie optimieren, indem Sie weitere Wörter im Suchausdruck verwenden oder logische Operatoren für Ihre Suchangaben nutzen.

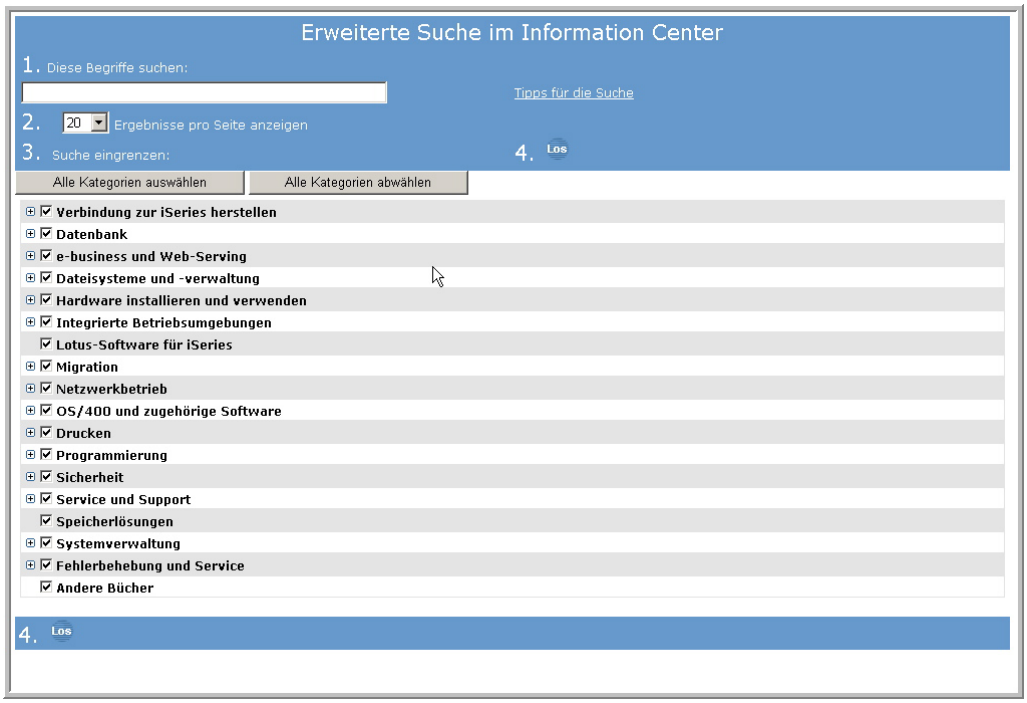
Boolescher Operator	Symbol	Erläuterung
AND	&	Die Angabe „embedded AND SQL“ sucht nach Dokumenten, die sowohl das Wort „embedded“ als auch das Wort „SQL“2 enthalten.
OR	/	Die Angabe „embedded OR SQL“ sucht nach Dokumenten, die mindestens eines der Wörter enthalten.
NOT	!	Dokumente, die den angegebenen Begriff enthalten, werden von der Ergebnisliste ausgeschlossen.

Sie können für Ihre Suche die Operatoren AND, OR, NOT oder die entspre-

2.1.4.3.2
Seite 2

chenden Symbole verwenden. Vor und nach dem Operator muss ein Wort angegeben werden. „!SQL“ ist somit kein gültiger Suchbegriff. Der Suchbegriff darf maximal fünf Wortfolgen und vier boolesche Operatoren enthalten.

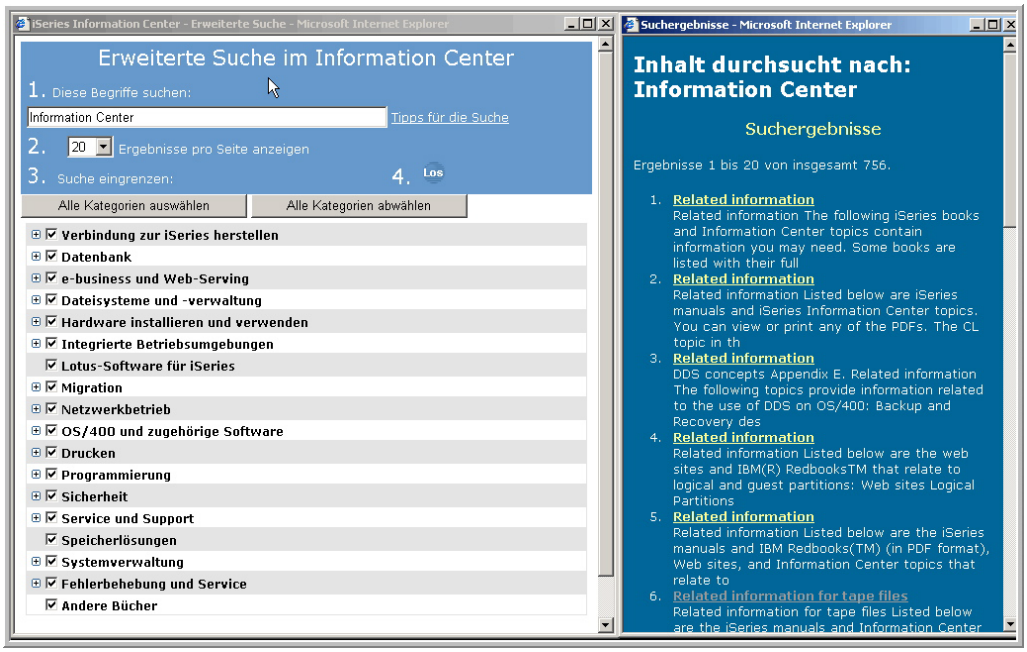
Wenn Sie die Suche auf einzelne Themen oder Themengruppen beschränken wollen, müssen Sie die erweiterte Suchfunktion verwenden.



Die erweiterte Suchfunktion

Geben Sie auch hier wieder ihren Suchbegriff ein, wählen Sie alle Kategorien ab und anschließend die Kategorien bzw. die einzelnen Themen aus, in denen gesucht werden soll. Falls Sie nur die iSeries-Handbücher nach dem Thema durchsuchen wollen, wählen Sie als Kategorie „Andere Bücher“. Für die Suche in einem bestimmten Handbuch müssen Sie die Suchfunktion des Acrobat Reader verwenden, d.h.: Sie müssen das Handbuch zunächst als PDF-Version herunterladen und anschließend im Acrobat Reader durchsuchen. Falls der Acrobat Reader nicht auf Ihrem Computer installiert ist, können Sie das Programm kostenlos auf der Adobe-Homepage herunterladen.

Mit der Schaltfläche „LOS“ wird der Suchvorgang gestartet. Während des Suchvorgangs können Sie den Suchvorgang über die Schaltfläche „Stoppen“ jederzeit abbrechen. Die Suchergebnisse werden in einem neuen Fenster angezeigt.



Ergebnisliste der Volltextsuche

Wenn Sie ein Dokument lesen möchten, klicken Sie einfach auf den Titel. Der Artikel wird dann im Browser angezeigt. Alternativ können Sie mit der rechten Maustaste auf den Titel klicken und das Dokument so in einem separaten Fenster öffnen. Am Ende der Ergebnisliste stehen Ihnen entsprechende Schaltflächen zur Verfügung, um die nächste Seite der Suchergebnisse anzuzeigen.

Für die Suche eines einzelnen CL-Befehls oder eines APIs gibt es eine separate Suchfunktion innerhalb der Navigationsleiste. Im Themengebiet „Programmierung“ finden Sie im Unterthema „CL“ bzw. „API“ den „CL Command Finder“ bzw. den „API Finder“.

2.1.4.3.2

Seite 4

Informationen im Information Center drucken

Für den Fall, dass Sie gefundene Informationen drucken wollen, haben Sie verschiedene Möglichkeiten:

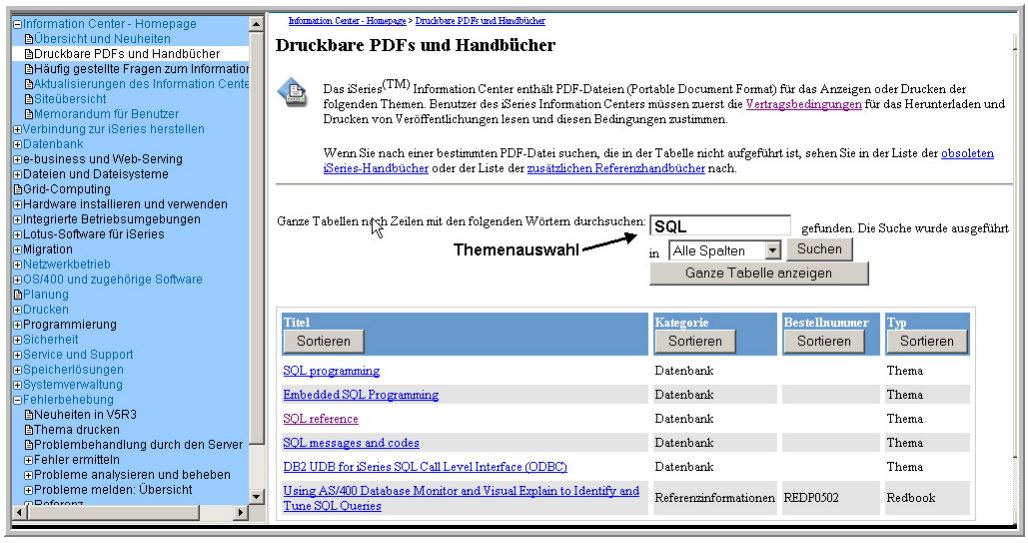
Einzelne Seiten drucken

Klicken Sie mit der Maus auf die zu druckende Seite und klicken Sie anschließend im Browser auf die Schaltfläche „Drucken“ oder wählen Sie im Menü „DATEI“ die Option „Drucken“. So können Sie den Artikel ohne die Informationen der Navigationsleiste drucken.

Hauptthemen drucken

Falls Sie einzelne Kapitel drucken möchten, klicken Sie auf der Seite oder in der Navigationsleiste auf den Link „Thema drucken“. Das Information Center verwendet hierfür entsprechende PDF- (Portable Document Format-) Dateien. Beachten Sie bitte, dass die Druckqualität möglicherweise nicht der Qualität im Browser entspricht.

Alternativ können Sie in der Navigationsleiste auf das Thema „Information Center – Homepage“ / „Druckbare PDFs und Handbücher“ wechseln. Alle Handbücher sind im PDF Format verfügbar und können online gelesen, gedruckt oder heruntergeladen werden.



Handbücher und PDFs drucken

Wenn Sie ein Handbuch auswählen, wird das Handbuch im Adobe Acrobat Reader angezeigt. Werden Sie bitte nicht ungeduldig, wenn das Laden des Handbuches etwas dauert. Anschließend können Sie das Handbuch auf Ihrem System speichern und drucken, d.h. so bearbeiten wie jedes andere PDF-File.

2.1.4.3.3
Seite 2

Links zu weiterführenden Informationen

Das Information Center enthält aber nicht nur spezifische iSeries-Informationen, sondern auch weiterführende Informationen. Aus diesem Grund enthält die Navigationsleite den Link zu weiterführenden Informationen. Hier finden Sie eine weitere Form der iSeries-Dokumentation: die so genannten „Redbooks“.

Die iSeries Redbooks

Redbooks enthalten in der Regel konkrete Beispiele für Aufgaben, die sich auf eine bestimmte Geschäftssituation beziehen. Die Beispiele können für das Verständnis eines bestimmten Themas sehr hilfreich sein. Sie sind allerdings nicht formal getestet und beziehen sich möglicherweise nur auf eine bestimmte Systemkonfiguration oder Situation. Über verfügbare Redbooks können Sie sich im Internet auch direkt informieren, indem Sie die folgende Internetadresse aufrufen:

http://www.redbooks.ibm.com.

Das Information Center ist mittlerweile die strategische Lösung der IBM für die Informationsveröffentlichung aller IBM Produkte. Außer den Redbooks sind möglicherweise auch andere IBM Veröffentlichungen für Ihre Arbeit hilfreich. Alle Details des Informationspools können wir an dieser Stelle sicherlich nicht erläutern. Ich hoffe trotzdem, dass ich Ihnen einen kleinen Überblick vermitteln konnte. Nehmen Sie sich einfach etwas Zeit für eigene Recherchen. Ich denke, es wird sich lohnen!

2.1.4.3.4
Seite 2

iSeries Navigator-Basiswissen

In den Kapiteln 2.1.1 bis 2.1.4 haben wir uns mit den iSeries-Grundlagen beschäftigt. Dort haben wir zunächst den traditionellen Weg über 5250-Bildschirme und Dialoge gewählt. Während früher alle AS/400-Funktionen mit 5250-Bildschirmen angezeigt und mit CL-Befehlen installiert, konfiguriert und administriert wurden, hat sich heute dieses Bild drastisch verändert:

Mit der Client Access-Version 3.2 sowie den Client Access-Express-Versionen 4.4, 4.5 und 5.1 kam das Produkt Operation Navigator auf dem iSeries-Desktop – auch wenn bis heute einige dieses Tool ignorieren. Die Dominanz des 5250-Bildschirms wurde seitens IBM damit beendet. Doch damit nicht genug: IBM integrierte in die grafische Oberfläche diverse Funktionsbereiche des OS/400, die ausschließlich mit iSeries Navigator zu erreichen sind. Fakt ist also: Es existiert ein iSeries Navigator, der für iSeries-Benutzer viele gute, arbeitsvereinfachende Funktionen bereithält. Diese Oberfläche wird auch im Laufe der Zeit nicht verschwinden! Vielmehr kündigt IBM zum nächsten Release eine noch weiter verbesserte grafische Oberfläche an, die weitere Funktionen bereithält. Es ist also an der Zeit, sich mit dieser Oberfläche etwas genauer zu beschäftigen.

Die Entwicklung des iSeries Navigator durchlief im Grunde genommen drei Phasen:

Phase 1 – Die Implementierung benutzernaher Funktionen

Die Funktionen entwickelten sich aus dem sogenannten System Object Access (SOA) des Client Access für Windows. Hier wurden lediglich sehr einfache Funktionen angeboten – wie „Nachrichten und Druckausgaben ansehen", um nur einige zu nennen. Zu dieser Zeit waren die Funktionen oftmals instabil und die Reaktionszeiten häufig indiskutabel. Vielleicht war das der Grund, warum sich viele Operators das Tool kurz anschauten und anschließend als unbrauchbar in der Schublade verschwinden ließen?

Phase 2 – Die Implementierung von beobachtenden Funktionen in der Mehrzahl

Mit den Releases 4.4 und 4.5 wurden weitere beobachtende Funktionen in die Oberfläche integriert: Jobs konnten zumindest teilweise beobachtet oder Benutzer angelegt werden. Erste Zugriffe auf Objekte des Integrated File System waren durch Drag&Drop möglich. Ich erinnere mich aber noch gut daran, dass mein Laptop – nach dem Einspielen des SP 2 für Windows XP – keine Lust mehr hatte, mit dem iSeries Navigator zu arbeiten. Auch der Viren-Scanner war nur schwer davon abzuhalten, die vielen kleinen Dateien des Navigators jedes Mal aufs Neue zu prüfen. Die Tasse Kaffee konnte man sich beim Start des Navigators nicht nur aus der Küche holen, man konnte sie häufig auch in Ruhe auch austrinken.

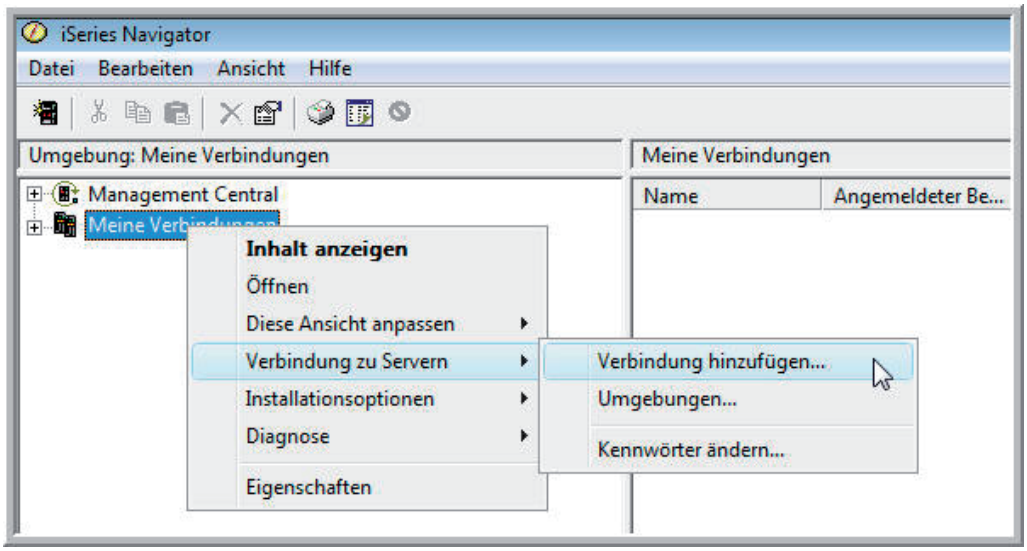
Phase 3 – Die Implementierung mehrheitlich steuernder Funktionen

Diese letzte Phase wurde mit der Veröffentlichung des iSeries-Betriebssystems OS/400 V5R1 eingeleitet. In diesem Umfeld können wir aus dem Operations Navigator heraus mit allen Systemwerten arbeiten, Benutzer komplett bearbeiten, die Datenbank komplett administrieren und visualisieren, das TCP/IP-Netzwerk komplett kontrollieren, integrierte Windows Server administrieren oder logische Partitionen verwalten. Spätestens seit Release V5R1 sind auch die anfänglichen Kinderkrankheiten der grafischen Benutzeroberfläche verschwunden.

Beachten Sie bitte, dass alle nachfolgend gezeigten Vorgänge mit dem Operations Navigator, Version 5.4 in Zusammenhang mit einer iSeries 520 unter OS/400 Version 5.4 gezeigt werden. Falls Sie mit älteren Versionen arbeiten, kann es sein, dass Sie nicht alle Vorgänge auf Ihrem System wiederfinden können.

iSeries Navigator-Basiskonfiguration

Ich gehe davon aus, dass Sie iSeries Access für Windows fehlerfrei mit den erforderlichen Komponenten installiert haben. Falls nicht, finden Sie im Kapitel 7 die ausführlichen Installationsanweisungen. Wenn Sie den iSeries Navigator zum ersten Mal öffnen, bestehen noch keine Verbindungen zu den iSeries Servern. Die Verbindungen werden lokal auf Ihrem PC eingerichtet. Unsere erste Aufgabe wird es sein, eine entsprechende Verbindung aufzubauen.

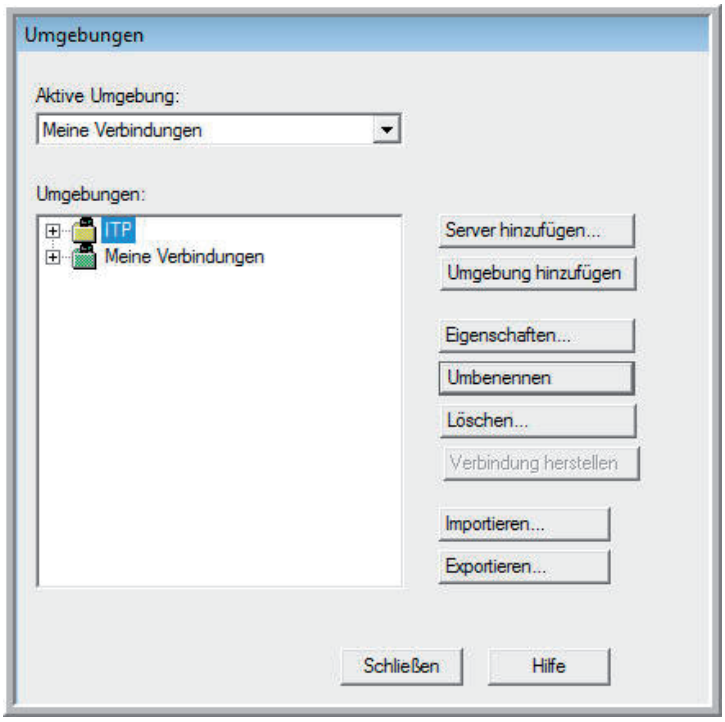


Verbindung hinzufügen

In der oberen Abbildung sehen Sie, dass der Eintrag „Meine Verbindungen“ markiert wurde; anschließend wurden die Funktionen „Verbindung zu Servern“ und „Verbindung hinzufügen“ über die Menü-Kontexttasten geöffnet. Die Konfiguration der Verbindung zum Server erfolgt in einer Verbindungsumgebung.

Schnell stellt sich die Frage, was ist eine Verbindungsumgebung? Eine Verbindungsumgebung stellt sich als ein übergeordnetes Verzeichnis dar, unter dem die Verbindung zu mehreren Systemen gleicher Art zusammengefasst werden können. Man kann auch sagen, sie ist eine Art Behälter für Ihre aktiven Verbindungen.

2.1.6.1
Seite 2



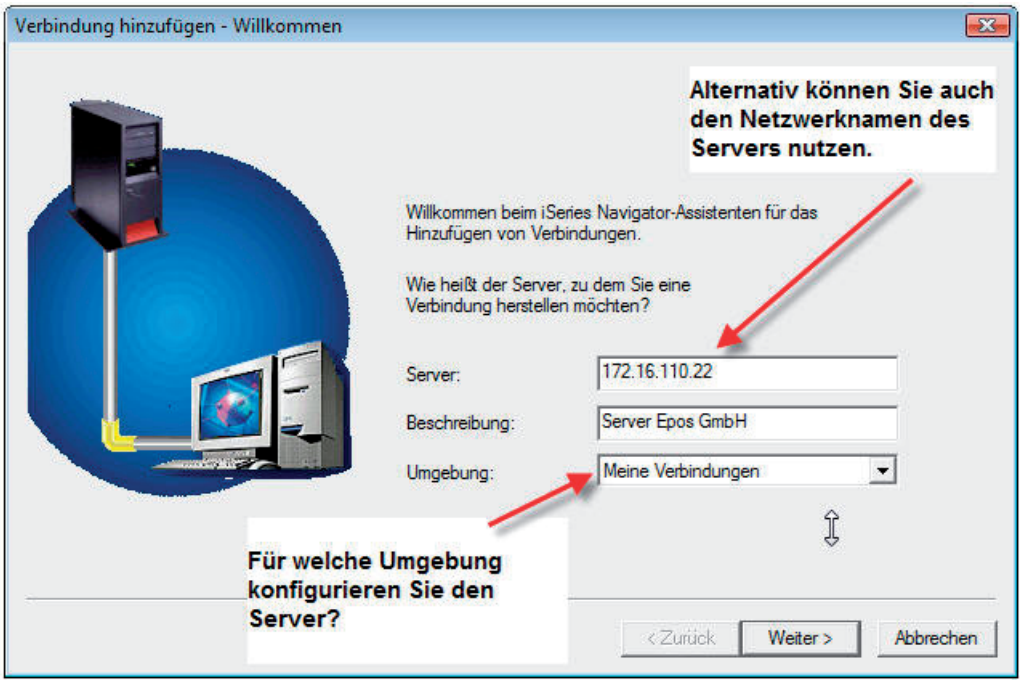
Umgebungen

Sie können verschiedene Umgebungen erstellen, die unterschiedliche iSeries-Verbindungen enthalten. Man kann also sagen, dass die Umgebungen lediglich ein Ordnungsinstrument darstellen, falls Sie eine Vielzahl von Servern in unterschiedlichen Organisationen verwalten möchten.

Diese Umgebungen lassen sich im Übrigen auch problemlos im- und exportieren. Beim Export wird die ausgewählte Umgebung einschließlich ihrer Verbindungen in einer Datei gesichert. Diese Datei können Sie anschließend jederzeit auf anderen PCs wieder importieren und ersparen sich dadurch die erneute Konfiguration der Verbindungen.

Auch wenn Sie mit verschiedenen Umgebungen arbeiten, kann es aktuell stets nur eine aktive Umgebung geben. In meinem Beispiel ist das die Standardumgebung „Meine Verbindungen“. Den Namen der Standardverbindung können Sie selbstverständlich individuell anpassen. Ich werde im weiteren Verlauf auf den Einsatz verschiedener Umgebungen verzichten und nutze einfach die Standardumgebung.

Jetzt werde ich eine „Neue Verbindung“ zu einer iSeries mit der IP-Adresse 172.16.110.22 aufbauen. Dazu wähle ich den Weg, der in Abbildung „Verbindung hinzufügen“ beschrieben worden ist. Anschließend öffnet sich ein neuer Dialog:



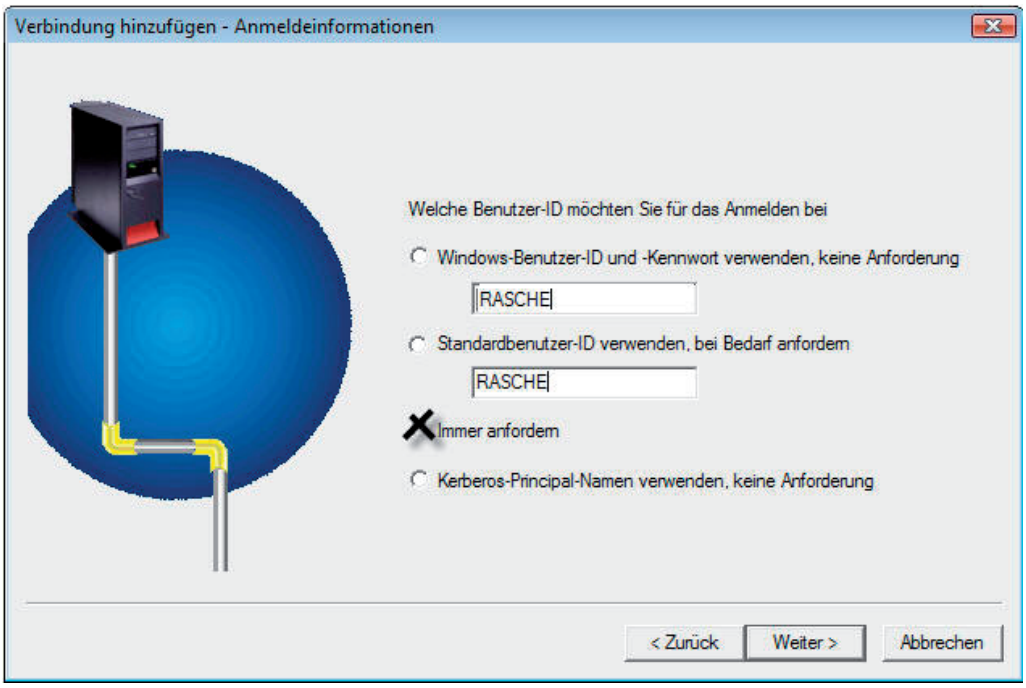
Verbindung konfigurieren – Teil 1

Im Parameter „Server" sollten Sie die IP-Adresse der iSeries eintragen, zu der die Verbindung aufgebaut werden soll – in meinem Beispiel also die IP 172.16.110.22. Generell ist es möglich, auch mit IP-Host-Namen zu arbeiten. Aber Vorsicht! Wenn Sie hier einen Host-Namen eintragen, dann muss dieser Host-Name durch das TCP/IP-Netzwerk aufgelöst werden. Das bedeutet, dass Sie entweder einen entsprechenden Eintrag in der Host-Tabelle Ihres Windows-PCs vornehmen oder dass Ihr PC auf einen im Netzwerk befindlichen Domain Name Server (DNS) zugreifen kann. Fragen Sie hierzu gegebenenfalls Ihren Netzwerkadministrator.

Ein beschreibender Text ist nicht zwingend notwendig. Als Letztes wählen Sie über ein Pull-Down-Menü den Namen einer Verbindungsumgebung. Ich belasse es in unserem Fall beim Standard. Anschließend kommen Sie zur zweiten Seite.

In diesem Dialog definiere ich die Anmeldung auf dem iSeries Server; außerdem lege ich fest, unter welchem Benutzerprofil ich auf der iSeries arbeiten werde.

2.1.6.1
Seite 4



Verbindung konfigurieren – Teil 2

Grundsätzlich lassen sich vier Anmeldeverfahren unterscheiden:

1. Variante: Windows-Benutzer und Kennwort verwenden

Sie können die Windows-Benutzernamen und deren Kennwörter für die Anmeldung an der iSeries verwenden. Vorausgesetzt: Sie pflegen auf Ihrer Windows-Arbeitsstation beziehungsweise in Ihrem Windows-Netzwerk Benutzerkonten, die bezüglich Ihres Namens und Kennworts identisch mit iSeries-Benutzerprofilen sind. Bei der Verbindungsaufnahme zur iSeries wird anschließend kein Anmeldedialog angezeigt. Die Anmeldung erfolgt implizit.

Sollte trotzt dieses Eintrags einmal kein identischer Benutzer vorliegen, wird ein Anmeldedialog angezeigt.

2. Variante: Standardbenutzer-ID verwenden:

Sie verwenden zur Anmeldung auf der iSeries einen Standardbenutzer. Jetzt wird nur bei der ersten Verbindungsaufnahme ein Anmeldedialog gezeigt, in dem der Name des Standardbenutzers bereits eingetragen ist. Anschließend muss einmalig das Kennwort eingegeben werden. Danach merkt sich der Operations Navigator diese Anmeldung. In der Folge sind die Anmeldedialoge also wieder implizit.

3. Variante: Immer anfordern:

Wenn Sie implizite Anmeldungen am iSeries Server verhindern wollen, müssen Sie den Punkt „Immer anfordern“ wählen. Anschließend erfolgt bei jeder

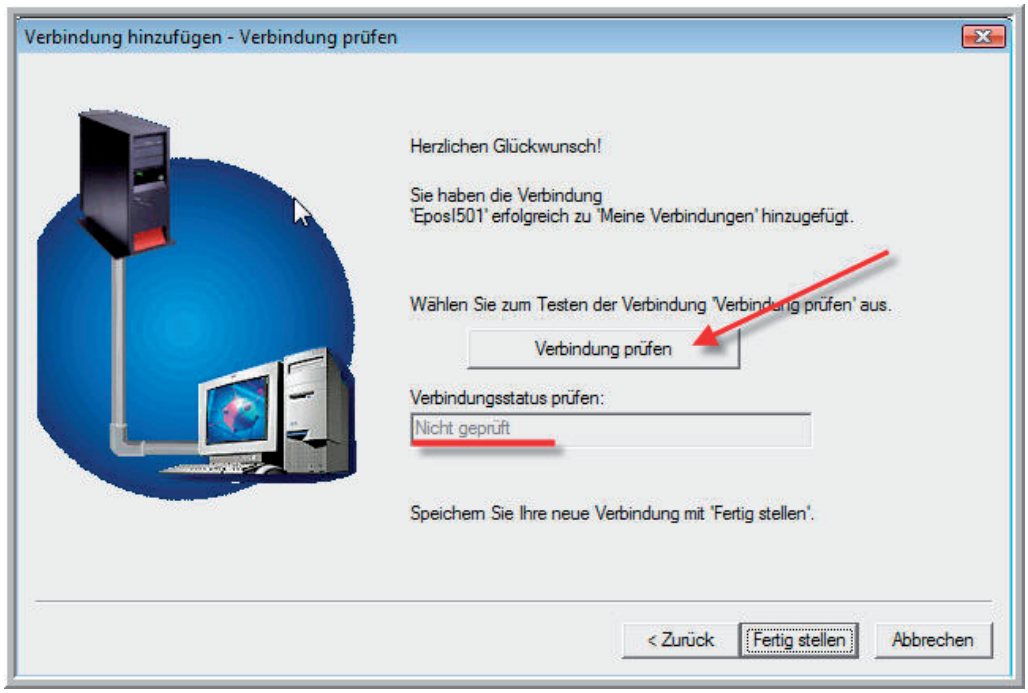
Verbindungsaufnahme ein Anmeldedialog analog der 5250-Anmeldebildschirme.

4. Variante: Kerberos-Principal-Namen verwenden, keine Anforderung

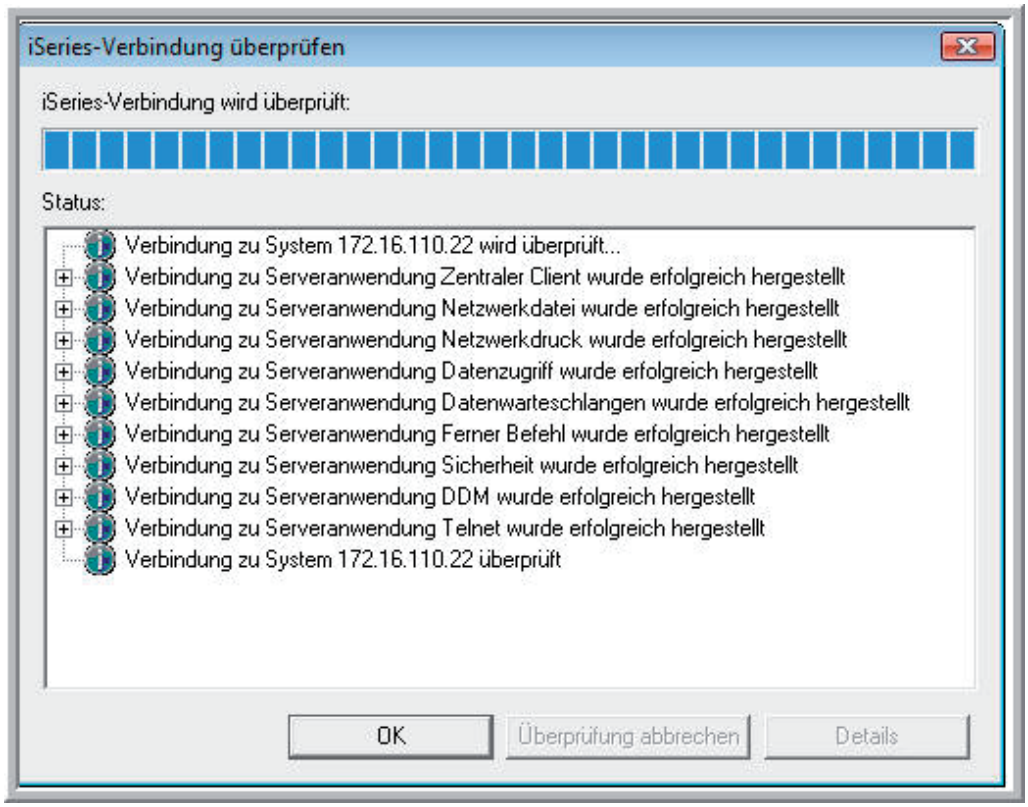
Die letzte Variante kommt nur dann in Frage, wenn Sie einen Kerberos Server für die Benutzeridentifikation verwenden. In diesem Fall wird der Kerberos-Principal-Name für die Anmeldung verwendet, der dann in einem zweiten Schritt von der iSeries in einen i5/OS-Benutzernamen aufgelöst wird, um die Zugriffsberechtigungen auflösen zu können.

Ich persönlich entscheide mich in den meisten Fällen für die Variante „Immer anfordern". Auf der einen Seite muss das jeder Benutzer tun, der eine 5250-Emulation startet; auf der anderen Seite verhindere ich damit, dass ein Benutzer einen hochgefahrenen PC nutzt, um sich auf der iSeries anzumelden und um mit zu hohen Berechtigungen im System zu agieren.

Im nächsten Dialog können Sie die bisherigen Verbindungsparameter auf ihre Gültigkeit überprüfen.

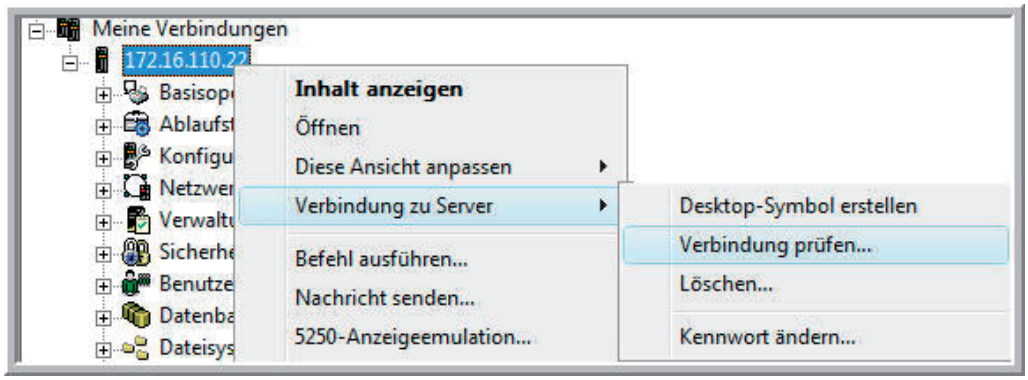


Verbindung konfigurieren – Teil 3



Verbindung prüfen

Die Verbindung zu einem Server können Sie auch jederzeit für bestehende Verbindungen aufrufen. Dafür müssen Sie lediglich im Kontextmenü einer bestehenden Server-Verbindung die Menüpunkte „Verbindung zu Server“ und anschließend „Verbindung prüfen“ angeben.



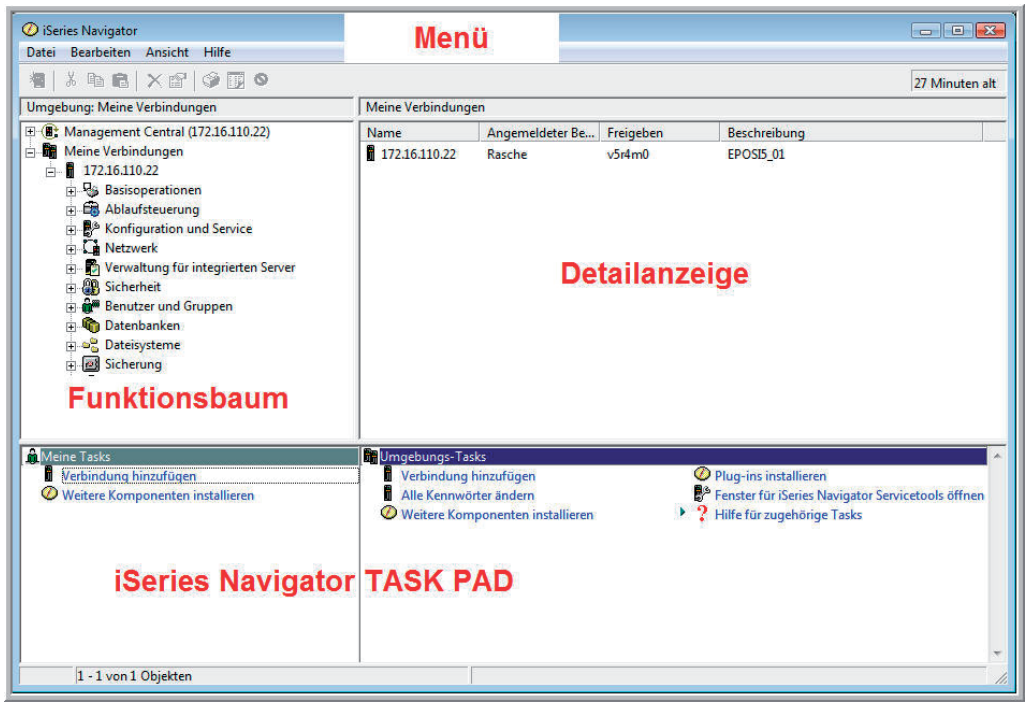
Bestehende Verbindung prüfen

Achten Sie darauf, dass nach einer komplett erfolgreichen Verbindungsprüfung die Leitung zwischen PC und iSeries die Farbe grün annimmt. Sollte die Prüfung nicht erfolgreich verlaufen sein, so erkennen Sie diesen Sachverhalt an einer geborstenen Leitung.

iSeries Navigator – Ein erster Überblick

Wie bei jeder anderen Software müssen Sie sich zuerst einmal mit der Bedienung des iSeries Navigator vertraut machen. Positiv ist hierbei, dass sich die IBM-Programmierer des Operations Navigator im Großen und Ganzen an Windows orientiert haben, so dass die Einarbeitung in den Operations Navigator relativ leicht fällt – sofern Sie natürlich mit der Bedienung von Windows-Programmen vertraut sind.

Schauen wir uns zunächst den Ausgangsdialog an:

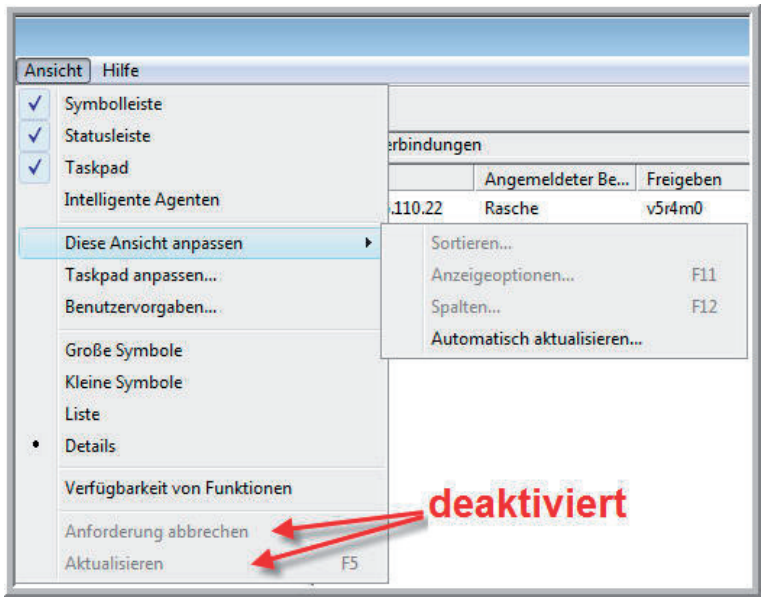


Ausgangsdialog

Wie es in Windows-Programmen üblich ist, finden Sie unterhalb der Titelleiste ein Menü. Wichtig ist, dass diese Menüs kontextsensitiv sind. Das heißt: In Abhängigkeit von Ihrer gerade ausgeführten Tätigkeit, werden in den sogenannten Submenüs der Menüpunkte „Datei“, „Bearbeiten“ und „Ansicht“ immer wieder andere Untermenüs sichtbar sein.

Schauen wir uns anhand eines Beispiels an, was gemeint ist:

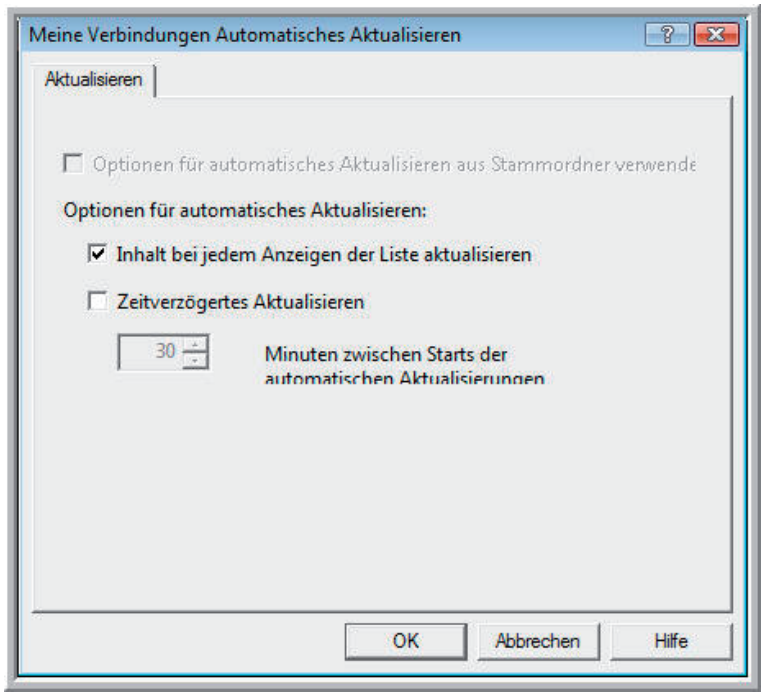
2.1.6.2
Seite 2



Menü „Ansicht“

In der Abbildung sehen Sie das Menü „Ansicht“. Einige Menüpunkte sind hellgrau dargestellt. Sie sind zurzeit nicht aktiviert, weil dieses Menü in einem bestimmten Sachzusammenhang aufgerufen wurde. Die Auswahl „Diese Ansicht anpassen…“ ist jedoch aktiv und kann verwendet werden.

Der Navigator kann übrigens automatisch die Bildschirmanzeigen aktualisieren, ohne dass Sie mit der Funktionstaste „F5“ die Aktualisierungen anfordern müssen.

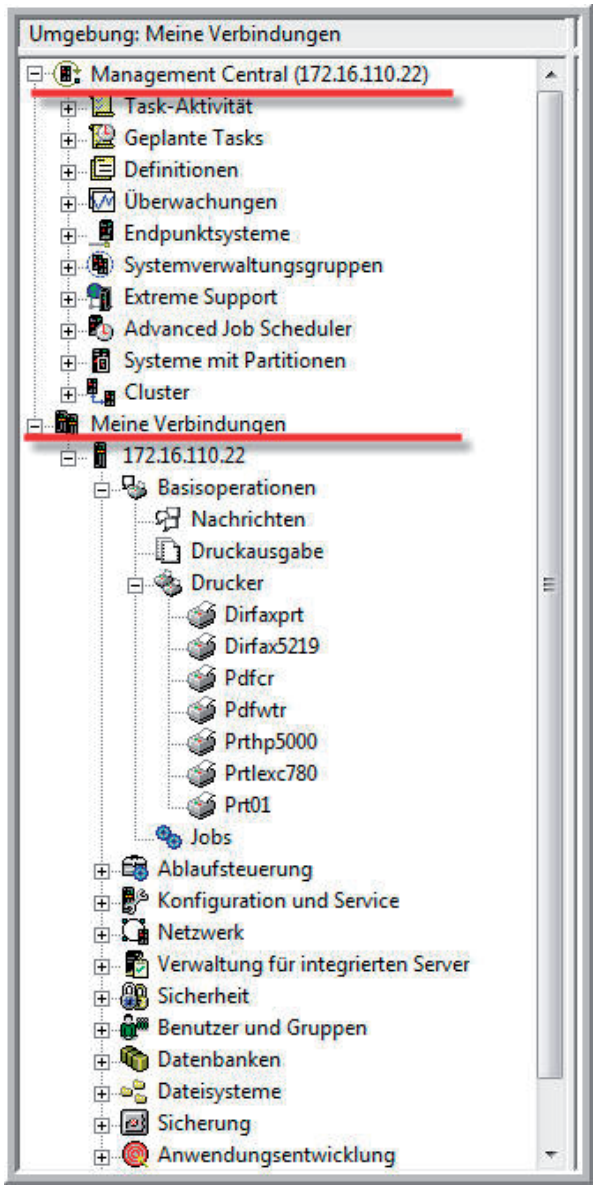


Bildschirmaktualisierungen

Es empfiehlt sich, die Aktualisierungen anzufordern, sobald Sie ein neues Fenster öffnen. Andernfalls kann es passieren, dass Sie die Veränderungen eines Fensters erst beim erneuten Aufruf sehen, wenn Sie Taste „F5“ drücken.

Ich möchte jetzt nicht jedes Untermenü detailliert erläutern, aber grundsätzlich lässt sich sagen, dass Sie über das Menü „Ansicht“ die iSeries Navigator-Oberfläche individuell gestalten können. Das Menü „Bearbeiten“ enthält die klassischen Windows-Funktionen „Suchen“, Bearbeiten“, „Ausschneiden“ und „Einfügen“. Das Menü „Datei“ ist dagegen sehr umfangreich und beinhaltet im Grunde genommen viele – um nicht zu sagen alle Funktionen, die Sie im Funktionsbaum selbst oder über entsprechende Kontextmenüs ausführen können. Auch hier ist die Parallele zur Windows-Welt sehr deutlich.

Am linken Bildschirmrand steht der Funktionsbaum:



Funktionsbaum

2.1.6.2
Seite 4

Diese Funktionen sind in Form einer Verzeichnisstruktur hierarchisch angeordnet. Im Funktionsbaum selbst lassen sich grundsätzlich zwei große Funktionsbereiche unterscheiden: „Einzelverbindungen“ und „Management Central“.

Im „Management Central“ werden iSeries-Systeme, zu denen Sie Einzelverbindungen unterhalten, als administratives Netzwerk angesehen, so dass Sie hier quasi gleichzeitig mit mehreren Systemen arbeiten können. In jedem neuen Release bietet Management Central noch weitere Funktionen, die Sie bei der Verwaltung Ihrer Systeme unterstützen. Management Central ermöglicht Ihnen unter anderem:

- das Synchronisieren von Funktionen über mehrere Systeme,
- die einheitliche Verwaltung von Benutzern und Gruppen über mehrere Systeme,
- das Überwachen der Systeme, Jobs, Nachrichten und B2B-Aktivitäten in Echtzeit,
- das Erstellen von Langzeitgrafiken der Systemleistung,
- das Verwalten von Fixes (PTFs) und Fixgruppen über mehrere Systeme,
- das Ausführen eines Befehls auf mehreren iSeries Servern.

Ohne detailliert auf die einzelnen Funktionsbereiche des Management Central einzugehen, sollten Sie zu Beginn Ihrer Arbeit mit dem iSeries Navigator bereits entscheiden, welches System Ihr zentrales System sein soll. Dazu einige Begriffsdefinitionen:

Das zentrale System ist das Hauptsystem. Es speichert alle Daten von Management Central und stellt eine Verbindung zu den Endpunktsystemen her. Es übernimmt die gesamte Datenübertragung mit den Endpunktsystemen. Beim zentralen System muss es sich um ein System handeln, das über iSeries Access verbunden ist, auf das über TCP/IP zugegriffen werden kann und auf dem i5/OS ab V4R4 installiert ist. Der Release-Stand des zentralen Systems und der Endpunktsysteme bestimmt, welche Funktionen im Management Central verfügbar sind.

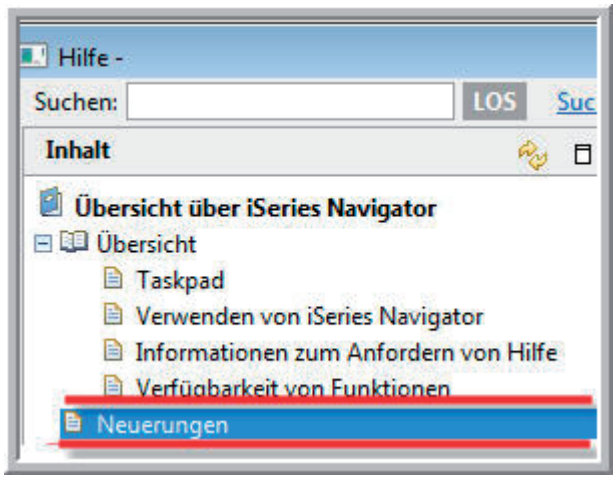
Als Endpunktsysteme bezeichnet man die iSeries Server, die mit dem Management Central verwaltet werden. Sie können die Endpunktsysteme zu Systemverwaltungsgruppen zusammenfassen, weil Ihr Unternehmen eventuell nach Funktion oder Region gegliedert ist. Um diese Struktur wiederzuspiegeln, erstellen Sie Systemverwaltungsgruppen, die auf Ihre Anforderungen abgestimmt sind – beispielsweise Buchungssysteme oder Systeme für Osteuropa.

Sie werden jetzt möglicherweise denken, dass für Sie Management Central nicht in Frage kommt, weil Sie ohnehin nur über einen iSeries Server verfügen? So einfach ist das jedoch nicht! Auch Einzelverbindungen greifen auf

Funktionen des Management Central zu, so dass die damit verbundenen Konfigurationsaufgaben in jedem Fall realisiert werden müssen.

Fazit: Die im Operations Navigator dargestellte Zweiteilung von Funktionen – in die des Management Central und in die der Einzelverbindungen – existiert in dieser Form nicht. Eigentlich kann man in diesem Punkt nur von einer groben Funktionsgruppierung sprechen.

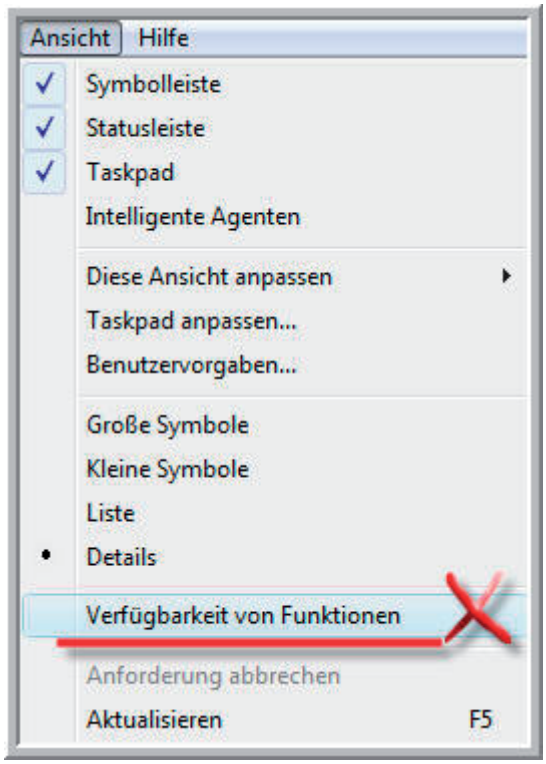
Über die Einzelverbindungen arbeiten Sie mit dem jeweiligen System. Daher ist es sinnvoll, für jeden Server eine Einzelverbindung aufzubauen. Wenn eines Ihrer Systeme nicht mit dem Operations Navigator verwaltet werden soll, verzichten Sie natürlich auf die entsprechende Einzelverbindung. Da jedes Release vom iSeries Navigator immer neue Funktionen bietet, müssen Sie, wenn Sie wissen wollen, welche Tasks Sie mit iSeries Navigator ausführen können, die Themen im linken Fensterbereich durchsuchen. Nach einem Release-Wechsel bietet die Hilfe des iSeries Navigator eine gute Orientierung über neue iSeries Navigator-Funktionen.



Neuerungen im iSeries Navigator

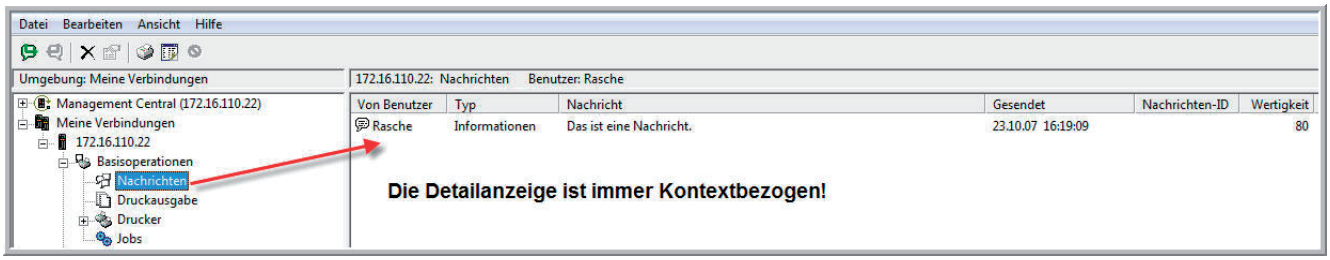
Sollten Sie den Eindruck haben, dass Ihnen der iSeries Navigator nicht alle Funktionen des Release zur Verfügung stellt, können Sie sich eine Übersicht der zur Verfügung stehenden Funktionen in Abhängigkeit Ihres Release-Standes anzeigen lassen.

2.1.6.2
Seite 6



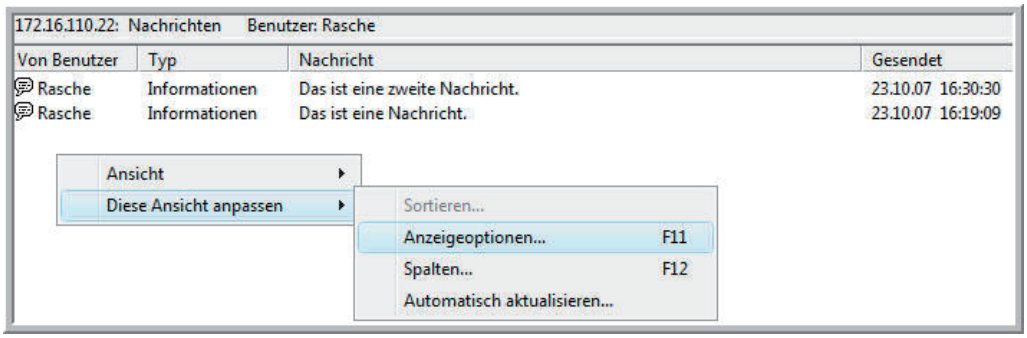
Verfügbarkeit von iSeries Navigator Funktionen

Auf der rechten Seite des Operations Navigator sehen Sie den Bereich der Detailanzeigen. Detailanzeigen verändern sich, je nachdem welche Funktion Sie im Funktionsbaum des Operations Navigator ausgewählt haben. In der folgenden Abbildung sehen Sie, dass ich den Eintrag „Nachrichten“ im Funktionsbaum ausgewählt habe. Daraufhin erscheinen im Detailfenster meine Nachrichten.



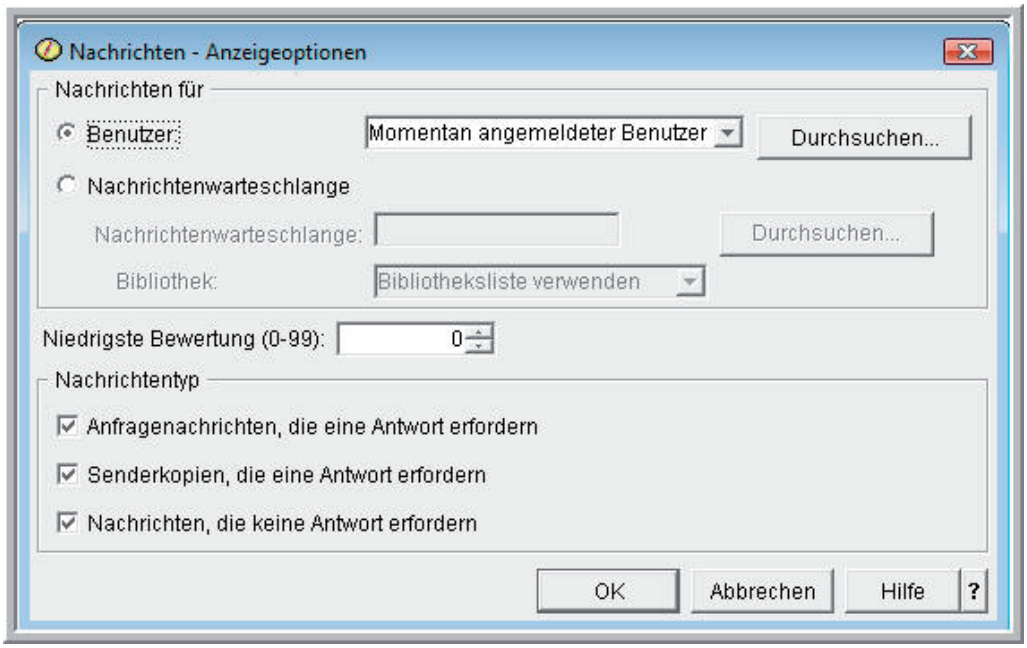
Detailanzeigen

Auch die Detailanzeigen lassen sich individuell anpassen. Sie können wieder über das Menü „Ansicht“ gehen oder Sie benutzen einfach die Kontexttaste (rechte Maustaste) im Fenster der Detailanzeige, so dass Sie folgende Auswahlmöglichkeiten erhalten:



Detailanzeigen individuell anpassen

Über den Dialog „Anzeigeoptionen…“ können Sie entscheiden, welche Detailinformationen im Fenster aufgenommen werden sollen. In unserem Beispiel könnten wir uns die Nachrichten des SYSOPR anzeigen lassen – oder wir könnten nur Nachrichten selektieren, die eine Antwort erfordern.



Anzeigeoptionen festlegen

Die Funktion des Dialogs „Anzeigeoptionen“ lässt sich am einfachsten mit der F4-Bedienerhilfe eines 5250-Dialogs der iSeries vergleichen. Wenn Sie eine bestimmte Funktion im Funktionsbaum des Operations Navigator angewählt haben und wenn Sie sich dann den Dialog „Anzeigeoptionen“ für diese Funktion anzeigen lassen, dann erhalten Sie eine Liste von Parametern. Welche Parameter angezeigt werden, hängt von der jeweiligen Funktion im Funktionsbaum ab. In der Abbildung „Anzeigeoptionen festlegen„ sehen Sie die Parameterliste für die Funktion „Nachrichten“. Spaßeshalber gebe ich in einem 5250-Bildschirm der iSeries den Befehl DSPMSG ein und prompte ihn mit F4.

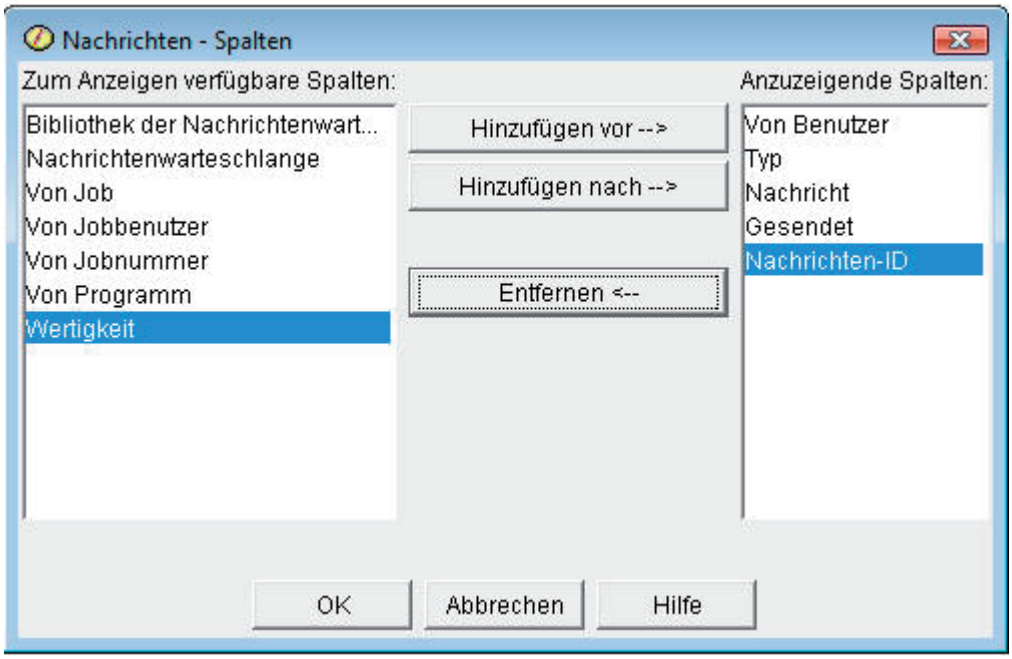
2.1.6.2
Seite 8

```
                        Nachrichten anzeigen (DSPMSG)

Auswahl eingeben und Eingabetaste drücken.

Nachrichtenwarteschlange . . . .   *WRKUSR___     Name, *WRKUSR, *SYSOPR...
  Bibliothek . . . . . . . . . .     __________    Name, *LIBL, *CURLIB
Ausgabe  . . . . . . . . . . . .   *_______       *, *PRINT, *PRTWRAP

                  Zusätzliche Parameter

Nachrichtenart . . . . . . . . .   *ALL_          *ALL, *INFO, *INQ, *COPY
Nachr. die zuerst angez. werd.     *LAST_         *LAST, *FIRST
Bewertungscodefilter . . . . . .   0_____         0-99, *MSGQ
Unterstützungsstufe  . . . . . .   *PRV_____      *PRV, *USRPRF, *BASIC...
```

Bedienerführung des Befehls DSPMSG

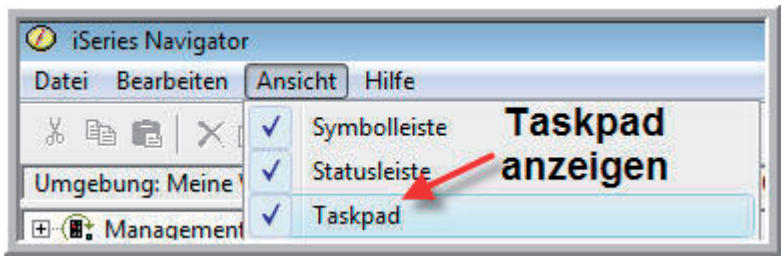
Die Übereinstimmungen sind leicht zu erkennen!

Der Dialog „Spalten“ bestimmt, welche Informationsspalten die Detailanzeige enthalten soll. Alle Spalten, die angezeigt werden, sind unter „Anzuzeigende Spalten“ aufgelistet. Zusätzlich verfügbare Spalten stehen unter „Zum Anzeigen verfügbare Spalten“. Über die Schaltflächen „Hinzufügen vor“ oder „Hinzufügen nach“ können Sie weitere Spalten in die Anzeige aufnehmen. Mit „Schaltflächen entfernen“ ist es möglich, die Spalten aus der Anzeige zu entfernen. Sollen jetzt zusätzliche Spalten in die Anzeige aufgenommen werden, so werden sie links markiert und je nach gewünschter Reihenfolge wird rechts ein Bezugsfeld markiert und dann mit „Hinzufügen vor“ bzw. „Hinzufügen nach“ auf die rechte Seite gebracht.



Spalten hinzufügen oder entfernen

Ein weiterer Hauptbestandteil die Operations Navigator ist das sogenannte Taskpad.



Taskpad

Diese visuelle Komponente wurde mit dem Operations Navigator V5R1 eingeführt. iSeries Navigator-Taskpad dient als Ausgangspunkt zum schnellen Starten von Tasks, die häufig ausgeführt werden.

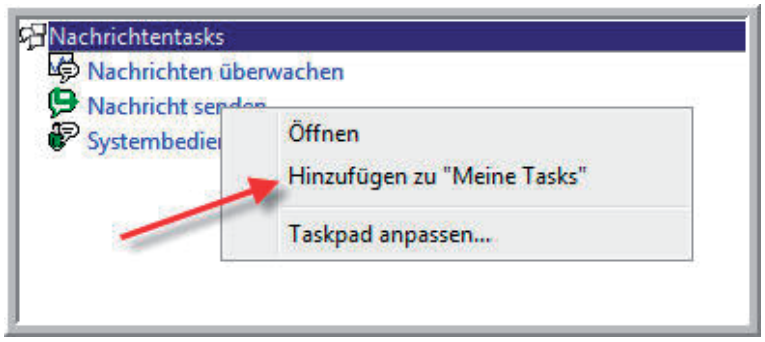
Falls Sie kein Taskpad angezeigt bekommen, müssen Sie die Menü-Anzeige „Ansicht“ aktivieren:

Taskpad anzeigen

Im Taskpad finden Sie sogenannte Wizzards, die Konfigurationsarbeiten durch zwangsgeführte Dialoge sowie die Bereitstellung von sinnvollen Default-Werten vereinfachen sollen. Sie können ein Task über das Taskpad starten, indem Sie auf das gewünschte Task klicken. Bei den im Taskpad automatisch angezeigten Tasks handelt es sich um die Tasks der iSeries Navigator-Funktion, mit der Sie gerade arbeiten. Sie können jedoch ein angepasstes Taskpad erstellen, das alle häufig verwendeten Tasks unabhängig von der Funktion enthält, zu der sie gehören.

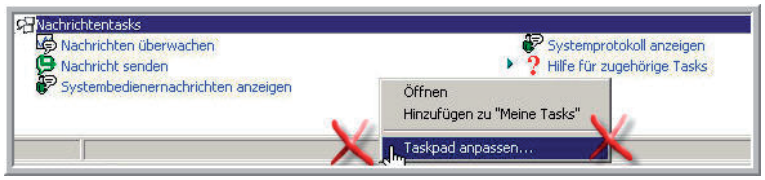
Gehen Sie dabei folgendermaßen vor:

Erweitern Sie zunächst den Ordner, mit dem Sie arbeiten möchten. Das Taskpad wird aktualisiert, wenn Sie die Ordner wechseln. Wenn Sie einen Task in Ihr angepasstes Taskpad aufnehmen möchten, klicken Sie mit der rechten Maustaste auf das gewünschte Task und wählen „Hinzufügen zu ‚Meine Tasks'" aus.



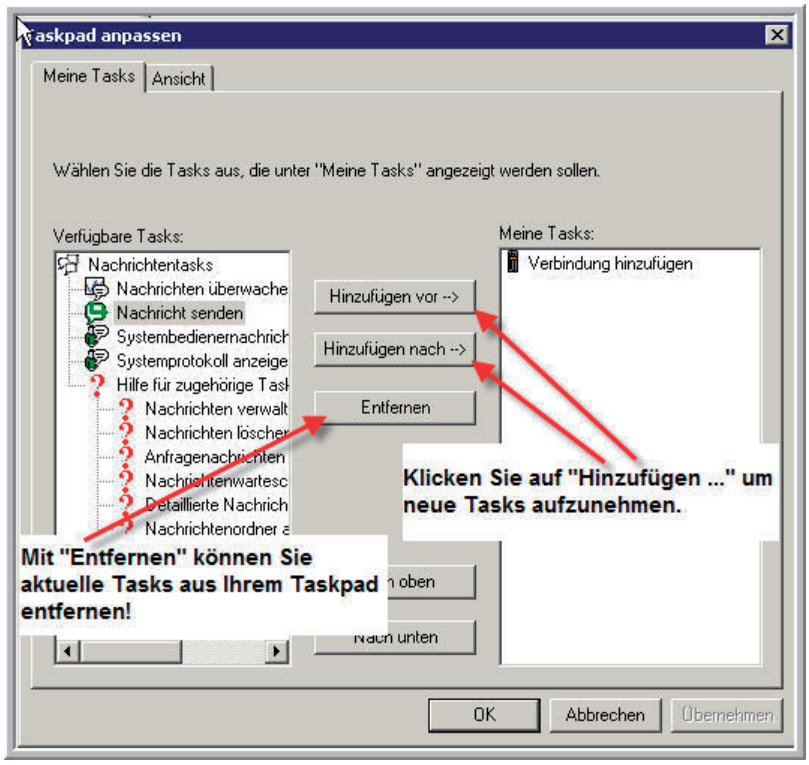
„Taskpad anpassen…“ – Variante 1

Wenn Sie mehrere Tasks einer Funktion hinzufügen möchten, klicken Sie mit der rechten Maustaste (Kontextfunktion) auf eine beliebige Stelle im Taskpad und wählen Sie „Taskpad anpassen…“ aus.



„Taskpad anpassen…“ – Variante 2

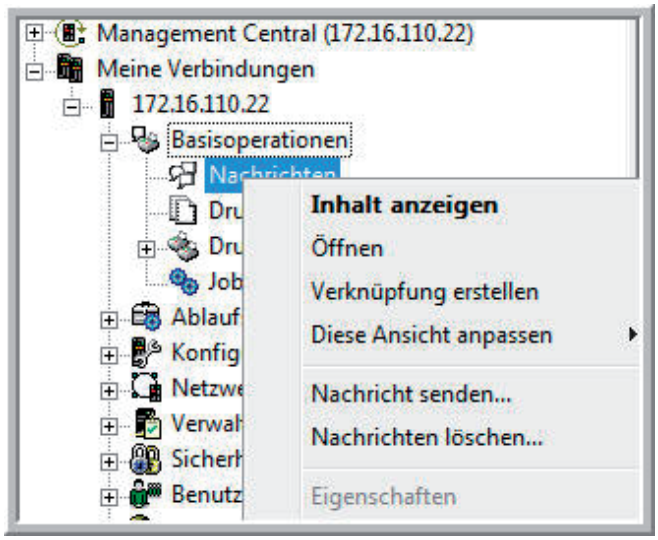
Anschließend öffnet sich ein neuer Dialog, der es Ihnen erlaubt, mehrere Tasks auszuwählen und in „Meine Tasks“ einzufügen.



Dialog „Taskpad anpassen“

Sie können mehrere Tasks auswählen und die Reihenfolge ändern, in denen sie angezeigt werden. Auf diese Weise gestalten Sie Ihre iSeries Navigator-Umgebung sehr individuell; sie müssen die momentan geöffneten Funktionen nicht mehr verlassen, um beispielsweise eine Nachricht zu senden oder zu lesen. Vergleichbar ist diese Funktion mit der Taste ESC im 5250-Dialog, die meistens das ASSIST-Menü startet, ohne dass Sie die eigentliche Arbeit unterbrechen müssen.

Ein weiteres grundlegendes Bedienungselement des iSeries Navigator sind die Kontextmenüs, die wir durch das Drücken der rechten Maustaste erreichen. Einige dieser Kontextmenüs habe ich bereits schon genutzt.



Kontextmenüs nutzen

Sie sehen hier, dass ich im Funktionsbaum auf der linken Seite des Operations Navigator „Nachrichten“ markiert habe; durch Drücken der rechten Maustaste wird dann ein Kontextmenü aufgerufen, das mir die entsprechenden Funktionen für das Arbeiten mit Nachrichten zur Verfügung stellt.

Kontextmenüs existieren für fast alle im Funktionsbaum dargestellten Funktionen, aber auch für die meisten Detailanzeigen.

Zum Schluss möchte ich Ihnen noch eine angenehme kleine Funktion der Kontextmenüs zeigen. In fast jedem funktionsbezogenen Kontextmenü – nicht in denen der Detailanzeigen(!) – finden Sie die Menüauswahl „Verknüpfung erstellen“. Diese Funktion basiert auf folgendem Grundgedanken:

Als Administrator benötigt man nicht ständig den kompletten Funktionsumfang des Operations Navigator. Meistens wird sich die tägliche Arbeit auf relativ wenige, jedoch oft benötigte Funktionen reduzieren. Warum sollten Sie also immer den gesamten Operations Navigator mit seinem gesamten Funktionsumfang öffnen? Ist es nicht besser gezielt auf eine Funktion zuzugreifen?

2.1.6.2
Seite 12

Hier nun setzt die Auswahl „Verknüpfung erstellen“ an. Wenn ich also weiß, dass nur die Basisoperationen benötigt werden, so kann diese Funktion eine Verknüpfung auf dem Desktop erstellen, die den Direktzugriff auf diese Funktion ermöglicht.

Direktzugriff auf die Basisoperationen

Sie sehen, dass es eine ganze Reihe Bedienelemente gibt, die sich an der Windows-Welt orientieren. Insgesamt ist also die Einarbeitungsphase in den Operations Navigator als erfreulich kurz anzusehen.

iSeries Navigator-Basisoperationen

Der Bereich der Basisoperationen umfasst die Systemfunktionen, die von vielen Benutzern der iSeries täglich ausgeführt werden. Hierzu gehören die folgenden Funktionen:

Basisoperationen des iSeries Navigator

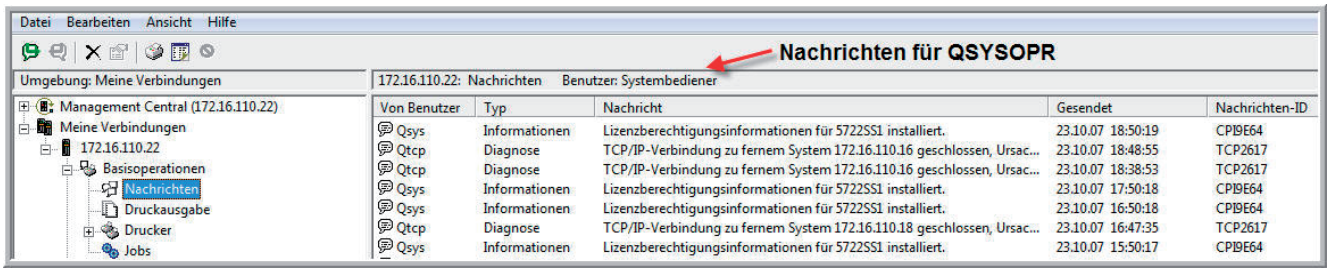
Die in der oberen Abbildung gezeigten Funktionen können im Prinzip als Alternativen für die iSeries CL-Befehle WRKMSG (Work with Messages), WRKSPL (Work with Spoolfiles) WRKWTR (Work with Writers) und WRKUSRJOB (Work with User Jobs) angesehen werden.

2.1.6.3

Seite 2

Mit Nachrichten arbeiten

Im iSeries Navigator können Sie sich eine Liste der Nachrichten anzeigen lassen. Sie legen fest, welche Nachrichten in der Liste angezeigt werden sollen. Sie können beispielsweise alle Nachrichten für den Systembediener aufnehmen. Sie können auch angeben, welche Spalten mit welchen Informationen in der Liste angezeigt werden sollen oder in welcher Reihenfolge diese Spalten angeordnet sein müssen. All das haben wir uns bereits detailliert im vorangegangenen Kapitel angeschaut.

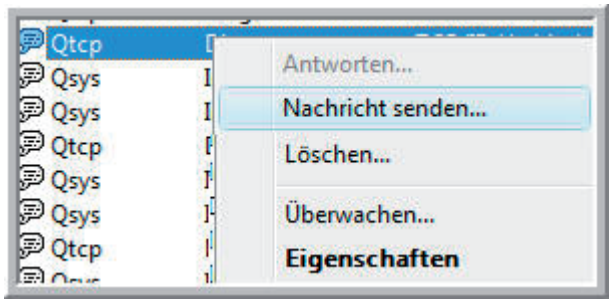


Mit Nachrichten arbeiten

Wenn wir die Funktion „Nachrichten“ markieren, sehen wir auf der rechten Seite des Operations Navigator die Nachrichtenanzeige. Ich lasse mir hier die Nachrichten von QSYSOPR anzeigen. Wir sehen, dass für den Operator eine Reihe von Nachrichten eingegangen sind. Die hier gezeigten Nachrichten sind vom Typ her Informations- oder Diagnose-Nachrichten. Am Nachrichtentext erkennen wir, dass sich die Nachrichten sowohl auf normal als auch abnormal beendete Jobs im System iSeries beziehen.

Der hier teilweise sichtbare Nachrichtentext wird als Text der ersten Ebene (First Level) bezeichnet. Aber als erfahrener iSeries-Anwender kennen Sie diese Anzeige ja aus dem Befehl DSPMSG (Display Message). Ansonsten finden Sie weiterführende Informationen im Kapitel 2.3.1.1 und 2.12.7. Ein Doppelklick auf eine der angezeigten Nachrichten würde dann auch den Nachrichtentext der zweite Ebene anzeigen.

Öffnen wir mit Hilfe der rechten Maustaste das Kontextmenü für eine Nachricht, so werden wir folgendes Bild sehen.



Kontextmenü einer Nachricht

2.1.6.3.1
Seite 2

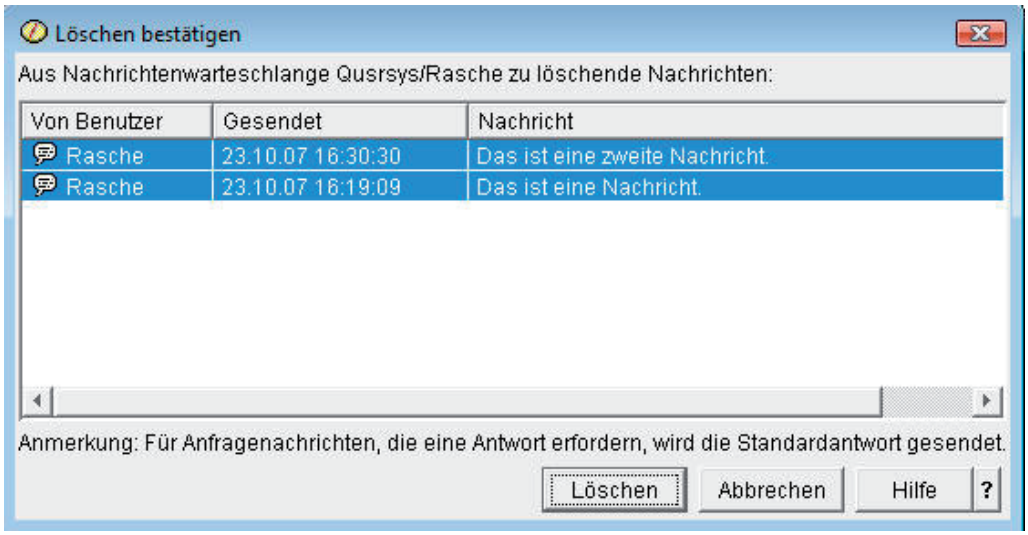
Falls Ihr Kontextmenü anders aussieht, liegt es am Release-Stand oder an eventuell nicht eingespielten Fixes, denn auch über die Fixes werden mittlerweile Funktionen in den iSeries Navigator integriert. Gehen wir einfach die Menüpunkte kurz durch:

Antworten

In meinem Beispiel ist der Menüpunkt deaktiviert, aber Sie vermuten richtig, dass Sie hier auf eine Abfragenachricht (Nachrichtenart *INQ) antworten können. Der Dialog ist im Übrigen selbsterklärend und beinhaltet keine Auswahlparameter.

Löschen

Das Dialogfeld „Löschen bestätigen“ erscheint, sobald die Nachrichten gelöscht werden sollen.



Nachrichten löschen bestätigen

Im Dialog sehen Sie alle Nachrichten, die gelöscht werden sollen. Sie kennen diese Dialoge von den 5250-Bildschirmen. Auch dort erhalten Sie ein separates Fenster, wenn Sie die Option 4 nutzen. Ähnlich dem 5250-Dialog können Sie Einträge zurücknehmen, indem Sie die STRG-Taste gedrückt halten und auf ein beliebiges Element in der Liste klicken, um die Auswahl aufzuheben. Falls Sie eine Nachricht löschen, die eine Antwort erwartet, wird natürlich der Standardwert als Antwort gesendet.

Eine letzte Anmerkung: Falls Sie eine Abfragenachricht oder die Antwort auf eine Abfragenachricht löschen, wird die Nachricht immer zusammen mit der entsprechenden Antwort gelöscht. Nachrichten und Antworten können somit nur paarweise gelöscht werden.

Überwachen

Sie können auch mit der rechten Maustaste auf eine Nachricht klicken und „Überwachen“ auswählen, um eine Nachrichtenüberwachung zu erstellen. Ob diese Funktion zu den Basisfunktionen zu zählen ist, wollen wir an dieser Stelle lieber nicht diskutieren. Sie können zum Beispiel Nachrichtenüberwachungen erstellen, um alle Nachrichtenwarteschlangen auf eine beliebig ausgewählte Nachrichtengruppe hin zu überwachen. Die Überwachung kann auf einem oder auch auf mehreren Systemen ausgeführt werden. Beispielsweise könnten sie die Nachrichtenwarteschlange für Systembediener in einer Gruppe von Systemen auf die Nachricht „CPI0953“ hin überwachen. Diese Nachricht wird gesendet, sobald der Schwellenwert eines Platten-Pools überschritten wird. Bei der Nachrichtenüberwachung können Sie außerdem einen Befehl angeben, der auf dem Server ausgeführt werden soll, wenn die Nachrichtenanzahl einen bestimmten Wert überschreitet. So können Sie beispielsweise angeben, dass nicht mehr benötigte Objekte aus dem Platten-Pool löscht werden, wenn die Nachrichtenüberwachung die Nachricht „CPI0953“ feststellt.

Eigenschaften

Auf der Seite Nachrichteneigenschaften können Sie die Basiseigenschaften der Nachricht anzeigen. Hierzu gehören: Nachrichten-ID, Nachrichtentyp, Sendedatum und weitere Job-Informationen.

Nachricht senden

Hinter dieser Funktion verbirgt sich eine Art grafischer SNDMSG- (Send Message-) Befehl. Aber lassen Sie uns an dieser Stelle ein wenig über die Wirkungsweise der Nachrichtenversendungsbefehle einer iSeries nachdenken.

Im System existieren zwei Nachrichtenbefehle, die dem Benutzerumfeld zuzuordnen sind:

- SNDMSG (Send Message)
- SNDBRKMSG (Send Breakmessage)

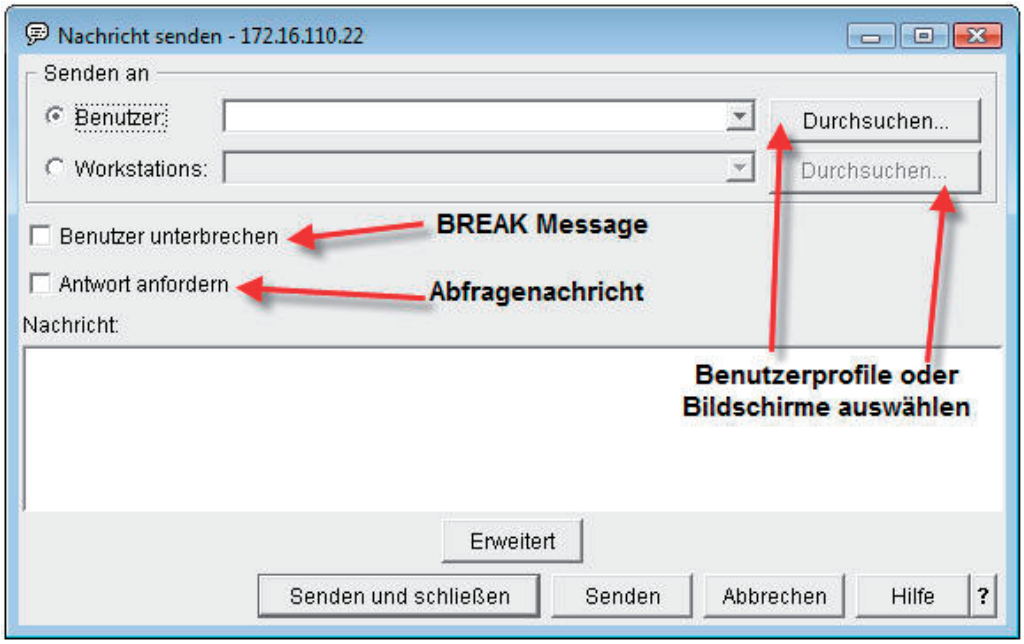
Beide Befehle haben Eigenarten, die Ihren Einsatz einschränken können. Der Befehl SNDMSG sendet eine Benutzernachricht an einen Benutzer. Dabei ist es unerheblich, an welchem Bildschirm sich der Benutzer angemeldet hat. Die Nachricht erreicht die Nachrichtenwarteschlange des Benutzerprofils auf alle Fälle. Jedoch hat die Sache einen Haken: Soll der adressierte Benutzer die Nachricht unbedingt lesen und möglicherweise darauf antworten, so kann der Absender dies nicht garantieren, denn wie die Nachricht vom Empfänger wahrgenommen wird, hängt von der Einstellung der Nachrichtenwarteschlange des Empfängerbenutzerprofils ab. Wenn sich die Nachrichtenwarteschlange des Empfängerprofils nicht im Modus *BREAK befindet, so führt

2.1.6.3.1
Seite 4

das Eintreffen der Nachricht bestenfalls zu einem Signalton (Modus *Notify) oder wird schlimmstenfalls überhaupt nicht wahrgenommen (Modus *Hold).

Soll ein Anwender des Systems iSeries eine Nachricht unbedingt lesen, so bietet sich als nächster Systembefehl der Befehl SNDBRKMSG an. Wie der Name schon sagt, sendet dieser Befehl eine Durchbruchsnachricht. Das heißt, dass der Bildschirm des Empfängers auf alle Fälle die Nachricht anzeigt. Aber auch hier gibt es einen Haken: Als Standard schickt der Befehl SNDBRKMSG seine Nachricht an die Nachrichtenwarteschlange eines Bildschirms – nicht an einen Benutzer. Sende ich also eine Nachricht an einen Bildschirm, so kann es mir passieren, dass der Bildschirm zwar die Nachricht sofort beim Eintreffen anzeigt. Ob aber der gewünschte Benutzer wirklich vor diesem Bildschirm sitzt, kann ich nicht garantieren.

An diesem Punkt setzt die Funktion „Nachrichten senden“ des iSeries Navigator ein.



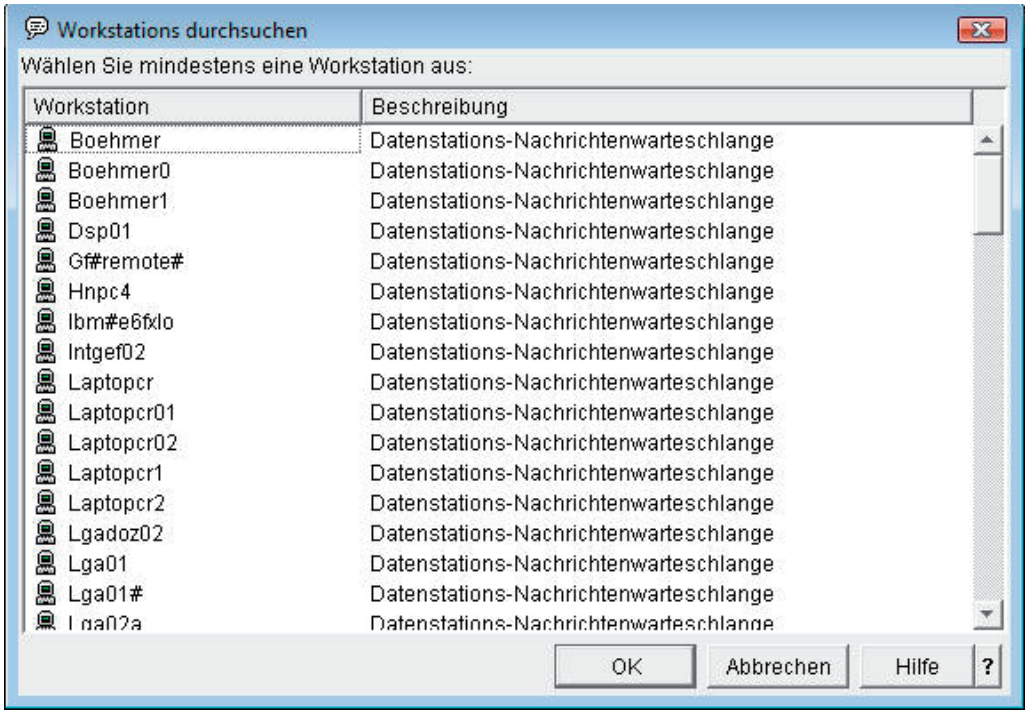
Nachrichten senden

Beabsichtigen Sie die Nachricht an einen Benutzer zu senden, so können Sie den Namen eines bekannten Benutzerprofils in das Dialogfeld „Benutzer“ direkt eintragen. Darüber hinaus stehen Ihnen über das Pulldown-Menü vordefinierte Benutzergruppen zur Verfügung.

Vordefinierte Benutzergruppen

Alternativ ist es auch möglich, den Namen eines iSeries-Benutzers mit dem Dialog „Durchsuchen“ ausfindig zu machen.

Neben der Funktion der Benutzernachrichten können auch Nachrichten an Datenstations-Nachrichtenwarteschlangen abgesetzt werden. Auch hier ist es möglich, die Nachricht direkt an einen Bildschirm zu senden, sofern der Name der Bildschirmeinheit bekannt ist. Als vordefinierte Gruppenadresse gibt es hier nur die Möglichkeit, allen Datenstationen eine Nachricht zu senden – analog dem CL-Befehl SNDBRKMSG. Und natürlich können Sie auch an dieser Stelle des Dialoges einen bestimmten Bildschirm mit der Funktion „Durchsuchen“ ermitteln.



Bildschirmeinheiten suchen

Aber Vorsicht! Sowohl der Dialog „Durchsuchen“ für Benutzer als auch der für Datenstationen haben einen Nachteil. Diese Funktionen können nicht zwischen aktiven Benutzern, Datenstationen und ‚Leichen’ unterscheiden. Nehmen wir als Beispiel die Suchfunktion für Datenstationen: Der Administrator dieses Systems weiß, dass die beiden Datenstationen „Laptopcr01“ und „Laptopcr02“ schon seit einiger Zeit nicht mehr Bestandteil des Systems sind.

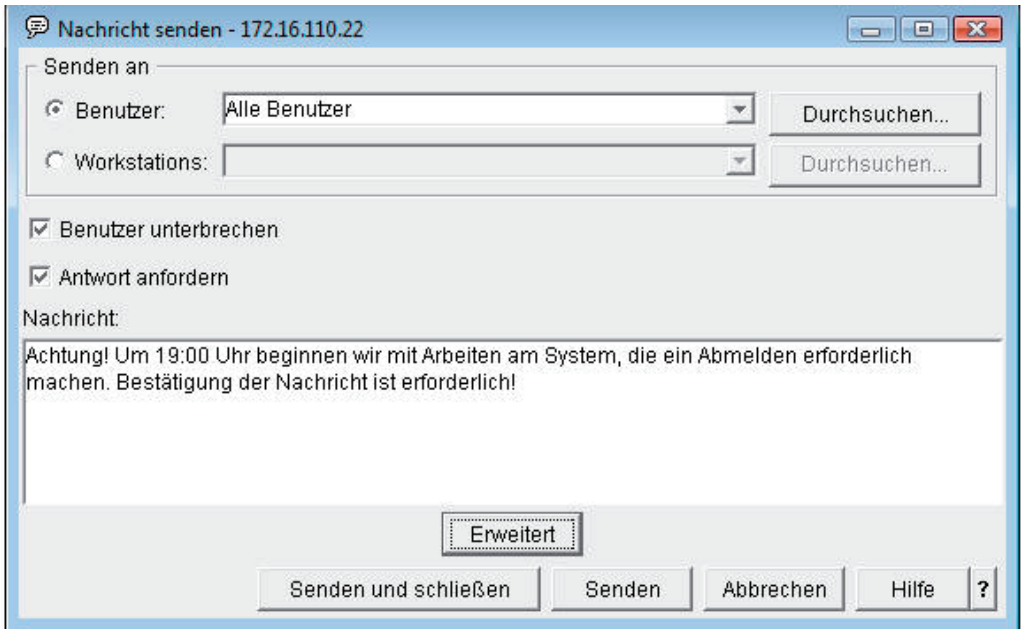
2.1.6.3.1
Seite 6

Diese Datenstationseinheiten wurden bei Versuchen, mittels VPN auf die iSeries zuzugreifen, testweise erstellt. Der Administrator hat jedoch vergessen, diese nicht mehr benötigten Einheiten zu löschen. An was man alles denken muss! Wenn wir jetzt eine Nachricht an diese Bildschirme senden würden, so landen sie im ‚Nirwana'.

Der Parameter „Benutzer unterbrechen" gilt sowohl für Benutzernachrichten als auch für Datenstations-Nachrichtenwarteschlangen. Diese Funktionalität stellt eine erhebliche Erweiterung der Funktion der Systembefehle der iSeries dar und übertrifft auch das Menü ASSIST. Benötigen Sie eine Antwort auf Ihre Nachricht, so markieren Sie den Parameter „Antwort anfordern" durch einen Haken. Nun dürfen Sie nur nicht vergessen, den Nachrichtentext zu tippen; anschließend können Sie den Dialog mit „Senden" oder mit „Senden und schließen" beenden.

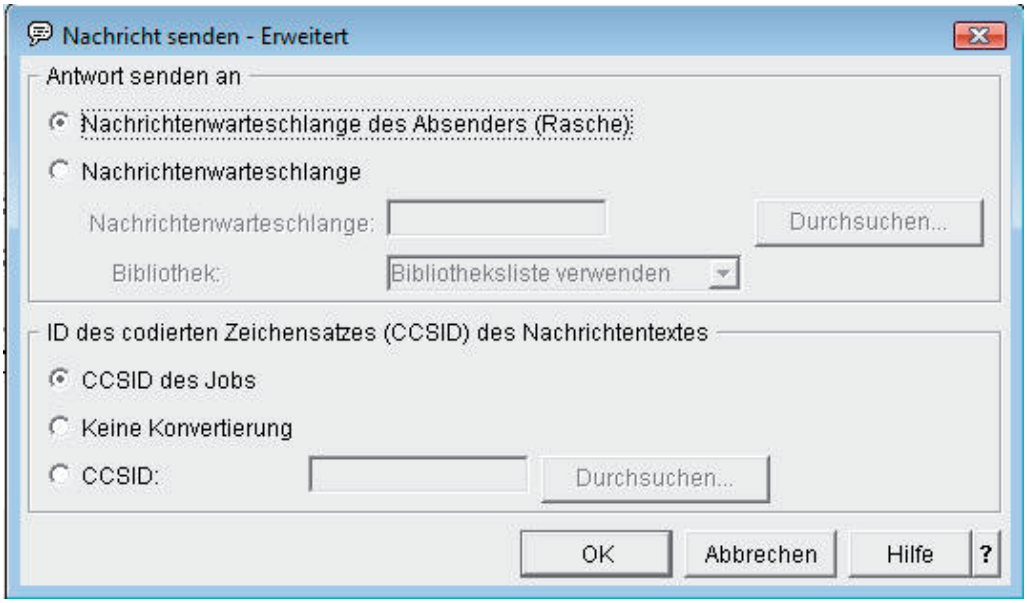
Lassen Sie uns ein kleines Beispiel entwickeln. Sie beabsichtigen um 19:00 Uhr mit Systempflegearbeiten zu beginnen. Darüber wollen Sie alle Systembenutzer informieren. Sie müssen sicherstellen, dass alle Benutzer diese Nachricht sofort beim Eintreffen lesen. Außerdem erwarten Sie von jedem Benutzer, dass er Ihnen die Nachricht bestätigt.

Ihr Dialogfenster sollte die folgenden Eingaben haben:



Beispiel

Das sollte reichen. Sie haben jetzt noch die Möglichkeit, durch Drücken des Buttons „Erweitert" zu definieren, an welchen Benutzer die Bestätigungsnachrichten zu senden sind und welche CCSID verwendet wird.



Erweiterte Attribute

Sie sehen, dass alle Antworten an das Benutzerprofil RASCHE gesendet werden. Standardmäßig erhält der Absender einer Nachricht auch die entsprechenden Antworten. Ist dies nicht gewollt, müssen Sie eine andere Nachrichtenwarteschlange auswählen.

Nachdem dieser Dialog mit „OK“ bestätigt und verlassen wurde, bleibt mir nur noch die Möglichkeit, die Nachricht zu versenden.

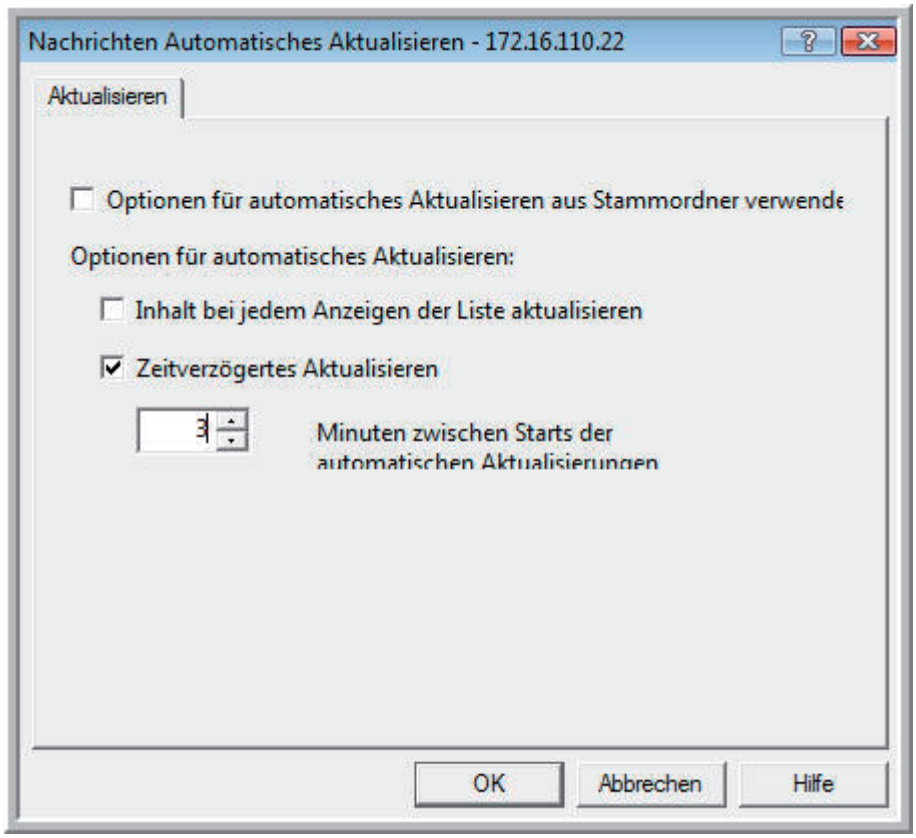
Zum Abschluss dieses Funktionsbereiches ein letzter Hinweis: Die Anzeigen des Operations Navigator sind insgesamt als statische Bildschirme anzusehen – das heißt: Die Anzeigen werden nur zu folgenden Zeiten aktualisiert:

- Wenn wir die Detailanzeige eines Funktionsbereichs öffnen.
- Wenn wir die F5-Taste oder die Funktion „Aktualisieren“ im Menü „Ansicht“ bedienen.

Gerade diese Statik kann aber bei der Nachrichtenfunktion zu Nachteilen führen. Stellen Sie sich vor, Sie bekommen eine wichtige Nachricht von einem Benutzer, der Ihnen mitteilt, dass er mit der Bedienung seines Druckers Probleme hat. Eventuell werden Sie die Nachricht nicht lesen, wenn Sie die Aktualisierungsfunktion der Anzeige nicht auslösen.

Um diesen Nachteil abzufedern, gibt es die Möglichkeit, eine automatische Aktualisierung anzufordern. Nutzen Sie hierfür das Kontextmenü der Nachrichtendetailanzeige und wählen Sie den Menüpunkt „Automatisches Aktualisieren“. Anschließend erscheint der folgende Dialog:

2.1.6.3.1
Seite 8



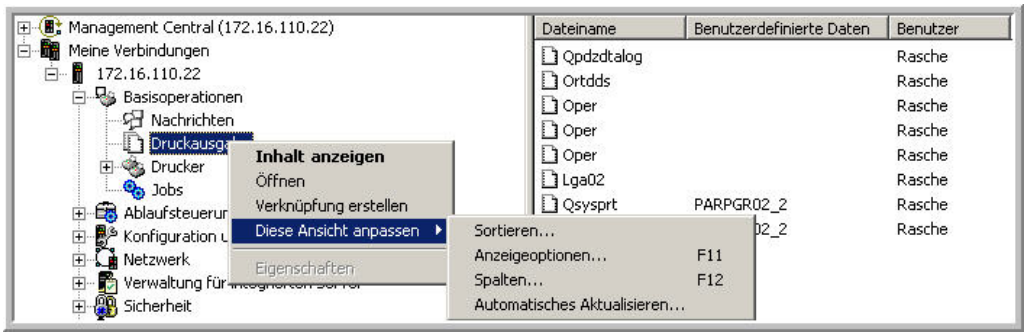
Nachrichten automatisch aktualisieren

Der Standardwert ist „Optionen für automatisches Aktualisieren aus Stammordner verwenden“. Nun bin ich aber nicht der Meinung, dass jeder Dialog in kurzen Intervallen sich eigenständig aktualisieren muss; ich verwende daher lieber – wie in der oberen Abbildung gezeigt – Angaben, die an den kritischen Stellen abweichen. Ich habe hier den Parameter „Zeitverzögertes Aktualisieren“ markiert und festgelegt, dass die Nachrichtenanzeige jeweils nach drei Minuten aktualisiert werden soll. Wählen Sie diesen Intervall aber nicht zu klein, denn durch zu häufige Aktualisierungen erzeugen Sie vermehrt Netzwerkverkehr.

iSeries Navigator Druckausgaben

In diesem Kapitel werden wir uns den Funktionsbereich „Druckausgaben im iSeries Navigator“ anschauen. Grundsätzlich lässt sich der Funktionsumfang mit dem 5250-Befehl WRKSPLF (Work with Spoolfiles) vergleichen.

Wenn Sie den Eintrag „Basisoperationen“ erweitern und dann den Menüpunkt „Druckausgabe“ auswählen, können Sie sich eine Liste der Druckausgaben im iSeries Navigator anzeigen lassen.

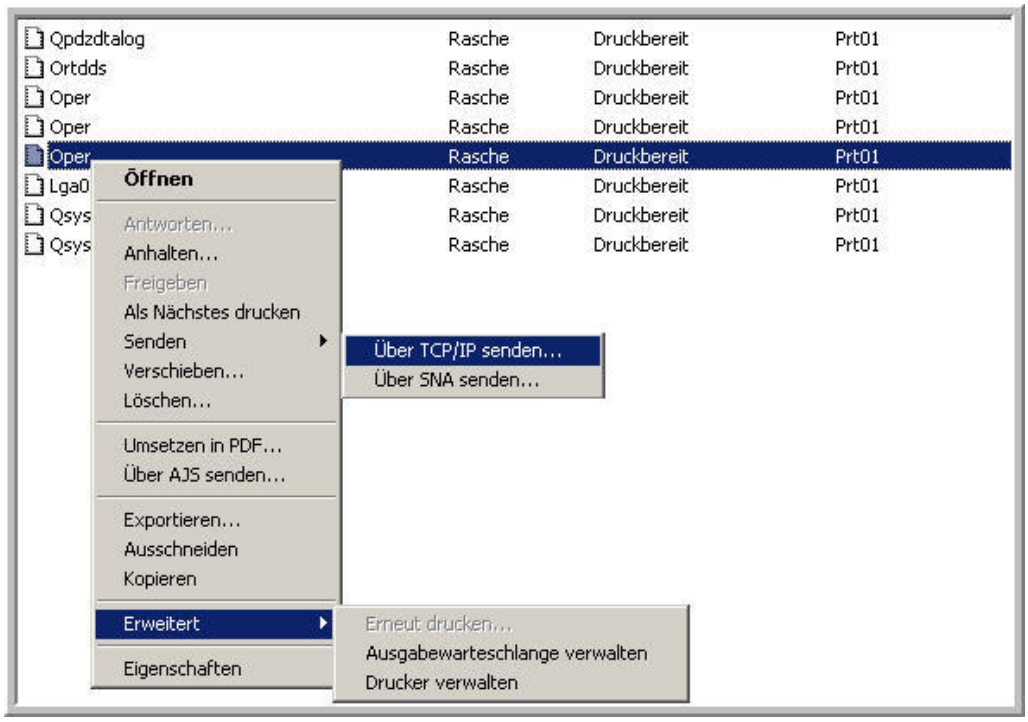


Basisoperationen: iSeries Navigator „Druckausgaben“

Wie bereits im vorangegangenen Kapitel erläutert wurde, können Sie auch hier wieder festlegen, welche Druckausgaben in der Liste angezeigt werden sollen. Die Anzeige erfolgt, wenn Sie den Menüpunkt „Anzeigeoptionen“ nutzen. Alternativ steht die Funktionstaste „F11“ zur Verfügung. Hier können Sie bis zu 20 Benutzer auswählen und diese dann auf die Ausgabewarteschlangen, auf die Drucker, die Jobnamen und die Formulartypen selektieren. Es stehen Ihnen aber auch Kriterien bezüglich des Erstellungsdatums und der -zeit zur Verfügung. Schauen Sie sich die Selektionskriterien einmal in Ruhe an. Sie werden viele Parallelen zum i5/OS-Befehl WRKSPLF erkennen.

2.1.6.3.2
Seite 2

Wenn Sie eine der Druckausgaben markieren und anschließend das Kontextmenü (rechte Maustaste) aufrufen, werden Sie viele Möglichkeiten finden, um die Druckausgaben zu bearbeiten:



Mit Druckausgaben arbeiten

Sie befinden sich jetzt in der Anzeige meiner eigenen Druckausgaben; eine Spool-Datei wurde markiert und die rechte Maustaste gedrückt. Dadurch hat sich das oben dargestellte Kontextmenü geöffnet. Sie können jetzt folgende Druckausgabefunktionen ausführen:

- Anzeigen des Inhalts einer Druckausgabedatei,
- Beantwortung einer Nachricht, die auf die Druckausgabe wartet,
- Anhalten einer Druckausgabedatei,
- Freigeben einer Druckausgabedatei,
- Löschen einer oder mehrerer Druckausgabedateien,
- Versetzen einer Druckausgabedatei an die oberste Stelle in der Ausgabewarteschlange,
- Versetzen einer Druckausgabedatei an eine andere Druckereinheit oder Ausgabewarteschlange.

Da ich davon ausgehe, dass Ihnen der Befehl WRKSPLF bekannt ist, werde ich nicht weiter auf diese Funktion eingehen. Vielmehr möchte ich auf folgende Funktionen aufmerksam machen:

- Änderungen der Spoolfile-Eigenschaften
- Anzeige von Spool-Dateien in der grafischen Umgebung eines Windows PC
- Exportieren einer Druckausgabedatei in Ihr PC-Dateisystem, so dass Sie diese an eine e-Mail anhängen oder in anderen PC-Anwendungen verwenden können
- Sie können eine Druckausgabedatei auch an einen anderen Benutzer oder an ein fernes System senden, auf dem TCP/IP ausgeführt wird.

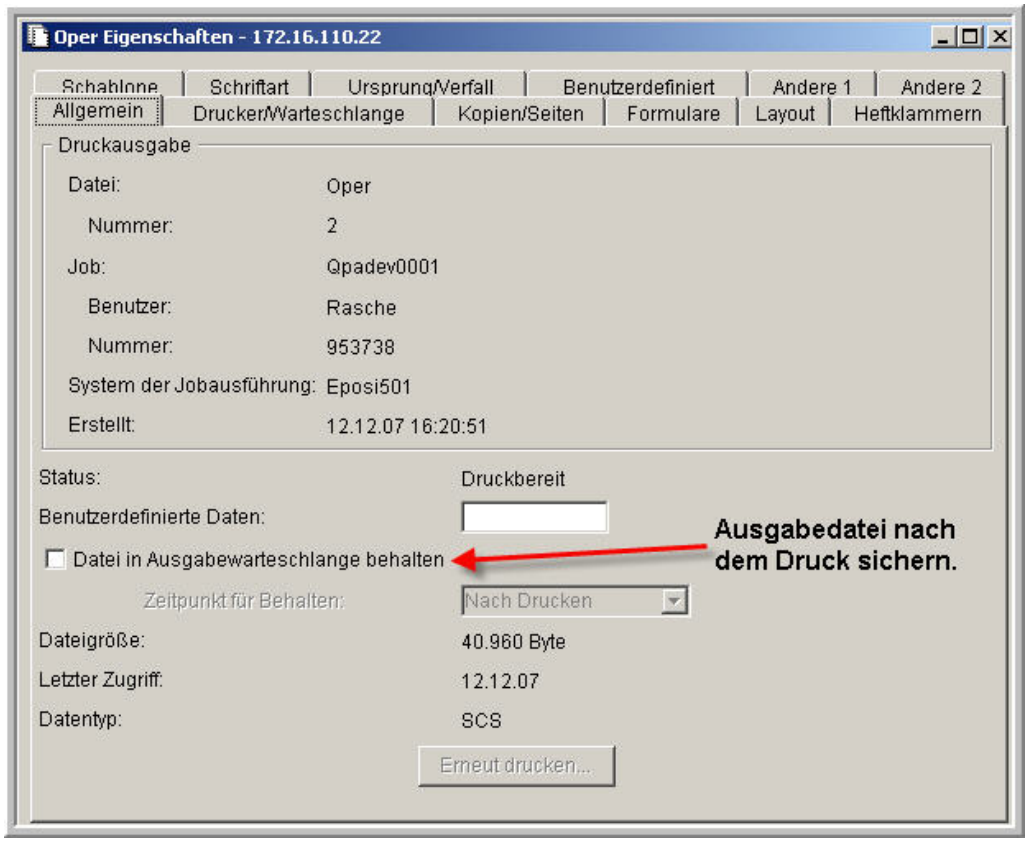
2.1.6.3.2

Seite 4

Druckausgaben – Eigenschaften

2.1.6.3.2.1
Seite 1

Hier beginne ich zunächst mit dem Eigenschaften-Dialog. In diesem Dialog können Sie die Attribute einer Spool-Datei ändern. Auf der iSeries nutzen Sie hierfür den Befehl CHGSPLFA (Change Spoolfile Attributes). Schauen wir uns schwerpunktbezogen mal einige der zahlreichen Register an:

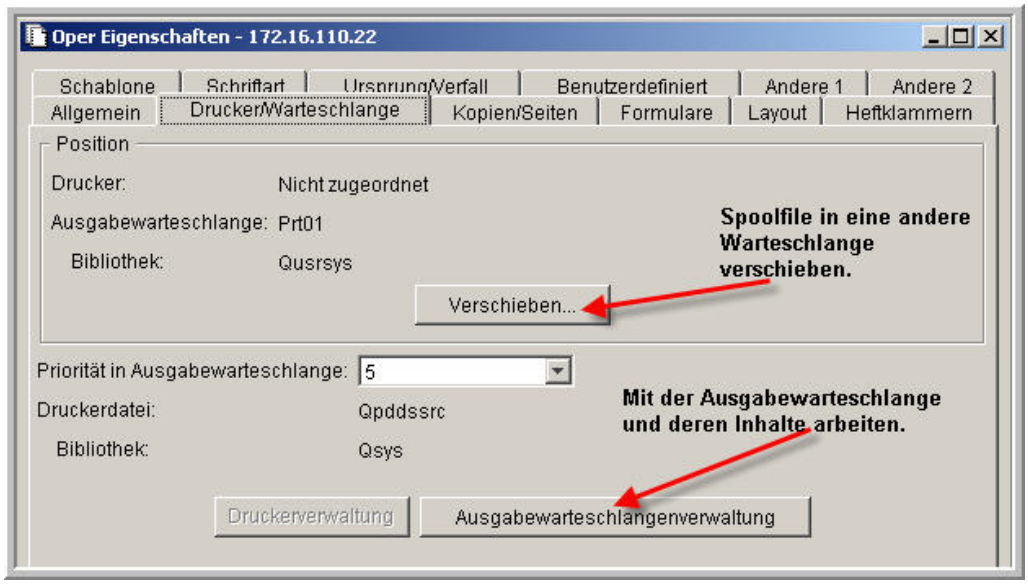


Dialog „Eigenschaften“ – Register „Allgemein“

Auf der ersten Seite finden Sie die Basiseigenschaften der Druckausgabe. Sie sehen den aktuellen Status der Druckausgabe – das heißt: Datum und Uhrzeit der Erstellung, Verarbeitungsbeginn sowie das Datum des letzten Zugriffs. Darüber hinaus können Sie in diesem Register das Attribut „SAVE“ setzen, damit die Druckausgabe nach dem Druck gesichert wird.

2.1.6.3.2.1
Seite 2

Als Nächstes öffne ich das Register „Drucker/Warteschlange“:



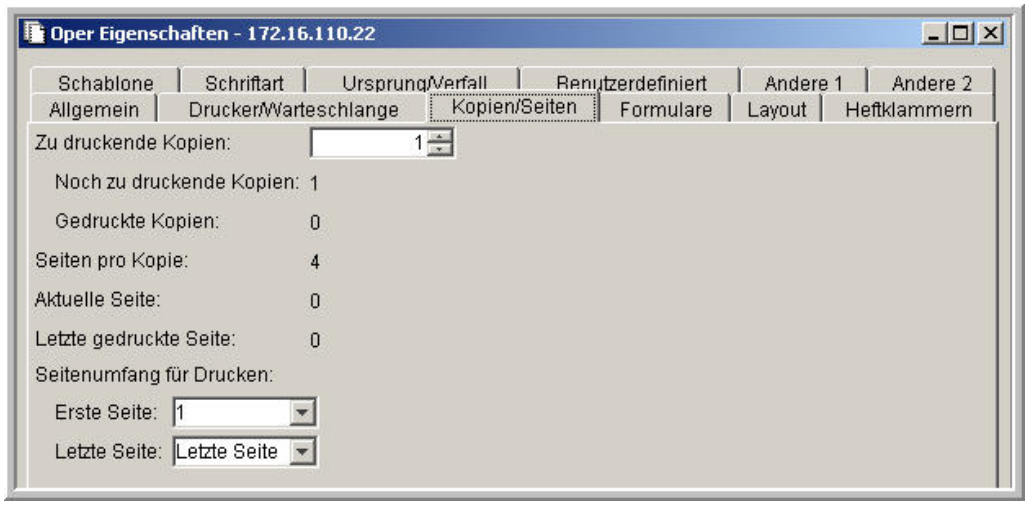
Register „Drucker/Warteschlange“

Auf dieser Seite finden Sie Informationen über die Druckeinheit und die Ausgabewarteschlange. Wenn Sie auf „Verschieben…“ klicken, lässt sich die aktuell markierte Druckdatei in eine andere Warteschlange verschieben.

Außerdem können Sie die Ausgabepriorität verändern und dadurch sowohl einen Druck in der Warteschlange vorziehen als auch in der Rangfolge weiter nach hinten stellen. Wenn Sie mit der Ausgabewarteschlange arbeiten wollen, die der Druckausgabe aktuell zugeordnet ist, dann klicken Sie auf die Schaltfläche „Ausgabewarteschlangenverwaltung“. Sie erhalten anschließend eine Liste aller Druckausgaben, die sich in der Warteschlange befinden. Den Inhalt der Warteschlange können Sie jetzt beliebig bearbeiten – das heißt: Sie können andere Spoolfiles löschen, verschieben usw. – genauso, wie Sie es mit dem Befehl WRKOUTQ gewohnt sind. Die Schaltfläche wird im Übrigen nur angezeigt, wenn die Ablaufsteuerungskomponente (Sonderberechtigung *SPLCTL und *JOBCTL) für den Benutzer aktiviert wurde und der Zugriff aufgrund der Objektberechtigungen zulässig ist.

Im Register „Kopien/Seiten“ finden Sie die Kopie- und Seiteneigenschaften.

2.1.6.3.2.1
Seite 3

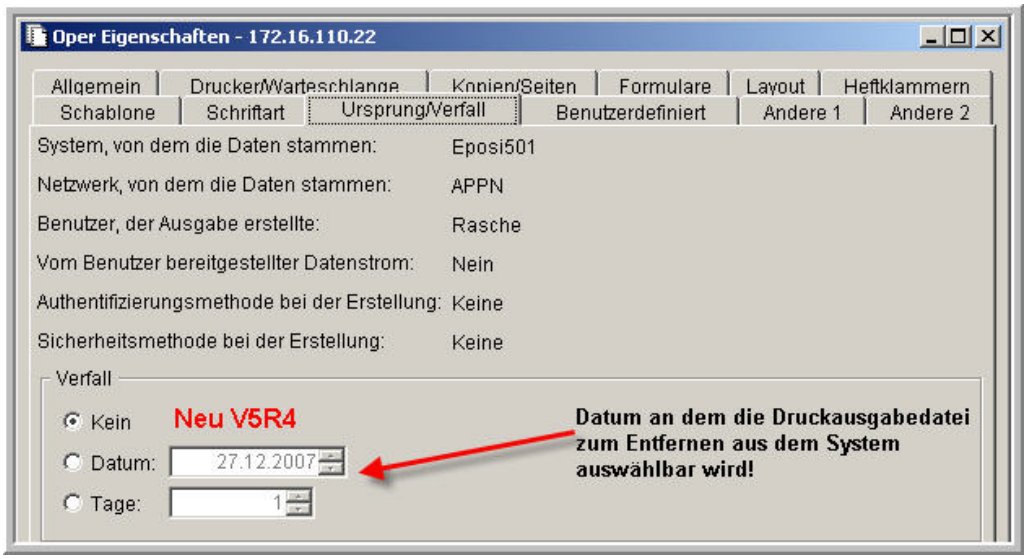


Register „Kopien/Seiten“

Der Parameter „Zu druckende Kopien“ (COPIES) bestimmt die Anzahl der Kopien, die zu drucken sind. Im Ausgabeparameter „Noch zu druckende Kopien“ wird Ihnen der Fortschritt des Druckens angezeigt. Wenn Sie beispielsweise beabsichtigen, fünf Kopien einer Ausgabe zu drucken, und wenn Sie bereits zwei Exemplare gedruckt haben, dann werden Sie in diesem Feld den Wert „3“ sehen. Das ist die Anzahl der noch zum Druck anstehenden Ausgabeexemplare. In diesem Register können Sie außerdem festlegen, ob alle Seiten oder nur ein Teilbereich gedruckt werden sollen. Hierfür nutzen Sie die Parameter „Erste Seite“ und „Letzte Seite“.

2.1.6.3.2.1
Seite 4

Interessant ist in diesem Zusammenhang jetzt nur noch das Register „Ursprung/Verfall“.



Register „Ursprung/Verfall“

In diesem Register werden Informationen über Art und Zeitpunkt der Druckausgabenerstellung angezeigt. Sie erhalten Informationen darüber, auf welchem System die Druckausgabe erzeugt wurde; Sie sehen weiterhin die Netzwerkkennung des erstellenden Systems und den Benutzernamen. Falls die Druckausgabe mit Hilfe eines API (Application Program Interface) erstellt wurde, enthält der Parameter „Vom Benutzer bereitgestellter Datenstrom“ ein „JA“; andernfalls weist der Parameter – wie in meinem Beispiel –den Wert „Nein“ aus. Auch sicherheitsrelevante Informationen können Sie dieser Seite entnehmen: Wenn bei der Erstellung der Druckausgabe Secure Socket Layer (SSL) verwendet wurde, würden Sie als Sicherheitsmethode „SSL3“ vorfinden. Neben den reinen Informationen gibt es seit kurzem im unteren Bildschirmabschnitt Informationen zum Verfallsdatum. Das Verfallsdatum besagt, wann die Druckausgabe zum Entfernen im System ausgewählt wird. Verfallene Druckausgabedateien können Sie mit dem Befehl DLTEXPSPLF (Delete Expired Spooled Files) interaktiv über die Befehlszeile oder über den iSeries Navigator löschen. Grundsätzlich können Sie zwischen zwei Varianten wählen:

- Falls Sie ein Datum angeben, verfällt die Druckausgabedatei um 23:59 Uhr des genannten Datums.
- Falls Sie eine bestimmte Anzahl von Tagen wählen, wird die Spool-Datei entsprechend viele Tage im System aufbewahrt. Beachten Sie bitte, dass das Verfallsdatum wie folgt berechnet wird: angegebene Anzahl Tage plus aktuelles Tagesdatum.

Die meisten der anderen Register enthalten lediglich weitere Informationen über die Druckdatei und deren Attribute; sie sind in der Regel nicht änderbar. Wenn Sie sich diese Parameterwerte in Ruhe ansehen, werden Sie die Bedeutung dieser Parameter aufgrund Ihrer Erfahrungen mit dem CHGSPLFA-Befehl schnell erkennen.

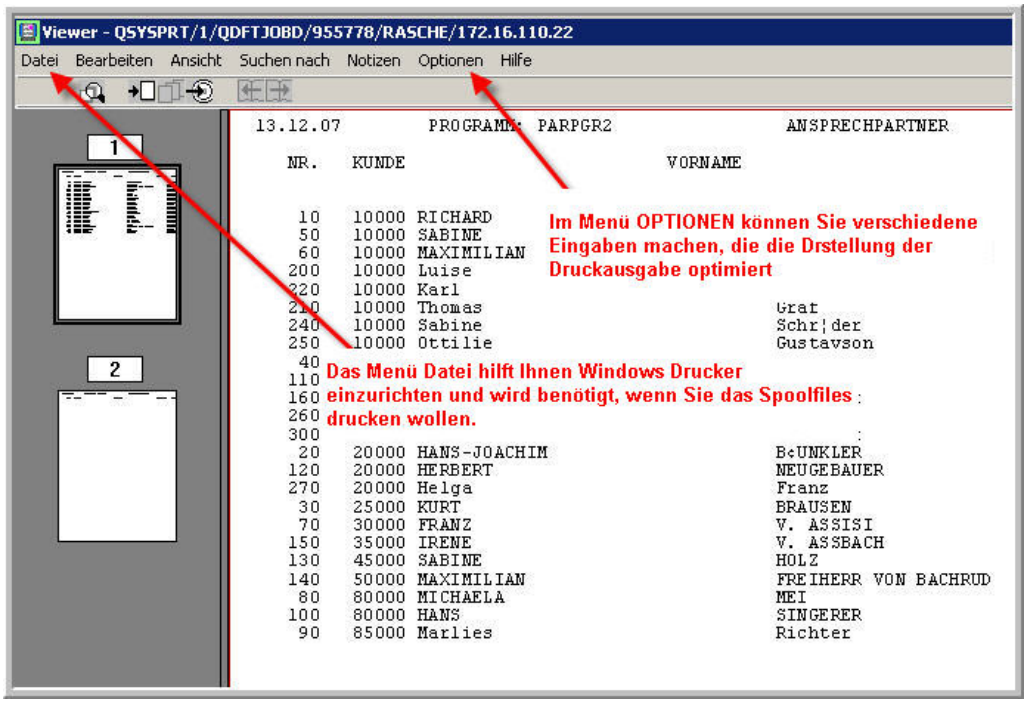
2.1.6.3.2.1

Seite 6

iSeries Navigator – Spool-Dateien anzeigen

Mit Hilfe des iSeries Navigators lassen sich Spoolfiles im grafischen Umfeld anzeigen. Verantwortlich hierfür ist die AFP (Advanced Function Presentation) Workbench für Windows. AFP Workbench ist eine Plattform für die Integration AFP-fähiger Anwendungen und Services. Der AFP Viewer ist eine Komponente dieser Workbench. Kurz gesagt: Diese Komponente des Client Access ermöglicht alle Funktionen, die auch der 5250-Befehl DSPSPLF (Display Spool File) zeigt. Allerdings geht der AFP Viewer weit über die Funktionen des DSPSPLF-Befehls hinaus. So werden Funktionen realisiert, die die Weitergabe von Spool-Dateien an PC-Anwendungen ermöglichen. Es können iSeries Spool-Dateien direkt über einen Windows-Drucker ausgegeben werden, was eine der besten Funktionen des AFP Viewer ist.

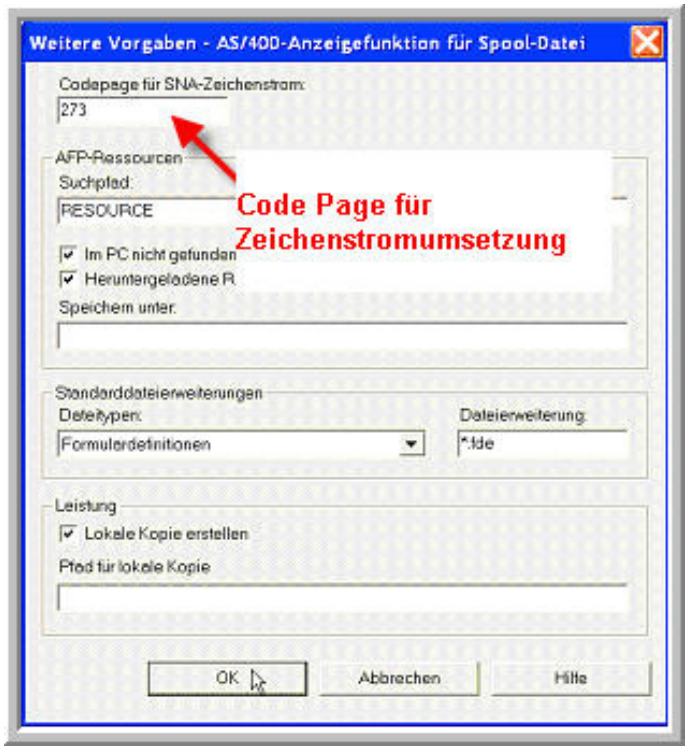
Sobald Sie im Kontextmenü einer Druckausgabe den Menüpunkt „Öffnen“ wählen oder ein Spoolfile einfach durch Doppelklick anzeigen, erscheint der AFP Viewer:



AFP Viewer

2.1.6.3.2.2
Seite 2

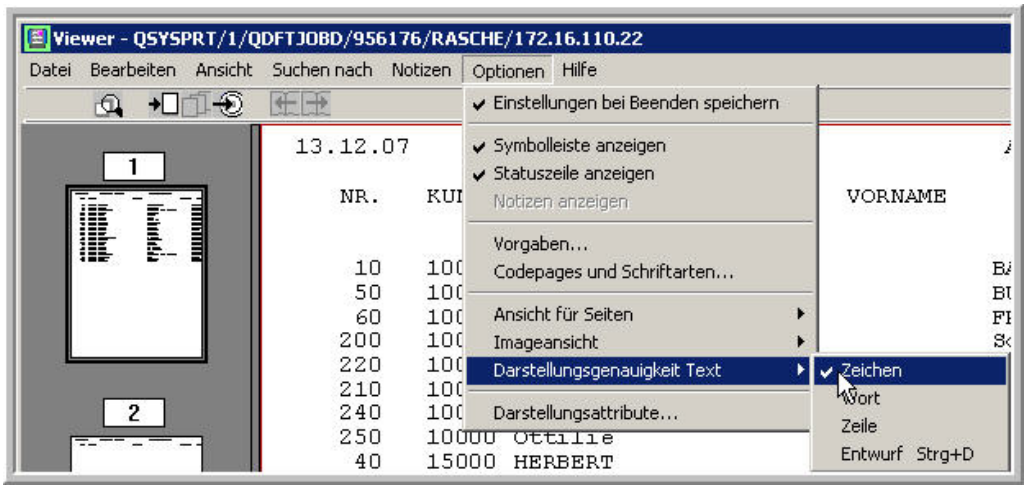
Der AFP Viewer hat eine eigene Menüleiste, die einige Funktionen verbirgt. Zu Beginn gleich zwei kleine Tipps, mit denen Sie die Anzeigen des AFP Viewer verbessern können: Ihnen ist vielleicht aufgefallen, dass die Umlautdarstellung des AFP Viewer in meinem Beispiel nicht korrekt erfolgt. Dieser Fehler lässt sich leicht beheben. Gehen Sie im AFP Viewer in das Menü „Optionen“ und wählen Sie anschließend den Punkt „Vorgaben“. Im Vorgaben-Dialog drücken Sie den Knopf „Weitere“. Danach sehen Sie folgenden Dialog:



Weitere Vorgaben

Tragen Sie in den Parameter „Codepage für SNA-Zeichenstrom“ den Wert „273“ ein. Diese Option wird vom Viewer verwendet, wenn iSeries Spool-Dateien mit dem Attribut *SCS angezeigt werden, um zu bestimmen, wie der Text in der Spool-Datei der PC-Codepage zugeordnet wird. Wird im Feld keine Codepage eingegeben (Default-Einstellung), verwendet der Viewer je nach verwendeter Windows-Version eine beliebige Codepage für die Umsetzung. Beenden Sie abschließend den Dialog mit „OK“. Schließen Sie jetzt auch den AFP Viewer. Beim nächsten Öffnen des Viewer sollten die Umlaute korrekt angezeigt werden.

Wenn Sie den Eindruck haben, dass die Darstellungsgenauigkeit des AFP Viewer zu wünschen übrig lässt, dann können Sie die Darstellungsgenauigkeit erhöhen, indem Sie die Option „Darstellungsgenauigkeit Text“ wählen.



Darstellungsgenauigkeit

Mit dieser Einstellung können Sie die Leistungsfähigkeit der Anzeige verbessern. Folgende Werte stehen Ihnen hierbei zur Verfügung:

- Zeichen
  Jedes Zeichen wird separat mit der besten Darstellungsgenauigkeit positioniert.
- Wort
  Jedes Wort wird mit guter Genauigkeit positioniert.
- Zeile
  Defaulteinstellung: Jeder Textblock wird mit normaler Darstellungsgenauigkeit positioniert.
- Entwurf
  Es findet keine Anpassung statt – das heißt: Es findet keine Anpassung zwischen den einzelnen Zeichen statt.

Die Auswahlmöglichkeiten stellen unterschiedliche Präzisionskategorien für die Textdarstellung im AFP Viewer dar. Nutzen Sie die Einstellung „Zeichen“, um eine optimale Darstellung des Textes zu gewährleisten.

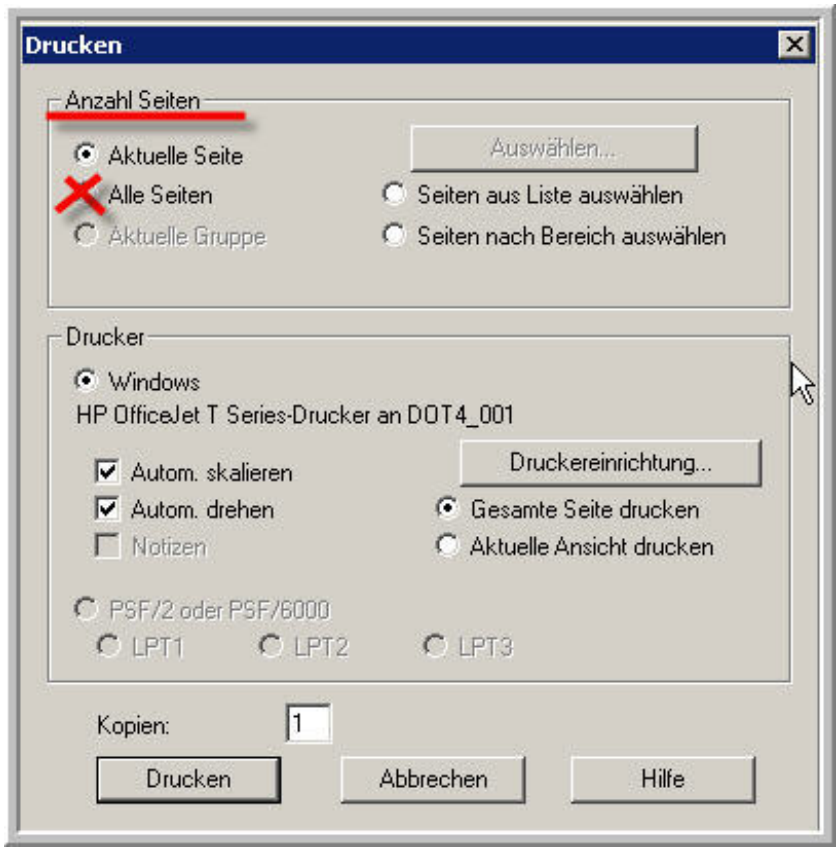
Wie eingangs bereits erwähnt, können Sie mit Hilfe des AFP Viewer auch iSeries Spool-Dateien direkt über einen Windows-Drucker ausgeben.

2.1.6.3.2.2
Seite 4

Öffnen Sie hierfür zunächst das Menü „Datei“ und wechseln Sie dann in den Menüpunkt „Druckereinrichtung“. Es sollte daraufhin folgender Dialog erscheinen:

Druckereinrichtung

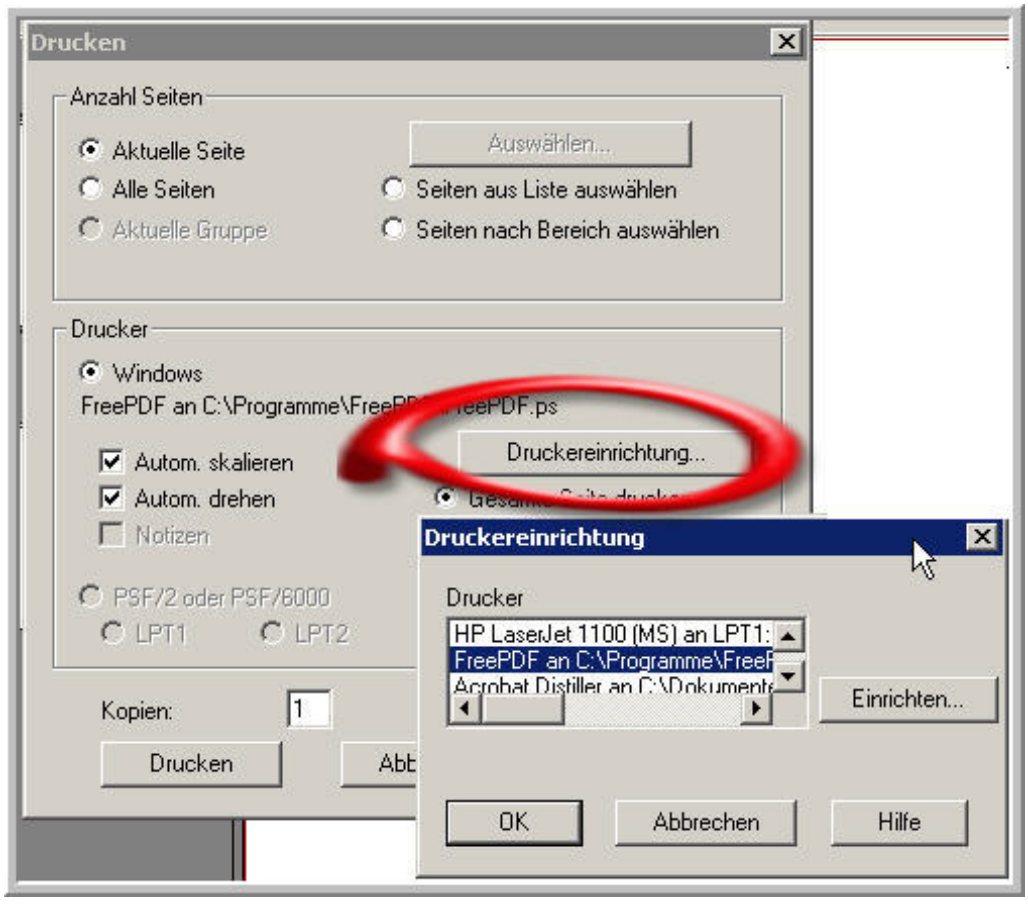
Sie sehen, dass ich hier „HP OfficeJet“ ausgewählt habe. Diese Auswahl bestätige ich mit „OK“; anschließend wechsle ich in das Menü „Datei/Drucken“.



Spoolfile drucken

Der Dialog erinnert stark an die üblichen Windows-Dialoge und sollte daher keine Probleme bereiten. Sie sollten allerdings darauf achten, dass der Viewer per Default nur die aktuelle Seite druckt. Wenn Sie also alle Seiten benötigen, ändern Sie die Selektionskriterien entsprechend. Sobald Sie die Anzeige mit der Auswahl „Drucken“ bestätigen, startet Windows den Druckprozess.

Auf diese Art und Weise lassen sich auch problemlos PDF-Dokumente erzeugen. Dieses Mal werde ich einen kürzeren Weg wählen, um das Spoolfile als PDF-File auszugeben. Dafür wechsle ich sofort in das Menü „Datei/Drucken“. Natürlich erscheint wieder der Dialog aus der oberen Abbildung. Hier gehe ich nun direkt in die Druckereinrichtung, indem ich den Knopf „Druckereinrichtung“ drücke. Es erscheint ein ebenfalls bekannter Dialog, der es uns ermöglicht, einen Windows-Drucker zu selektieren:

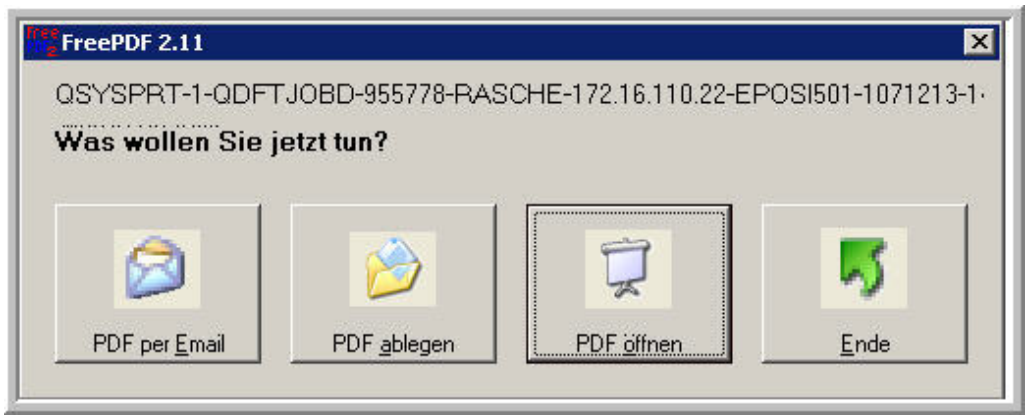


PDF-File erstellen

Ich entscheide mich im Beispiel für den Drucker „FreePDF“. Voraussetzung für die PDF-File-Erstellung ist ein PDF-Konverter. Sie können sich auch für ein kostenfreies Produkt entscheiden, um PC-Ausgaben ins PDF-Format zu konvertieren, oder Sie können den Acrobat PDFwriter nutzen, der Bestandteil des kostenpflichtigen Produkts Acrobat ist. Welche Software Sie unter Windows einsetzen, ist für das Ergebnis unbedeutend. Der Druckprozess kann wieder auf die bereits beschriebene Weise gestartet werden. Je nachdem mit

2.1.6.3.2.2
Seite 6

welchem Produkt Sie arbeiten, erhalten Sie nun den normalen Standarddialog des PDF-Writer.



PDF-File verarbeiten

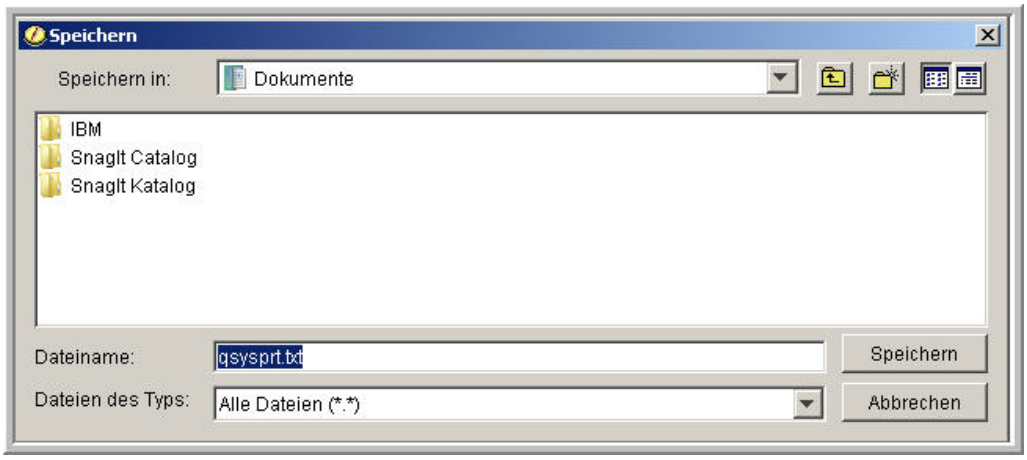
In meinem Beispiel werde ich aufgefordert, das PDF-File zu speichern, per e-Mail zu versenden oder das PDF-File zu öffnen. Ich entscheide mich für die letzte Variante, denn schließlich will ich ja sehen, ob die Konvertierung funktioniert hat.

Das Ergebnis

Sie sehen, dass der AFP Viewer viele kleine nützliche Funktionen hat, deren Einsatz lohnenswert sind.

iSeries Navigator – Druckausgaben exportieren

Mit Hilfe des iSeries Navigator lassen sich die Druckausgaben aber auch ganz einfach in das PC-Dateisystem exportieren. Sie können dort anschließend mit Windows-Programmen bearbeitet oder als e-Mail-Anhang verschickt werden. Die Druckausgaben werden beim Export automatisch in ASCII-Textdateien umgewandelt. Gehen Sie wie folgt vor, um ein Spoolfile zu exportieren: Klicken Sie mit der rechten Maustaste auf ein Spoolfile und wählen Sie anschließend im Kontextmenü „Exportieren“ aus. Daraufhin erscheint folgender Windows-Dialog:



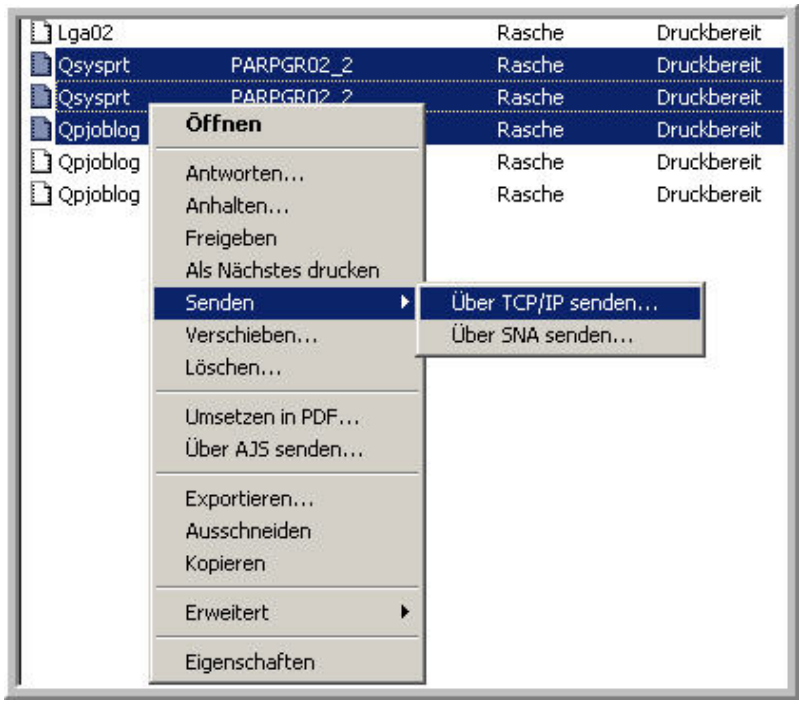
Exportverzeichnis wählen

Sie können ein beliebiges Windows-Verzeichnis und auch den Namen der Textdatei frei wählen. Danach klicken Sie auf „Speichern“, um die ASCII-Datei auf Ihrem PC zu erstellen.

Abschließend ein kleiner Tipp: Sie können das Spoolfile auch durch „Drag&Drop“ oder mit „Kopieren“ und „Einfügen“ in ein Windows-Verzeichnis speichern. Die Umsetzung in ASCII erfolgt dabei auch automatisch.

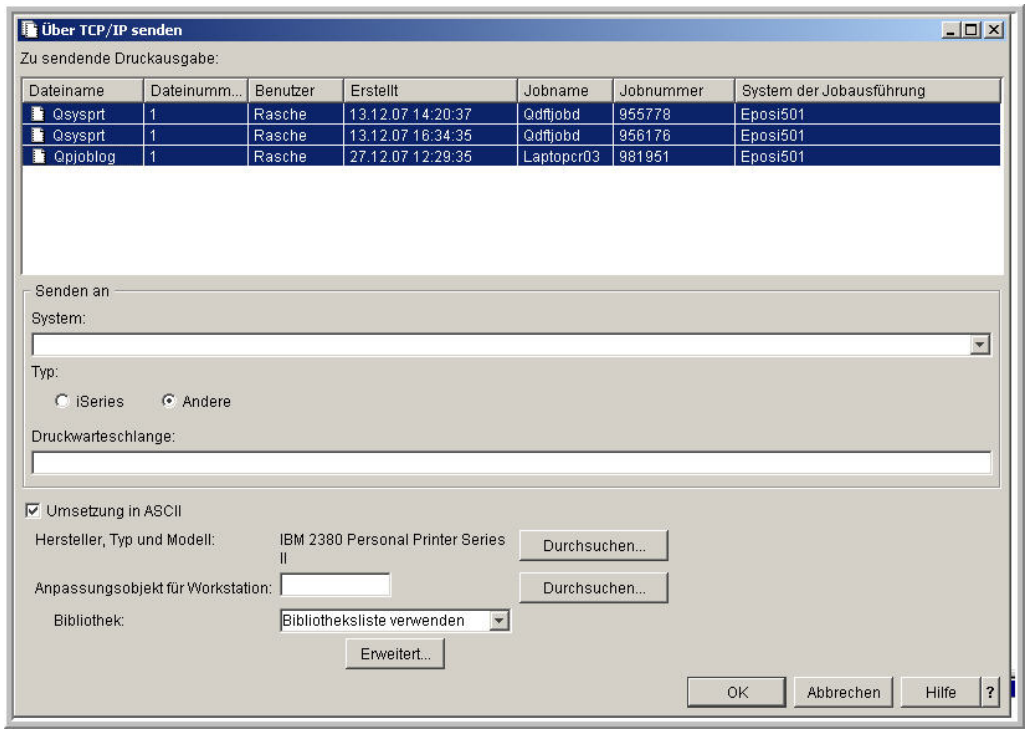
Etwas komplexer ist der Dialog „Über TCP/IP senden“ gestaltet. Er erlaubt aber dafür auch den Austausch der Spoolfiles zwischen zwei iSeries-Servern oder mit einem anderen Server-System. Auch die Umsetzung in ASCII kann individuell gesteuert werden. Um ein oder mehrere Spoolfiles über TCP/IP zu senden, sollten Sie zunächst wieder eine oder mehrere Spoolfiles markieren.

2.1.6.3.2.3
Seite 2



Spoolfiles auswählen und über TCP/IP senden

Anschließend erhalten Sie eine Anzeige der Druckausgaben, die Sie senden wollen:



Parametereinstellungen

Wenn Sie einen Eintrag in der Liste aufheben wollen, haben Sie jetzt die Möglichkeit dazu, indem Sie die Taste STRG gedrückt halten und gleichzeitig mit der Maus auf den entsprechenden Listeneintrag klicken.

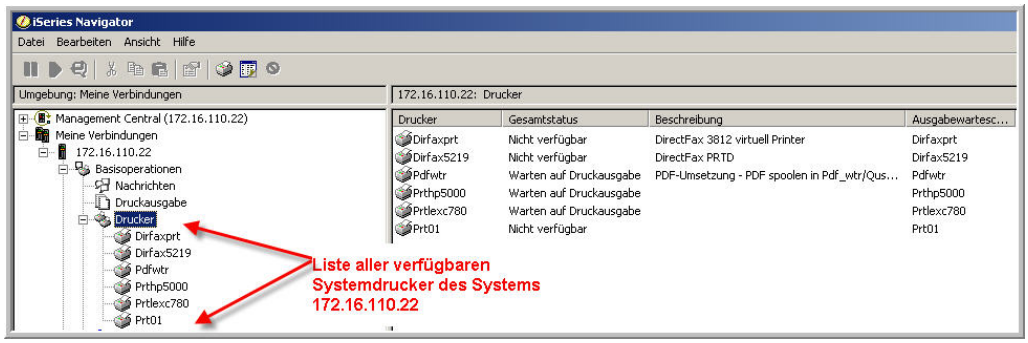
Im Parameter „Senden an“ müssen Sie den Namen des fernen Systems angeben. Dabei ist es unerheblich, ob Sie den Netzwerknamen oder die IP-Adresse verwenden. Außerdem benötigen Sie den Namen der Ausgabewarteschlange, wenn es sich beim Zielsystem um einen iSeries Server handeln sollte. Wenn Sie als Zielsystem ein anderes System wählen, benötigen Sie auch hierfür den Namen einer Druckerwarteschlange. Dieser Name ist abhängig vom Zielsystem, allerdings achten manche Systeme auf Groß- und Kleinschreibung. Wenn der Name der Druckerwarteschlange „case-sensitive“ ist, sollten Sie den Namen in einfache Anführungszeichen einschließen. Abschließend geben Sie noch den Hersteller, den Typ und das Modell des Druckers an, der für die Druckerdatenstromumsetzung im Host verwendet wird. Detaillierte Informationen zum Datenaustausch über TCP/IP können Sie dem Kapitel „Client-Connectivity“ entnehmen.

2.1.6.3.2.3

Seite 4

iSeries Navigator – Mit Druckern arbeiten

Im iSeries Navigator können Sie sich eine Liste der Systemdrucker anzeigen lassen. Wie immer haben Sie die Möglichkeit, die anzuzeigenden Drucker über zahlreiche Selektionskriterien auszuwählen (F11). Grundsätzlich ähnelt dieser Funktionsbereich dem 5250-Befehl WRKWTR. Sie können auch hier Drucker starten und stoppen, Nachrichten beantworten, einen Drucker anhalten, freigeben oder sich die Druckereigenschaften anzeigen lassen. Der iSeries Navigator-Funktionsbereich „Drucker“ ermöglicht somit die Steuerung von iSeries-Systemdruckern.



Liste aller Systemdrucker

Sie sehen in der oberen Abbildung alle Systemdrucker, die am System 172.16.110.22 angeschlossen sind. Im rechten Fensterbereich können Sie den Druckerstatus erkennen, der Ihnen zeigt, ob ein Drucker gestartet oder beendet ist.

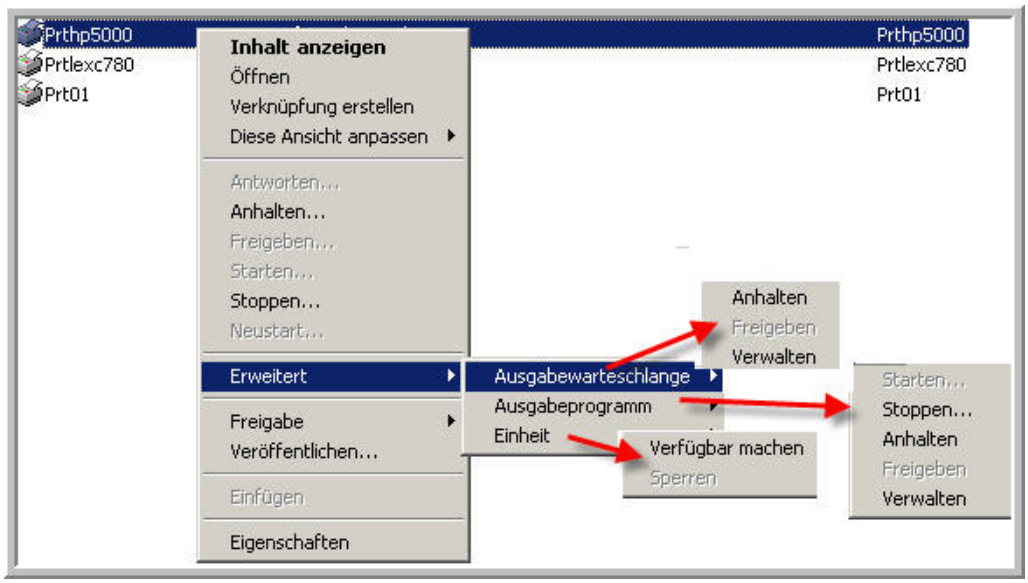
Schauen wir uns zum Vergleich noch einmal die Ausgabe des Befehls WRKWTR (Work Writer) an:

Ausgabe des Befehls WRKWTR

2.1.6.3.3
Seite 2

Sie sehen, dass der Status „END“ eines Druckers im iSeries Navigator mit „Nicht verfügbar“ abgebildet wird. Dagegen ist der Status „STR“ gleichbedeutend mit „Warten auf Druckausgabe“. Doch eines unterscheidet die beiden Darstellungen: Im Navigator können Sie sehen, mit welcher Ausgabewarteschlange der Drucker verbunden ist.

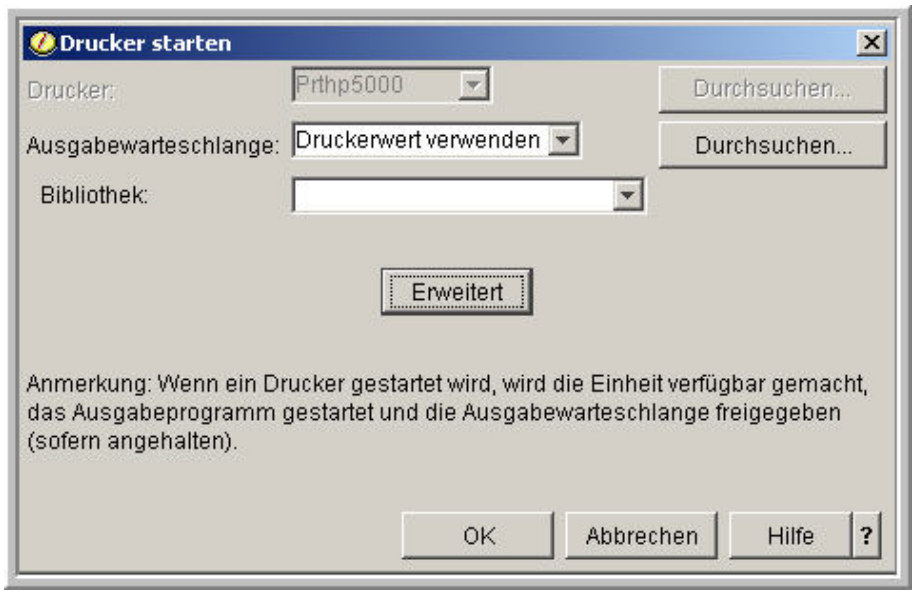
Schauen wir uns jetzt das Kontextmenü des Druckers PRTHP5000 im iSeries Navigator im direkten Vergleich zum Befehl WRKWTR an:



Kontextmenü des Druckers PRTHP5000

- Inhalt anzeigen bzw. Öffnen
  In beiden Fällen werden die zur Verarbeitung anstehenden Spoolfiles dieses Druckers angezeigt. Allerdings wird beim Menüpunkt „Öffnen“ hierfür ein eigenständiges Fenster geöffnet, während der Menüpunkt „Inhalt anzeigen“ die rechte Fensterhälfte für die Darstellung nutzt. Diese Menüpunkte entsprechen wohl am ehestem der Option 8 „Mit Ausgabewarteschlange arbeiten“.
- Verknüpfung erstellen
  Das ist – wie ich finde – eine recht nützliche Funktion. Sie erlaubt es nämlich, ein Icon für diese Funktion in einem Ordner oder auf dem Desktop zu erstellen. So können Sie sich bei Problemen mit diesem Drucker oder einer Druckausgabe ganz einfach den Inhalt anzeigen lassen, ohne den Navigator starten zu müssen. Eine vergleichbare Funktion fehlt dafür natürlich im 5250-Umfeld.
- Antworten
  Entspricht der Option 7 „Nachrichten anzeigen“ des WRKWTR-Befehls. Hierüber können Sie sich einfach die Nachrichten eines Druckers anzeigen lassen und – falls erforderlich – auch gleich beantworten.

- Anhalten
  Dieser Menüpunkt ist selbsterklärend. Er entspricht der Option 3 des 5250-Befehls.
- Freigeben
  Hierüber können Sie einen angehaltenen Drucker wieder freigeben. Dies entspricht der Option 6 des WRKWTR-Befehls.
- Starten
  Das Druckausgabeprogramm wird im Subsystem QSPL gestartet, falls der Drucker derzeit gestoppt ist.



Drucker starten

Sie wundern sich vielleicht über die wenigen Startparameter? Keine Sorge – die Angaben wurden nicht vergessen, sondern nur hinter dem Button „Erweitert“ versteckt. Falls Sie also Angaben zum Formulartyp oder den Ausgabeprogrammen machen wollen, schauen Sie sich einfach mal diese erweiterten Parameter an. Dort werden Sie finden, was Sie jetzt noch vermissen. Es gibt keinen Unterschied zur Option 1 „Starten“ des WRKWTR Befehls.

- Stoppen
  Das Druckausgabeprogramm wird genauso beendet, als hätten Sie die Option 4 verwendet oder den Befehl ENDWTR.
- Erweitert
  Im Kontextmenü „Erweitert“ wurden gleich mehrere Funktionen zusammengefasst. So können Sie über den Eintrag „Ausgabewarteschlange“ mit der zugeordneten Ausgabewarteschlange arbeiten – analog dem Befehl WRKOUTQ. Diese Funktion entspricht wohl am ehesten der Option 8 „Mit Ausgabewarteschlange arbeiten“. Allerdings ist das Kontextmenü umfangreicher, da es zusätzlich erlaubt, die Ausgabewarteschlange

anzuhalten beziehungsweise eine angehaltene Ausgabewarteschlange wieder freizugeben. Auch der Befehl CLROUTQ wurde im Kontextmenü integriert. Außerdem können Sie hier mit dem Ausgabeprogramm arbeiten, es aber auch stoppen und starten. Zusätzlich ist es möglich, die zugehörige Einheit anzuhängen, um Sie verfügbar zu machen.

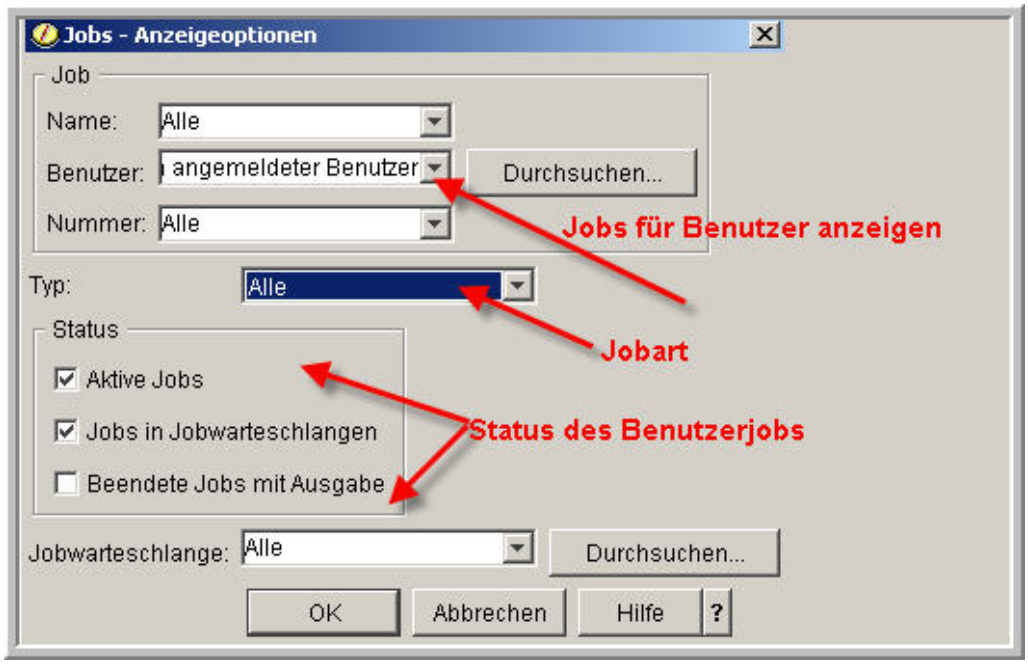
- Freigabe
  Ermöglicht die Druckerfreigabe durch iSeries NetServer in einer Windows-Netzwerkumgebung. In den folgenden Dialogen können Sie die Eigenschaften einer bestehenden Freigabe anzeigen und konfigurieren. Auch hierfür existiert keine alternative Option im Befehl WRKWTR.
- Veröffentlichen
  Voraussetzung für die Veröffentlichungen eines Druckers ist die Konfiguration eines LDAP-Servers. Anschließend können Sie die Einstellungen über diesen Eintrag anzeigen oder ändern. Eine entsprechende Option im Befehl WRKWTR existiert nicht.
- Eigenschaften
  Hier erhalten Sie alle Informationen zu einem Drucker: Informationen über die Ausgabewarteschlange, zum Ausgabejob und zur Druckeinheit aber auch Informationen zur aktuellen Datei, die derzeit gedruckt wird und deren Eigenschaften. Die Informationen sind vergleichbar mit denen, die Sie erhalten, wenn Sie die Option 5 „Arbeiten mit“ verwenden.

Sie sehen, dass Sie keine Funktion der 5250-Bildschirme im iSeries Navigator vermissen werden. Im Gegenteil: Häufig bietet der iSeries Navigator mehr Komfort. Er ist aufgrund der grafischen Darstellung übersichtlicher und beinhaltet sogar Funktionen, die in den entsprechenden 5250-Bildschirmen nicht integriert werden. Es könnte sich durchaus lohnen, den iSeries Navigator häufiger zu nutzen.

iSeries Navigator – Mit Benutzerjobs arbeiten

Die Funktion „Jobs“ in den Basisoperationen des Operations Navigator entspricht weitgehend den Funktionen des CL-Befehls WRKUSRJOB. Dieser Befehl dient im i5/OS der Suche nach Jobs, die sich im System befinden. Wir nutzen den Befehl WRKUSRJOB auch im Kapitel 2.6.6 für diese Aufgabe. Auch jetzt möchte ich wieder nach allen Jobs des Benutzers „Rasche“ auf dem System suchen. Die Suche betrifft sowohl Jobs, die in einer Jobwarteschlange auf ihre Verarbeitung warten, als auch solche, die derzeit aktiv vom System bearbeitet werden. Weiterhin gehören Jobs dazu, die auf ihre Ausgabe in einer Ausgabewarteschlange warten.

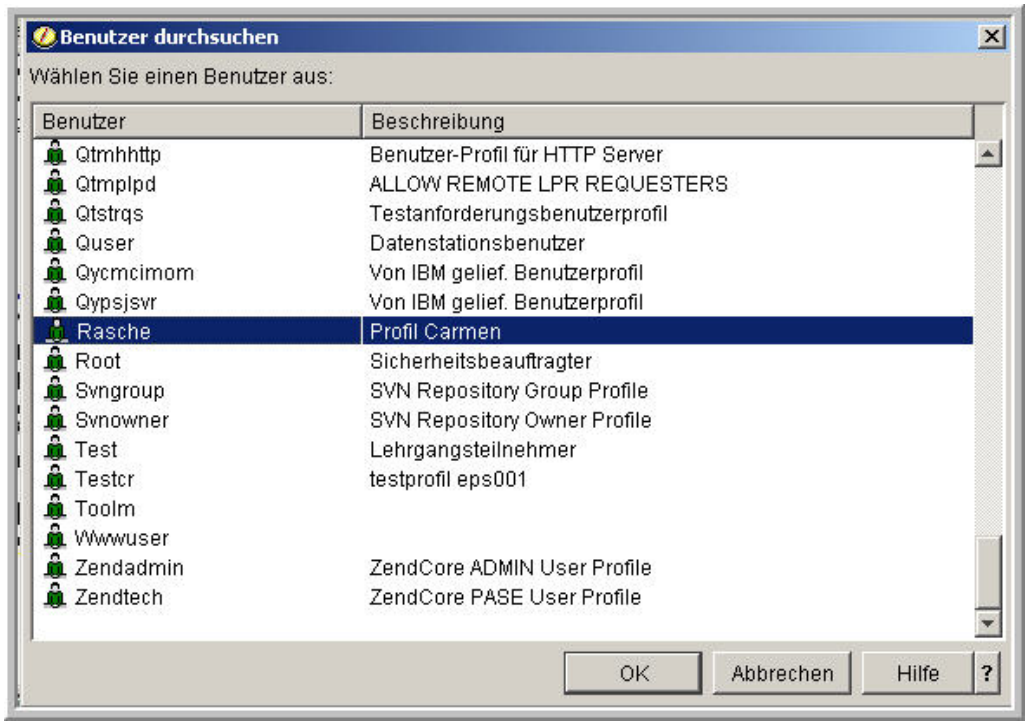
Wie gehe ich vor? Zuerst einmal muss ich dafür sorgen, dass ausschließlich die Jobs für den Benutzer „Rasche“ angezeigt werden. Ich gehe daher zunächst einmal in die Anzeigeoptionen (F11). Der jetzt folgende Dialog hat sehr viel Ähnlichkeit mit dem Bedienerführungsdialog des CL-Befehls WRKUSRJOB (vgl. 2.6.6):



Job-Anzeigeoptionen

Innerhalb der Parametergruppe „Job“ gibt es den Einzelparameter „Benutzer“, in dem der Standardwert „Aktueller Benutzer“ eingetragen ist. Das bedeutet, dass als Standard die Jobs angezeigt werden, die dem angemeldeten Benutzer zuzuordnen sind. Diesen Wert werde ich ändern, indem ich den Knopf „Durchsuchen“ nutze, so dass sich der Dialog „Benutzer“ öffnet.

2.1.6.3.4
Seite 2



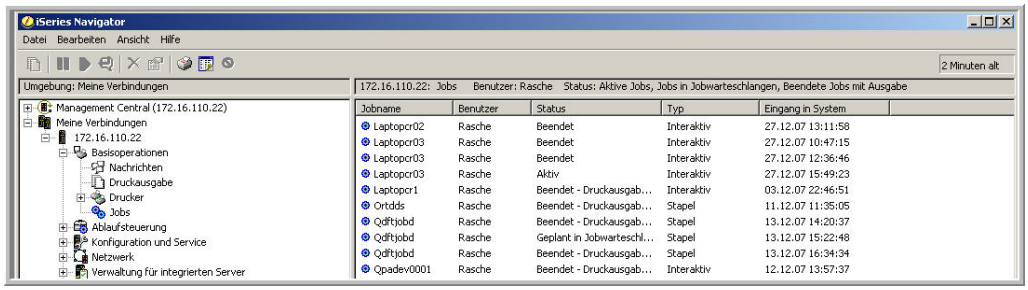
Benutzer suchen

Hier wähle ich den Benutzer „Rasche“ und bestätige die Eingabe mit „OK“.

In der Parametergruppe „Status“ aktiviere ich anschließend noch die folgenden Parameter:

- Aktive Jobs anzeigen – Ja
- Jobs in Jobwarteschlange anzeigen – Ja
- Beendete Jobs mit Druckausgabe anzeigen – Ja

Danach kehre ich in die iSeries Navigator-Anzeige zurück. In der Anzeige erhalte ich jetzt alle Jobs des Benutzers „Rasche“.



Jobs des Benutzers „Rasche“

In der Anzeige auf der linken Seite erkennen Sie die Jobs des Benutzers „Rasche“ im System 172.16.110.22. Sie können auch sehen, dass einige der angezeigten Jobs zur Druckausgabe anstehen. Einige dieser Jobs sind als Stapeljobs (Batchjobs) durch das System gelaufen. Einer der Jobs – Jobtyp „Geplant in Jobwarteschlange“ – ist ein Scheduler Job, der auf die Verarbeitung wartet. Das ist jetzt interessant. Denn wir sehen nach, ob dieser Job ein Jobprotokoll hat – und außerdem welcher Aufruf sich dahinter verbirgt. Dazu markiere ich den Job mit der rechten Maustaste, es öffnet sich das Kontextmenü.

Kontextmenü eines Jobs

Folgende Funktionen sind je nach Jobstatus möglich:

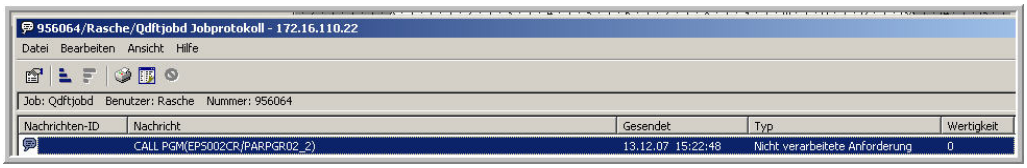
- Überwachen:
  Der Job ist auf seinen Ressourcenverbrauch hin zu überwachen.
  Diese Funktion werde ich zu einem späteren Zeitpunkt im Rahmen des Management Central besprechen.
- Druckausgabe:
  Druckausgaben für einen Job anzeigen.
- Jobprotokoll:
  Jobprotokoll für einen Job anzeigen.
- Details
- Antworten:
  Anstehende Nachrichten für einen Job beantworten.
- Anhalten:
  Laufenden Job anhalten.
- Freigeben:
  Angehaltenen Job wieder freigeben.

- Verschieben:
  Job in eine andere Jobwarteschlange, in ein anderes Subsystem oder in eine andere Ausgabewarteschlange verschieben.
- Löschen/Beenden:
  Job sofort oder kontrolliert beenden.
- Eigenschaften:
  Eigenschaften eines Jobs anzeigen. Aus diesem Dialog entnehmen Sie die genauen Kenndaten des beobachteten Jobs.

Jobeigenschaften

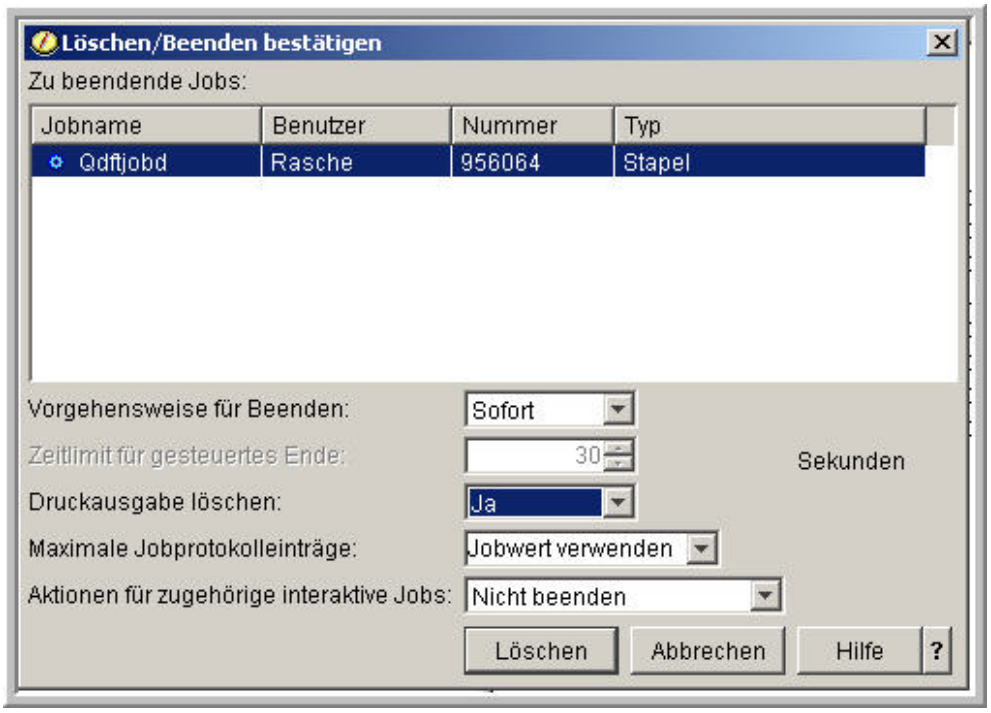
Sie können erkennen, dass der Job vom Benutzer „Rasche“ am 12.12.07 ins System gestellt und bereits einmal interaktiv ausgeführt wurde. Schauen sie sich bei Gelegenheit die anderen Register und deren Informationen an. Vieles erinnert an die Informationen, die Sie erhalten, wenn sie mit einem Job im WRKACTJOB-Befehl arbeiten.

Mich interessiert aber noch, welcher Programmaufruf sich hinter dem Scheduler-Job befindet. Deshalb öffne ich jetzt das Jobprotokoll:



Jobprotokoll des Scheduler-Jobs

Jetzt können wir erkennen, dass das Programm „PARPGR02_2“ in der Bibliothek „EPS002CR“ aufgerufen wird. Hierbei handelt es sich um ein Demo-Programm, dass ich in einem meiner Seminare einsetze. Der Job wird also momentan nicht gebraucht; er kann gelöscht beziehungsweise beendet werden.

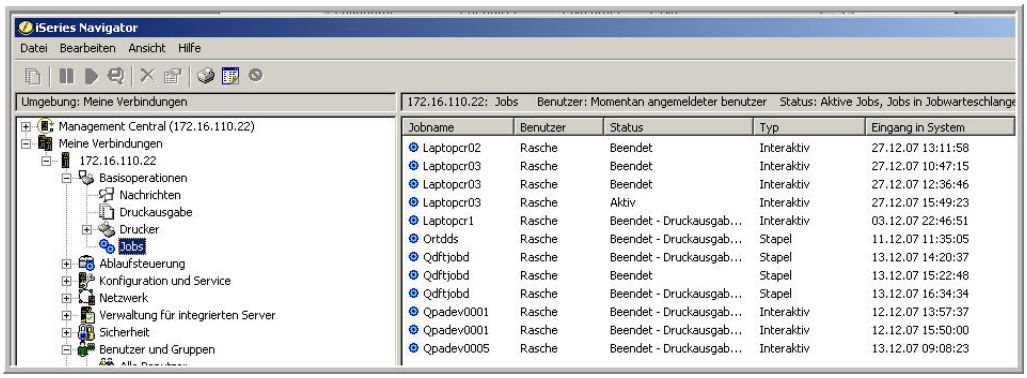


Job beenden

Ich habe mich dafür entschieden, den Job sofort (*IMMED) zu beenden. Die kontrollierte Beendigung steht selbstverständlich auch zur Verfügung. Außerdem möchte ich alle Druckausgaben, die dieser Job erstellt hat, sofort löschen.

2.1.6.3.4
Seite 6

Wenn Sie das Ergebnis kontrollieren möchten, dann schauen Sie sich einfach die aktualisierte Ausgabe an.



Ergebnis

Sie sehen, dass bereits die Basisoperationen des iSeries Navigator viele komfortable Möglichkeiten bieten – angefangen von der Nachrichtenverwaltung über die Verwaltung der Druckausgaben und Drucker bis hin zur Verwaltung der Benutzerjobs. Vermutlich werden Sie sich jetzt fragen, warum dies zu den Basisoperationen zählt. Außerdem werden Sie feststellen, dass einige der gezeigten Funktionen sehr weit reichen und auf keinen Fall für den Anwender geeignet sind. Nun – darüber lässt sich sicherlich lange und ausführlich diskutieren. Fakt ist, dass es der Anwender sehr viel einfacher hat, seine Nachrichten und Druckausgaben zu verwalten, da viele Bildschirme des Navigators intuitiv zu bedienen sind. Die Gefahr kann dahingehend beschränkt werden, dass Sie dem Anwender über das Benutzerprofil nur Zugriff auf wenige iSeries Navigator-Funktionen erlauben. Darüber hinaus ist es für den Anwender nicht möglich, fremde Jobs und fremde Druckausgaben zu steuern, wenn Sie den Ressourcenschutz aktiviert haben. Unter diesen Voraussetzungen ist der Einsatz des Navigators sicherlich zu überdenken. Sie könnten sich und Ihren Kollegen zukünftig viel Arbeit und Zeit sparen, wenn einige der Operator-Aufgaben auf andere Ebenen verlagert werden. Es könnte sich zumindest lohnen einmal darüber nachzudenken!

Das System starten und stoppen

iSeries-Server sind dafür bekannt, dass sie selten gestoppt und erneut gestartet werden müssen. Für manche Wartungsvorgänge oder Systemänderungen kann es aber doch einmal erforderlich sein, die Verarbeitung des iSeries-Server zu stoppen und anschließend ein einleitendes Programmladen (IPL) durchzuführen. Das Booten einer iSeries wird IPL genannt. IPL ist die Abkürzung für „Initial Program Load“ oder einleitendes Programmladen. Sie können den Startvorgang einer iSeries mit dem Bootvorgang Ihres PC vergleichen. Allerdings werden Sie feststellen, dass ein IPL sehr viel mehr Zeit in Anspruch nimmt. Denn während der iSeries-Server startet:

- wird der Plattenspeicher neu organisiert,
- wird neue Hardware erkannt,
- werden neue Lizenzprogramme erkannt,
- werden PTFs aktiviert,
- werden Zugriffspfade geprüft und bei Bedarf neu erstellt,
- werden geänderte Systemwerte erkannt u.v.m.

Dies sind nur einige Aufgaben, die Ihre iSeries ausführt, während sie startet.

Während des IPL erhalten Sie am „Control Panel“, der Steuerkonsole der iSeries, Informationen darüber, welche Aufgaben durchgeführt werden. Lassen Sie uns an dieser Stelle zunächst den Begriff der Steuerkonsole bzw. des Control Panel klären.

7. Ergänzung 06/2006

2.2
Seite 2

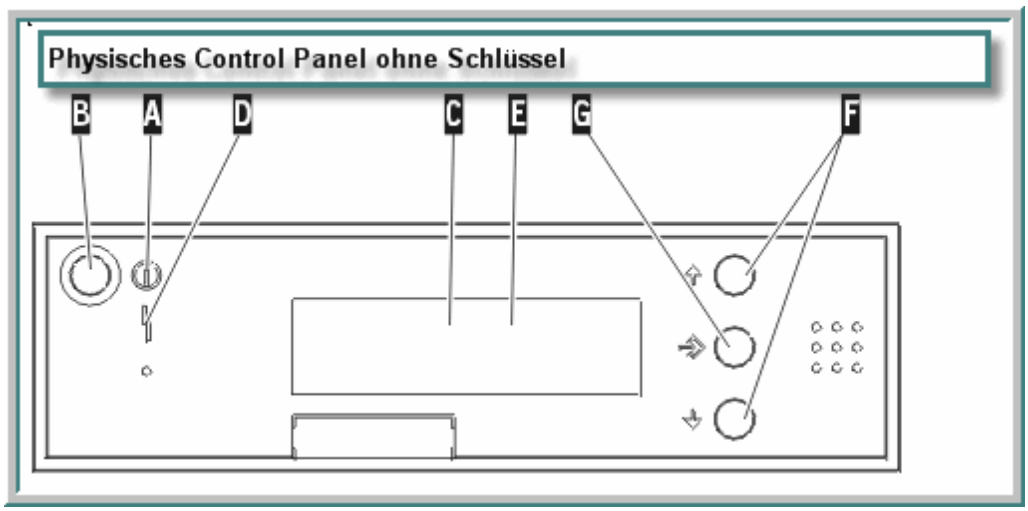
Das Control Panel

Das Control Panel ist ein Interface zur iSeries. Mit Hilfe des Control Panel können Sie den Server ein- und ausschalten, die IPL-Art und den Betriebsmodus bestimmen und viele weitere Funktionen durchführen. Je nach Konfiguration unterscheidet man zwischen physischem Control Panel, Remote Control Panel und virtuellem Control Panel.

2.2.1 Seite 2

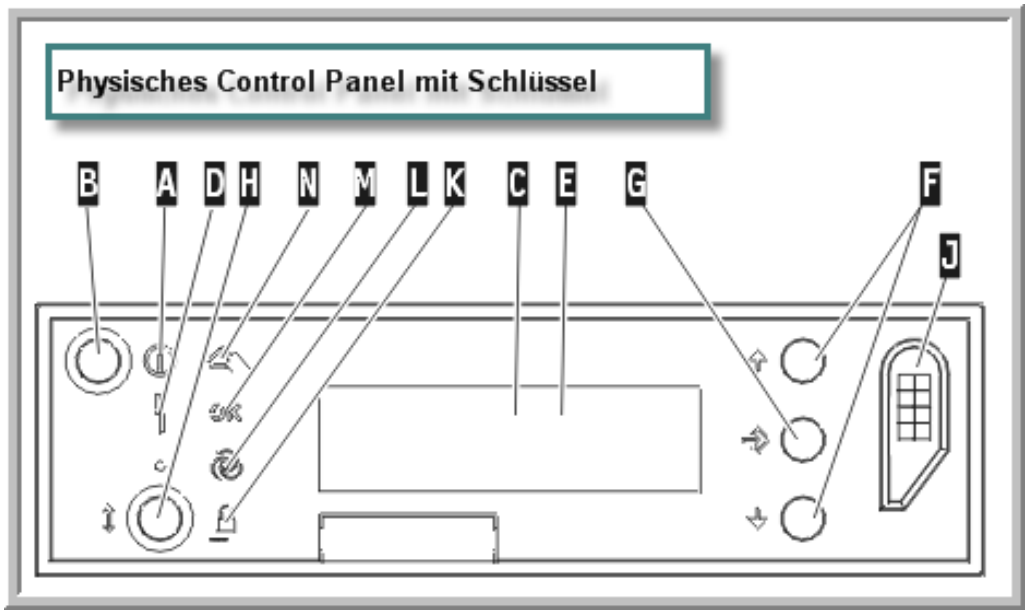
Physisches Control Panel

Das physische Control Panel ist das direkte Interface zur iSeries, da es sich an der Maschine selbst befindet.



Physisches Control Panel ohne Schlüssel

(A)	System an/aus Leuchte
(B)	Netzschalter
(C)	Datenanzeige
(D)	Achtung! Bedienereingriff erforderlich
(E)	Anzeige der gewählten Funktion
(F)	Funktionen auswählen (vorwärts und rückwärts blättern)
(G)	Funktionsauswahl bestätigen



Physisches Control Panel mit Schlüssel

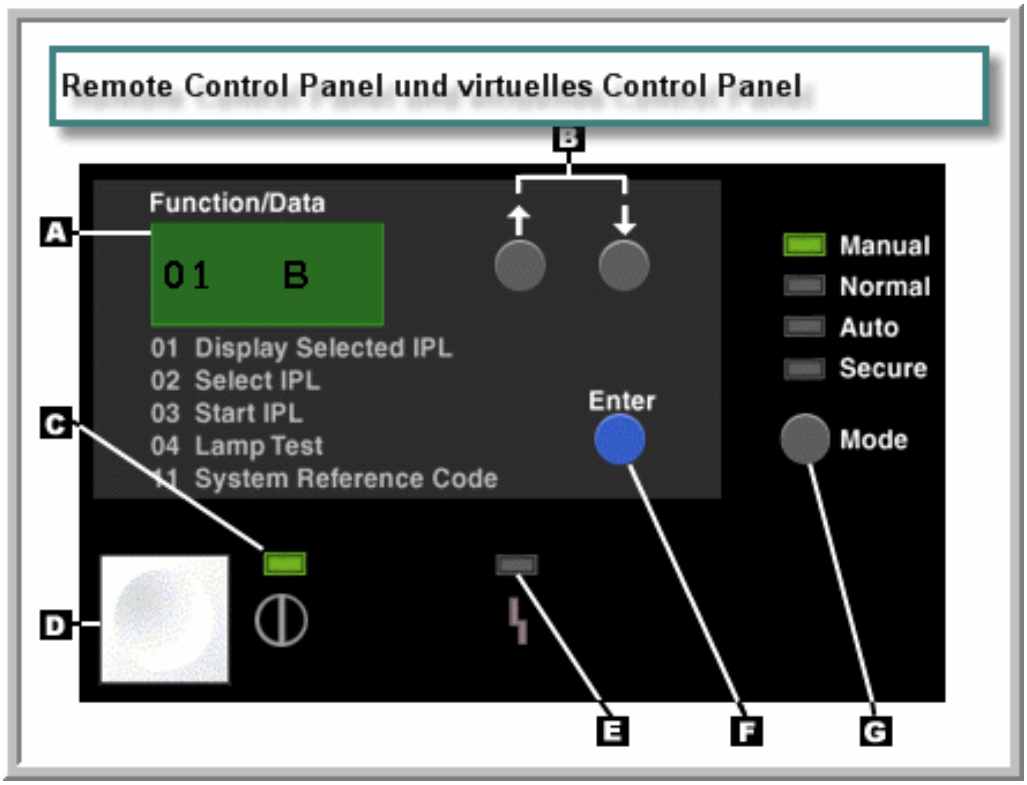
2.2.1.1
Seite 2

(A)	System an/aus Leuchte
(B)	Netzschalter an/aus
(C)	Prozessor Aktivität
(D)	Achtung! Bedienereingriff erforderlich
(E)	Anzeige der gewählten Funktion/Datenanzeige
(F)	Funktion auswählen (vorwärts und rückwärts blättern)
(G)	Funktionswahl bestätigen
(H)	Modus auswählen
(J)	Electronic Keystick Slot
(K)	Modus gesichert
(L)	Modus Auto
(M)	Modus Normal
(N)	Modus Manuell

Remote Control Panel und virtuelles Control Panel

Das Remote Control Panel und das virtuelle Control Panel können Sie während des Installationsvorgangs der „Operation Console“ konfigurieren. Operation Console ist eine Komponente von iSeries Access für Windows. Weiterführende Informationen und Problemlösungen finden Sie im Internet unter

www-03.ibm.com/servers/eserver/iseries/access/console/#header_0



Remote/virtuelles Control Panel

(A)	Anzeige der gewählten Funktion/Datenanzeige
(B)	Funktion auswählen (vorwärts und rückwärts blättern)
(C)	System an/aus Lampe
(D)	Netzschalter
(E)	Achtung! Bedienereingriff erforderlich
(F)	Funktionswahl bestätigen
(G)	Modusauswahl

2.2.1.2
Seite 2

Control Panel-Funktionen

Die Control Panel-Funktionen lassen sich in drei Gruppen unterteilen:

Die Funktionen 01 bis 49 werden als normale Control Panel-Funktionen bezeichnet. Dies sind Funktionen, die der IPL-Einstellung dienen oder z.B. die „Dedicated Service Tool“ (DST)-Umgebung starten. Es handelt sich hierbei um die gebräuchlichsten Funktionen, wobei der Bereich 21 bis 49 nur im Betriebsmodus „manuell“ gewählt werden kann. Der Bereich 50 bis 70 umfasst die erweiterten Control Panel-Funktionen und die Funktionen 57 bis 70 sind Low Level Debug Panel-Funktionen.

Funktions-Code	Remote Control Panel-Code	Virtuelle Control Panel-Funktion?	Funktionsbeschreibung
01	01	Ja	Zeigt die aktuell gewählte Art des IPL.
02	02	Ja	Ermöglicht die IPL-Art zu wählen.
03	03	Ja	IPL wird durchgeführt aufgrund der gewählten IPL-Optionen.
04	04	Ja	Lampenfunktionstest.
05	05	Nein	Zeigt den SPCN (System Power Control Network) Systemreferenzcode (SRC) im Control Panel an.
07	Nicht unterstützt	Nicht unterstützt	Erlaubt SPCN (System Power Control Network) Service-Funktionen auszuwählen.
08	08	Ja	Ausschalten des Server.
09 – 10	Nicht unterstützt	Nicht unterstützt	Reserviert.
11 – 19	11 bis 19	Ja	Der Systemreferenz-Code wird angezeigt.
20	20	Ja	Zeigt den Maschinentyp, das Modell, den Prozessor Feature-Code und die Prozessor-Klasse an.
21	21	Ja	DST- (Dedicated Service Tool) Umgebung aufrufen.
22	22	Ja	Erzwingt einen Hauptspeicher-Dump.

Eine vollständige Beschreibung aller Funktions-Codes finden Sie im iSeries Information Center unter dem Suchbegriff „control panel functions“.

2.2.1.3
Seite 2

IPL durchführen für ein nicht partitioniertes System

Die Vorgehensweise beim Starten und Stoppen des Server ist davon abhängig, ob Sie ein partitioniertes System haben oder nicht. Ich werde mich im weiteren Verlauf auf einen iSeries-Server ohne Partitionen beziehen. Unabhängig von der Konfigurierung Ihrer iSeries muss das Starten und Stoppen des Server allerdings immer sehr sorgfältig geplant werden, um Datenverluste zu verhindern und die Integrität des Systems zu schützen. Dazu später mehr (Kapitel 2.2.6). Betrachten wir zunächst die verschiedenen IPL-Arten und die unterschiedlichen Betriebsmodi:

2.2.2
Seite 2

IPL-Arten

Der lizenzierte interne Code einer iSeries wird in zwei Speicherbereichen abgelegt. Im Bereich A befindet sich eine permanente Kopie, die keine temporären Fixes (PTF) enthält. Im Speicherbereich B liegt stets die aktuellste Kopie mit den temporär eingespielten PTFs. Sie bestimmen also über die IPL-Art welche Kopie des lizenzierten internen Codes für das IPL verwendet wird. Normalerweise verwenden Sie stets die IPL-Art B, da dieser Speicherbereich die neueste Kopie des lizenzierten internen Codes enthält. Nur wenn

- die IPL-Art B fehlschlägt,
- Sie von einer Prozedur aufgefordert werden, die IPL-Art A zu verwenden
- oder Sie vermuten, dass es Probleme mit den temporären Fixes für den lizenzierten internen Code gibt

verwenden Sie den Speicherbereich A.

Die IPL-Art C ist für den technischen Außendienst reserviert. Verwenden Sie diesen Speicherbereich bitte nicht. Bei falscher Verwendung kann es zu schwerwiegenden Datenverlusten kommen.

Die IPL-Art D schließlich lädt die Systemprogramme von einer alternativen IPL-Einheit, zum Beispiel von einem Bandlaufwerk oder von einer CD-ROM. In der Regel verwendet das IPL einen Speicherbereich, der in der primären IPL-Einheit also einem Plattenlaufwerk gespeichert ist. Manchmal ist es aber notwendig, das IPL von einer anderen Quelle aus durchzuführen, z.B. wenn Programme auf einem Band gespeichert sind. Dann verwenden Sie die IPL-Art D als alternative IPL-Einheit.

2.2.2.1 Seite 2

Die Betriebsmodi

Anhand des Betriebsmodus bestimmen Sie einerseits, ob während des IPL ein Eingriff des Benutzers erforderlich ist. Andererseits können Sie über den Modus die Steuerkonsole sperren, um ein unbefugtes oder unbeabsichtigtes IPL von der Steuerkonsole aus zu verhindern.

Es gibt vier verschiedene Betriebsmodi:

Betriebsmodus Normal (unbeaufsichtigt)

Wenn Sie das System im normalen Modus einschalten, wird ein IPL ausgeführt und anschließend wird die Anzeige zum Anmelden an allen verfügbaren Datensichtgeräten angezeigt. Der Bediener kann während des IPL keine Änderungen am System vornehmen. DST (Dedicated Service Tools) und das Betriebssystem bieten Ihnen während dieses IPL keinen Eingriff über eine Bildschirmanzeige an der Systemkonsole. Die Systemkonsole oder auch Operator Konsole ist ein ganz normaler Bildschirm über den Sie auf die iSeries zugreifen können. Allerdings ist es der Bildschirm, der mit der ersten physikalischen Adresse des Ein-/Ausgabeprozessors verbunden ist. Daher nennt man die Operator Konsole auch DSP01. Der DSP01 steht meistens in der Nähe Ihrer iSeries im Maschinenraum und wird in der Regel für die Datenerfassung und Programmierung nicht genutzt. Alle Jobs, die an diesem Arbeitsplatz gestartet werden, werden im steuernden Subsystem (Controlling Subsystem) ausgeführt. Das Controlling Subsystem ist im Systemwert QCTLSBSD hinterlegt. Meistens steht hier das Subsystem QCTL.

Betriebsmodus Manuell (beaufsichtigt)

Während eines manuellen IPL erhalten Sie vEingabeaufforderungen, über die Sie Änderungen an der internen Systemumgebung vornehmen können. Unter anderem kann der Debug-Modus für die IBM-Unterstützungsservices eingegeben werden, um schwierige Probleme zu analysieren. Während eines IPL im manuellen Modus können Sie u. a. die folgenden Aktionen ausführen:

- IPL-Optionen ändern (einschließlich Systemwerte)
- Das Betriebssystem installieren
- Fixes laden (PTFs)
- Einige Arten von Hardware-Upgrades im System durchführen
- DST benutzen
- Probleme analysieren

2.2.2.2
Seite 2

Betriebsmodus Auto (automatisch)

Im Modus „Auto“ können Sie ein automatisches fernes IPL, ein automatisches IPL an einem bestimmten Datum und einer bestimmten Uhrzeit, sowie ein automatisches IPL nach einem Stromausfall durchführen.

Betriebsmodus Gesichert

Mit Hilfe des Modus „Gesichert“ können Sie verhindern, dass über die Steuerkonsole ein IPL ausgeführt wird. Dieser Modus ist keine besondere Form eines IPL, sondern verhindert nur, dass ein unbefugtes oder unbeabsichtigtes IPL von der Steuerkonsole aus durchgeführt wird.

Betriebsmodi und IPL-Arten ändern

Wie wählen Sie die IPL-Art und den Betriebsmodus? Zunächst einmal müssen wir unterscheiden zwischen einer iSeries mit und ohne elektronischem Schlüssel.

Falls Ihr Server einen elektronischen Schlüssel benötigt, stecken Sie ihn als erstes in das dafür vorgesehene Slot des Control Panels. Die optische Anzeige am Control Panel zeigt Ihnen die aktuellen Einstellungen. Mit Hilfe des Modusknopfs blättern Sie durch die vier Betriebsmodi (Manuell, Normal, Gesichert und Auto) und wählen den manuellen Modus. Jetzt können Sie die Funktion 02 mit den Auswahlknöpfen (Aufwärtspfeil/Abwärtspfeil) auswählen. Die Funktion 02 steht nur im manuellen Betriebsmodus zur Verfügung. Wenn Sie die Funktion mit der Eingabetaste bestätigt haben, können Sie anschließend mit den Auswahlknöpfen die gewünschte IPL-Art einstellen und bestätigen. Vergessen Sie aber nicht, den Betriebsmodus wieder zurückzusetzen falls Sie im Modus „Normal“ arbeiten wollen.

Für iSeries Server ohne Modus-Knopf können Sie die IPL-Art ändern, indem Sie die Funktion 02 auswählen. Mit den Knöpfen zum Erhöhen/Verringern von Werten (Aufwärts-/Abwärtspfeil) bestimmen Sie anschließend den Bereich A oder B und den IPL-Modus normal bzw. manuell. An dem Control Panel werden sowohl die IPL-Art als auch der gewünschte Betriebsmodus angezeigt. Die Angabe B_N bedeutet zum Beispiel, dass die IPL-Art B im normalen Modus gewählt ist.

Sie können auch ein langsames oder schnelles IPL auswählen. Dieses ist allerdings an dem Control Panel nur möglich, wenn der Server ausgeschaltet ist. Wählen Sie auch hierfür die Funktion 02 und drücken Sie zweimal die Eingabetaste. Anschließend wählen Sie S (Schnell), L (Langsam) oder W (Wert von IPL-Attributen) aus. Das IPL-Attribut des Hardware-Diagnoseprogramms bestimmt die Art des nachfolgenden IPL. Die empfohlene Einstellung lautet *MIN. Wenn jedoch Hardwarefehler erwartet werden, geben Sie im Parameter für das Hardware-Diagnoseprogramm *ALL an. Die IPL-Attribute können Sie mit dem Befehl CHGIPLA (change IPL-Attribute) ändern.

7. Ergänzung 06/2006

2.2.2.3
Seite 2

Unbeaufsichtigtes IPL

Hierbei handelt es sich um die gebräuchliche Art zum Starten einer iSeries. Ein unbeaufsichtigtes IPL wird auch normales IPL genannt. Egal welchen Namen Sie verwenden, gemeint ist ein IPL, das im Betriebsmodus „Normal“ ausgeführt wird. Während des IPL wird der Systemspeicher automatisch zurückgesetzt und alle Konfigurationsänderungen werden verarbeitet, ohne dass ein Benutzereingriff erforderlich ist. Die erforderliche Zeit hängt von der Größe und Komplexität des Systems ab und kann wenige Minuten oder mehrere Stunden in Anspruch nehmen. Nach Beendigung eines unbeaufsichtigten IPL wird an allen Bildschirmen die Anmeldemaske angezeigt. Falls Sie ein nicht überwachtes IPL durchführen wollen, sollten Sie zunächst überprüfen, ob:

- der Systemwert QIPLTYPE auf 0 gesetzt ist,
- der IPL Modus auf Normal gesetzt ist, d.h. im Control Panel steht ein N.

Wenn Sie das IPL dann durchführen, kann sich Ihre iSeries in zwei unterschiedlichen Zuständen befinden:

1. Das System ist heruntergefahren

Starten Sie das IPL an dem Control Panel, indem Sie den Netzschalter drücken. Natürlich steht in der Panel-Anzeige der Buchstabe N für normales IPL, was gleichbedeutend mit dem unbeaufsichtigten IPL ist.

2. Das System ist hochgefahren

Nutzen Sie in diesem Fall den folgenden Befehl:

```
PWRDWNSYS  OPTION(*IMMED) RESTART(*YES)
```

Aber Vorsicht!

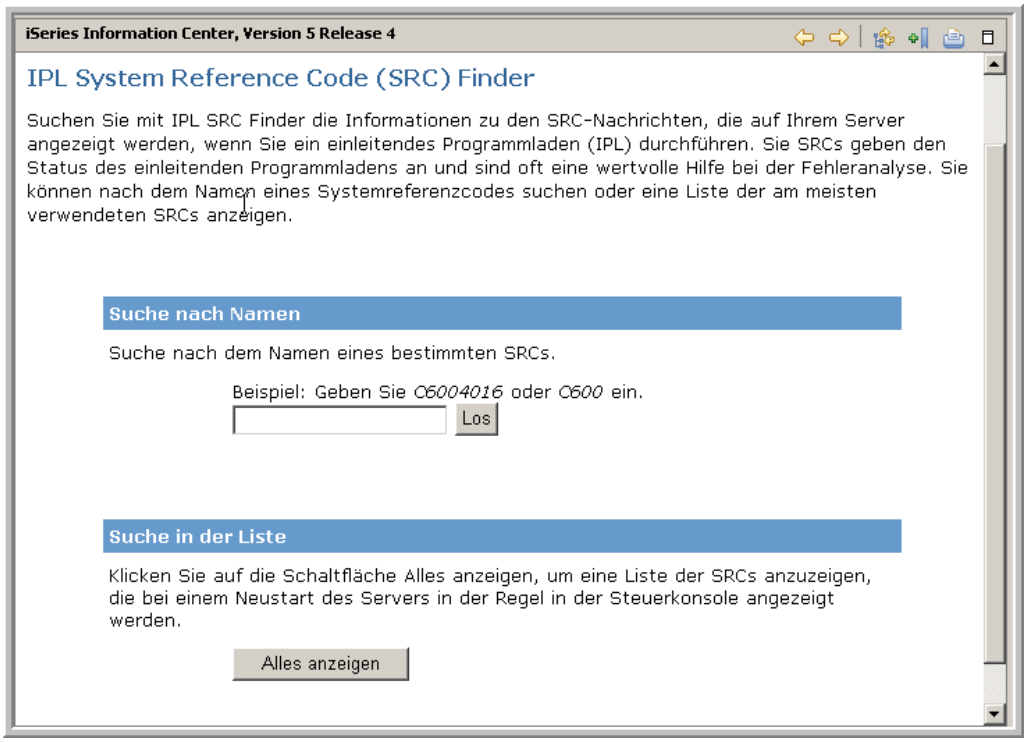
Dieser Befehl veranlasst die iSeries sofort zu stoppen und anschließend aufgrund der Systemwerteinstellungen und der Panel-Anzeige das IPL durchzuführen. Sie sollten sich also vorher vergewissern, dass keine Jobs mehr ausgeführt werden und alle Benutzer sich vom System abgemeldet haben.

Während das System startet, erhalten Sie im Control Panel eine Reihe von Referenz-Codes angezeigt. Anhand der Referenz-Codes können Sie erkennen, welche Aufgaben durchgeführt werden. Ich möchte Ihnen an dieser Stelle nur einen groben Überblick über die Bedeutung des Referenz-Codes geben:

2.2.2.4
Seite 2

Referenz-Code	Bedeutung
C1XX BXXX	Testen des Ein-/Ausgabeprozessors
C1XX 1XXX	Laden des Ein-/Ausgabeprozessors
C3XX 3XXX	Testen des Systemprozessors
C1XX 2XXX	Laden des Systemprozessors
C1XX 202E	Testen des Hauptspeichers
C1XX D009	Systemhardware bereit
C1XX 2034	IPL-Kontrolle an Systemprozessor übergeben
C6XX 4XXX	Testen der Systemkonfiguration
C6XX 4260	Wiederherstellen der Daten
C9XX 2XXX	Starten des Betriebssystems
C900 29C0	Wiederherstellen des Betriebssystems

Falls es aber einmal erforderlich sein sollte, die einzelnen Systemreferenz-Codes zu analysieren, steht Ihnen im Information Center der „IPL System Reference Code (SRC) Finder“ zur Verfügung.



Der IPL Reference Code Finder

Sobald das IPL abgeschlossen ist, erhalten Sie den Signon-Bildschirm und können sich am System anmelden.

Beaufsichtigtes IPL

In manchen Situationen ist es gegebenenfalls erforderlich, Daten oder geänderte Systemwerte während des einleitenden Programmladens (IPL) einzugeben oder mit den Service Tools (DST) zu arbeiten. In diesen Fällen führen Sie ein manuelles oder beaufsichtigtes IPL durch. Von einem beaufsichtigten IPL spricht man, wenn das IPL ausgeführt wird während der Betriebsmodus „Manuell“ gewählt ist. Überprüfen Sie vor dem IPL, ob

- der Systemwert QIPLTYPE auf 1 gesetzt ist,
- der IPL Modus auf „Manuell“ gesetzt ist, d.h. im Control Panel steht ein M.

Auch bei einem beaufsichtigten IPL kann Ihr Server entweder aus- oder eingeschaltet sein. Das Vorgehen in beiden Fällen ist identisch mit einem normalen IPL. Der Unterschied liegt nur in der Panel-Anzeige und in der veränderten Systemwerteinstellung und natürlich in der Wirkungsweise, d.h. wenn Sie sich an der Operatorkonsole (DSP01) anmelden, während das IPL ausgeführt wird, erhalten Sie ein Menü. Wird die Option 1 ausgewählt, können einige IPL-Einstellungen verändert werden. Die dann folgende Anzeige ist meiner Meinung nach selbsterklärend. Welchen Bildschirm Sie als nächstes sehen, ist davon abhängig, was Sie in den folgenden Parametern festlegen:

- Hauptangaben für System festlegen.
- System bei IPL definieren oder ändern.

Die folgenden Parameter werden nur angezeigt, wenn Sie für die Systemänderungen erforderlich sind.

- Zugriffspfade bei beaufsichtigtem IPL bearbeiten.
- Für Prüfung anstehende Integritätsbedingungen bei beaufsichtigtem IPL editieren.

Wird für alle Parameter N eingegeben, wird das IPL fortgesetzt und Sie erhalten anschließend Ihren Anmeldebildschirm. Andernfalls erhalten Sie entsprechend Ihrer Eingaben weitere Eingabebildschirme.

7. Ergänzung 06/2006

2.2.2.5
Seite 2

Hauptangaben für System festlegen

In der Anzeige „Hauptangaben für System festlegen“ können Sie die automatische Einheitenkonfiguration, die Namenskonventionen für die automatisch generierten Einheiten sowie die Sonderumgebung für die Ausführung der Programme auswählen. D.h. Kunden, die immer noch das Betriebssystem SSP auf der iSeries nutzen müssen, erhalten eine /36-Umgebung für die Ausführung ihrer Programme. Falls Sie die Default-Einstellungen verändern wollen, überschreiben Sie einfach die Angaben mit den neuen Werten. Folgende Einstellungen sind möglich:

Automatische Konfiguration aktivieren

J (Ja): Lokale Einheiten werden automatisch konfiguriert.

N (Nein): Es soll keine automatische Konfiguration erfolgen.

Einheitenkonfiguration benennen

*NORMAL: Benennt Einheiten mit einem aus drei Zeichen bestehenden Präfix und einem numerischen Suffix. Bildschirme erhalten dann das Präfix DSP, für Drucker wird das Präfix PRT, für Bandlaufwerke TAP und für CD-ROM Einheiten wird OPT genutzt.

*DEVADR: Die Einheitennamen werden von den Ressourcennamen abgeleitet. Beispielsweise steht DSP010203 für ein Datensichtgerät, PRT010203 für einen Drucker und TAP01 bzw. OPT01 für Band- bzw. CD-ROM-Einheiten.

Standardsonderumgebung

*NONE: Es soll keine Sonderumgebung verwendet werden.

2.2.2.5.1
Seite 2

System bei IPL definieren oder ändern

Es ist möglich, während eines IPL die Systemkonfiguration, Systemwerte, Netzwerkattribute, Benutzerprofile und Objekt- oder Dateiattribute zu ändern.

Mit Hilfe der Anzeige „Systemwertbefehle“ können Sie Systemwerte ändern, die das IPL oder andere Bereiche des Systems beeinflussen. Bei einigen Systemwerten wirkt sich eine Änderung erst nach dem nächsten einleitenden Programmladen (IPL) aus, bei anderen wird eine Änderung sofort wirksam. Um Systemwerte ändern zu können, müssen Sie im System als QPGMR, QSYSOPR oder QSRV angemeldet sein oder über die Berechtigung für alle Objekte (*ALLOBJ) verfügen. Einige Systemwerte können nur durch einen Sicherheitsbeauftragten, d.h. einem Benutzer mit der Sonderberechtigung *ALLOBJ und der Sonderberechtigung *SECADM geändert werden. Falls Sie jetzt vorhaben Systemwerteinstellungen zu verändern, gehen Sie wie folgt vor:

- Wählen Sie Auswahl 3 aus (Mit Systemwerten arbeiten).
- Wählen Sie Auswahl 2 in der Anzeige „Mit Systemwerten arbeiten“ aus.
- Überschreiben Sie die alten Systemwerteinstellungen und drücken Sie die Eingabetaste.
- Verlassen Sie jetzt alle Anzeigen mit F3, um das IPL fortzusetzen.

2.2.2.5.2
Seite 2

Zugriffspfade bei beaufsichtigtem IPL bearbeiten

Es kann sein, dass die iSeries während eines IPL Dateizugriffspfade wiederherstellen muss. In diesem Fall wird die Anzeige „Wiederherstellung der Zugriffspfade editieren“ angezeigt. Sie erhalten eine Liste der wiederherzustellenden Zugriffspfade. Die IPL-Schwelle bestimmt, welche Zugriffspfade für die Wiederherstellung vorgesehen sind. Alle Zugriffspfade mit einer Folgenummer, die kleiner oder gleich der angegebenen IPL-Schwelle ist, werden verarbeitet. Sie können die IPL-Schwelle ändern, indem Sie einen Wert zwischen 1 und 99 wählen. Der Standardwert ist 50. Wird die IPL-Schwelle geändert, werden alle Zugriffspfade mit dem Status IPL und AFTIPL entsprechend dem neuen Status der IPL-Schwelle behandelt. Wenn Sie die Reihenfolge nicht ändern möchten, drücken Sie einfach die Eingabetaste und es wird wieder die Anzeige „Zugriffspfad anzeigen“ aufgerufen, falls noch Zugriffspfade zur Wiederherstellung anstehen. Die Anzeige wird alle 5 Sekunden automatisch mit der aktuellen Laufzeit aktualisiert. Nachdem alle Zugriffspfade wiederhergestellt sind, deren Folgenummer kleiner oder gleich der IPL-Schwelle ist, wird das IPL fortgesetzt. Natürlich müssen Sie auf die Beendigung der Zugriffspfadwiederherstellung nicht warten. Wenn Sie die Funktionstaste F3 (Verlassen und IPL fortsetzen) drücken, werden die Zugriffspfade wiederhergestellt, während das IPL fortgesetzt wird. Drücken Sie die Funktionstaste F12 (Abbrechen), um die Anzeige „Wiederherstellung der Zugriffspfade editieren“ erneut aufzurufen.

2.2.2.5.3
Seite 2

Für Prüfung anstehende Integritätsbedingungen editieren

Falls Sie mit Integritätsbedingungen(Primary Key Constraint, referentielle Integritätsbedingungen, Check Constraint) arbeiten, wird die iSeries während des IPL einige der Dateibedingungen überprüfen. In diesem Fall wird die Anzeige „Für Prüfung anstehende Integritätsbedingungen“ bei einem manuellen IPL angezeigt. Der Bildschirm listet die zur Prüfung anstehenden Integritätsbedingungen auf. Die IPL-Schwelle bestimmt dabei, welche Integritätsbedingung für die Prüfung vorgesehen ist. Alle Integritätsbedingungen mit einer Folgenummer, die kleiner oder gleich der angegebenen IPL-Schwelle ist, werden geprüft. Sie können die IPL-Schwelle ändern, indem Sie einen Wert zwischen 1 und 99 wählen. Der Standardwert ist 50. Wird die IPL-Schwelle geändert, werden alle Integritätsbedingungen mit dem Status IPL und AFTIPL entsprechend dem neuen Status der IPL-Schwelle behandelt. Wenn Sie die Reihenfolge nicht ändern möchten, drücken Sie einfach die Eingabetaste und es wird wieder die vorherige Anzeige aufgerufen, falls noch Integritätsbedingungen zur Prüfung anstehen. Die Anzeige wird alle 5 Sekunden mit der aktuellen Laufzeit aktualisiert. Nachdem alle Integritätsbedingungen überprüft sind, deren Folgenummer kleiner oder gleich der IPL-Schwelle ist, wird das IPL fortgesetzt. Natürlich müssen Sie auf die Beendigung der Prüfung nicht warten. Wenn Sie die Funktionstaste F3 (Verlassen und IPL fortsetzen) drücken, wird die Prüfung durchgeführt, während das IPL fortgesetzt wird. Drücken Sie die Funktionstaste F12 (Abbrechen), um die Anzeige „Für Prüfung anstehende Integritätsbedingungen editieren“ erneut aufzurufen.

2.2.2.5.4

Seite 2

Das Systemkennwort

Seit der Version 4 des Betriebssystems OS/400 gibt es ein internes Systemkennwort. Das Kennwort tritt nicht sehr häufig in Erscheinung und manch einer weiß gar nicht, dass dieses Kennwort existiert. Das interne Kennwort kann aber in bestimmten Situationen während eines IPL angefordert werden. In diesen Fällen ist es günstig, das Kennwort parat zu haben. Falls das System feststellt, dass

- sich das Modell geändert hat,
- sich die Eigentumsrechte geändert haben,
- oder bestimmte Wartungsbedingungen vorliegen,

werden Sie zur Eingabe aufgefordert. Damit das IPL vollständig durchgeführt werden kann, müssen Sie das gültige Systemkennwort eingeben. Falls Sie das Kennwort nicht kennen, können Sie die Eingabe des Systemkennworts für einen begrenzten Zeitraum umgehen. Vergessen Sie aber bitte nicht, den zuständigen Vertriebsbeauftragten zu informieren, um ein neues Kennwort anzufordern. Sie erleichtern dem zuständigen Mitarbeiter die Arbeit, wenn Sie ihn bitten, RPQ S40346 zu bestellen.

2.2.3
Seite 2

Die Systemkennworteingabe umgehen

Falls Sie Ihr Systemkennwort nicht wissen, können Sie die Kennworteingabe umgehen. Drücken Sie die Eingabetaste. Es erscheint der Bildschirm „Verification of System Password Failed“. Wählen Sie anschließend die Option 2 „Bypass the System Password“. Lesen Sie die Informationen und denken Sie daran, dass der Vertriebsbeauftragte sofort benachrichtigt werden muss, damit das Systemkennwort verfügbar ist, bevor der Zeitraum abgelaufen ist, in dem das Systemkennwort umgangen werden kann. Anschließend drücken Sie F9 (Bypass), um das IPL fortzusetzen. Wenn das IPL vollständig durchgeführt ist, erhalten Sie stündlich Nachrichten, in denen Sie darüber informiert werden, wie lange der Umgehungszeitraum noch andauert. Eine kleine Anmerkung zum Schluss. Natürlich können Sie versuchen, das Kennwort zu erraten. Falls Sie allerdings fünf Mal ein falsches Kennwort eingeben, müssen Sie ein erneutes IPL durchführen.

7. Ergänzung 06/2006

2.2.3.1
Seite 2

Das Systemkennwort ändern

Wenn Sie die Kennworteingabe umgangen haben oder aus anderen Gründen das Systemkennwort verändern wollen, gehen Sie wie folgt vor:

Führen Sie ein beaufsichtigtes oder ein unbeaufsichtigtes IPL durch. In beiden Fällen erscheint der Bildschirm „Verification of System Password Failed“. Wählen Sie die Option 1 (Change the System Password) aus. Es wird die Anzeige „Change the System Password“ aufgerufen, die die folgenden Angaben zum System enthält:

- Seriennummer des Systems,
- Nummer des Systemtyps,
- Modellnummer des Systems,
- Version des Systemkennworts,
- Seriennummer der Prozessorkarte.

Geben Sie jetzt das Kennwort ein und bestätigen Sie Ihre Eingabe.

2.2.3.2
Seite 2

Das IPL-Startprogramm

Lassen Sie uns nun die Frage klären, warum Drucker und Subsysteme während des IPL gestartet werden und vor allem welche Subsysteme werden gestartet?

Verantwortlich ist das im Systemwert QSTRUPPGM hinterlegte CL-Programm. Dieser Systemwert benennt das Startprogramm, das von einem automatisch gestarteten Job aufgerufen wird, sobald das steuernde Subsystem gestartet wird. Dadurch erhält dieses Programm die Steuerung. Der Systemwert QSTRUPPGM kann nur von einem Benutzer mit der Berechtigung eines Sicherheitsbeauftragten geändert werden. Eine Änderung dieses Systemwerts wird bei der nächsten Durchführung eines IPL wirksam. Für QSTRUPPGM sind die folgenden Werte möglich:

Programmname Bibliothek

Das angegebene Programm wird ausgeführt, weil die Steuerung von dem automatisch gestarteten Job im steuernden Subsystem an dieses Programm übergeben wird.

*NONE

Der automatisch gestartete Job im Controlling Subsystem wird normal beendet, ohne dass ein Startprogramm aufgerufen wird.

Welche Aufgaben ausgeführt werden ist also vom Startprogramm abhängig. Das Standardstartprogramm ist das Programm QSTRUP in der Bibliothek QSYS. Wenn Sie wissen wollen, was in dem Programm hinterlegt ist, können Sie die Quelle mit dem folgenden Befehl abrufen:

```
RTVCLSRC PGM(QSYS/QSTRUP) SRCFILE(YOURLIB/QCLSRC))
```

Als Ergebnis erhalten Sie die CL-Quelle in der Quellendatei QCLSRC in der angegebenen Bibliothek.

```
PGM
DCL VAR(&STRWTRS) TYPE(*CHAR) LEN(1)
DCL VAR(&CTLSBSD) TYPE(*CHAR) LEN(20)
DCL VAR(&CPYR) TYPE(*CHAR) LEN(90) VALUE(' +
5722-SS1 (C) COPYRIGHT IBM CORP 1980, 2000. +
LICENSED MATERIAL - PROGRAM PROPERTY OF IBM')
QSYS/STRSBS SBSD(QSERVER)
MONMSG MSGID(CPF0000)
```

2.2.4 Seite 2

```
QSYS/STRSBS SBSD(QUSRWRK)
MONMSG MSGID(CPF0000)
QSYS/RLSJOBQ JOBQ(QGPL/QS36MRT)
MONMSG MSGID(CPF0000)
QSYS/RLSJOBQ JOBQ(QGPL/QS36EVOKE)
MONMSG MSGID(CPF0000)
QSYS/STRCLNUP
MONMSG MSGID(CPF0000)
QSYS/RTVSYSVAL SYSVAL(QCTLSBSD) RTNVAR(&CTLSBSD)
IF ((&CTLSBSD *NE 'QCTL QSYS ') *AND (&CTLSBSD *NE 'QCTL QGPL '))
GOTO DONE
QSYS/STRSBS SBSD(QINTER)
MONMSG MSGID(CPF0000)
QSYS/STRSBS SBSD(QBATCH)
MONMSG MSGID(CPF0000)
QSYS/STRSBS SBSD(QCMN)
MONMSG MSGID(CPF0000)
DONE:
QSYS/STRSBS SBSD(QSPL)
MONMSG MSGID(CPF0000)
QSYS/RTVSYSVAL SYSVAL(QSTRPRTWTR) RTNVAR(&STRWTRS)
IF (&STRWTRS = '0') GOTO NOWTRS
CALL PGM(QSYS/QWCSWTRS)
MONMSG MSGID(CPF0000)
NOWTRS:
RETURN
CHGVAR VAR(&CPYR) VALUE(&CPYR)
ENDPGM
```

Das Standardstartprogramm QSTRUP in der Bibliothek QSYS führt die folgenden Aktionen aus:

- Das Subsystem QSPL für Spool-Aufgaben wird gestartet.
- Das Subsystem QSERVER für Dateiserver-Aufgaben wird gestartet.
- Das Subsystem QUSRWRK für Benutzer-Aufgaben wird gestartet.
- Die Jobwarteschlangen QS36MRT und QS36EVOKE werden freigegeben, sofern sie angehalten wurden. Diese Jobwarteschlangen werden von der System/36-Umgebung verwendet.

- Die Bereinigung der Oberfläche für Anwendungen wird – sofern zulässig – gestartet.
- Alle Druckausgabeprogramme werden gestartet, sofern in der Anzeige für die IPL-Optionen nichts anderes angegeben wurde.
- Ist das steuernde Subsystem das Subsystem QCTL, werden die Subsysteme QINTER, QBATCH und QCMN gestartet.

Sie können den Quelltext natürlich verändern oder ergänzen und erstellen anschließend das Programm mit dem Befehl CRTCLPGM (Create Control Language Program) in Ihrer eigenen Bibliothek. Testen Sie aber bitte das Programm ausreichend, damit beim IPL keine Fehler auftreten. Erst wenn Sie sich diesbezüglich sicher sind, ändern Sie das Startprogramm, indem Sie den Programmnamen und die Bibliothek im Systemwert QSTRUPPGM hinterlegen.

2.2.4
Seite 4

Systemwerte zur Steuerung des IPL

Es gibt einige Systemwerte, die in direktem Zusammenhang mit einem IPL stehen. Lassen Sie sich die Systemwerteinstellungen anzeigen (DSPSYSVAL) und ändern (CHGSYSVAL) Sie die Einstellungen gegebenenfalls. Natürlich können Sie auch den Befehl WRKSYSVAL(work with system values) verwenden, um die Werte zu ändern oder anzuzeigen.

Geplanten Neustart zulassen (QIPLDATTIM)

Hierbei handelt es sich um das Datum und die Uhrzeit, zu dem ein automatisches IPL durchgeführt werden soll. Die Standardeinstellung ist *NONE. Zum Festlegen des Datums- und Zeitformats verwenden Sie die Systemwerte für Datum und Zeit (QDATFMT, QDATSEP, QTIMSEP).

Vorheriger Neustarttyp (QIPLSTS)

Anhand dieses Systemwerts können Sie erkennen, wie das letzte IPL durchgeführt wurde. Sie können diesen Systemwert nicht ändern.

Wert	Bedeutung	Erläuterung
0	Neustart der Steuerkonsole	Das IPL erfolgte aufgrund einer Anforderung von der Steuerkonsole oder von Dedicated Service Tools (DST) für eine sekundäre Partition.
1	Automatischer Neustart nach Stromausfall	Das IPL erfolgte automatisch nach einem Stromausfall.
2	Neustart	Nicht überwachtes IPL, nachdem der Befehl PWRDWNSYS RESTART (*YES) eingegeben wurde.
3	Uhrzeit Neustart	Nicht überwachtes geplantes IPL, das mit dem Systemwert QIPLDATIM festgelegt wurde.
4	Ferner Neustart	Nicht überwachtes IPL, wobei der Systemwert QRMTIPL auf 1 gesetzt wurde.

2.2.5
Seite 2

Typ des Neustarts (QIPLTYPE)

Der Systemwert definiert den IPL-Typ, der ausgeführt werden soll.

Wert	Bedeutung	Erläuterung
0	Unbeaufsichtigt	Es wird ein unbeaufsichtigtes IPL durchgeführt, so dass während des Neustarts niemand anwesend sein muss. Ist im Control Panel allerdings „Manuell" gewählt, wird stattdessen ein beaufsichtigtes IPL durchgeführt.
1	Beaufsichtigt	Es wird ein beaufsichtigtes IPL durchgeführt, so dass während des Neustarts ein Operator anwesend sein muss. Alle Funktionen der dedizierten Servicetools, sowie alle Neustartanzeigen sind verfügbar. Wenn es sich um ein fernes IPL, ein IPL nach Datum und Uhrzeit oder um ein IPL nach einem Stromausfall handelt, wird ein unbeaufsichtigtes IPL durchgeführt.
2	Beaufsichtigt, Konsole im Debug-Modus	Beaufsichtigtes IPL im Debug-Modus. Das System wird erneut gestartet und die Steuereinheit QCTL und Einheit QCONSOLE bleiben angehängt. Wählen Sie diese Option nur zur Fehleranalyse aus, da hierbei die Verwendung anderer Einheiten auf der Datenstations-steuereinheit verhindert wird.

Automatischer Neustart nach Stromausfall (QPWRRSTIPL)

Mit dem Systemwert für „Automatischer Neustart nach Stromausfall" können Sie angeben, ob nach einem Stromausfall ein automatischer Neustart zulässig sein soll. Falls Sie die Einstellung „0" wählen, wird nach einem Stromausfall kein automatisches IPL durchgeführt. Der Wert „1" erlaubt einen Neustart sobald die Stromverbindung wiederhergestellt wurde.

Fernes Einschalten und Neustart zulassen (QRMTIPL)

Mit dem Systemwert „Fernes Einschalten und Neustart zulassen" können Sie ein fernes System mit Hilfe eines Telefons und eines Modems, oder eines SPCN-Signals starten. Das bedeutet, dass ein beliebiger Telefonanruf einen Neustart des Systems verursachen kann. Der Wert „0" verhindert das IPL, der Wert „1" erlaubt ein Remote-IPL.

Aktion bei Stromausfall (QUPSDLYTIM)

Dieser Systemwert kontrolliert die Zeitspanne, die das System wartet, wenn die Stromversorgung unterbrochen wird. Wird die Stromversorgung vor Ablauf der Verzögerungszeit wiederhergestellt, arbeitet die iSeries weiter. Läuft die Zeit ab, so beginnt das System den Hauptspeicher zu sichern oder verwendet hierfür die unterbrechungsfreie Stromversorgung.

Wert	Bedeutung	Erläuterung
0	Gesamtes System automatisch ausschalten	Bei einem Stromausfall wird das System automatisch ausgeschaltet.
1-99999	System nach Zeitintervall ausschalten	Gibt die Verzögerungszeit in Sekunden an, bis das System abgeschaltet wird.
*BASIC	System ausschalten, Hauptturm eingeschaltet lassen	Nur der Prozessor, E/A-Prozessorkarten und IPL-Einheiten werden ausgeschaltet. Die entsprechende Wartezeit in Sekunden wird berechnet. (Sollte nur verwendet werden, wenn nicht jedes Gehäuse an die Einheit zur Notstromversorgung oder an eine unterbrechungsfreie Stromversorgung angeschlossen ist.)
*CALC	System ausschalten, System berechnet Verzögerungszeit	Die entsprechende Wartezeit in Sekunden wird berechnet. Dieser Wert darf nur dann verwendet werden, wenn Sie ein System 9402 oder 9404 mit einer Einheit zur Notstromversorgung haben.
*NOMAX	System nicht automatisch ausschalten	*NOMAX wird verwendet, wenn ein vom Benutzer zur Verfügung gestelltes Programm das Abschalten des Systems steuert oder ein Generator genug Strom zur Verfügung stellt. Das System führt keine Aktion durch.

Nachrichtenwarteschlange und Bibliothek (QUPSMSGQ)

Mit dem Systemwert können Sie eine weitere Nachrichtenwarteschlange (zusätzlich zur Nachrichtenwarteschlange des Systembedieners) bestimmen, die die IPL-Nachrichten erhält. Die Standardwerte sind QSYSOPR für die Nachrichtenwarteschlange und QSYS für die Bibliothek.

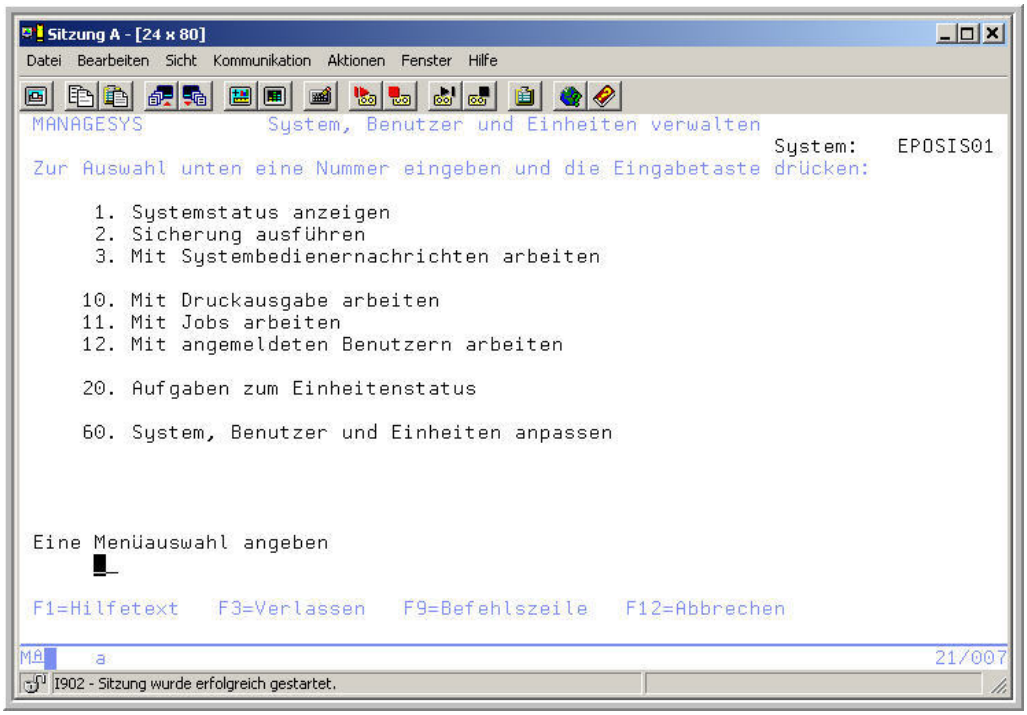
2.2.5
Seite 4

Das System ausschalten

Wir haben bereits vieles über den Start eines iSeries-Server gehört. Doch bevor die iSeries startet, muss sie gestoppt werden. Gehen Sie beim Ausschalten des Systems immer mit großer Sorgfalt vor. Wenn Sie das System einfach abschalten, könnten Daten verloren gehen oder es kommt zu unvorhersehbaren Systemreaktionen. Daher sollten Sie folgende Hinweise beachten:

Vergewissern Sie sich, dass alle Stapeljobs beendet sind und sich alle Benutzer vom System abgemeldet haben.

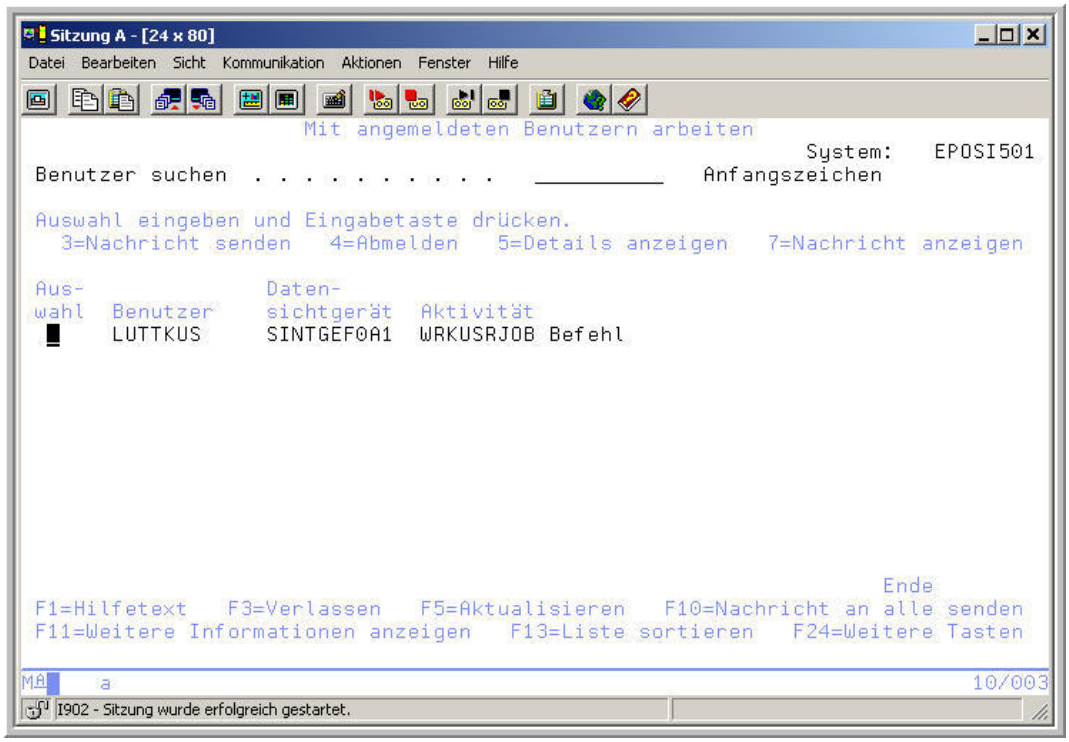
Senden Sie eine Unterbrechungsnachricht an alle angemeldeten Benutzer mit der Bitte, sich abzumelden. Sie können die Nachricht über das Menü MANAGESYS senden.



Das Menü Managesys

Wählen Sie in diesem Menü die Option 12 „Mit angemeldeten Benutzern arbeiten“. Falls die Anzeige „Mit Benutzerjobs arbeiten“ angezeigt wird, müssen Sie mit der Funktionstaste F21 zur Basisunterstützungsstufe wechseln.

2.2.6
Seite 2

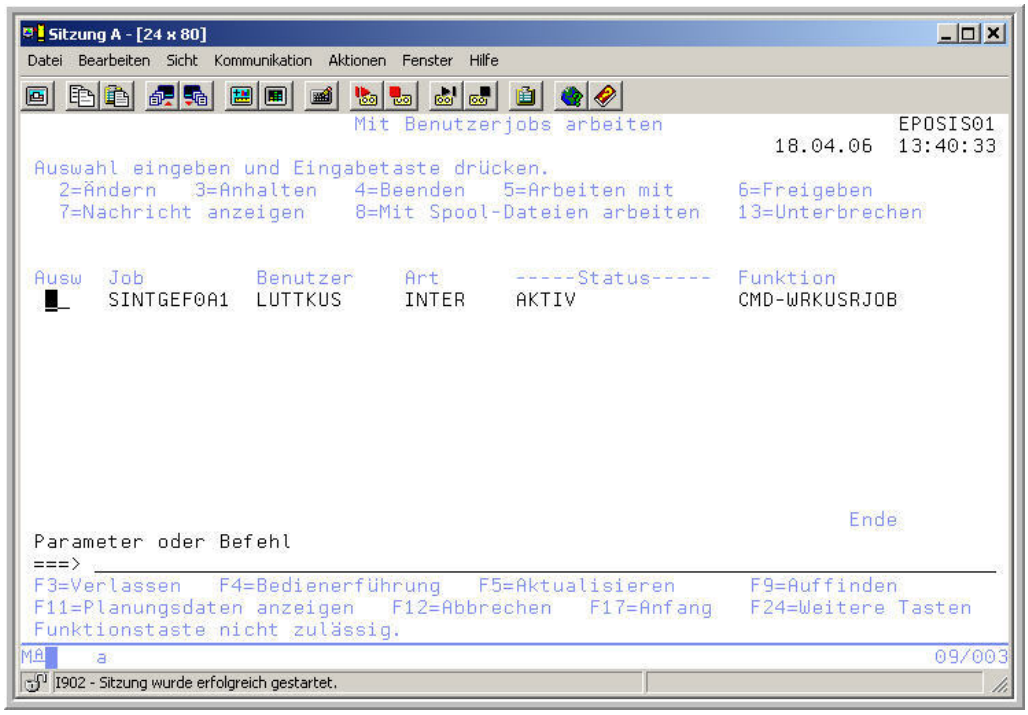


Mit angemeldeten Benutzern arbeiten

Versenden Sie jetzt die Nachricht mit der Funktionstaste F10. Geben Sie den Nachrichtentext ein, und drücken Sie erneut die Funktionstaste F10, um die Nachricht zu senden. Warten Sie, bis sich die Benutzer abgemeldet haben. Aktualisieren Sie hierfür den Bildschirm „Mit angemeldeten Benutzern arbeiten“, um zu prüfen, ob die Benutzer abgemeldet sind. Wenn dies der Fall ist, darf die Anzeige keine Benutzernamen mehr enthalten. Sie können die Benutzer mit der Auswahl 4 abmelden, falls keine Reaktion erfolgt. Wenn außer dem steuernden System noch andere interaktive Subsysteme vorhanden sind, können Sie die interaktiven Subsysteme beenden, nachdem sich die Benutzer abgemeldet haben. Dies verhindert, dass sich die Benutzer erneut anmelden, bevor das System gestoppt ist.

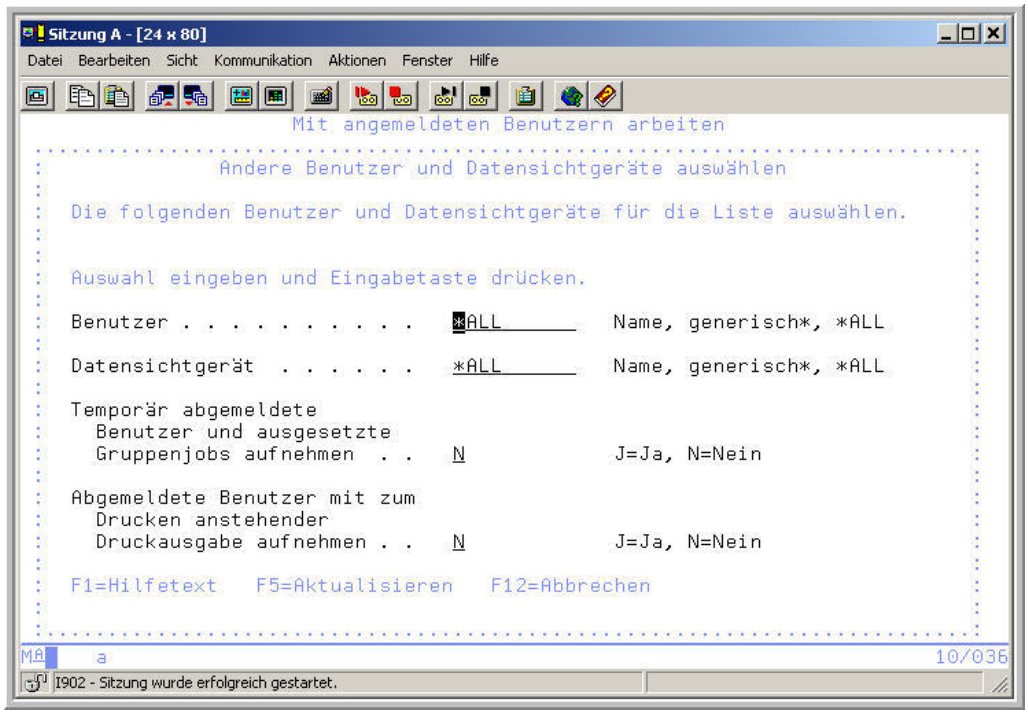
Prüfen Sie den Status der Stapeljobs, die vom Stoppen des Systems betroffen sein könnten.

Es bietet sich an, auch für diese Aufgabe das Menü MANAGESYS zu verwenden. Wählen Sie dieses Mal die Option 11 „Mit Jobs arbeiten“.



Auswahl 11 „Mit Jobs arbeiten“

Drücken Sie die Funktionstaste F14 „Andere Jobs auswählen“ und geben Sie *ALL im Feld „Benutzer“ ein. Geben Sie in allen anderen Feldern außer in den Feldern „Vorhandene Nachricht“, „Verarbeitung“ und „Angehaltene Jobs“ ein N ein.



Andere Jobs auswählen

2.2.6
Seite 4

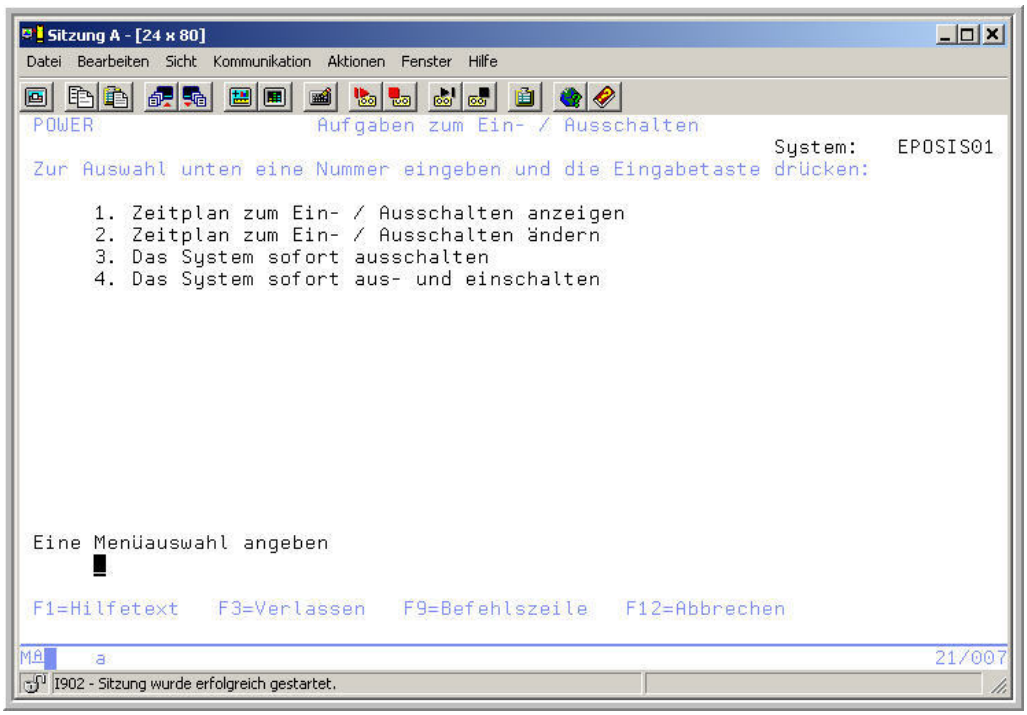
Die Anzeige „Mit Jobs arbeiten“ wird erneut aufgerufen. Sie enthält jetzt eine Liste aller Stapeljobs. Wenn sich in den Jobwarteschlangen noch Jobs befinden, drücken Sie die Funktionstaste F22 „Mit Jobwarteschlangen arbeiten“. Halten Sie alle Jobwarteschlangen an, die noch Jobs enthalten. Aber denken Sie daran, die Jobs freizugeben, nachdem das System gestartet wurde. Kehren Sie zur Anzeige „Mit Jobs arbeiten“ zurück und warten Sie, bis die Verarbeitung aller Stapeljobs vollständig beendet ist.

Auf austauschbare Datenträger prüfen

Stellen Sie sicher, dass sich in den Bandeinheiten keine Bänder und in den optischen Einheiten keine CD-ROMs mehr befinden. Entfernen Sie alle Bänder oder CD-ROMs.

Das Menü „Power“

Erst wenn alle interaktiven und alle Batch-Jobs beendet sind, sollten Sie die iSeries stoppen. Hierfür stehen verschiedene Möglichkeiten zur Verfügung, die alle im Menü „Power“ zusammengefasst sind. Geben Sie den Befehl „Go power“ in die Befehlszeile ein, um das Menü „Aufgaben zum Ein-/Ausschalten“ aufzurufen. Sie erhalten anschließend den folgenden Bildschirm:



Das Menü „Power“

Wählen Sie Auswahl 3, wenn Sie das System sofort ausschalten wollen. Der Menüpunkt 3 entspricht dem Befehl

```
PWRDWNSYS OPTION(*IMMED)
```

Wenn Sie die Auswahl 4 wählen, wird die iSeries gestoppt und anschließend wieder gestartet. Der Befehl

```
PWRDWNSYS OPTION(*IMMED) RESTART(*YES)
```

ist in der Wirkungsweise identisch.

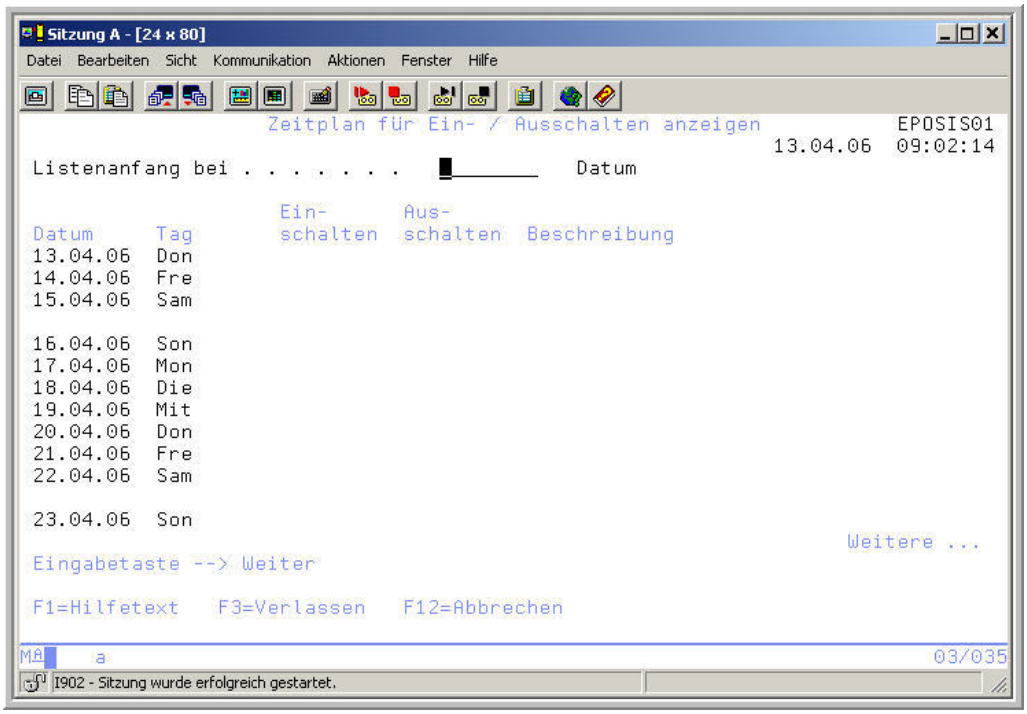
Wenn Sie das Menü „Power“ nicht benutzen können, um das System zu stoppen, können Sie das System auch mit Hilfe des Netzschalters ausschalten. Drücken Sie hierfür den Netzschalter am Control Panel. In der Anzeige „Funktion/Daten“ blinkt das Zeichen „0“. Drücken Sie den Netzschalter erneut. Jetzt blinkt die Betriebsanzeige, während das System ausgeschaltet wird. Wenn das Ausschalten des Systems abgeschlossen ist, erlischt die Betriebsanzeige. Um den Server über den Netzschalter zu stoppen, muss der Betriebsmodus der iSeries allerdings auf „Manuell“ gesetzt sein. Die Verwen-

2.2.6.1
Seite 2

dung des Netzschalters ist allerdings nicht empfehlenswert, weil dies zu Fehlern auf dem System führen kann. Außerdem wird das nächste IPL mehr Zeit in Anspruch nehmen.

Systemabschluss und Neustart über Zeitplan

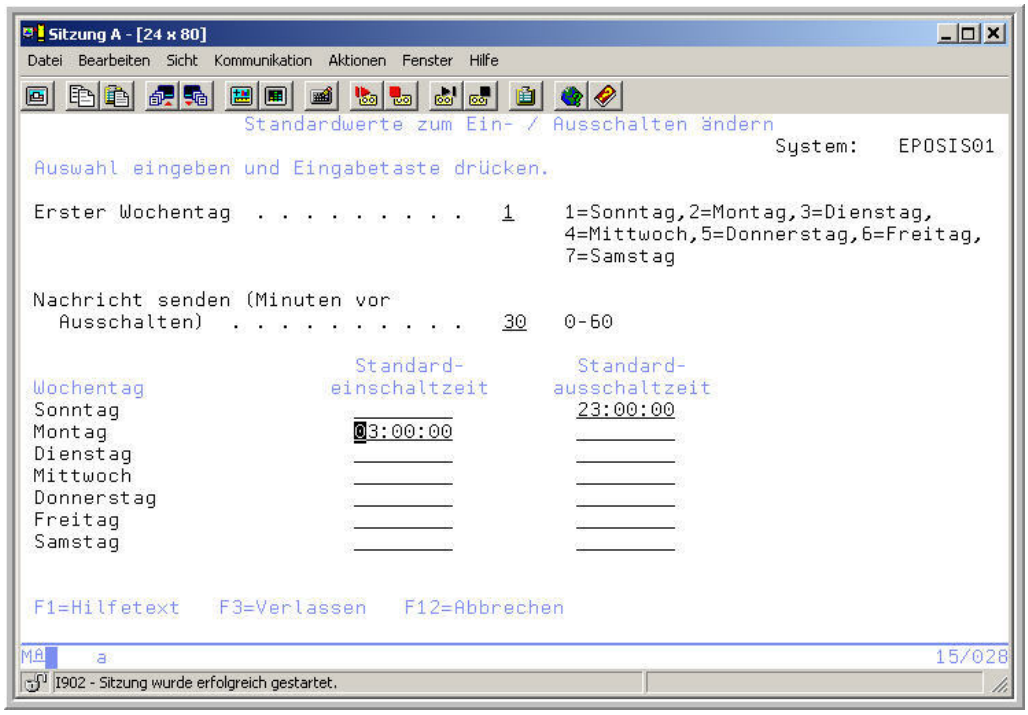
Für manche Systeme kann es sinnvoll sein, regelmäßig einen Systemabschluss und -neustart zu planen. Dafür können Sie einen Zeitplan definieren, der automatisch abgearbeitet wird. Die Benutzer werden vor dem Systemabschluss informiert und das System verzögert den Abschluss, damit die Benutzer ihre Arbeit beenden und sich abmelden können. Der Zeitplan für Ein- und Ausschalten bewirkt somit, dass das System zu bestimmten Tageszeiten automatisch ein- und ausgeschaltet wird. Sie können sich den aktuellen Zeitplan anzeigen lassen, indem Sie die Auswahl „1“ im Menü „Power“ nutzen:



Der aktuell hinterlegte Zeitplan

Wenn Sie den Zeitplan anpassen wollen, rufen Sie zunächst den Menüpunkt 2 „Zeitplan für Ein-/Ausschalten ändern“ im Menü „Power“ auf. Natürlich können Sie nicht für jeden einzelnen Tag den Ein- und Ausschaltzeitpunkt hinterlegen. Darum hinterlegen Sie Standardwerte. Rufen Sie mit der Funktionstaste F10 den folgenden Bildschirm auf.

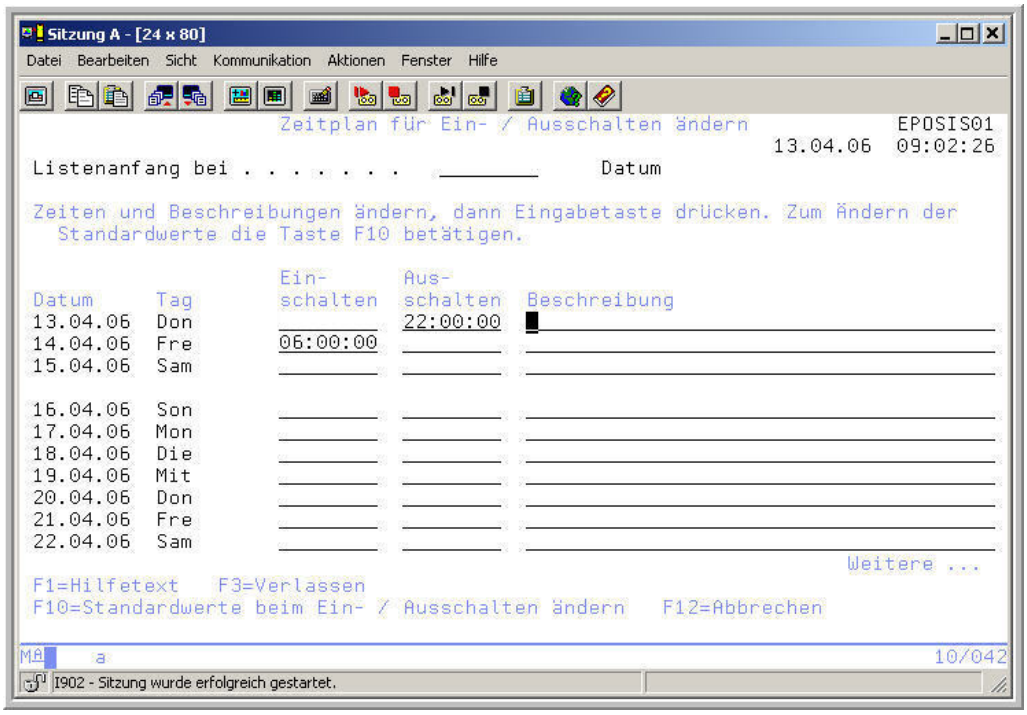
2.2.7
Seite 2



Standardwerte im Zeitplan hinterlegen

In den Standardwerten bestimmen Sie für einzelne Wochentage die Standardeinschalt- und Ausschaltzeitpunkte. In diesem Beispiel soll der Server jeweils am Sonntag um 23:00 Uhr ausgeschaltet und am Montagmorgen um 03:00 Uhr wieder gestartet werden. Außerdem können Sie den ersten Wochentag ändern, indem Sie die entsprechende Wochentagszahl im Feld „Erster Wochentag“ eingeben. Das System sendet in diesem Beispiel 30 Minuten vor dem Ausschalten an alle Benutzer eine Nachricht, in der mitgeteilt wird, dass das System abgeschaltet wird. Im Feld „Nachricht senden“ (Minuten vor dem Ausschalten) bestimmen Sie somit, wie viele Minuten vor dem Ausschalten die Benutzer benachrichtigt werden sollen. Der geplante Zeitpunkt für das Ausschalten kann um 30 Minuten bis 3 Stunden verzögert werden.

Selbstverständlich können Sie den Zeitplan zum Ein-/Ausschalten auch für einen einzigen Tag ändern unabhängig davon, ob Standardwerte hinterlegt wurden oder nicht. Als Beispiel sollen am Donnerstag, den 13.04.2006, die Zeiten für das Ein- und Ausschalten geändert werden, damit die zuvor eingespielten PTFs aktiviert werden.



Den Zeitplan anpassen

Geben Sie 22:00 Uhr in der Spalte „Ausschalten“ ein, damit das System spätestens um 22:30 Uhr am 13.04 abgeschaltet wird. Geben Sie den Anlass für die Zeitplanänderung in der Spalte „Beschreibung“ für das entsprechende Datum ein, und drücken Sie die Eingabetaste. Jetzt hinterlegen Sie für den 14.04 die Einschaltzeit, damit das System am Freitag wieder gestartet ist.

Zum Schluss noch einige Hinweise, die helfen könnten, wenn der Zeitplan bei Ihnen nicht ordnungsgemäß ausgeführt wird:

- Prüfen Sie, ob der Befehl STRCLNUP (Start Clean-up) Teil des Startprogramms ist. Für die Verwaltung des Zeitplans ist der Job QSYSSCD verantwortlich. Dieser Job wird aber nur gestartet, wenn der Befehl STRCLNUP (Bereinigung starten) im Startprogramm enthalten ist.
- Stellen Sie sicher, dass im Befehl CHGCLNUP (Change Clean-up) „Ja“ angegeben ist, damit die automatische Bereinigung auch möglich ist. Der Job QSYSSCD kann nur dann gestartet werden, wenn die automatische Bereinigung erlaubt ist.
- Wenn alle Bedingungen erfüllt sind und der Zeitplan dennoch nicht ordnungsgemäß ausgeführt wird, können Sie prüfen, ob der Job QSYSSCD ausgeführt wird. Er könnte in einer Jobwarteschlange angehalten worden sein.
- Stellen Sie außerdem sicher, dass für die Jobwarteschlange, an die der Befehl STRCLNUP (Start Clean-up) übergeben wird, der Parameter für die maximale Anzahl der Jobs auf *NOMAX oder eine Zahl größer „1“

2.2.7
Seite 4

gesetzt ist. Da der Job QSYSSCD immer ausgeführt wird, können die anderen Jobs, die Bereinigungsaufgaben oder Ausschaltfunktionen durchführen, nicht gestartet werden, wenn der Parameter für die maximale Anzahl der Jobs auf „1“ gesetzt wird. Sie können den Parameter für die maximale Anzahl der Jobs im Jobwarteschlangeneintrag mit dem Befehl CHGJOBQE (Jobwarteschlangeneintrag ändern) ändern.

- Außerdem müssen Sie darauf achten, dass der Betriebsmodus auf „Normal“ oder „Auto“ gesetzt ist.

Nachrichten und Protokolle

Nachrichten dienen im System iSeries der Kommunikation zwischen dem System und dem Systemanwender oder dem Informationsaustausch zwischen den Systemobjekten. Zusätzlich zu den Nachrichten arbeitet das Betriebssystem der iSeries mit vier verschiedenen Arten von Protokollen, die Sie in Ihrer täglichen Arbeit unterschiedlich einsetzen können. Beginnen wir zunächst mit den Nachrichten:

2.3
Seite 2

Nachrichten

Nachrichten werden immer gesendet und empfangen. Wird eine Nachricht vom System erstellt und an den Empfänger gesendet, so wird die Nachricht vom System als zugestellt definiert. Nachrichten senden Sie mit den Befehlen SNDMSG (send message) oder SNDBRKMSG (send break message). Eine detaillierte Beschreibung beider Befehle finden Sie im Kapitel 2.12.7.5. Die empfangende Stelle nimmt die Nachricht automatisch entgegen. Allerdings kann es passieren, dass die Nachrichten vom Empfänger nicht sofort gelesen werden. Meistens sammelt das Betriebssystem die Nachrichten zunächst bis der Empfänger Zeit hat sie zu lesen. Dieses Verfahren kennen wir auch aus anderen Kommunikationsbereichen. Denken Sie an Ihren Anrufbeantworter: Er sammelt die eingehenden Nachrichten und stellt sie bei Bedarf zur Verfügung. Ähnlich verhält es sich bei der iSeries: Sie sammelt die eingehenden Nachrichten in Nachrichtenwarteschlangen. Einträge in den Nachrichtenwarteschlangen werden nicht automatisch entfernt, wenn Sie die Einträge gelesen haben. Ähnlich Ihrem Anrufbeantworter bleiben die Nachrichten als „alte“ Nachrichten erhalten, noch nicht gelesene Einträge werden als „neue“ Nachrichten definiert. Unter einer Nachrichtenwarteschlange verstehen wir somit einen Bereich, in dem Nachrichten aufbewahrt werden. Weitere Informationen über Nachrichtenwarteschlangen lesen Sie im Kapitel 2.12.7.2.

Jeder Benutzer und jeder Bildschirm verfügt im System iSeries über eine eigene Nachrichtenwarteschlange, die für gewöhnlich den gleichen Namen trägt wie das zugehörige Benutzerprofil oder der zugehörige Bildschirm. Es gibt im Verhalten dieser Nachrichtenwarteschlangen allerdings einen deutlichen Unterschied. Stellen wir uns folgende Situation vor: Sie wollen das System runterfahren und versenden eine Nachricht an die Benutzernachrichtenwarteschlangen, die dazu auffordert, sich in fünf Minuten vom System abzumelden. Wenn Sie die Nachricht versenden, lesen alle aktuell angemeldeten Anwender die Nachricht. Falls Sie dieselbe Nachricht an die Datenstationsnachrichtenwarteschlangen versenden, erhalten alle angemeldeten Datenstationen die Nachricht sofort, alle Benutzer, die sich zu diesem Zeitpunkt bereits abgemeldet hatten, werden die Nachricht bei ihrer nächsten Anmeldung lesen, also ggf. am nächsten Morgen. Die Aufforderung „Bitte melden Sie sich ab! Das System wird in fünf Minuten beendet!“ könnte in diesem Fall für viel Verwirrung sorgen.

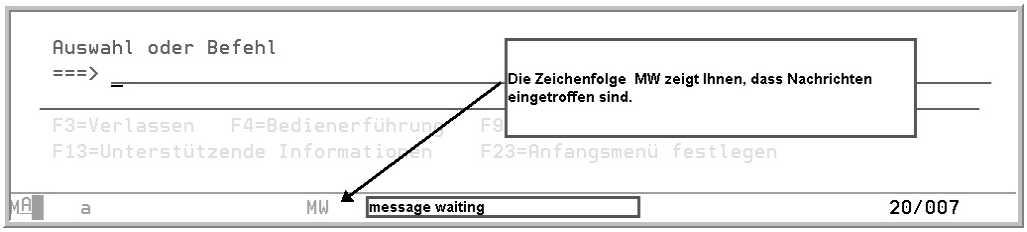
2.3.1 Seite 2

Nachrichtenarten

Es gibt drei wesentliche Nachrichtenarten, die Sie während Ihrer Arbeit erhalten: Informations-, Unterbrechungs- und Programmnachrichten. Eine detaillierte Darstellung der unterschiedlichen Nachrichtenarten enthält das Kapitel 2.12.7.3.

Unterbrechungsnachrichten können Sie nicht verpassen. Sie erhalten diese Nachrichten stets als Vollbildschirm. Man sendet Unterbrechungsnachrichten normalerweise mit dem Befehl SNDBRKMSG. Allerdings können Sie Ihre Nachrichtenwarteschlange auch so konfigurieren, dass jede Nachricht als Unterbrechungsnachricht zugestellt wird. Selbst harmlose Mitteilungen zwingen Sie dann dazu, Ihre Arbeit zu unterbrechen. Wenn Sie alle Nachrichten als Unterbrechungsnachrichten zustellen wollen, müssen Sie nur den Parameter „Nachrichtenzustellung“ Ihrer Nachrichtenwarteschlange auf *BREAK abändern.

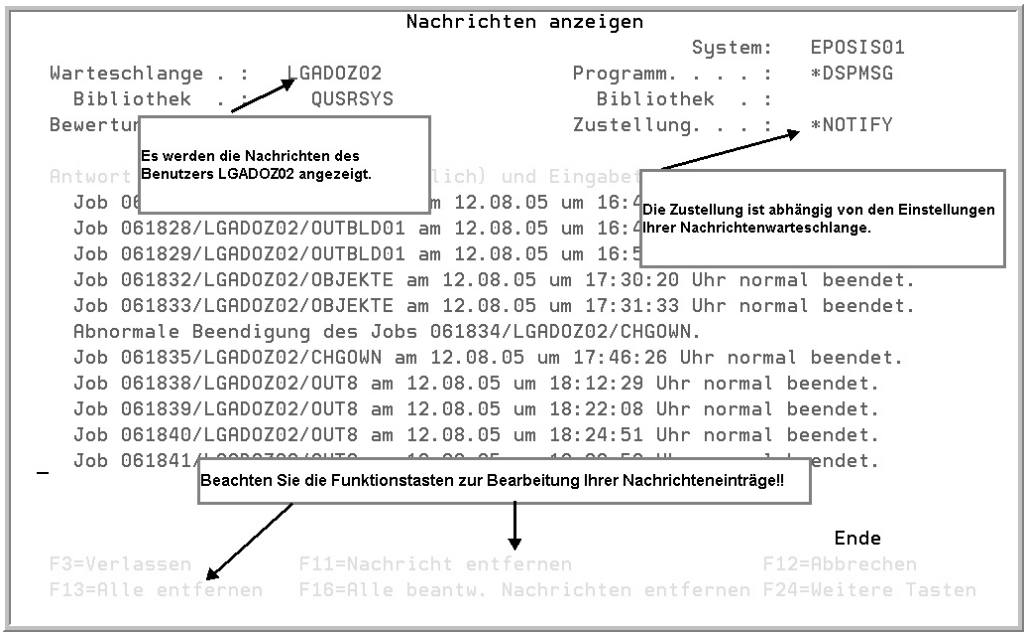
Informationsnachrichten können Ihnen leichter entgehen, da die Zustellung von den Einstellungen Ihrer Nachrichtenwarteschlange abhängt. Häufig zeigt nur ein kleines Symbol, dass eine Nachricht eingetroffen ist.



Informationsnachricht anstehend

Dabei hängt es von der Terminalemulation ab, welches Symbol verwendet wird. Sobald Sie eine Nachricht erhalten haben, können Sie die Nachricht mit dem Befehl DSPMSG anzeigen.

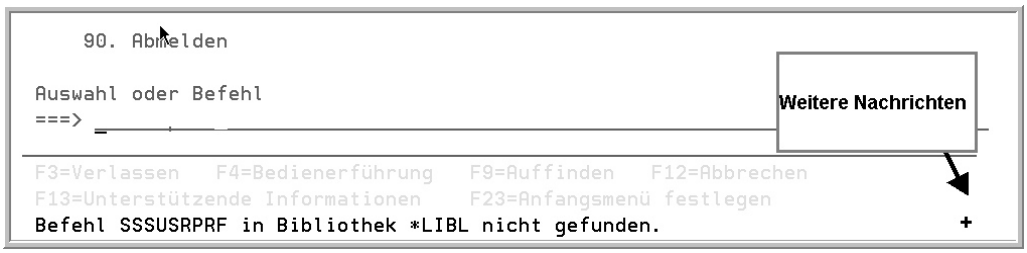
2.3.1.1
Seite 2



Ergebnis des Befehls DSPMSG

Es werden alle für Sie verfügbaren Nachrichten angezeigt. Wenn Sie weitere Informationen wünschen, können Sie den Cursor auf eine der Nachrichten positionieren und anschließend die Funktionstaste F1 drücken. Es erscheint – falls vorhanden – eine ausführliche Beschreibung.

Programm-/Systemnachrichten erscheinen in der untersten Zeile Ihres Bildschirms, nachdem Sie einen Befehl oder ein Programm aufgerufen haben. In der Statuszeile können maximal 80 Zeichen dargestellt werden. Auch hier werden detaillierte Informationen zur Verfügung gestellt, wenn Sie den Cursor auf die Nachricht positionieren und die Funktionstaste F1 nutzen. Beachten Sie das kleine Pluszeichen, dass evtl. rechts neben den Programmnachrichten angezeigt wird.



Mehrere Informationsnachrichten verfügbar

Dieses Pluszeichen informiert Sie darüber, dass weitere Nachrichten eingetroffen sind. Sie können auf diese Nachrichten zugreifen, indem Sie den Cursor auf die Nachricht positionieren und die Bild-Ab-Taste drücken.

Die Nachrichtenwarteschlange QSYSOPR

Die Nachrichtenwarteschlange QSYSOPR ist Bestandteil des Betriebssystems und enthält z.B. Nachrichten über Batch-Jobs oder Probleme im System. Allerdings lösen viele wichtige und unwichtige Ereignisse Nachrichten aus, die an QSYSOPR geschickt werden. Dadurch gehen manchmal wichtige Mitteilungen zwischen den vielen unwichtigen Nachrichten verloren. Die iSeries bietet verschiedene Lösungen für dieses Problem an. Sie können z.B. eine spezielle Nachrichtenwarteschlange für wichtige Systemmitteilungen erstellen. Mit dem folgenden Befehl erstellen Sie eine separate Nachrichtenwarteschlange:

```
CRTMSGQ  QSYS/QSYSMSG  TEXT('wichtige Operatornachrichten')
```

Der Name der Nachrichtenwarteschlange und die Bibliothek ist nicht frei wählbar. Sobald die Nachrichtenwarteschlange QSYSMSG existiert, wird die iSeries bestimmte Nachrichten an diese Nachrichtenwarteschlange senden und nutzt nicht mehr die Warteschlange QSYSOPR. Eine Dokumentation der Nachrichten, die in die Nachrichtenwarteschlange QSYSMSG gestellt werden, enthält das IBM-Handbuch OS/400 CL-Programming.

2.3.1.2
Seite 2

Die Systemantwortliste

Viele Nachrichten, die das System Ihnen sendet, sind sicherlich nützlich. Aber es gibt auch Mitteilungen die eher lästig sind, da sie wiederholt auftreten und Programme unterbrechen bis der Anwender oder der Operator auf die Nachricht reagiert. Zudem sind die Antworten auf diese Nachrichten immer nahezu identisch. Ein typisches Beispiel ist die Nachricht, die zum Formularwechsel am Drucker auffordert.

Mit der Systemantwortliste können Sie die unbeliebten Abfragenachrichten automatisch beantworten. Allerdings müssen Sie zunächst ein wenig Arbeit investieren. Stellen wir uns die folgende Situation vor: Sie haben einige Drucker, die nur ein Papierformat nutzen. Für diese Drucker wollen Sie die Nachricht zum Wechseln der Formularart unterdrücken. Allerdings besitzen Sie auch einen Listendrucker „PRT01“, dessen Nachrichten nicht automatisch beantwortet werden dürfen, da dort verschiedene Formulararten verwendet werden und Sie die Nachrichteninformationen benötigen.

Als erstes müssen Sie Ihre Jobs veranlassen, die Systemantwortliste zu verwenden. Dies geschieht über den Parameter INQMSGRPY. Der Parameter „Antwort auf Abfragenachricht“ (INQMSGRPY) wird für Batch-Jobs im SBMJOB-Befehl hinterlegt. Mit dem Befehl CHGJOB (change job) können Sie diesen Parameter für Ihre interaktiven Jobs verändern. Falls Sie dauerhaft den Parameter ändern wollen, sollten Sie den Parameter INQMSGRPY in den entsprechenden Jobbeschreibungen auf den Wert *SYSRPYLE abändern, damit die Liste verwendet wird.

Bevor Sie die Systemantwortliste modifizieren, müssen Sie sich zunächst die entsprechende Nachricht ansehen, die vom System gesendet wird und die Sie abfangen wollen. In unserem Fall ist es die Nachricht CPA4002. Falls Sie noch nicht mit Nachrichtendateien gearbeitet haben, sollten Sie zunächst das Kapitel 2.12.7.4 lesen.

2.3.1.3
Seite 2

```
                    Mit Nachrichtenbeschreibungen arbeiten
                                                          System:   EPOSIS01
Nachrichtendatei:   QCPFMSG        Bibliothek:   QSYS

Listenanfang bei  . . . .  _______   Nachrichten-ID

Auswahl eingeben und Eingabetaste drücken.
  2=Ändern   4=Löschen   5 Details anzeigen   6=Drucken

Aus-  Nachrichten
wahl      ID       Bewertung  Nachrichtentext
 =      CPA4002        99     (I G N R E C ) Ausrichtung am Drucker &3 prüfen.
 _      CPA4003        99     Bei Drucker &3 Formularvorschubmodus &4 erforder
 _      CPA4004        99     Farbbandende bei Drucker &3. (C H I PAGE 1-99999
 _      CPA4005        99     Farbbandende bei Drucker &3. (C I)
 _      CPA4006        99     Dichte des Datenträgers &5 nicht korrekt. (C R)
 _      CPA4007        99     Papierstau auf Drucker &3. (C R)
 _      CPA4008        99     Offene Abdeckung auf Drucker &3. (C H PAGE 1-999
                                                                  Weitere ...
Parameter oder Befehl
===> _____________________________________________________________________
F3=Verlassen   F5=Aktualisieren   F6=Hinzufügen   F12=Abbrechen
F24=Weitere Tasten
```

Ergebnis des Befehls WRKMSGD CPA4002

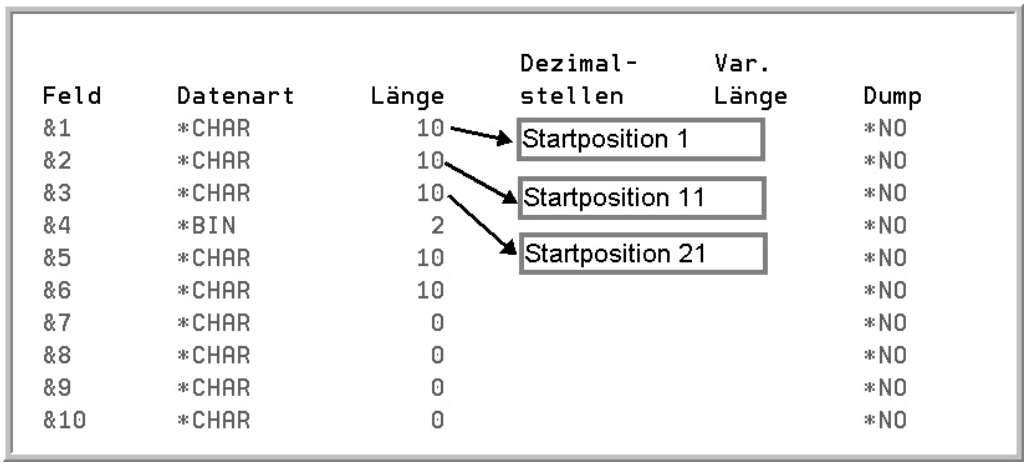
Sie sollten sich als erstes die detaillierten Informationen der Nachricht mit der Option 5 anzeigen lassen. In dem dann folgenden Bildschirm wählen Sie die Option 1. Es erscheint der Nachrichtentext der Nachricht CPA4002.

```
                    Formatierten Nachrichtentext anzeigen

Nachrichten-ID . . . . . :   CPA4002
Nachrichtendatei . . . . . :   QCPFMSG
  Bibliothek . . . . . :     QSYS

Nachricht . . . :   (I G N R E C ) Ausrichtung am Drucker &3 prüfen.
Ursache . . . . . :   Die Formulare sind möglicherweise nicht richtig
  ausgerichtet.  Die erste Zeile für die Datei ist &4.
Auswahlmöglichkeiten  . . . . . :
  I -- Um mit dem Druck ausgerichteter Formulare bei der nächsten Zeile der
    Datei fortzufahren, I eingeben.
  G -- Um mit dem Druck ausgerichteter Formulare auf einer neuen Seite
    fortzufahren, wobei die erste Zeile erneut gedruckt wird, G eingeben.
  N  -- Um die erste Zeile auf der nächsten Seite noch einmal zu drucken und
    um die Ausrichtung zu prüfen,
    1. Stop nur dann drücken, wenn Start und Stop zwei verschiedene Tasten
  sind; andernfalls die Grundstellungstaste drücken.
    2. Das Papier durch Drücken von Papiervorschub/Neue Seite zum nächsten
                                                                  Weitere ...
Eingabetaste --> Weiter

F3=Verlassen   F11=Unformatierten Nachrichtentext anzeigen   F12=Abbrechen
```

Die Variable &3 enthält den Namen des Druckers.

Nachrichtentext der Nachricht CPA4002

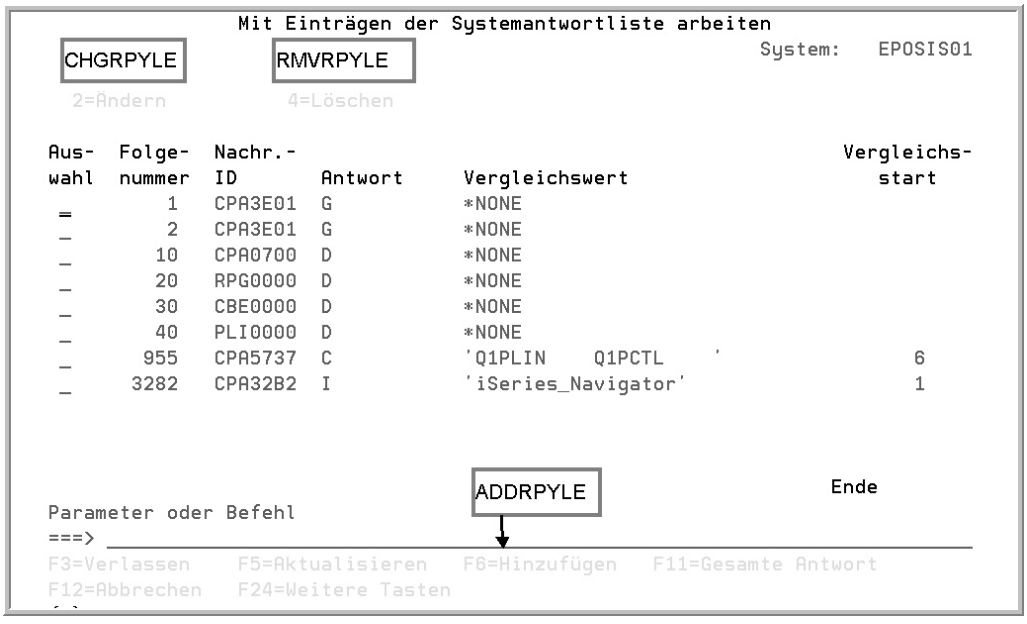
Sie wissen jetzt, welche Ersatzvariable den Druckernamen enthält. Als nächstes benötigen Sie die Startposition der Ersatzvariablen. Nutzen Sie hierfür in dem vorherigen Bildschirm die Option 2.



Liste der Ersatzvariablen

Sie können sehen, dass die Variable „&3“ an dritter Stelle in der Liste aufgeführt wird. Der Name des entsprechenden Druckers beginnt somit in der 21. Spalte. Jetzt haben Sie alle Informationen und können Ihren Eintrag der Systemantwortliste hinzufügen.

Vorher schauen wir uns erst einmal die Systemantwortliste etwas genauer an. Verwenden Sie hierfür den Befehl WRKRPYLE (work replay list entry).



Die Systemantwortliste

Der Eintrag mit der Folgenummer 10 veranlasst, dass jede Fehlernachricht, die im Bereich CPA0700 bis CPA0799 liegt, vom System mit dem Wert „D“ (= Speicherauszug) beantwortet wird. Manchmal ist die Eintragsreihenfolge schwierig zu bestimmen, da sich der Gültigkeitsbereich der Nachrichten-IDs stets aus den letzten vier Ziffern ergibt, d.h. die Folgenummer 10 betrifft alle Nachrichten mit einer ID zwischen CPA0700 und CPA0799.Die Folgenum-

mer 20 betrifft den Bereich RPG0000 bis RPG9999. Da das System die Liste sequentiell abarbeitet und nicht weiter untersucht, sobald eine Nachricht ausgegeben werden kann, ist die Eintragsreihenfolge von entscheidender Bedeutung. Für unser Beispiel bedeutet das, dass wir zuerst den Eintrag für den Drucker PRT01 erfassen und erst danach die allgemeine Antwort hinterlegen können. Den Eintrag erfassen wir mit dem Befehl ADDRPYLE (add reply list entry). Der Befehl benötigt drei Parameter:

Der Parameter Nachrichten-ID
Dieser kennzeichnet die Abfragenachricht, für die das System automatisch tätig werden soll.

Der Parameter Vergleichswert
Er enthält die Zeichenfolge, die mit den Nachrichtendaten der Abfragenachricht verglichen wird.

Der Parameter Nachrichtenantwort
Der unterstützt die folgenden Werte:

*DFT	Es wird die Standardantwort auf die Abfragenachricht gesendet.
*RQD	Eine bestimmte Antwort ist auf die Abfragenachricht erforderlich.
Zeichenfolge	Zeichenfolge, die als Antwort auf die Abfragenachricht gesendet wird.

In unserem Beispiel soll der Operator die Nachricht erhalten, wenn es sich um den Drucker „PRT01“ handelt, daher wählen wir als Nachrichtenantwort den Wert *RQD.

```
                    Antwortlisteneintr. hinzufügen (ADDRPYLE)

 Auswahl eingeben und Eingabetaste drücken.

 Folgenummer  . . . . . . . . . . > 12          1-9999
 Nachrichten-ID . . . . . . . . . > CPA4002     Zeichenwert, *ANY
 Vergleichsdaten:
   Vergleichsdaten  . . . . . .    PRT01
   Startposition Nachrichtendaten  21          1-999, *NONE
 Nachrichtenantwort . . . . . . > *RQD
```

Systemlisteneintrag hinzufügen

Unser zweiter Eintrag sieht keinen Namensvergleich vor und bezieht sich damit auf alle anderen Drucker.

```
ADDRPYLE  SEQNBR(14)  MSGID(CPA4002)  RPY(G)
```

Wir bestimmen, dass die Zeichenfolge „G“ (go) als automatische Antwort auf die Abfragenachricht CPA4002 gesendet wird.

Nach den Änderungen sollte die Systemantwortliste folgendermaßen aussehen:

```
                Mit Einträgen der Systemantwortliste arbeiten
                                                          System:   EPOSIS01
Auswahl eingeben und Eingabetaste drücken.
  2=Ändern   4=Löschen

Aus-  Folge-  Nachr.-                                          Vergleichs-
wahl  nummer  ID       Antwort     Vergleichswert                 start
 =        1   CPA3E01  G           *NONE
 _        2   CPA3E01  G           *NONE
 _       10   CPA0700  D           *NONE
 _       12   CPA4002  *RQD        'PRT01'                         21
 _       14   CPA4002  G           *NONE
 _       20   RPG0000  D           *NONE
 _       30   CBE0000  D           *NONE
 _       40   PLI0000  D           *NONE
 _      955   CPA5737  C           'Q1PLIN    Q1PCTL    '           6
 _     3282   CPA32B2  I           'iSeries_Navigator'             1

                                                                 Ende
Parameter oder Befehl
===> ____________________________________________________________________
F3=Verlassen    F5=Aktualisieren   F6=Hinzufügen   F11=Gesamte Antwort
F12=Abbrechen   F24=Weitere Tasten
```

Die modifizierte Systemantwortliste

Jetzt werden wir die unliebsame Nachricht zum Wechseln der Formularart nie wieder erhalten, da die Antwort automatisch aufgrund der Systemantwortliste geliefert wird.

2.3.1.3
Seite 6

Protokolle

Das System iSeries verwendet aber nicht nur Nachrichten für die Kommunikation, sondern es stehen zusätzlich vier Protokolle zur Verfügung, die Sie zur Informationsgewinnung nutzen können. Einige der Protokolle laufen automatisch und stehen Ihnen somit sofort zur Verfügung. Andere Protokolle müssen Sie zunächst konfigurieren. Betrachten wir die Protokollarten zunächst in der Übersicht, bevor wir uns dann detailliert mit der Nutzung der einzelnen Protokollarten befassen.

Protokollname	Verfügbarkeit	Speicherort
Systemprotokoll (QHST)	Automatisch	Physische Datei
Jobprotokoll	Automatisch; aufzuzeichnende Ereignisse können manuell gesteuert werden	Jobspool-Dateien, Bestandteil des aktiven Jobs
Jobabrechnung	Manuell	Journal
Sicherheitsprotokoll	Manuell	Journal

2.3.2
Seite 2

Das Systemprotokoll QHST

Alle Aktivitäten des Betriebssystems iSeries lösen Nachrichten aus. Einige dieser Nachrichten werden nicht nur dem entsprechenden Job zugestellt, sondern zusätzlich in physischen Dateien gespeichert. Hier finden Sie z.B. Informationen über den Systemstart, die Anmeldung Ihrer Anwender, den Gerätestatus u.v.m. Ein spezielles Betriebssystemprogramm empfängt diese Nachrichten und speichert sie in der physischen Datei QHSTxxxxxx in der Bibliothek QSYS. Diese Funktion wird allgemein als History-Protokoll bezeichnet. Die Größe der physischen Datei QHST legen Sie in dem Systemwert QHSTLOGSIZ fest, bzw. Sie bestimmen mit dem Wert die Menge an Sätzen, die in der physischen Datei gespeichert werden sollen.

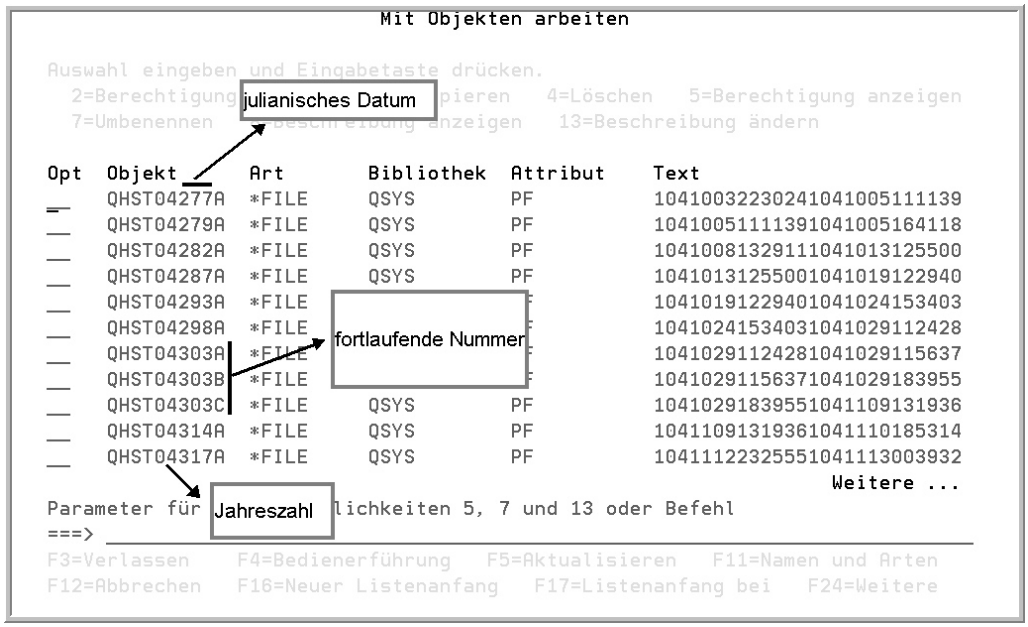
```
Systemwert . . . . . :   QHSTLOGSIZ
Beschreibung . . . . . :   Max. Systemprotokollsätze

Anzahl Datensätze  . . . . . :   5000       1-65535
```

Der Systemwert QHSTLOGSIZ

Wenn eine Version voll ist, d.h. die maximale Anzahl an Sätzen erreicht ist, wird eine neue Version erstellt. Die volle Version können Sie sichern und anschließend löschen. Je niedriger dieser Wert, desto häufiger muss eine neue Version des Systemprotokolls erstellt werden. Prüfen Sie doch einfach mal, wie viele Protokollversionen auf Ihrem System verfügbar sind, indem Sie den folgenden Befehl verwenden:

```
WRKOBJ   QSYS/QHST*    *FILE
```

2.3.2.1
Seite 2




```
                          Mit Objekten arbeiten

Auswahl eingeben und Eingabetaste drücken.
  2=Berechtigung ändern   3=Kopieren   4=Löschen   5=Berechtigung anzeigen
  7=Umbenennen   8=Beschreibung anzeigen   13=Beschreibung ändern

Opt  Objekt      Art       Bibliothek  Attribut    Text
     QHST04277A  *FILE     QSYS        PF          104100322302410410051111 39
     QHST04279A  *FILE     QSYS        PF          104100511113910410051641 18
     QHST04282A  *FILE     QSYS        PF          104100813291110410131255 00
     QHST04287A  *FILE     QSYS        PF          104101312550010410191229 40
     QHST04293A  *FILE     QSYS        PF          104101912294010410241534 03
     QHST04298A  *FILE     QSYS        PF          104102415340310410291124 28
     QHST04303A  *FILE     QSYS        PF          104102911242810410291156 37
     QHST04303B  *FILE     QSYS        PF          104102911563710410291839 55
     QHST04303C  *FILE     QSYS        PF          104102918395510411091319 36
     QHST04314A  *FILE     QSYS        PF          104110913193610411101853 14
     QHST04317A  *FILE     QSYS        PF          104111223255510411130039 32
                                                                  Weitere ...
Parameter für Optionen 5, 7 und 13 oder Befehl
===>
F3=Verlassen    F4=Bedienerführung   F5=Aktualisieren   F11=Namen und Arten
F12=Abbrechen   F16=Neuer Listenanfang   F17=Listenanfang bei   F24=Weitere
```

Ergebnis des Befehls WRKOBJ QSYS/QHST *FILE*

Stellen Sie etwas fest? Es gibt tatsächlich mehrere physische Dateien mit dem Namen QHST. An den Dateiendungen können Sie feststellen, wann die Protokollversion erstellt wurde. Alte Protokollversionen sollten Sie löschen. Insbesondere vor einem Release-Wechsel bzw. vor einer Systemsicherung können Sie dadurch wertvolle Zeit sparen.

Selbstverständlich müssen Sie die aktuellen Protokollinformationen nicht auf diese Art und Weise suchen. Wenn Sie das aktuelle History-Protokoll lesen wollen, verwenden Sie den Befehl DSPLOG (display log).

```
                        Protokoll anzeigen (DSPLOG)

Auswahl eingeben und Eingabetaste drücken.

Protokoll  . . . . . .   QHST          QHST
Zeitraum für Protokollausgabe:
  Startzeit und Startdatum:
  Startzeit  . . . . . .   *AVAIL       Zeit, *AVAIL
  Startdatum . . . . . .   *CURRENT     Datum, *CURRENT, *BEGIN
  Endzeit und Enddatum:
  Endzeit  . . . . . .     *AVAIL       Zeit, *AVAIL
  Enddatum . . . . . .     *CURRENT     Datum, *CURRENT, *END
Ausgabe  . . . . . .       *           *, *PRINT, *PRTWRAP...

                Zusätzliche Parameter

Anzuzeigende Jobs  . . . . . .   *NONE          Name, *NONE
  Benutzer . . . . . .                         Name
  Nummer . . . . . .                           000000-999999
     + für weitere Werte
Nachrichten-ID . . . . . .       *ALL          Name, *ALL
```

Der Befehl DSPLOG

Sie sehen, dass es verschiedene Möglichkeiten gibt, die anzuzeigenden Daten zu selektieren. Entweder Sie suchen die Informationen für einen bestimmten Zeitraum und/oder für einen bestimmten Job. Anschließend erhalten Sie die Ergebnisse.

```
                          Protokollinhalt anzeigen

  Job 066194/LGADOZ02/QPADEV0006 im Subsystem QINTER in QSYS am 22.10.05 um 15:
  Antwortlisteneintrag mit Folgenummer 12 zur Systemantwortliste hinzugefügt.
  Antwortlisteneintrag mit Folgenummer 14 zur Systemantwortliste hinzugefügt.
  Antwortlisteneintrag mit Folgenummer 12 aus Systemantwortliste gelöscht.
  Antwortlisteneintrag mit Folgenummer 14 aus Systemantwortliste gelöscht.
```

Ergebnis des Befehls DSPLOG

Auf den ersten Blick können die Informationen sehr verwirren, insbesondere wenn Sie die Auswahl nicht begrenzen. Wenn Sie aber den Cursor auf eine der Nachrichten positionieren und anschließend die Hilfetaste (F1) drücken, erhalten Sie für jeden Eintrag weitere Informationen.

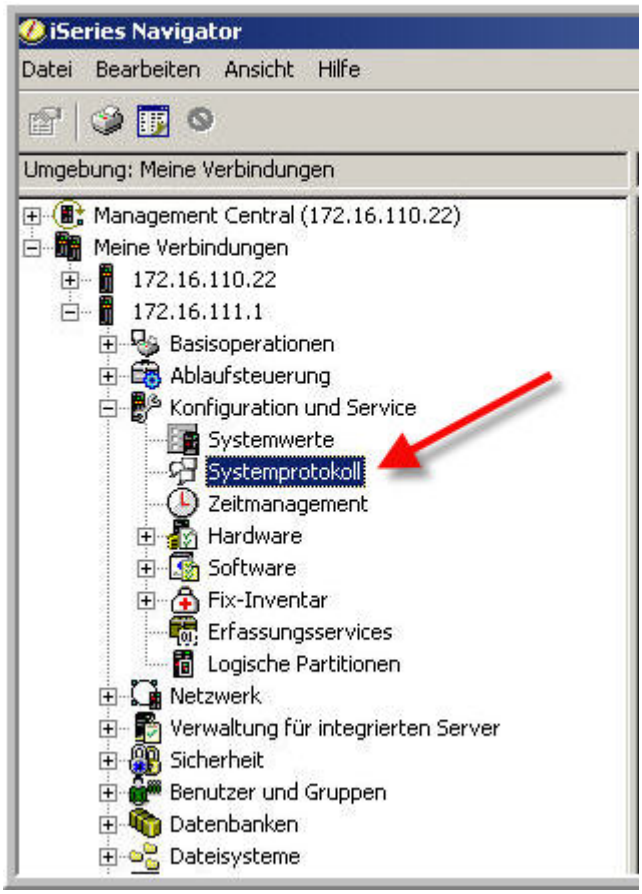
```
                       Weitere Nachrichteninformationen

 Nachrichten-ID . . . . :   CPC2403
 Sendedatum . . . . . . :   22.10.05      Sendezeit  . . . . . . :   15:42:27

 Nachricht . . . :   Antwortlisteneintrag mit Folgenummer 12 zur
   Systemantwortliste hinzugefügt.

 Ursache  . . . . :   Der Befehl ADDRPYLE (Antwortlisteneintrag hinzufügen)
   wurde erfolgreich beendet.
```

Erweiterte Nachrichteninformationen

Wie sollten Sie die Informationen der History-Datei einsetzen? Was fangen Sie mit der Vielzahl der Informationen an? Jeden Morgen alle Nachrichten lesen? Nun, das könnte eine Möglichkeit sein. Besser wäre es allerdings Sie automatisieren die Prüfung der Datei QHST mit Hilfe eines geeigneten CL-Programms, das die Datei z.B. nach sicherheitsrelevanten Nachrichten durchsucht. Ein entsprechendes Beispielprogramm finden Sie im IBM Handbuch „Work Management“ im Information Center.

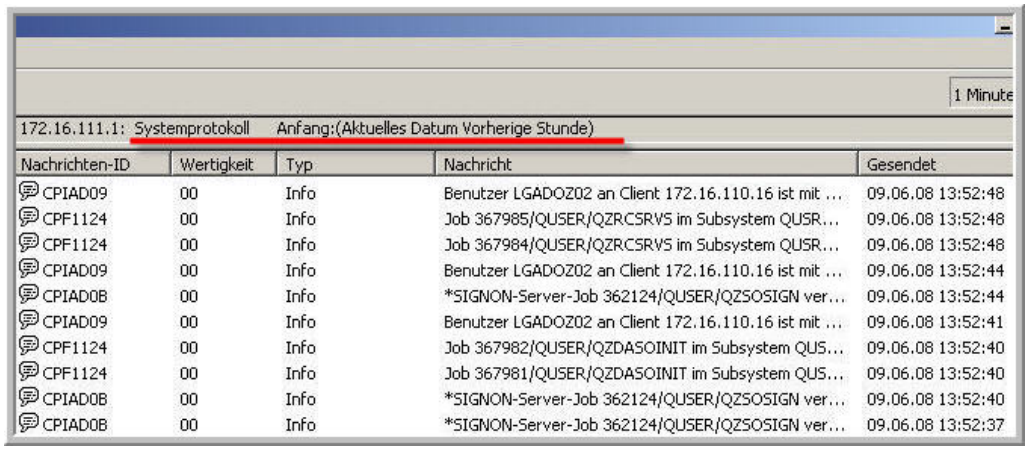
2.3.2.1
Seite 4

Sie können auch den iSeries Navigator einsetzen, um sich das Systemprotokoll anzeigen zu lassen. Dafür erweitern Sie im iSeries Navigator den Eintrag „Konfiguration und Service“.



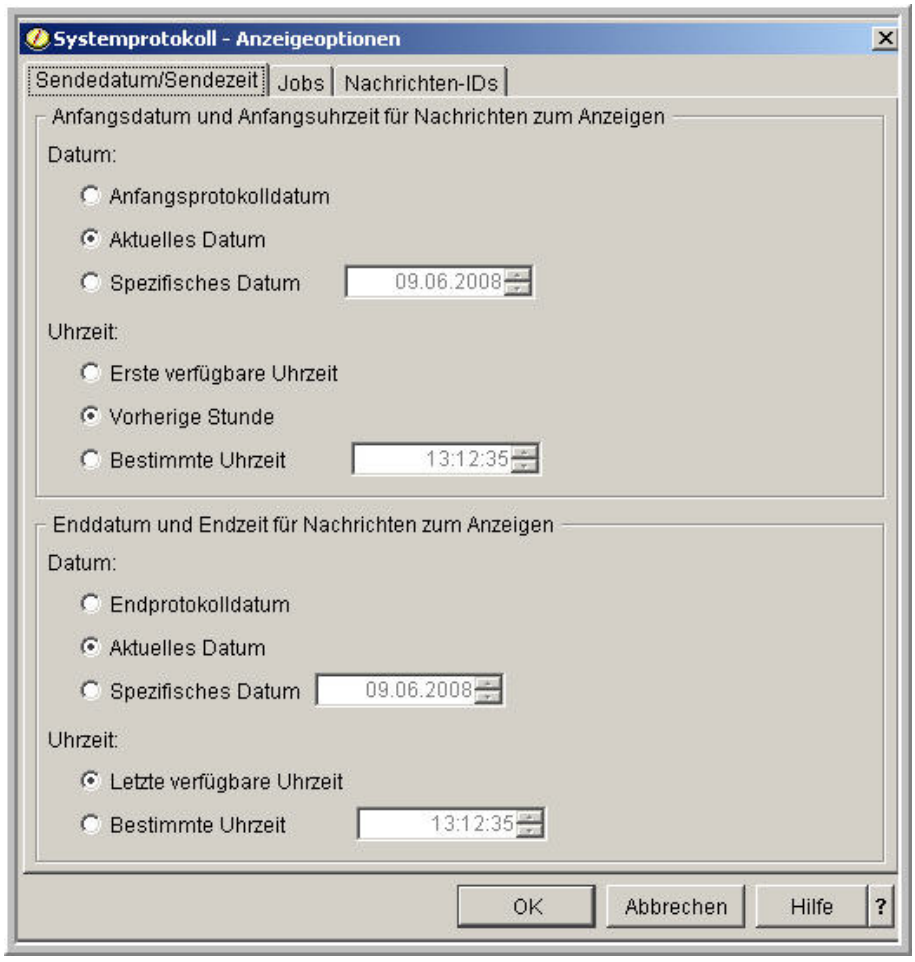
Systemprotokoll mit iSeries Navigator anzeigen

Sie sehen, ich öffne die Verbindung zu einem iSeries Server und erweitere den Menüpunkt „Konfiguration und Service“ und wähle anschließend das Systemprotokoll aus. Daraufhin werden im rechten Arbeitsbildschirm die aktuellen Systemprotokolleinträge angezeigt:



Die Protokolleinträge ansehen

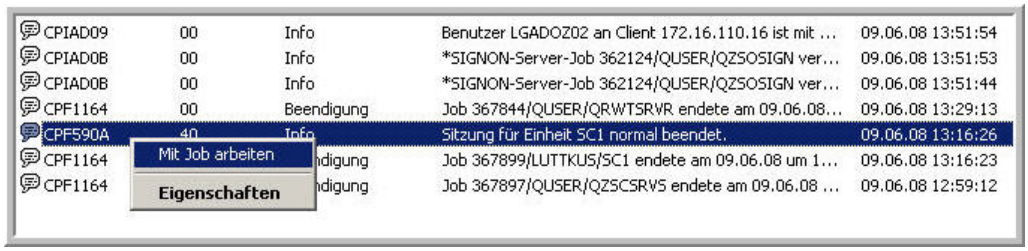
Normalerweise zeigt der iSeries Navigator Ihnen die Protokolleinträge der vergangenen Stunde. Diese Selektionskriterien können Sie aber selbstverständlich ändern. Hierfür nutzen Sie die Funktionstaste F11. Daraufhin erscheint ein neues Dialogfenster, das die Eingabe verschiedener Filterkriterien ermöglicht.



Systemprotokollmeldungen selektieren

2.3.2.1
Seite 6

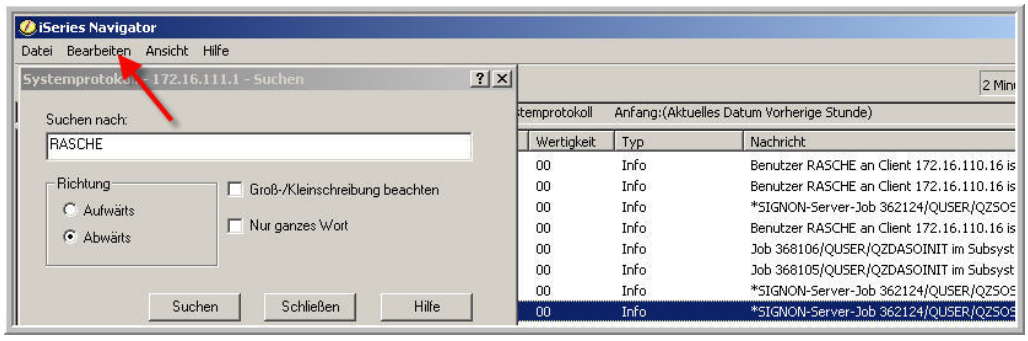
Im iSeries Navigator können Sie aber auch einzelne Systemprotokolleinträge markieren und die Kontextfunktion „Mit Job arbeiten“ nutzen, um direkt mit dem Job zu arbeiten, der die Systemmeldung verursacht hat.



Mit dem zugehörigen Job arbeiten

Falls der entsprechende Job mittlerweile beendet wurde, erhalten Sie als Ergebnis im Übrigen eine leere Bildschirmausgabe.

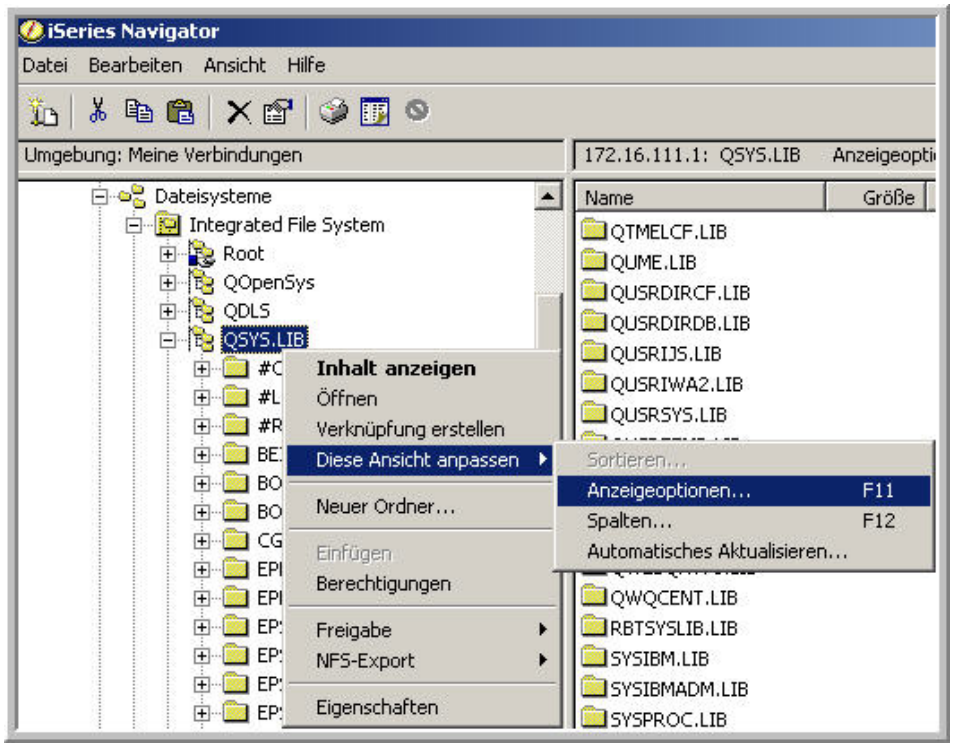
Nützlich ist in diesem Zusammenhang auch die Suchfunktion im iSeries Navigator. Rufen Sie hierfür im Dialog „Bearbeiten“ die Funktion „Suchen“ auf. Daraufhin erscheint die Suchmaske.



Informationen suchen

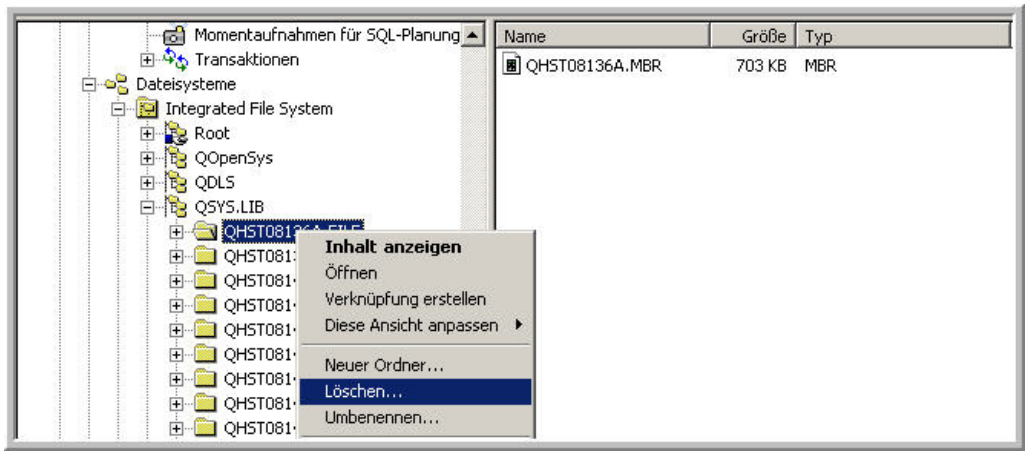
Mit dem Dialog „Suchen“ können Sie in der Liste nach einer Zeichenfolge suchen. Der Dialog „Suchen“ bleibt solange über der Liste geöffnet, bis Sie auf „Schließen“ klicken. Dies ermöglicht Ihnen ein mehrmaliges Suchen. Sie können die Taste F3 drücken, um unter Verwendung der aktuellen Suchoptionen erneut eine Suche auszuführen oder Sie nutzen einfach erneut den Button „Suchen“.

Systemprotokolle zeichnen bestimmte Betriebs- und Statusnachrichten auf, die alle Jobs des Systems betreffen. Um den Umfang der auf dem System verwendeten Ressourcen zu reduzieren, empfiehlt es sich, die Systemprotokolle von Zeit zu Zeit löschen, wenn sie nicht mehr benötigt werden. Gehen Sie hierzu wie folgt vor: Öffnen Sie in der Kategorie „Dateisysteme“ das „Integrated File System“. Klicken Sie anschließend mit der rechten Maustaste auf den Eintrag QSYS.LIB, und wählen Sie „Diese Ansicht anpassen“ aus.



Anzeige der History-Dateien in der Bibliothek QSYS

In dem nun folgenden Bildschirm, geben Sie für die Anzeigebedingungen QHST*.* ein. Sobald Sie auf „OK“ klicken, werden nur noch die Systemprotokolle in der Bibliothek QSYS angezeigt.



Die History-Dateien löschen

2.3.2.1
Seite 8

Jetzt können Sie eine oder mehrere Systemprotokolldateien löschen, indem Sie mit der rechten Maustaste auf das gewünschte Systemprotokoll klicken und den Eintrag „Löschen“ auswählen. Vor dem Löschen eines Systemprotokolls sollten Sie die Systemprotokolldatei ggf. sichern.

Das Jobprotokoll

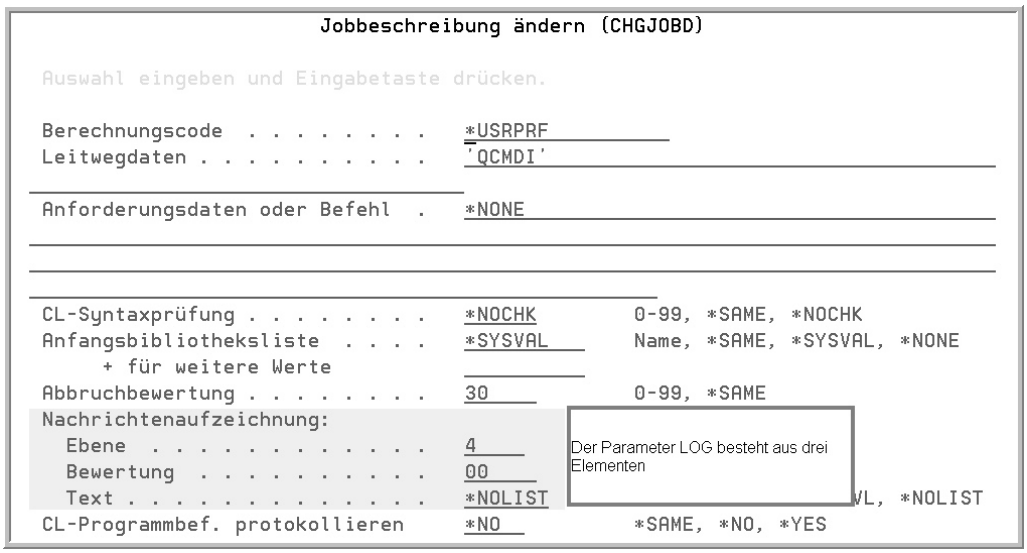
Ein weiteres Protokoll, das Ihnen wichtige Informationen liefern kann, ist das Jobprotokoll. Jeder Job, der auf der iSeries ausgeführt wird, speichert während der Ausführung für den Anwender unsichtbar Informationen in einer internen Protokolldatei. Nach dem Ende des Jobs kann der Inhalt des Jobprotokolls als Spool-Datei in die Ausgabewarteschlange QEZJOBLOG oder QPJOBLOG gestellt werden. Sie bestimmen die zu protokollierende Datenmenge und Sie entscheiden, ob die Protokollierung am Ende des Jobs gespeichert werden soll. Wenn Sie die Protokollierung deaktivieren, erhalten Sie – falls ein Job abstürzt – trotzdem ein Jobprotokoll. Dennoch ist die Deaktivierung nicht empfehlenswert, da Sie häufig nur über das Jobprotokoll Probleme mit Datenzugriffen und Programmaufrufen lokalisieren können, denn schließlich führt nicht jeder Fehler unmittelbar zum abnormalen Jobende.

2.3.2.2
Seite 2

Die Protokollstufe

2.3.2.2.1
Seite 1

Mit der Protokollstufe bestimmen Sie, welche Nachrichten und wie viele Nachrichten erfasst werden sollen. Die Protokollstufe (LOG) wird für Batchjobs im Befehl SBMJOB hinterlegt. Mit dem Befehl CHGJOB können Sie die Protokollstufe Ihres interaktiven Jobs verändern. Falls Sie dauerhaft eine andere Protokollstufe vereinbaren möchten, sollten Sie den Parameter LOG in den entsprechenden Jobbeschreibungen ändern.



Der Parameter LOG

Wie Sie in der Abbildung erkennen, besteht der Parameter aus drei Elementen: die Nachrichtenebene bzw. Protokollierungsebene, die Nachrichtenbewertung und die Nachrichtentextebene.

Das Element Ebene (LEVEL) (Wertebereich 0 bis 4)

0 Deaktiviert die Protokollierung

1 Es werden alle Nachrichten protokolliert, die an die externe Nachrichtenwarteschlange des Jobs geschickt werden und deren Nachrichtenprotokollierungsebene größer oder gleich der angegebenen Nachrichtenbewertung ist.

2 Alle Informationen der Aufzeichnungsebene 1. Zusätzlich werden Anforderungsnachrichten einer höheren Ebene mit einer Nachrichtenbewertung größer oder gleich der Nachrichtenprotokollierungsbewertung aufgezeichnet. Sowohl die Anforderungsnachricht als auch alle dazugehörigen Nachrichten werden beachtet.

3 Bestimmt, dass alle von einem CL-Programm ausgegebenen Befehle protokolliert werden, wenn die Attribute des CL-Programms die Protokollierung erlauben.

2.3.2.2.1
Seite 2

4 Alle Anforderungsnachrichten, alle Nachrichten mit einer Nachrichtenbewertung größer oder gleich der Nachrichtenprotokollierungsebene und alle CL-Befehle eines CL-Programms werden aufgezeichnet, wenn die Attribute des Programms dies erlauben.

Das Element Bewertung (SEVERITY)

Alle Nachrichten im System iSeries verfügen über einen Bewertungsschlüssel oder anders gesagt: Die Wichtigkeit einer Nachricht wird durch die Werte 00 – 99 festgelegt. Die letzte Nachricht vor einem abnormalen Programmende hat z.B. in der Regel die Wertigkeit 99. Informationsnachrichten werden dagegen mit dem Wert 00 gesendet. Wichtige Jobnachrichten haben meistens einen Wert 30 oder höher.

Das Element TEXT

Eine iSeries-Nachricht besteht immer aus zwei Ebenen. Der Nachrichtentext der 1. Ebene ist maximal 132 lang. Der Nachrichtentext der 2. Ebene kann bis zu 3000 Zeichen enthalten. Vergleichen Sie hierzu Kapitel 2.12.7.4. Welcher Text protokolliert werden soll, bestimmen Sie mit dem Element TEXT.

*MSG Der Nachrichtentext der 1. Ebene wird protokolliert.

*SECLVL Der Nachrichtentext der 2. Ebene wird protokolliert.

*NOLIST Nachrichten werden nur beim Programmabsturz protokolliert.

In stabilen Arbeitsumgebungen mit entsprechend stabilen Programmen wird für den Parameter LOG häufig folgender Eintrag verwendet:

LEVEL: 4

SEVERITY: 00

TEXT: *NOLIST

Die Standardjobbeschreibungen QCTL, QBATCH, QINTER, QPGMR enthalten ebenfalls die Standardeinstellung LOG (4 0 *NOLIST), d.h. alle Befehle und Nachrichten mit einer Wertigkeit größer 0 werden protokolliert, falls das Programm/Job abstürzt. Bedenken Sie bei Ihren interaktiven Jobs, dass diese Jobs normalerweise mit dem Befehl SIGNOFF beendet werden. Der Standardwert dieses Befehls ist *NOLIST. Dadurch wird die Protokollierung unterbunden, egal welchen Wert Sie in der Jobbeschreibung verwenden. Wenn Sie die Protokollierung für interaktive Jobs aktivieren wollen, müssen Sie nicht nur die Jobbeschreibung bzw. die Jobparameter ändern, sondern auch den Standardwert des Befehls SIGNOFF auf LOG(*LIST) setzen. Verwenden Sie hierfür den folgenden Befehl:

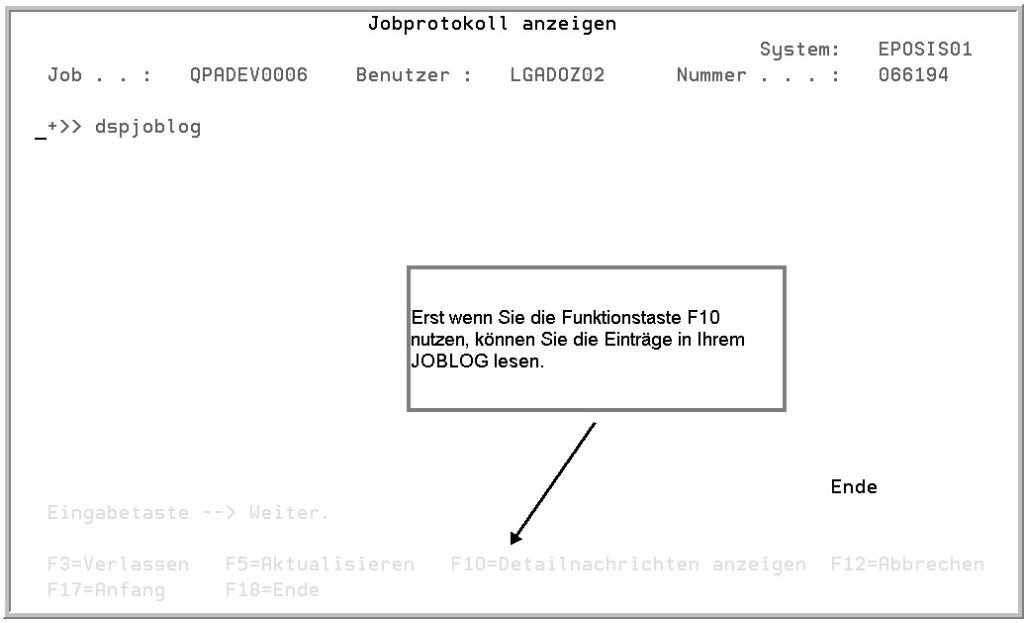
```
CHGCMDDFT  SIGNOFF  NEWDFT('LOG(*LIST)')
```

Jobprotokolle anzeigen

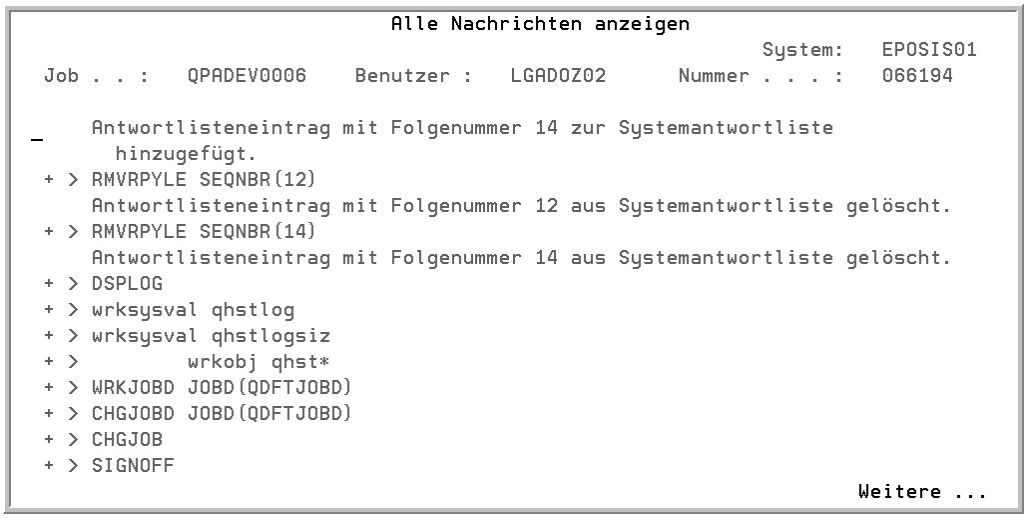
2.3.2.2.2
Seite 1

Wenn Sie sich das Jobprotokoll eines interaktiven Jobs ansehen wollen, gehen Sie wie folgt vor:

- Geben Sie den Befehl DSPJOBLOG (display job log) in der Befehlszeile ein. Anschließend drücken Sie die Funktionstaste F10 und die Bild-Ab-Taste.



Der Befehl DSPJOBLOG



Detailnachrichten anzeigen

- Geben Sie in der Befehlszeile den Befehl WRKJOB ein. Wählen Sie die Option 10 für den entsprechenden Job und drücken Sie anschließend die Funktionstaste F10.

2.3.2.2.2
Seite 2

- Falls der Job zurzeit noch aktiv ist, können Sie alternativ auch den Befehl WRKUSRJOB oder den Befehl WRKACTJOB verwenden. Falls Sie den Befehl WRKACTJOB verwenden, nutzen Sie die Option 5 (Arbeiten mit) und anschließend die Option 10 (Jobprotokoll anzeigen).

Falls der Job nicht mehr aktiv ist, gehen Sie wie folgt vor:

- Geben Sie den Befehl WRKOUTQ QPJOBLOG oder QEZJOBLOG. Die Liste der Spool-Dateien, die Ihnen angezeigt wird, kann sehr lang sein. Drücken Sie die Funktionstaste F18, um schnell an das Ende der Anzeige zu gelangen, wo sich die aktuellen Jobprotokolle befinden.

Jobaufzeichnung

Die Jobaufzeichnung erfasst jobspezifische Informationen, die Auskunft darüber geben, wer welche Ressourcen im System verbraucht. Diese Erfassung müssen Sie allerdings manuell starten. Die Daten werden dann in Journalen gespeichert und verwaltet. Jobaufzeichnungen können interessant sein, weil Sie Ihnen Informationen über die Jobs Ihres Systems liefern ohne dass Sie die Leistungsüberwachung einsetzen müssen. Außerdem ist die Erfassung von Statistiken möglich wie z.B. die Anzahl der Seiten eines Druckjobs, der Zeitpunkt, zu dem ein Job im System ankommt und wann er startet.

2.3.2.3
Seite 2

Jobaufzeichnung konfigurieren

Die Konfiguration der Jobaufzeichnung entspricht größten Teils dem Prozess der Journalerstellung. Sie benötigen also zunächst einmal ein Journal. Die Verwendung einer seperaten Bibliothek mit einem eindeutigen Journalempfänger wirkt sich positiv auf die Systemleistung aus. Erstellen Sie also eine Bibliothek mit dem Befehl CRTLIB (create library).

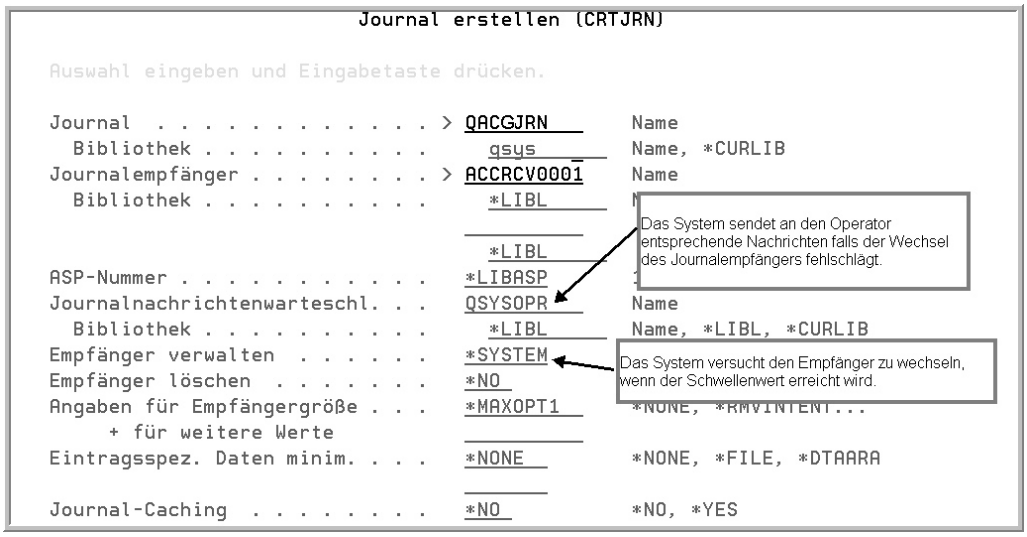
Als nächstes brauchen wir einen Journalempfänger. Mit dem Befehl CRTJRNRCV (create journal receiver) wird ein Journalempfänger erstellt. Der Name des Journalempfängers sollte 10 Zeichen (6 Buchstaben und 4 Ziffern) lang sein. Ich verwende den Namen ACCJRN0001. So ist der Name leicht der Funktion zuzuordnen und das System kann selbst einen neuen Empfänger generieren, wenn der Schwellenwert erreicht wird.

```
                    Journalempfänger erstellen (CRTJRNRCV)

 Auswahl eingeben und Eingabetaste drücken.

 Journalempfänger . . . . . > ACCRCV0001     Name
   Bibliothek . . . . . .       *CURLIB      Name, *CURLIB
 ASP-Nummer . . . . . .       *LIBASP      1-32, *LIBASP
 Journalempfängerschwelle . . .  1500000     1-1000000000, *NONE
 Text 'Beschreibung'  . . . . .  *BLANK
 _____

                 Zusätzliche Parameter

 Bevorzugte Speichereinheit . . .  *ANY       1-255, *ANY
 Berechtigung . . . . . .      *LIBCRTAUT   Name, *LIBCRTAUT, *CHANGE...

                                                                  Ende
```

Der Befehl CRTJRNRCV

Anschließend erstellen Sie das Journal. Für das Journal der Jobaufzeichnung dürfen Sie weder die Bibliothek noch den Namen des Journals frei wählen. Verwenden Sie einfach den folgenden Befehl für die Erstellung, falls das Journal noch nicht existiert.

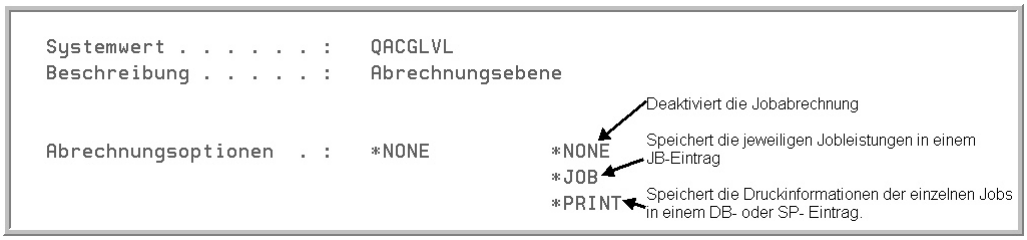
2.3.2.3.1
Seite 2



Der Befehl CRTJRN

Sobald ein Empfänger mit dem Befehl CRTJRN oder CHGJRN an ein Journal angehängt wird, können Einträge in den Empfänger gestellt werden.

Anschließend müssen Sie den Systemwert QACGLVL setzen. Er steuert die Protokollierung der Jobabrechnung. Beachten Sie bitte, dass das Journal zu diesem Zeitpunkt bereits existieren muss, da Sie sonst eine entsprechende Fehlermeldung erhalten würden.

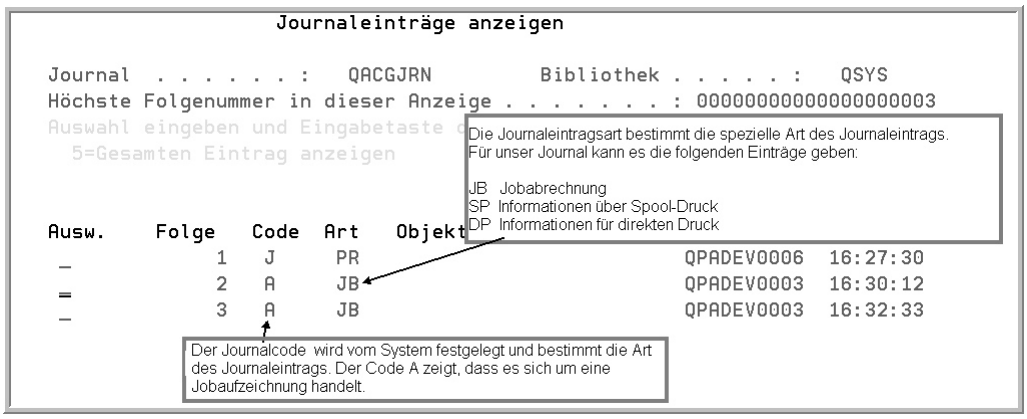


Der Systemwert QACGLVL

Sobald Sie den Systemwert verändert haben, wird die Aufzeichnung gestartet.

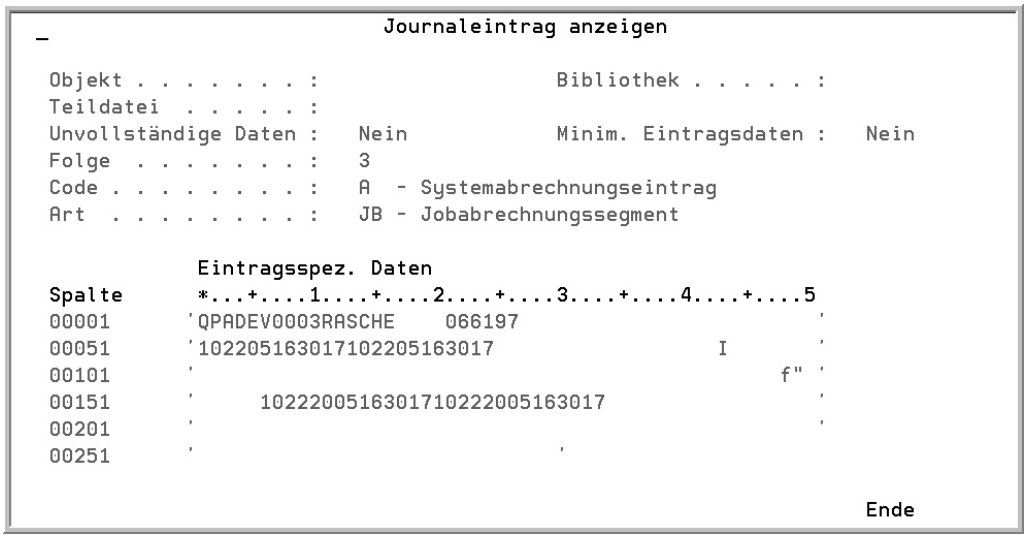
Jobaufzeichnungsdaten anzeigen

Etwas schwieriger wird es jetzt bei der Anzeige der Daten. Natürlich können Sie sich mit dem Befehl DSPJRN (display journal) die Daten im Journal anzeigen lassen. Der Befehl ist allerdings nicht sonderlich komfortabel, da Journale in der Regel für die Datenbankwiederherstellungen genutzt werden und da muss halt nur die iSeries die Informationen verstehen. Schauen wir uns dennoch den Befehl DSPJRN an, denn schließlich wollen wir ja unsere Jobaufzeichnungsdaten auch verwenden.



Journaleinträge anzeigen (DSPJRN)

Die Anzeige zeigt Ihnen eine Liste der Journaleinträge mit den entsprechenden Basisinformationen. Um alle Informationen anzuzeigen, geben Sie eine 5 vor dem Eintrag an. Anschließend können Sie die detaillierten Informationen lesen.



Detaillierte Aufzeichnungsinformationen

2.3.2.3.2
Seite 2

Spätestens jetzt werden Sie verstehen, warum diese Anzeige wenig geeignet ist, Sie mit den nötigen Informationen zu versorgen. Um die Bedeutung der einzelnen Einträge zu kennen, benötigen Sie zwei Referenzdateien:

```
QSYS/ QAJBACG (Format QWTJAJBE)  und QSYS/QAPTACG (Format QSPJAPTE)
```

Die Jobabrechnungsprotokolle werden aufgrund dieser Dateien erstellt. Sie könnten natürlich die Felder aus diesen Referenzdateien ziehen, wenn Sie Ihre Daten lesen wollen. Das ist allerdings sehr umständlich. Daher sollten Sie besser eine Kopie (CRTDUPOBJ) von den entsprechenden physischen Dateien machen. Anschließend geben Sie den folgenden Befehl ein, um die benötigten Datensätze zu selektieren:

```
DSPJRN  JRN(QACGJRN)  JRNCDE(A)  ENTTYP(JB)  OUTPUT(*OUTFILE)  +
OUTFILE(bibliothek/ihrDuplikat)
```

Mit diesem Befehl extrahieren Sie die Jobdatensätze in eine physikalische Datei, die Sie anschließend mit Hilfe logischer Dateien, SQL-Abfragen oder HLL-Programmen auswerten können.

```
                          Daten anzeigen
                                                Datenbreite . . . . . :     821
Anfang auf Zeile . . . . . _____      Anfang in Spalte . . . . . .  _____
....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....
EINTRAGS-          FOLGE-    CODE  ART  EINTRAGS-  EINTRAGS-  RESERVIERT
LÄNGE              NUMMER                DATUM      ZEIT

     125               1    J     PR    221005     162.730   QPADEV0006LGADOZ02
     404               2    A     JB    221005     163.012   QPADEV0003LGADOZ02
     404               3    A     JB    221005     163.233   QPADEV0003RASCHE
     404               4    A     JB    221005     163.411   QPADEV0003LGADOZ02
```

SQL-Auswertung der Journaleinträge (1)

```
                    Daten anzeigen
                                          Datenbreite . . . . . :     821
f Zeile . . . . . _____      Anfang in Spalte . . . . . .  _____
..25....+...26....+...27....+...28....+...29....+...30....+...31....+.
 ZEIT         DATUM       ZEIT           ÜBERTRAGUNG    ÜBERTRAGUNGEN
 EINGEGEBEN   GESTARTET   GESTARTET      ZEIT
 SYSTEM
                                                  0               0
   162958     102205      162958                 2               2
   163017     102205      163017                 0               3
   163242     102205      163242                 0               3
 Datenende    ********
```

Auswertung 2 der Journaleinträge

Jetzt können Sie die Daten lesen und natürlich auch auswerten.

Das Sicherheitsprotokoll

Im Gegensatz zu den History- und Jobprotokollen müssen Sie auch das Sicherheitsprotokoll zunächst konfigurieren und aktivieren. Mit dem Sicherheitsprotokoll können Sie die Aktivitäten eines einzelnen Benutzers, eines Objektes oder den Zugriff eines Benutzers auf ein bestimmtes Objekt protokollieren. Sicherheitsprotokolle können helfen, Sicherheitsprobleme zu lokalisieren, da sie nicht nur die erfolgreiche Ausführung sondern jeden versuchten Funktionsaufruf registrieren. Anders gesagt, falls Sie z.B. einen Benutzer verdächtigen, die System Service Tools (SST) zu verwenden, um Ihre Sicherheitseinstellungen zu umgehen, prüfen Sie diesen Benutzer auf *SERVICE, um sich hoffentlich alle erfolglosen Versuche dieses Benutzers anzeigen zu lassen. Das Sicherheitsprotokoll aktiviert man meistens nur für kurze Zeiträume, nämlich immer dann, wenn spezifische Probleme auftreten.

2.3.2.4
Seite 2

Einrichtung des Sicherheitsprotokolls

1. Erstellen Sie zunächst mit dem Befehl CRTLIB (create Library) eine Bibliothek.

```
CRTLIB  SECURITY
```

2. Als nächstes müssen Sie wiederum einen Journalempfänger erstellen, denn auch das Sicherheitsprotokoll wird in Journalen aufgezeichnet:

```
CRTJRNRCV   JRNRCV(security/secaud0001)  AUT(*EXCLUDE)    +
TEXT („Sicherheitsprotokollempfänger“)
```

3. Jetzt können Sie das Journal erzeugen, wenn es noch nicht existiert. Der Name ist allerdings wiederum nicht frei wählbar. Verwenden Sie einfach den folgenden Befehl:

```
CRTJRN QSYS/QAUDJRN  JRNRCV(security/SECAUD0001)    +
AUT(*EXCLUDE)  TEXT („Sicherheitsjournal“)
```

2.3.2.4.1

Seite 2

Systemwerte

Das Sicherheitsprotokoll wird durch vier Systemwerte bestimmt:

Der Systemwert QAUDCTL (audit control)
Dieser startet und beendet das Protokoll. Sobald Sie einen anderen Wert als *NONE angeben, wird die Aufzeichnung gestartet. Wenn Sie den Wert wieder auf *NONE zurücksetzen, wird die Aufzeichnung gestoppt.

Der Systemwert QAUDLVL (audit level)
Der bestimmt die Objektaktionen, die überwacht werden.

Wert	Bedeutung
*AUTFAIL	Alle Berechtigungsfehler werden protokolliert, z.B. unzulässige Kennworteingabe, Jobübergaben.
*CREATE	Alle Objekterstellungen mit Ausnahme von Objekten in der Bibliothek QTEMP.
*DELETE	Alle Objektlöschungen werden protokolliert. Objekte der Bibliothek Qtemp werden nicht aufgezeichnet.
*JOBDTA	Alle Jobveränderungen werden aufgezeichnet: starten, stoppen, ändern eines Jobs usw.
*NETBAS	Netzbasisoperationen werden protokolliert, z.B. Socketverbindungen.
*NETCLU	Operationen für Cluster werden protokolliert, z.B. erstellen, löschen usw.
*NETCMN	Netzbasisfunktionen, Netzclusteroperationen, Netzfehler, Socketfunktionen.
*NETFAIL	Netzfehler werden protokolliert.
*NETSCK	Sockettasks, z.B. zugeordnete DHCP-Adressen, Connect usw.
*OBJMGT	Generische Objekttasks, z.B. umbenennen und übertragen von Objekten.
*OPTICAL	Alle optischen Funktionen werden protokolliert, z.B. Kopieren, Versetzen oder Umbenennen einer optischen Datei.
*PGMADP	Die Übernahme der Programmeignerberechtigung wird protokolliert.
*PGMFAIL	Programmfehler werden protokolliert.
*PRTDTA	Druckfunktionen werden protokolliert, z.B. Drucken einer Spool-Datei.
*SAVRST	Informationen, die das Sichern und Zurückspeichern betreffen, werden protokolliert, z.B. wenn sich der Eigner oder Berechtigungen bei zurück gespeicherten Objekten ändert.
*SECCFG	Die Sicherheitskonfiguration wird protokolliert, z.B. Änderungen der Systemwerte, Umgebungsvariablen und Netzwerkattribute.
*SECDIRSRV	Änderungen oder Updates bei der Ausführung von Verzeichnis-Servicefunktionen werden protokolliert.

2.3.2.4.2
Seite 2

*SECIPC	Änderungen der Interprozesskommunikation werden protokolliert, z.B. Änderung des Eigentumsrechts oder der Berechtigung eines IPC-Objekts.
*SECNAS	Aktionen des Netzwerkauthentifizierungsservice werden protokolliert, z.B. Service-Ticket gültig, Diskrepanz bei Ticket-IP-Adresse usw.
*SECRUN	Sicherheitslaufzeitfunktionen werden protokolliert, z.B. Änderungen des Objekteignerrechts, Änderungen der Berechtigungsliste oder der Objektberechtigung usw.
*SECSCKD	Socket-Deskriptoren werden protokolliert.
*SECURITY	*SECURITY setzt sich aus folgenden Werten zusammen: *SECCFG, *SECDIRSRV, *SECIPC, *SECNAS, *SECRUN, *SECSCKD, *SECVFY, *SECVLDL
*SERVICE	Servicebefehle und API-Aufrufe werden protokolliert.
*SPLFDTA	Spool-Dateifunktionen werden protokolliert, z.B. Erstellen, Löschen, Anzeigen, Kopieren, Anhalten, Freigeben einer Spool-Datei usw.
*SYSMGT	Systemverwaltungstasks werden protokolliert , z.B. Änderungen der Systemantwortliste

Eine vollständige Beschreibung der Protokollierungsangaben entnehmen Sie bitte den IBM Handbüchern.

Der Systemwert QAUDFRCLVL (audit force level)
Dieser bestimmt die Anzahl der Protokolleinträge, die in das Sicherheitsprotokoll geschrieben werden können, bevor die Journaldaten auf die Festplatte gestellt werden.

Der Eintrag *SYS legt fest, dass die Journaleinträge nur dann in den Zusatzspeicher geschrieben werden, wenn das System feststellt, dass dies abhängig von der internen Systemverarbeitung erforderlich ist. Diese Angabe liefert zwar die besten Protokollierungsergebnisse, kann aber auch zum höchsten Datenverlust bei einem abnormalen Ende führen. Ein numerischer Wert zwischen 1 und 100 legt die Anzahl der Einträge fest, die im Sicherheitsprotokolljournal notiert werden, bevor die Protokollierungsdaten in den Zusatzspeicher geschrieben werden. Je kleiner diese Anzahl ist, umso größer sind die Auswirkungen auf die Systemleistung.

Der Systemwert QAUDENDACN (audit end action)
Der Wert gibt die Aktion an, die vom System ausgeführt wird, wenn keine Protokollsätze an das Protokolljournal gesendet werden können, da beim Senden des Journaleintrags Fehler aufgetreten sind.

Der Befehl CHGOBJAUD

Mit dem Befehl CHGOBJAUD (change object audit) stellen Sie einzelne oder eine Gruppe von Objekten unter Überwachung.

```
                    Objektprotokollierung ändern (CHGOBJAUD)

 Auswahl eingeben und Eingabetaste drücken.

 Objekt . . . . . . . . . . . . . > PAANS00      Name, generisch*, *ALL
   Bibliothek . . . . . . . . . .     *LIBL      Name, *LIBL, *USRLIBL...
 Objektart  . . . . . . . . . . . > *FILE       *ALL, *ALRTBL, *AUTHLR...
 ASP-Einheit  . . . . . . . . . .   *            Name, *, *SYSBAS
 Objektprotokollierungswert . . . > *CHANGE     *NONE, *USRPRF, *CHANGE, *ALL
```

Objektprotokollierung ändern (CHGOBJAUD)

Die Datei PAANS00 wird in diesem Beispiel dahingehend überwacht, ob Veränderungen am Objekt vorgenommen werden. Hierbei werden nicht nur die erfolgreichen, sondern selbstverständlich auch die abgewiesenen Versuche protokolliert.

Der Parameter Objektprotokollierungswert (OBJAUD)
Dieser bestimmt, welche Aktionen geprüft werden sollen.

*NONE	Deaktiviert die Prüfung für das Objekt.
*USRPRF	Zeichnet die Prüfdaten auf, die im jeweiligen Benutzerprofil im Parameter AUDLVL abgerufen werden.
*CHANGE	Zeichnet alle Änderungen oder Änderungsversuche am Objekt auf.
*ALL	Überwacht alle Versuche, das Objekt zu ändern oder zu verwenden.

2.3.2.4.3
Seite 2

Der Befehl CHGUSRAUD

Mit dem Befehl CHGUSRAUD (change user audit) überwachen Sie einzelne Benutzer. Dabei bestimmt der Parameter AUDLVL (audit level), welche Aktionen geprüft werden sollen. Sie können die Überprüfungsattribute im Benutzerprofil anzeigen lassen, indem Sie den Befehl DSPUSRPRF (display user profil) verwenden. Aktiviert wird die Überwachung allerdings erst durch den Systemwert QAUDCTL.

```
                Benutzerprotokollierung ändern (CHGUSRAUD)

Auswahl eingeben und Eingabetaste drücken.

Benutzerprofil . . . . . > LGADOZ02     Name
     + für weitere Werte
Objektprotokollierungswert . . . > *CHANGE      *SAME, *NONE, *CHANGE, *ALL
Benutzeraktionsprotokollierung     *none_       *SAME, *NONE, *CMD...
     + für weitere Werte
```

Benutzerprotokollierung ändern (CHGUSRAUD)

Der Parameter Objektprotokollierungswert (OBJAUD)

Dieser ist nur wirksam, wenn für das Objekt, auf das zugegriffen wird, der Wert *USRPRF für den Parameter OBJAUD (Objektprotokollierungswert) angegeben ist. Der Parameter unterstützt die folgenden Werte:

*NONE	Deaktiviert die Prüfung für das Objekt.
*USRPRF	Zeichnet die Prüfdaten auf, die im jeweiligen Benutzerprofil im Parameter AUDLVL abgerufen werden.
*CHANGE	Zeichnet alle Änderungen oder Änderungsversuche am Objekt auf.
*ALL	Überwacht alle Versuche, das Objekt zu ändern oder zu verwenden.

2.3.2.4.4
Seite 2

Der Parameter Benutzeraktionsprotokollierung (AUDLVL)

Er bestimmt die Stufe der Aktivität, die für dieses Benutzerprofil protokolliert wird. Die Systemwerte QAUDLVL und QAUDLVL2 werden in Verbindung mit diesem Parameter benutzt. Wird z.B. für QAUDLVL der Wert *DELETE und für AUDLVL2 der Wert *CREATE angegeben, werden sowohl *DELETE als auch *CREATE für den Benutzer protokolliert.

Wert	Bedeutung
*NONE	Es ist keine Protokollierungsstufe angegeben. Die Protokollierungsstufe für diesen Benutzer wird den Systemwerten entnommen.
*CMD	Protokolliert alle Befehle, die der Benutzer verwendet, einschließlich Prozeduren der System/36-Umgebung.
*CREATE	Protokolliert alle Objekterstellungen des Benutzers.
*DELETE	Protokolliert alle Objektlöschungen des Benutzers.
*JOBDTA	Protokolliert alle Jobaktionen, z.B. Starten und Beenden eines Jobs.
*OBJMGT	Protokolliert alle Objekte, die von dem Benutzer umbenannt oder verschoben werden.
*OFCSRV	Protokolliert Office-Vivion-Benutzeraktionen.
*PGMADP	Protokolliert die Ausführung von Programmen, die eine Zugriffsberechtigung erfordern.
*SAVRST	Protokolliert Sicherungs-/Wiederherstellungsfunktionen.
.SECURITY	Protokolliert Sicherheitsfunktionen.
*SERVICE	Protokolliert Zugriffsversuche auf SST oder DST.
*SPLFDTA	Protokolliert Spool-Dateiaktivitäten.
*SYSMGT	Protokolliert Systemverwaltungsaktivitäten.

Abschließende Überlegungen

Bei der Sicherheitsüberwachung wird oftmals der Fehler gemacht, dass zu viele Daten angefordert werden, die für die Auswertung nicht geeignet sind, oder es werden komplizierte Prüfregeln aufgestellt, die Daten aber niemals ausgewertet. So sollte es natürlich nicht sein!! Deshalb sollten Sie Ihre Sicherheitsaufzeichnungen gründlich planen, denn die Aufzeichnung verursacht natürlich Kosten in Bezug auf Leistung und verbrauchten Plattenplatz. Folgende Fragen könnten Ihnen helfen:

- Welcher Benutzer soll überwacht werden?
- Welche Objekte sollen überwacht werden?
- Welche Informationen benötige ich?
- Wie lange benötige ich die Informationen?

Implementieren Sie anschließend genau diese Aspekte für einen entsprechend kurzen Zeitraum, um das vermutete Problem zu lösen. Danach deaktivieren Sie die Überwachung wieder und analysieren Ihre Daten.

Zum Abschluss zwei Beispiele, die Ihnen evtl. helfen können, Ihre Überwachungen zu aktivieren:

Ein Objekt – alle Benutzer

Stellen Sie sich vor, dass Sie eine spezifische Datei überprüfen wollen. Wenn Sie z.B. die Datei „PAANS00“ auf Änderungen überwachen wollen, geben Sie den folgenden Befehl ein:

```
CHGOBJAUD  OBJ(PAANS00)  OBJAUD(*CHANGE)
```

Stellen Sie außerdem sicher, dass der Systemwert QAUDCTL auf *OBJAUD gesetzt ist.

Jeder Benutzer, der jetzt die Datei verändert oder versucht, Änderungen am Objekt vorzunehmen, wird protokolliert.

Mehrere Objekte – ein Benutzer

Jetzt wollen wir prüfen, ob der Benutzer LGA09 versucht, eine der Dateien, die mit der Zeichenfolge „PA“ beginnen, zu verändern. Geben Sie hierfür den folgenden Befehl ein.

```
CHGOBJAUD  OBJ(*LIBL/PA*)  OBJAUD(*USRPRF)
```

Beachten Sie, dass der Befehl CHGOBJAUD sowohl generische Bibliotheks- als auch generische Objektnamen akzeptiert. Das erspart Ihnen Zeit, Ihre Sicherheitsrichtlinien zu implementieren. Arbeiten Sie aber nicht mit dem

2.3.2.4.5
Seite 2

Wert *ALL/*ALL. Die Implementierung durch das System könnte Stunden dauern und später das Leistungsverhalten der iSeries negativ beeinflussen.

Nachdem Sie die zu überwachenden Objekte bestimmt haben, bestimmen Sie abschließend das Benutzerprofil.

```
CHGUSRAUD  USER(LGA09)  OBJAUD(*CHANGE)
```

Der Systemwert QAUDCTL muss wiederum auf *OBJAUD gesetzt sein.

Objekt- und Benutzerverwaltung

Der iSeries-Server kann sehr vielfältig eingesetzt werden. Doch egal, ob Sie die iSeries als reinen Datenbank-Server oder als Applikations-Server einsetzen, eine zentrale Rolle Ihrer Tätigkeiten und Überlegungen sollte der Objekt- und Benutzerverwaltung gelten. Auf einem kleinen System gibt es vielleicht nur ein oder zwei Personen, die Zugang zu allen Objekten erhält und es werden nur wenige Benutzerprofile und -gruppen benötigt. In großen Unternehmen hingegen werden Sie mit einer Vielzahl von Objekten konfrontiert, die verwaltet und für unterschiedliche Benutzergruppen berechtigt werden müssen. Doch egal wie komplex Ihr System ist, Sie benötigen verschiedene Benutzerprofile, die die individuellen Rechte und Zugangsberechtigungen für die Objekte bestimmen. Die iSeries unterstützt Benutzergruppen und Objektberechtigungskonzepte, die so einfach oder so komplex sind wie Sie es wünschen. Dabei ist das Benutzer- und Objektberechtigungskonzept abhängig von verschiedenen Systemwerten, wobei der Systemwert QSECURITY sicherlich eine entscheidende Rolle spielt.

11. Ergänzung 06/2007

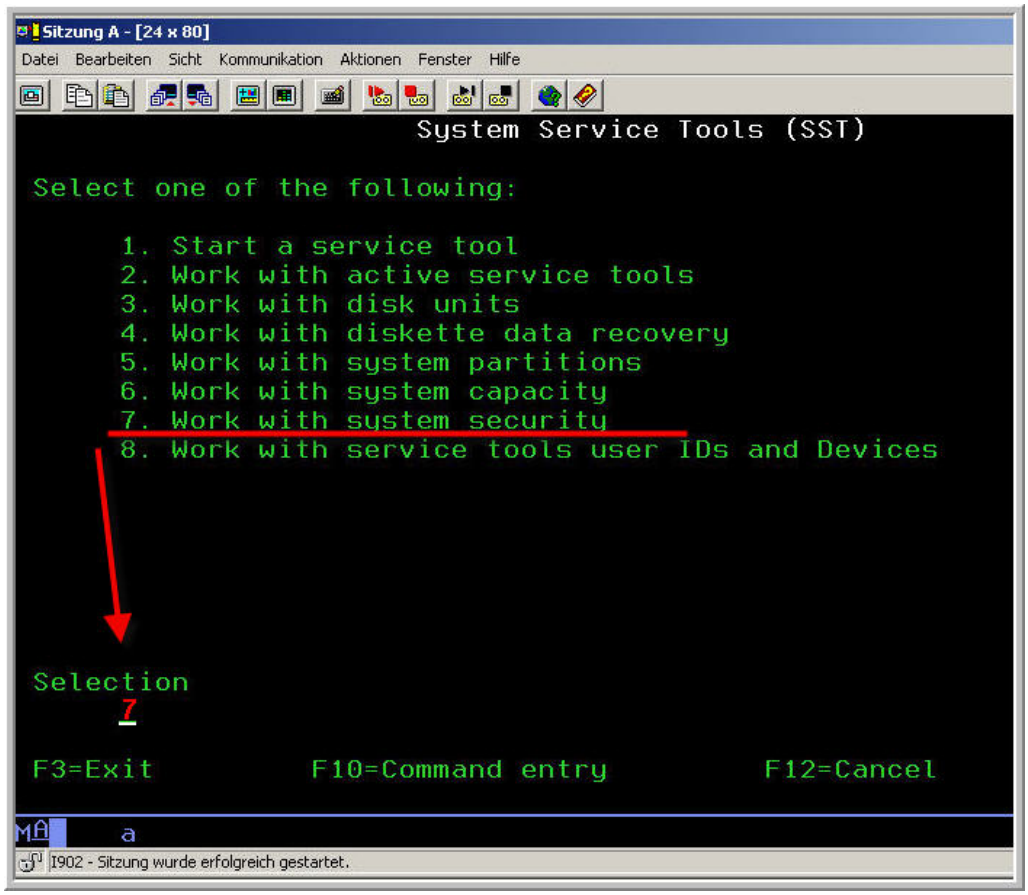
2.4
Seite 2

Systemwerte

Systemwerte sind globale Parameter, die für das gesamte System geltende Vorgaben enthalten.

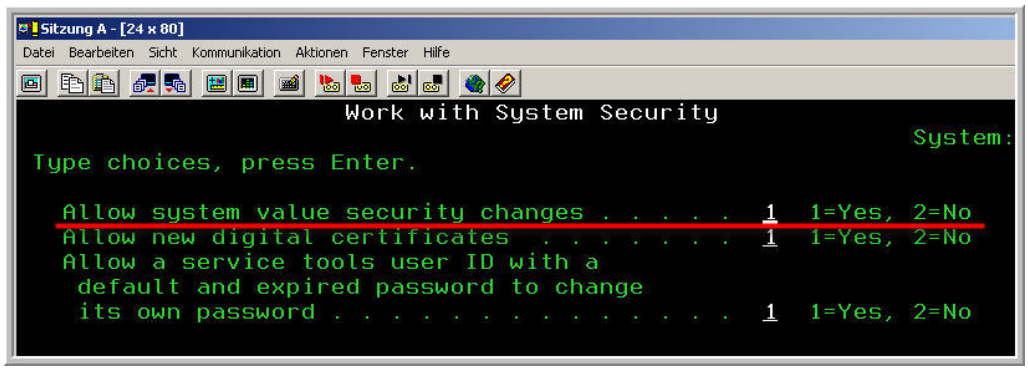
Aber Vorsicht! Systemwerte beeinflussen das gesamte System. Die eingestellten Systemwerte können Sie mit i5/OS-Befehlen oder mit dem iSeries Navigator abfragen und/oder verändern. Ein Benutzer kann Systemwerte ändern, um z. B die Arbeitsumgebung zu definieren oder Sie können beispielsweise als Systemwert für das Datumsformat JJ/MM/TT, MM/TT/JJ, TT/MM/JJ oder julianisch angegeben werden. Auch die Bibliotheksliste ist ein Beispiel für die Verwendung von Systemwerten. Wann eine Systemwertänderung wirksam wird, ist vom jeweiligen Systemwert abhängig. Nicht alle Systemwertänderungen werden sofort nach ihrer Änderung vom System übernommen. Bei einigen Werten, wie etwa der Sicherheitsstufe QSECURITY wird die Änderung erst beim nächsten IPL des Servers wirksam. Informationen darüber, wann ein bestimmter Systemwert wirksam wird, finden Sie im entsprechenden Hilfetext für den Systemwert. Natürlich stellt sich auch die Frage „Wer darf Systemwerte ändern?“. Und auch darauf lässt sich keine eindeutige Antwort geben. Die Sonderberechtigungen im jeweiligen Benutzerprofil legen fest, ob ein Benutzer den entsprechenden Systemwert ändern darf. Welche Sonderberechtigungen für einen bestimmten Systemwert erforderlich sind, müssen Sie wiederum dem Hilfetext für den entsprechenden Systemwert entnehmen. Sie können zusätzlich über die Systemservice-Tools (SST) sicherheitsrelevante Systemwerte für Änderungen freigeben oder sperren. Hierzu öffnen Sie zunächst eine 5250-Emulation und geben den Befehl STRSST (start system servicetool) ein. Daraufhin öffnet sich ein Dialog für die Eingabe Ihres Benutzernamens und Kennwortes. Anschließend gelangen Sie in den folgenden Dialog:

2.4.1
Seite 2



System Service Tools

Wenn Sie jetzt die Option 7 „Arbeiten mit Systemsicherheit“ wählen, erscheint ein weiterer Bildschirm.



Work with System Security

Hier können Sie im Feld „Ändern sicherheitsbezogener Systemwerte zulassen“ entweder den Wert 1 = Ja angeben, um die sicherheitsbezogenen Systemwerte zu entsperren, oder eine 2 = Nein eintragen, um die sicherheitsbezogenen Systemwerte zu sperren. Die Default-Einstellung erlaubt im Übrigen die Veränderung systemrelevanter Systemwerte. Falls Sie die Defaulteinstellung verändern und es wird anschließend versucht einen der sicherheitsrelevanten Systemwerte zu ändern, wird die Änderung zurückgewiesen und der Benutzer erhält die Nachricht „CPF18C0“. Sicherheitsrelevante Systemwerte sind in diesem Zusammenhang die folgenden Systemwerte:

Sicherheitsrelevante Systemwerte

Jedes Mal wenn Sie einen Systemwert ändern, sollten Sie anschließend die neuen Systemwerteinstellungen sichern, damit Ihnen keine Systemwertdaten verloren gehen, falls Ihre Systemwerte beschädigt werden. Systemwerte werden in der Systembibliothek QSYS gespeichert. Sie müssen also die Bibliothek QSYS sichern, um die aktuellen Systemwerteinstellungen zu sichern. Es ist egal, ob Sie eine Systemsicherung durchführen oder nur die Systemdaten sichern. Auch der Befehl SAVSYSINF (save systeminformation) ist zur Sicherung der Systemwerteinstellungen ausreichend. Falls Sie mit BRMS arbeiten, verwenden Sie die Sicherungsrichtlinie *SYSTEM oder *SYSGRP.

16. Ergänzung 9/2008

2.4.1
Seite 4

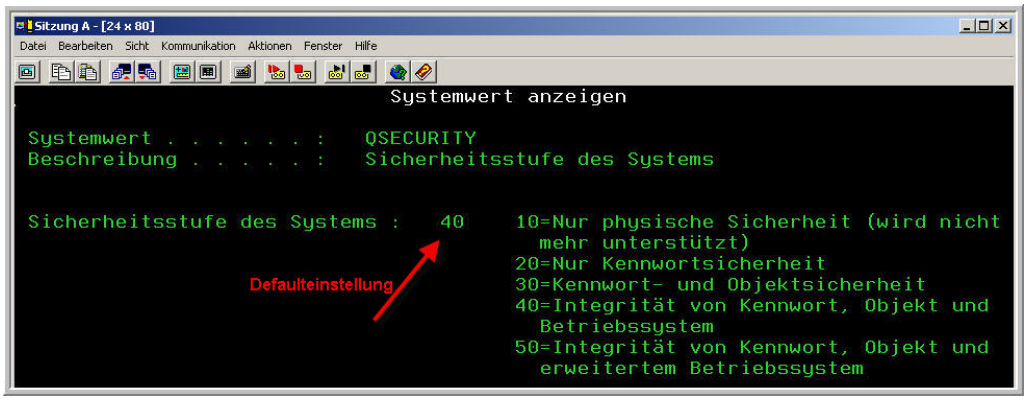
Systemwerteinstellungen mit i5/OS-Befehlen bearbeiten

Sie können jederzeit mit dem i5/OS-Befehl DSPSYSVAL (display system value) aktuelle Systemwerteinstellungen abfragen und – falls Sie die nötigen Berechtigungen besitzen – die Werte mit dem Befehl CHGSYSVAL (change system value) ändern. Außerdem bietet sich der Befehl WRKSYSVAL (work with system values) an, um mit Systemwerten zu arbeiten. Geben Sie auf einer Kommandozeile den Befehl WRKSYSVAL QPRTDEV ein. Daraufhin sollte der folgende Bildschirm erscheinen:

Der Befehl WRKSYSVAL QPRTDEV

Mit der Option „5 = Anzeigen“ können Sie jetzt den Standarddrucker Ihres Systems ermitteln.

Schauen wir uns auch noch den Systemwert QSECURITY etwas genauer an:



Ausgabe des Befehls DSPSYSVAL QSECURITY

Der Systemwert QSECURITY legt die grundlegenden Sicherheitseinstellungen Ihres Servers fest. Vor dem Release V3R7 lieferte die IBM die Systeme in der Sicherheitsstufe 10 aus, um die Installation zu erleichtern. Die Sicherheitsstufe 10 wird allerdings seit dem Release V4R3 nicht mehr unterstützt. Alle Kunden mussten bei der Installation des Release V4R3 automatisch auf die Sicherheitsstufe 20 umstellen. Heute werden die Systeme in der Sicherheitsstufe 40 ausgeliefert.

2.4.1.1
Seite 2

Sicherheitsstufe 10
Bei der Sicherheitsstufe 10 gab es weder Kennwort- noch Objektschutz. Jede beliebige Eingabe führte dazu, dass ein Benutzerprofil generiert wurde, sofern das Profil noch nicht existierte. Nach der Anmeldung konnte der Anwender ohne Einschränkung mit dem System arbeiten.

Sicherheitsstufe 20
Bei der Sicherheitsstufe 20 ist nur der Kennwortschutz aktiv. Der Benutzer kann sich nur anmelden, wenn das Benutzerprofil dem System bekannt ist. Objektschutz besteht nicht. Der Benutzer kann alle System- und Datenobjekte kopieren, löschen oder ändern.

Sicherheitsstufe 30
Bei der Sicherheitsstufe 30 sind sowohl der Kennwort- als auch der Objektschutz aktiviert. Der Systemwert beinhaltet die Sicherheitsstufe 20. Außerdem sind für jedes Objekt Berechtigungen hinterlegt. Nur entsprechend berechtigte Benutzer können die Objekte bearbeiten.

Sicherheitsstufe 40
Diese Stufe entspricht der Sicherheitsstufe 30. Zusätzlich jedoch werden unberechtigte Programmzugriffe über interne Systemschnittstellen abgewiesen. Der Schutz der internen Schnittstellen beinhaltet, dass Sie z. B. das Befehlsverarbeitende Programm (CPP) eines Betriebssystembefehls nicht direkt aufrufen können. Die Zeigerfunktionalitäten der Sprachen C, PASCAL oder Assembler können nicht eingesetzt werden, um die offiziellen IBM-Systemschnittstellen zu umgehen. In Jobbeschreibungen können keine Benutzerberechtigungen mehr übernommen werden, wenn der Benutzer nicht sowohl für die Jobbeschreibung als auch für das Benutzerprofil die Berechtigung *USE besitzt. In der Sicherheitsstufe 30 ist es bei bestimmten Subsystembeschreibungen möglich, sich ohne Benutzer-Kennung anzumelden. Auf der Stufe 40 werden diese Möglichkeiten ausgeschlossen.

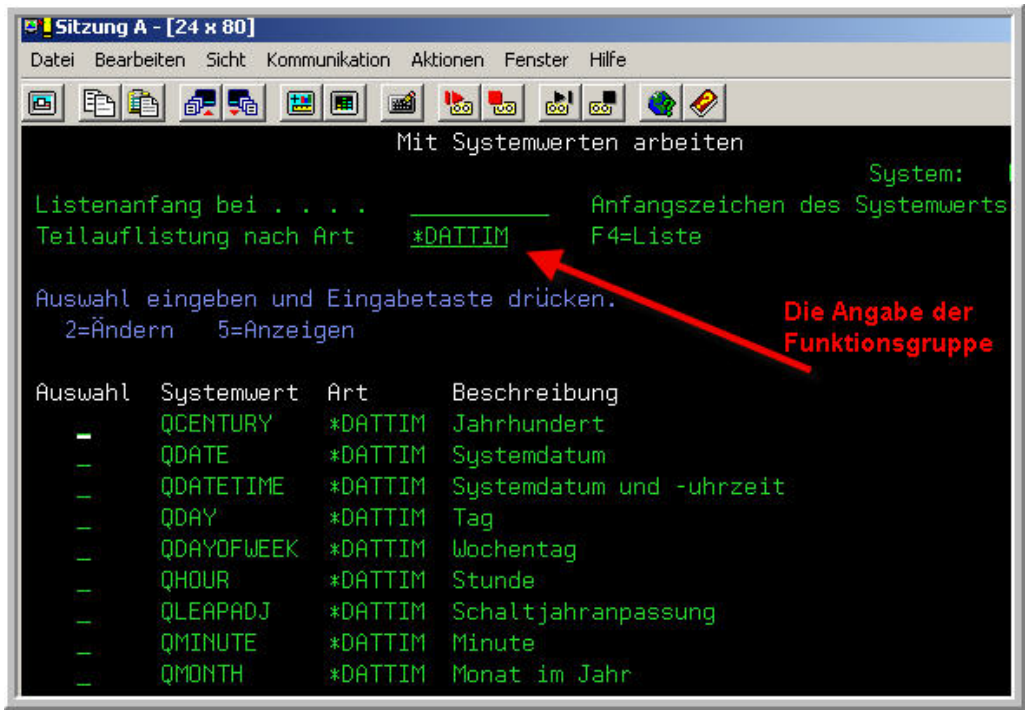
Sicherheitsstufe 50
Die Sicherheitsstufe 50 wurde entwickelt, um die Anforderungen des C2-Sicherheitsstandards zu erfüllen, das vom US-Verteidigungsministerium definiert wurde. Die Stufe 50 bietet erweiterten Integritätsschutz für Konfigurationen mit strengen Sicherheitsanforderungen. Diese Sicherheitsstufe wird vor allem dort eingesetzt, wo das iSeries System einen offenen Zugriff auf Fernbenutzer ermöglicht.

Es gibt derzeit über 150 verschiedene Systemwerte, die in Funktionsgruppen zusammengefasst sind. Die Anzeige „Art des Systemwerts auswählen“ enthält diese Funktionsgruppen. Die Anzeige rufen Sie durch Drücken der Funktionstaste F4 auf, wenn der Positionsanzeiger im Feld „Teilauflistung nach Art...“ in der Anzeige „Mit Systemwerten arbeiten“ steht.

Die Systemwertarten – 5250-Ausgabe

Wenn Sie sich für die Kategorie *ALC entscheiden, enthält die Liste der Systemwerte anschließend nur die Zuordnungssystemwerte. Hierzu gehört der Systemwert QTOTJOB „Anfängliche Anzahl Jobs“ genauso wie der Systemwert QJOBMSGQMX „Maximale Größe der Jobnachrichtenwarteschlange“. *EDT fasst alle Editiersystemwerte zusammen. Die Gruppe *LIBL enthält u.a. die Systemwerte QSYSLIBL und QUSRLIBL. Beide Systemwerte gemeinsam bestimmen den grundlegenden Aufbau der Bibliothekssuchlisten auf Ihrem System. Die Systemwerte der Art *MSG definieren zahlreiche Protokolleinstellungen. Die Kategorie *SEC enthält die Sicherheitssystemwerte. Hier finden Sie auch den Systemwert QSECURITY. Grundlegende Einstellungen des Arbeitsspeichers werden mit Hilfe der Systemwerte der Art *STG vorgenommen und die Gruppe *SYSCTL steht synonym für die Systemsteuerungs-Systemwerte. Sie können die Einzelwerte einer Kategorie anzeigen, indem Sie einfach die entsprechende Systemwertart im Parameter „Teilauflistung nach Art“ eintragen. Ich lasse mir im nachfolgenden Bildschirm alle Systemwerte der Gruppe *DATTIM anzeigen:

2.4.1.1
Seite 4

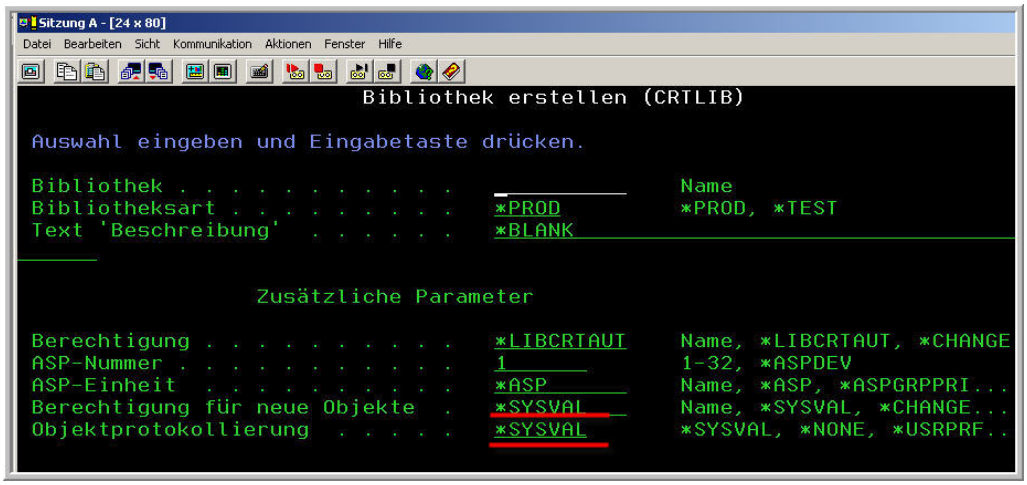


*Die Funktionsgruppe *DATTIM*

In der Funktionsgruppe *DATTIM werden alle relevanten Systemwerte zur Einstellung von Datum und Uhrzeit zusammengefasst.

Jede dieser Funktionsgruppen enthält also eine mehr oder minder große Anzahl Einzelwerte. Ich möchte an dieser Stelle nicht jede Kategorie detailliert erläutern, sondern werde hin und wieder in den entsprechenden Kapiteln auf die jeweils relevanten Systemwerte verweisen. Hinzu kommt, dass die Systemwerte mit fast jedem neuen Release erweitert bzw. einzelne Werte verändert werden. So hat IBM im Release V5R4 z. B. die Kategorie „Jobs“ um zwei neue Systemwerte erweitert. Mit dem ersten Systemwert QALWJOBITP, „Unterbrechen von Jobs zulassen“, können Sie festlegen, ob neue aktive Jobs unterbrochen werden dürfen, um benutzerdefinierte Exit-Programme auszuführen. Mit dem zweiten neuen Systemwert, „Druckausgabe für Jobprotokoll erstellen“ (QLOGOUTPUT), können Sie angeben, ob eine Druckausgabe für ein Jobprotokoll erstellt wird. Außerdem wurde der Systemwert „Aktionsüberwachung aktivieren“ (QAUDCTL) dahingehend geändert, dass Sie jetzt auch die Möglichkeit haben, Abrufereignisse zu überwachen und der Systemwert „Maximale Größe des Jobprotokolls“ (QJOBMSGQMX) erlaubt jetzt die Angabe, 64 MB für die Größe der Jobprotokolldateien zu nutzen. Außerdem wurde der Systemwert „Größe der Systemprotokolldatei“ (QHSTLOGSIZ) in der Kategorie „Nachrichten und Service“ um einen neuen Wert erweitert. Seit dem Release V5R4 kann das System täglich ein neues Systemprotokoll erstellen, anstatt immer nur dann ein neues Protokoll zu erstellen, wenn das aktuelle Protokoll voll ist. Sie sehen es gibt eine Vielzahl verschiedener Systemwerte, die sich zudem auch noch je nach Release unterscheiden können.

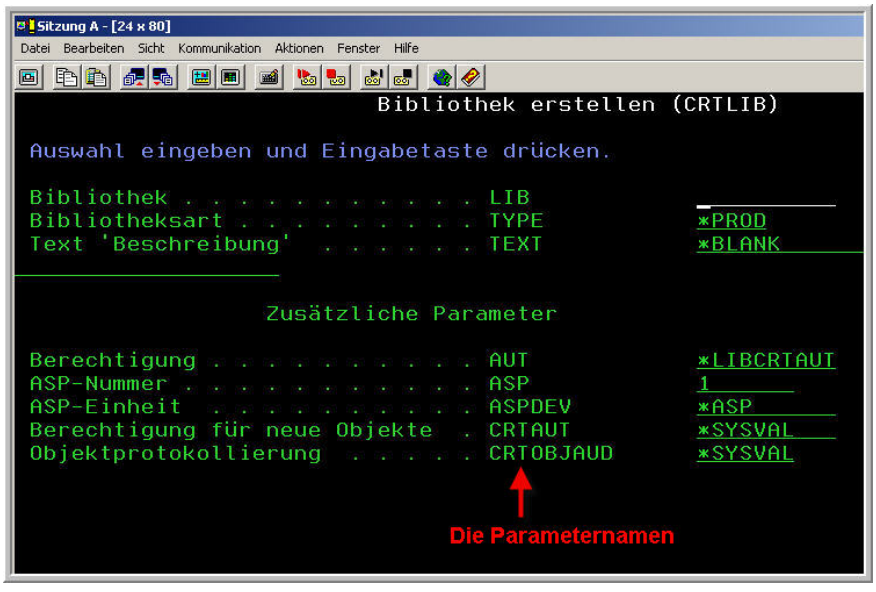
Viel wichtiger erscheint mir daher an dieser Stelle die Frage, wie finden Sie möglichst schnell die entsprechende Systemwerteinstellung, wenn Sie i5/OS-Befehle Systemwerteinstellungen nutzen? Schauen Sie sich dazu die folgende Abbildung an:



Der Befehl CRTLIB

Der Befehl CRTLIB (create libary) verwendet sowohl für die Einstellung der Objektprotokollierung als auch für die Berechtigungssteuerung Systemwerteinstellungen. Es ist naheliegend zu fragen: „Welche Systemwerte werden verwendet?“. Es scheint gar nicht so einfach, die entscheidenden Systemwerte zu finden, oder haben Sie Lust 150 Systemwerte zu durchsuchen? Aber es geht auch einfacher:

Lassen Sie sich zunächst die Parameternamen des Befehls CRTLIB anzeigen. Hierzu drücken Sie die Funktionstaste F11. Daraufhin sollte sich Ihre Bildschirmausgabe verändern:



Anzeige der Parameternamen

2.4.1.1
Seite 6

Sie können jetzt den Namen des entsprechenden Parameters lesen. Der Parameter CRTAUT steuert die „Berechtigung für neue Objekte“ und CRTOBJAUD bestimmt die Objektprotokollierung. Beide Parameterwerte werden von Systemwerten abgeleitet. Die hierfür verwendeten Systemwerte tragen den Namen des jeweiligen Parameters mit dem vorangestellten Buchstaben Q.

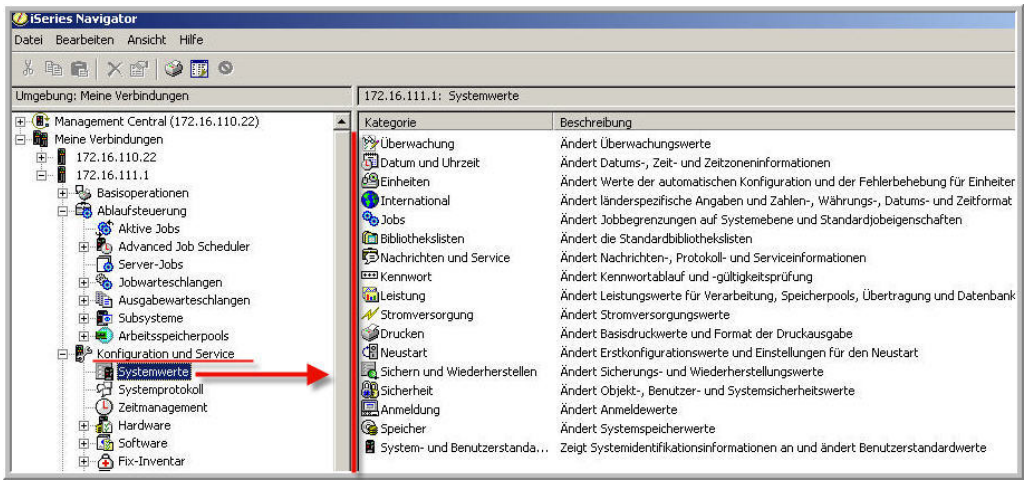
```
Auswahl  Systemwert  Art      Beschreibung
   -       QCRTAUT     *SEC     Allgem. Berechtigung für neue Objekte erstellen
   -       QCRTOBJAUD  *SEC     Objektprüfprotokollierung erstellen
```

Die zugehörigen Systemwerte

Sie sehen, dass die Einstellung des Systemwertes QCRTOBJAUD festlegt, ob Objekte in der neuen Bibliothek aufgezeichnet werden und dass der Systemwert QCRTAUT die Berechtigung der neuen Objekte in der zu erstellenden Bibliothek bestimmt. Dies ist im Übrigen der Grund, warum die Öffentlichkeit *PUBLIC stets per Default ein CHANGE-Recht an allen Objekten einer Bibliothek erhält. Der Wert *CHANGE ist im Systemwert QCRTAUT hinterlegt!

Systemwerteinstellungen mit iSeries Navigator bearbeiten

Auch der iSeries Navigator bietet die Möglichkeit, mit Systemwerten zu arbeiten. Systemwerteinstellungen finden Sie im iSeries Navigator in der Rubrik „Konfiguration und Service“. Wenn Sie diese Rubrik öffnen, erhalten Sie die folgende Bildschirmausgabe:

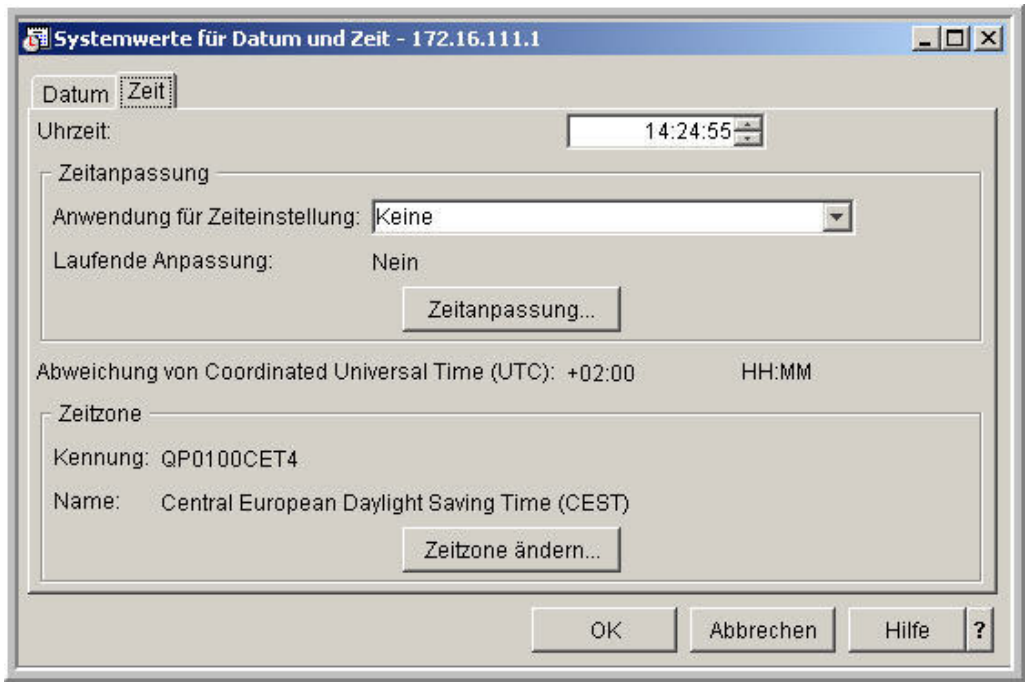


Mit Systemwerteinstellungen im iSeries Navigator arbeiten

Sie sehen, der iSeries Navigator fasst die Systemwerte ebenfalls in Funktionsgruppen zusammen. Hierbei geht der Operations Navigator aber weit über das Gruppenraster hinaus, das uns aus dem Befehl WRKSYSVAL bekannt ist. Es ist sicherlich leicht zu erkennen, dass hier eine differenzierte Gruppierung der Systemwertfunktionen vorgenommen wurde. Auch jetzt werde ich nicht alle oben gezeigten Gruppen detailliert mit Ihnen zusammen durchgehen, sondern nur stellvertretend einige Gruppen exemplarisch vorstellen.

2.4.1.2
Seite 2

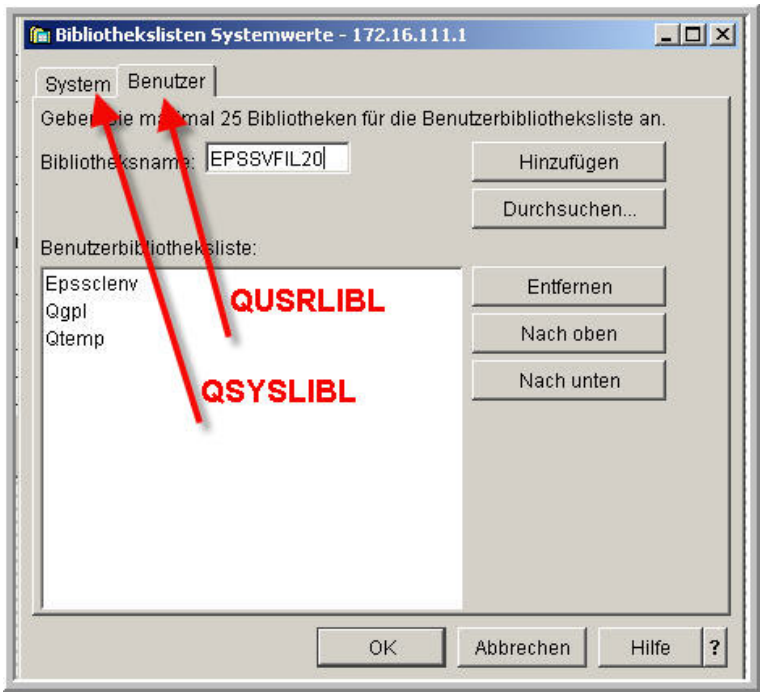
Ich öffne wie schon im 5250-Dialog zunächst die Funktionsgruppe „Datum und Uhrzeit“. So erhalten wir einen direkten Vergleich zwischen der 5250-Darstellung und dem iSeries Navigator:



Die Gruppe der Datums- und Zeitwerte

Die verschiedenen Systemwerte werden im iSeries Navigator nochmals über Register gruppiert. Dies erspart uns die Kenntnis der einzelnen Systemwertnamen. Der Systemwert QDATE wird u.a. im Register „Datum“ und der Systemwert QTIME im Register „Zeit“ dargestellt. Das Register „ZEIT“ gibt die Uhrzeit im Format HHMMSS an. Hierbei gilt: HH = Stunden, mm = Minuten und SS = Sekunden. Wenn Sie die Uhr mit diesem Systemwert einstellen, springt die Uhr entweder vorwärts oder rückwärts. Achtung! Dies kann zu unvorhersehbaren Ergebnissen führen, wenn Programme vor und nach der Änderung dieses Systemwerts auf die Uhr zugreifen. Zum Ändern der Systemzeit ist eine umfassende Kenntnis der Systemumgebung erforderlich. Wenn dieser Systemwert geändert wird, während eine automatische Zeitanpassung vorgenommen wird, wird die laufende Anpassung gestoppt. Eine automatische Zeitanpassung ist der manuellen Einstellung des Systemwerts QTIME stets vorzuziehen, denn die automatische Zeitanpassung stellt sicher, dass die Zeit nicht vorwärts oder rückwärts springt und so unvorhersehbare Fehler verursacht. Sie beschleunigt oder verlangsamt die Zeit eher ein wenig, bis die gewünschte Zeit erreicht ist.

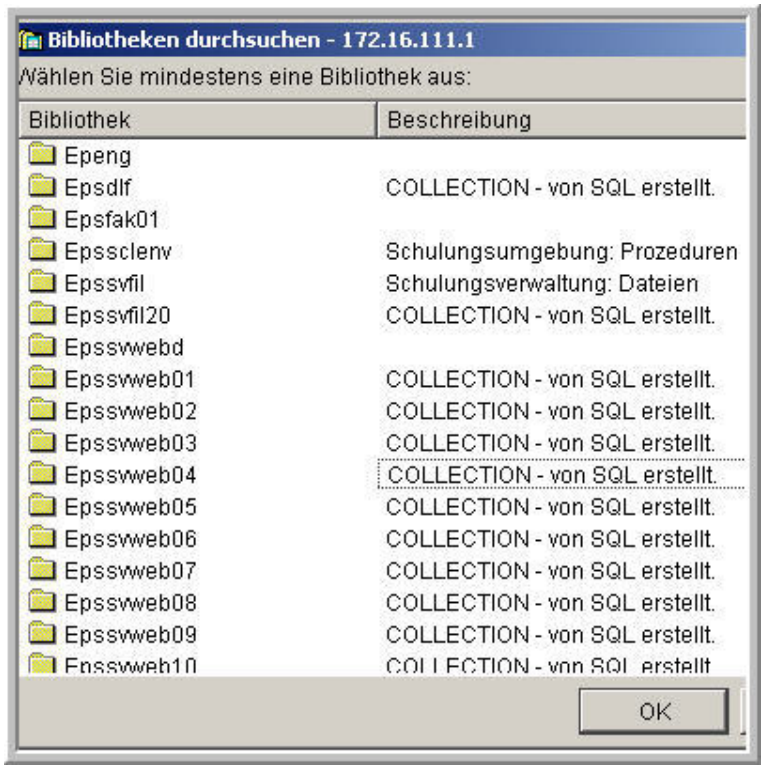
Als nächstes bearbeiten wir die Systembibliothekssuchliste. Dafür öffnen wir selbstverständlich die Kategorie „Bibliothekslisten“. Zwei Systemwerte – QSYSLIBL und QUSRLIBL – bestimmen maßgeblich den Aufbau der Bibliothekssuchliste.



Systemwerte der Bibliotheksliste QSYSLIBL und QUSRLIBL

Hinter dem gezeigten Dialog steht die Bearbeitung des Systemwerts QUSRLIBL. Soll der Benutzerteil der Bibliothekssuchliste geändert werden, so geben Sie den Namen der gewünschten Bibliothek in den Parameter Bibliotheksname ein und drücken dann „Hinzufügen“. Ist Ihnen der Name der Bibliothek jedoch nur teilweise oder gar nicht bekannt, gehen Sie in den Dialog „Durchsuchen“. Daraufhin öffnet sich folgender Dialog:

2.4.1.2
Seite 4



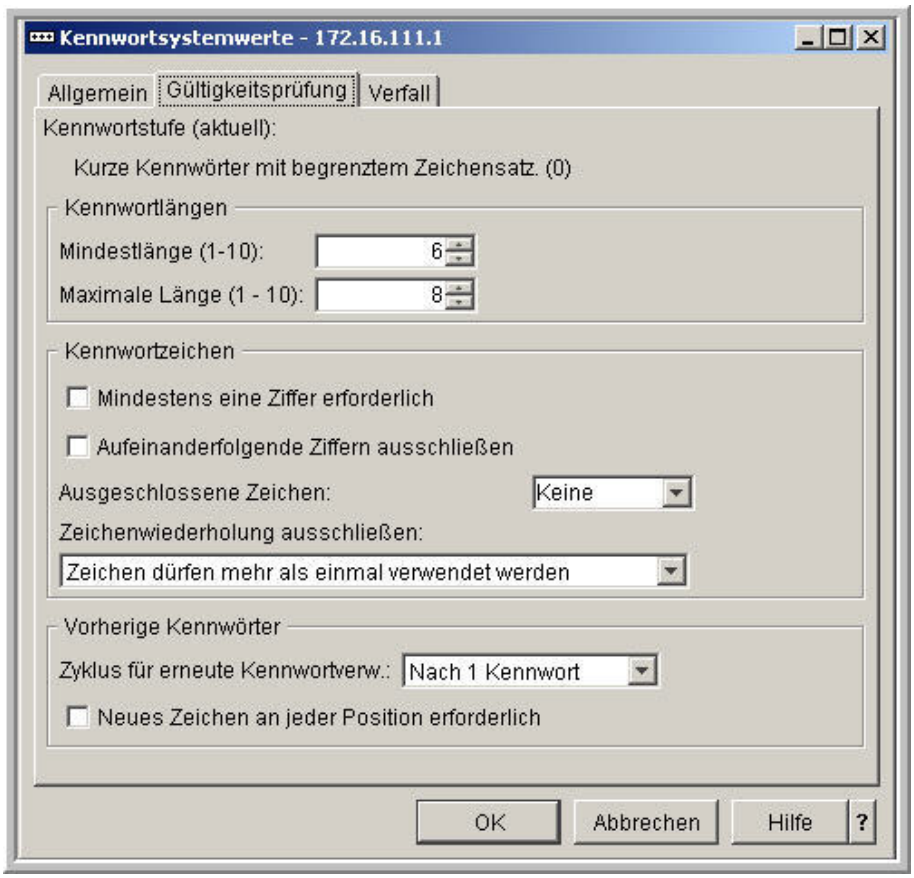
Bibliothek suchen

Wählen Sie aus der gezeigten Liste die gewünschte Bibliothek aus. Auch Mehrfach-Auswahlen sind möglich. Aber Achtung! Es werden hier nur die Bibliotheken des Systems angezeigt, für die Sie ein Leserecht besitzen. Nachdem Sie die betreffende Bibliothek markiert haben, beenden Sie den Dialog durch „OK“. Danach erscheint die ausgewählte Bibliothek im Register „Benutzer“. Verlassen Sie auch diesen Dialog mit dem Button „OK“, damit Ihre Änderungen übernommen werden.

Auch den Standarddrucker finden Sie im iSeries Navigator wieder. Dafür öffnen Sie die Gruppe „Drucker“.

Der Systemwert QPRTDEV

Abschließend möchte ich gern die Systemwertgruppe „Kennwort“ betrachten, da es mir immer wieder passiert, dass ich auf die unzulänglichen Möglichkeiten der Kennwortgestaltung einer iSeries angesprochen werde. Ein Vorwurf der nicht gerechtfertigt ist, wie Sie gleich sehen werden:



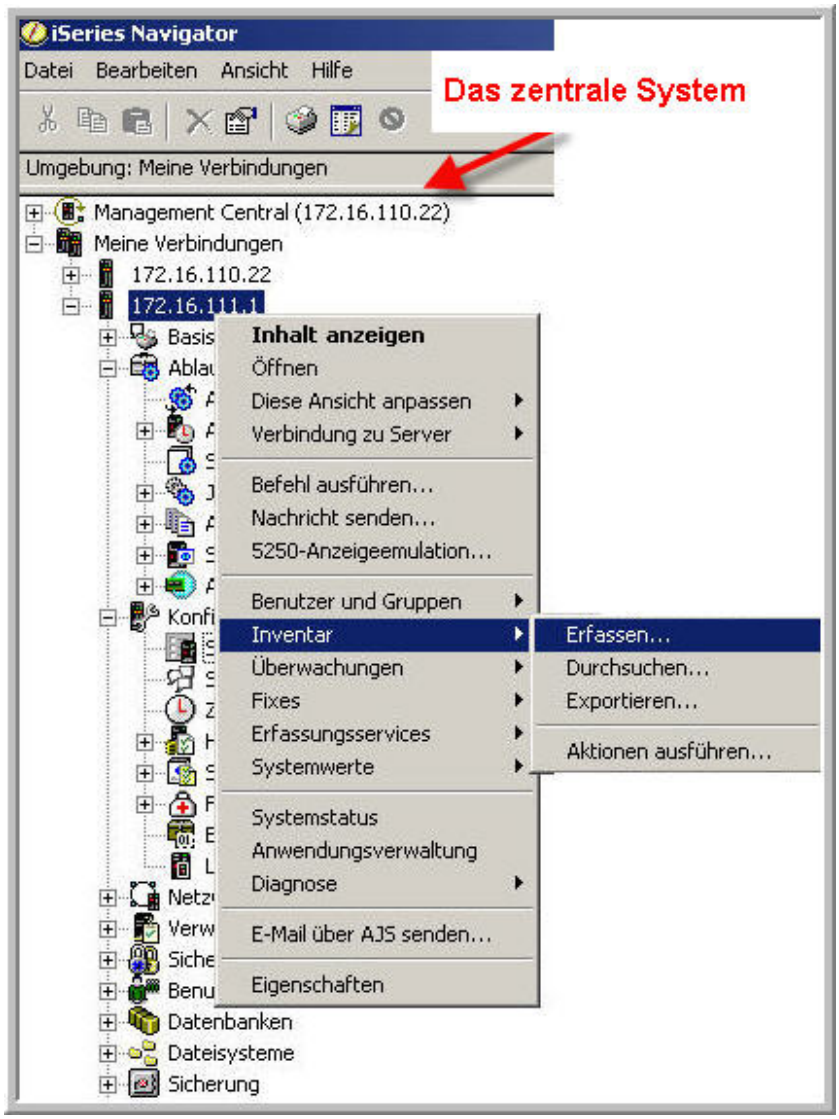
Die Kennwortsystemwerte

Kennwörter können auf einem iSeries Server ebenso komplex gestaltet werden wie auf jedem Windows System. Die Einstellungen für die Kennwortregeln sind allerdings von einer Vielzahl verschiedener Systemwerte abhängig. Allein das Register „Gültigkeitsprüfung“ fasst ca. zehn verschiedene Systemwerte zu einer Gruppe zusammen. In diesem Register legen Sie die Mindestanzahl der Zeichen (Systemwert QPWDMINLEN) für ein Kennwort fest. Die gültigen Werte sind abhängig von der Kennwortstufe Ihres Systems. Wird die Kennwortstufe 0 oder 1 angezeigt, lauten die gültigen Werte für die Mindestlänge 1 bis 10. Wird die Kennwortstufe 2 oder 3 angezeigt, lauten die gültigen Werte für die Mindestlänge 1 bis 128. Die Kennwortstufe wählen Sie im Register „Allgemein“ dieser Anzeige. Auch die maximale Länge des Kennwortes ist abhängig von der Kennwortstufe Ihres Systems. Wird Kennwortstufe 0 oder 1 angezeigt, lauten die gültigen Werte für die maximale Länge 1 bis 10. Wird Kennwortstufe 2 oder 3 angezeigt, lauten die gültigen Werte für die maximale Länge 1 bis 128. Der Standardwert beträgt 8 Zeichen. Neben

der Längenangabe können Sie auch den inhaltlichen Aufbau des Kennwortes bestimmen, d.h. Sie können die Verwendung numerischer Zeichen erzwingen oder die aufeinander folgende Verwendung numerischer Zeichen in einem Kennwort unterbinden. Diese Option bietet zusätzliche Sicherheit, da Benutzer in einem Kennwort keine Geburtsdaten, Telefonnummern oder Ziffernfolgen verwenden können. Außerdem können Sie Zeichen grundsätzlich verbieten. Verbieten Sie z. B. die Verwendung von Vokalen, kann der Benutzer tatsächlich existierende Wörter nicht als Kennwort verwenden. Alle anderen Einstellungen sind selbsterklärend und müssen an dieser Stelle nicht weiter erläutert werden.

Der iSeries Navigator bietet aber noch ein weiteres nützliches Feature, das es in dieser Form im 5250-Umfeld nicht gibt. Mit iSeries Navigator können Sie Ihre Systemwerteinstellungen innerhalb des Netzwerks über mehrere Systeme hinweg vergleichen und aktualisieren. Als Administrator müssen Sie ggf. die Systemwerteinstellungen auf vielen verschiedenen Systemen verwalten. Für diesen Fall ist es hilfreich sich ein Modellsystem aufzubauen, um dann die Systemwerte auf Ihrem Modellsystem mit den Werten auf einem oder mehreren Zielsystemen zu vergleichen. Es ist sogar möglich, die Systemwerte auf den Zielsystemen so zu aktualisieren, dass sie mit den Einstellungen des Modellsystems identisch sind. Wenn Sie möchten, können Sie auch eine Liste mit den Wertdifferenzen zwischen Modell- und Zielsystem generieren, statt die Werte auf dem Zielsystem tatsächlich zu ändern.

Eine Besonderheit gilt es hierbei jedoch zu beachten: Die Systemwerteinstellungen des Zielsystems werden nicht aktuell vom iSeries Server abgerufen, sondern sind in einem Inventar hinterlegt. Sie können an jedem Endpunktsystem seit V5R1 oder einer neueren Betriebssystemversion ein Inventar der Systemwerte anlegen. Erst wenn Sie diese Inventardaten zusammengestellt haben, sind die Informationen in iSeries Navigator für den Abgleich der Systemwerte verfügbar. Lassen Sie uns also zunächst ein entsprechendes Inventar erzeugen. Hierzu führen Sie entweder im Taskpad des Operations Navigator einen Doppelklick auf die Funktion „Inventar erfassen“ aus oder wählen über die Kontexttaste den Eintrag „Inventar erfassen“ aus.



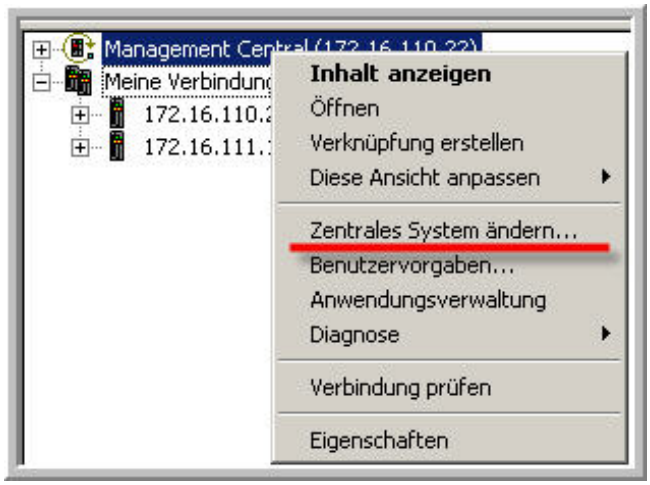
Inventar erfassen

Zunächst erscheint ein Informationsbildschirm, auf dem Ihnen mitgeteilt wird, dass eine Verbindung zum zentralen System hergestellt ist.

Der Informationsbildschirm

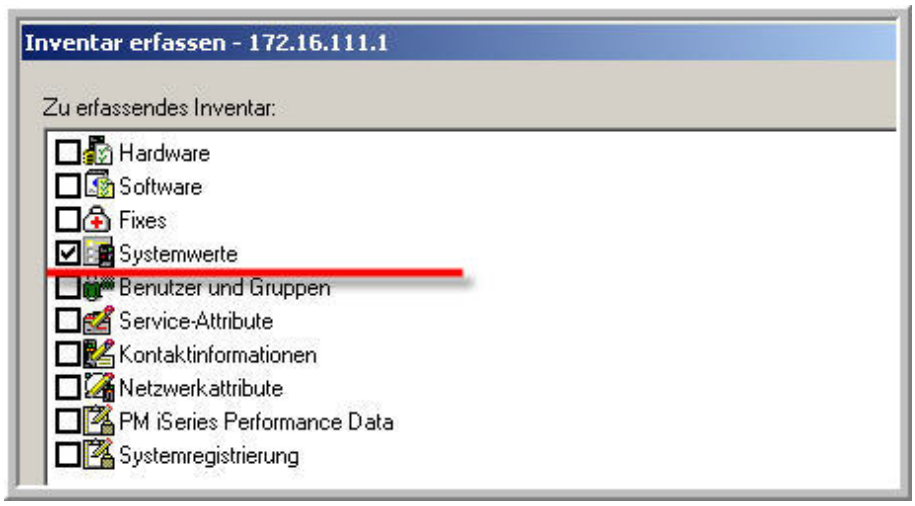
2.4.1.2
Seite 8

Die Inventarisierung und auch der spätere Abgleich sind Funktionen des Management Central, einem besonderen Feature des iSeries Navigator, das wir bislang nicht genutzt haben. Schauen Sie sich die Abbildung „Inventar erfassen“ noch einmal an. Dort sehen Sie die IP-Adresse des zentralen Systems. Das zentrale System wird als eine Art Hauptsystem im Netzwerk verstanden. Es übernimmt die gesamte Inventarisierung der Daten und die Datenübertragung zwischen den Endpunktsystemen. Sie legen das zentrale System fest, indem Sie den Eintrag „Management Central“ markieren und dann die rechte Maustaste drücken. Hier finden Sie den Menüpunkt „Zentrales System ändern“.



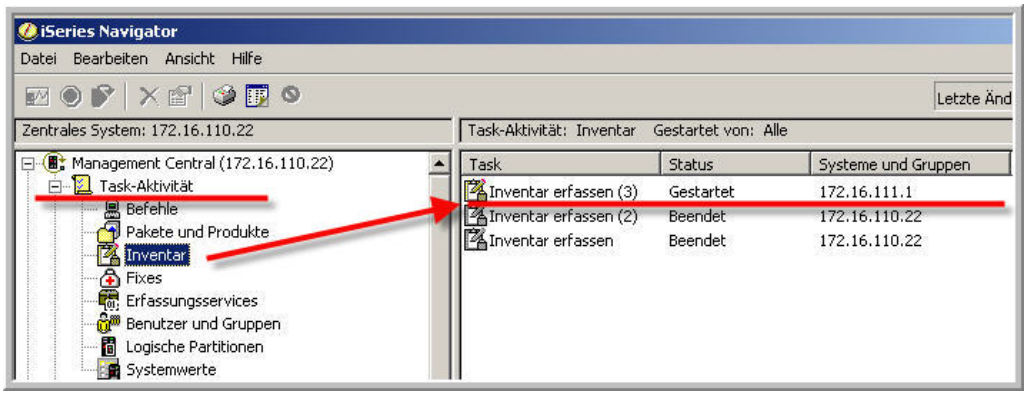
Das zentrale System festlegen

Zu diesem System wird für die Inventarisierung Ihrer Daten jetzt eine Verbindung hergestellt. Sie benötigen daher entsprechende Rechte für das zentrale System, andernfalls scheitert die Anforderung. Sobald die Verbindung hergestellt werden konnte, erscheinen die eigentlichen Dialogbildschirme:



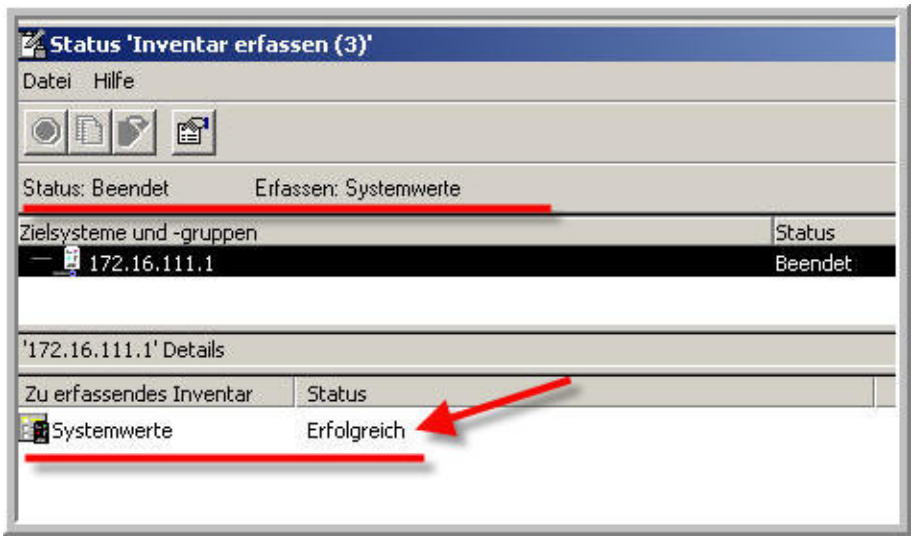
Inventarisierungsumfang festlegen

Im Dialog „Inventar erfassen“ können Sie eine Auswahl der Komponenten treffen, die inventarisiert werden sollen. Sie können einen oder mehrere Inventartypen zur Erfassung auswählen. Für den Moment ist die Inventarisierung der Systemwerte ausreichend. Die Erfassung des Inventars kann sofort ausgeführt werden oder für einen späteren Zeitpunkt geplant werden. Ich starte die Inventarisierung sofort und erhalte daraufhin ein Dialog-Fenster mit der Information, dass ein entsprechender Job auf dem zentralen System gestartet wurde. Den Inventarisierungsjob finden Sie im Management Central. Öffnen Sie „Management Central“ und den Eintrag „Task-Aktivität“. In der Kategorie „Inventar“ finden Sie die aktuellen Inventarisierungen, aber auch alte Jobs, die bereits beendet wurden:



Task-Aktivitäten anzeigen

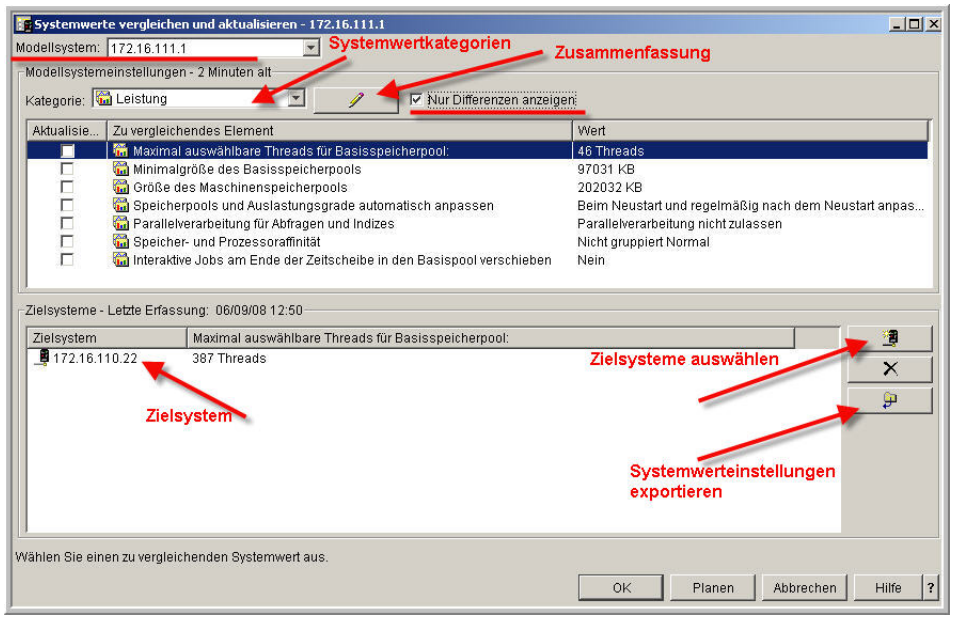
Sie müssen sich die Task-Aktivitäten aber nicht unbedingt anzeigen lassen. Denn sobald die Erfassung abgeschlossen ist, erhalten Sie einen entsprechenden Statusbildschirm. Auf diesem wird Ihnen mitgeteilt, dass die Inventarisierung abgeschlossen wurde. Diesem Bildschirm können Sie auch entnehmen, ob die Inventarisierung erfolgreich verlief:



Informationsbildschirm über den Abschluss der Erfassung

Meine Inventarisierung der Systemwerte wurde erfolgreich beendet, so dass ich jetzt mit dem Vergleich auf verschiedenen Systemen beginnen kann. Vorher aber noch eine kurze Anmerkung: Die Gesamterfassung des Inventars ist keine kurzfristige Aktion. Je mehr Komponenten des Systems erfasst werden, desto länger läuft der Erfassungsprozess. Laufzeiten von mehreren Stunden sind keine Seltenheit. Legen Sie deshalb die Inventarerfassung ggf. in die Nachtverarbeitung.

Ich benötige derzeit keine weiteren Inventarisierungen und öffne jetzt den Dialog für den Systemwertvergleich. Dafür markieren Sie zunächst eines der Endpunktsysteme. Im Kontextmenü wählen Sie „Systemwerte“. Daraufhin können Sie in der Unterauswahl „Vergleichen und aktualisieren“ auswählen, so dass sich der folgende Dialog öffnet:



Systemwerte vergleichen

Für diesen Vorgang benötigen Sie immer ein Modellsystem. Das Modellsystem entspricht einem System mit den optimalen Einstellungen für die Systemwerte. In den folgenden Beispielen ist das System 172.16.111.1 mein Modellsystem. Sie können auswählen, dass die Systemwerte des Modellsystems mit einem oder mehreren Zielsystemen verglichen werden sollen. Es wird zu jedem Zielsystem die aktuelle Einstellung für den Systemwert angezeigt, der in den Einstellungen für das Modellsystem ausgewählt wurde. Die Systemwerte der Zielsysteme werden vom zentralen System abgerufen. Damit diese Daten angezeigt werden, müssen Sie das Systemwerteinventar erfassen, das auf den Zielsystemen vorhanden ist. Wenn einer der Systemwerte nicht erfasst wurde, wird als Beschreibung für die letzte Erfassung der Status „Ein Teil des Inventars wurde nie erfasst“ ausgegeben, und der Wert lautet „Wert nicht erfasst“. Wenn keiner der Systemwerte für das Zielsystem erfasst wurde, lautet die Beschreibung der letzten Erfassung „Letzte Erfassung: Nie“. Den Status „Wert nicht erfasst“ erhalten Sie auch für Systemwerte, die vom Betriebssystem-Release des Zielsystems nicht unterstützt werden. Um ein Zielsystem hinzufügen, klicken Sie auf die Schaltfläche „Hinzufügen“. Anschließend wird eine Liste der verfügbaren Systeme angezeigt. Wählen Sie die gewünschten Systeme aus, und klicken Sie anschließend auf „Hinzufügen“. Um ein Zielsystem wieder aus der Liste zu entfernen, klicken Sie einfach auf die Schaltfläche „Aus Liste entfernen“. Nun müssen Sie auch noch die Kategorie der Systemwerte auswählen, die Sie anzeigen, vergleichen und aktualisieren wollen. Sie können die Systemwerte nach Kategorien anzeigen oder die Kategorie „Zusammenfassung“ auswählen, um alle Systemwerte anzuzeigen, die zur Aktualisierung ausgewählt wurden. Nachdem Sie einen der zu vergleichenden Werte ausgewählt haben, werden die entsprechenden Werte auf den Zielsystemen in der Liste für die Zielsysteme angezeigt. Nach Klicken auf die Schaltfläche „Export“ können Sie das Systemwerteinventar der Zielsysteme auch in einer Datei auf dem PC speichern.

Nützlich ist auch die Option „Nur Differenzen anzeigen“, um nur die Systemwerte und die Zielsysteme anzuzeigen, auf denen die Einstellung des Systemwerts vom Modellsystem abweicht. Falls Sie die Systemwerte jetzt abgleichen wollen, müssen Sie das Kontrollkästchen neben dem entsprechenden Systemwert aktivieren, der aktualisiert werden soll. Einige Systemwerte sind allerdings schreibgeschützt. Dies bedeutet, dass Sie den Wert nicht auswählen und aktualisieren können.

2.4.1.2
Seite 12

Das Objektkonzept

Alle auf einer iSeries gespeicherten Elemente werden als Objekte bezeichnet. Objekte haben eine eindeutige Beschreibung und werden in Bibliotheken gespeichert. Im System iSeries werden somit zusammengehörige Informationen als Objekt zusammengefasst und verwaltet. Ein Objekt kann unter einem definierten Namen angesprochen und mit entsprechenden Systembefehlen bearbeitet werden. Alle Einzelinformationen – wie z.B. der Name, der Ersteller, der Objekteigner, das Datum der letzten Sicherung, Berechtigungen usw. – können nur über das Objekt bearbeitet werden. Die möglichen Objektarten (=-typen) werden automatisch vom System vergeben. Alle Objekte werden in Bibliotheken organisatorisch verwaltet und durch ihre Objektart klassifiziert. Objekte gleichen Namens sind in getrennten Bibliotheken erlaubt. Gleichnamige Objekte in derselben Bibliothek müssen sich durch ihre Objektart unterscheiden.

Zur Veranschaulichung:

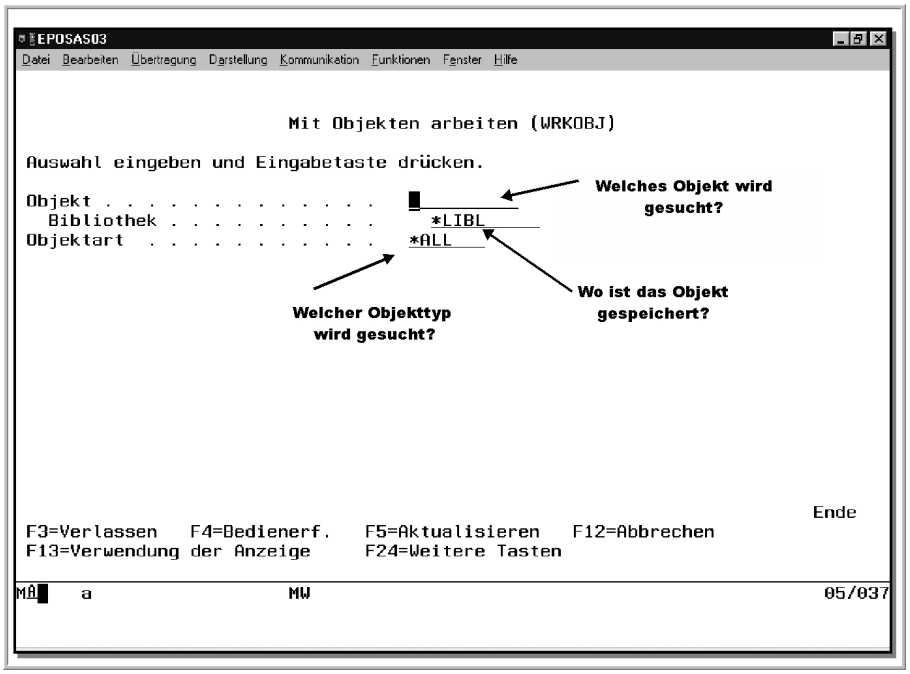
Sie erstellen eine Datei namens „KD001“ und ein Programm, das auch „KD001“ heißen soll. Die Datei bekommt vom System die Objektart *FILE. Nach der fehlerfreien Übersetzung Ihres Programms erhalten Sie das Objekt der Art *PGM. Beide Objekte können aufgrund der unterschiedlichen Objektart in einer Bibliothek verwaltet werden.

*LIB	Bibliotheken	*FILE	Dateien
*MSGF	Nachrichtendateien	*CMD	Befehle
*PGM	Programme	*JOBD	Jobbeschreibung
*JOBQ	Jobwarteschlangen	*OUTQ	Ausgabewarteschlangen

2.4.2
Seite 2

Zentrale Objektverwaltungsbefehle

Immer wieder kommt es vor, dass Sie eine Gruppe von Objekten bearbeiten möchten, oder aber Sie suchen ein ganz bestimmtes Objekt und können es nicht finden. Diese Probleme werden mit dem Befehl WRKOBJ gelöst.



Suchmaske für Objekte

Stellen Sie sich die folgenden Szenarien vor:

Sie möchten wissen, ob das Objekt „KUNDE“ existiert:

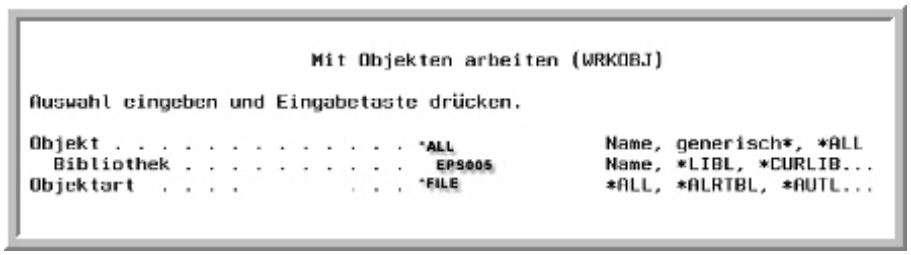
Suche nach einem bestimmten Objekt

Alle Objekte, die den Namen Kunde haben, werden angezeigt.

Wenn Sie die Objektart eingrenzen wollen, müssen Sie den gesuchten Objekttyp im Parameter Objektart eintragen.

2.4.2.1
Seite 2

Sie möchten wissen, welche Dateien in der Bibliothek „EPS005“ gespeichert sind:



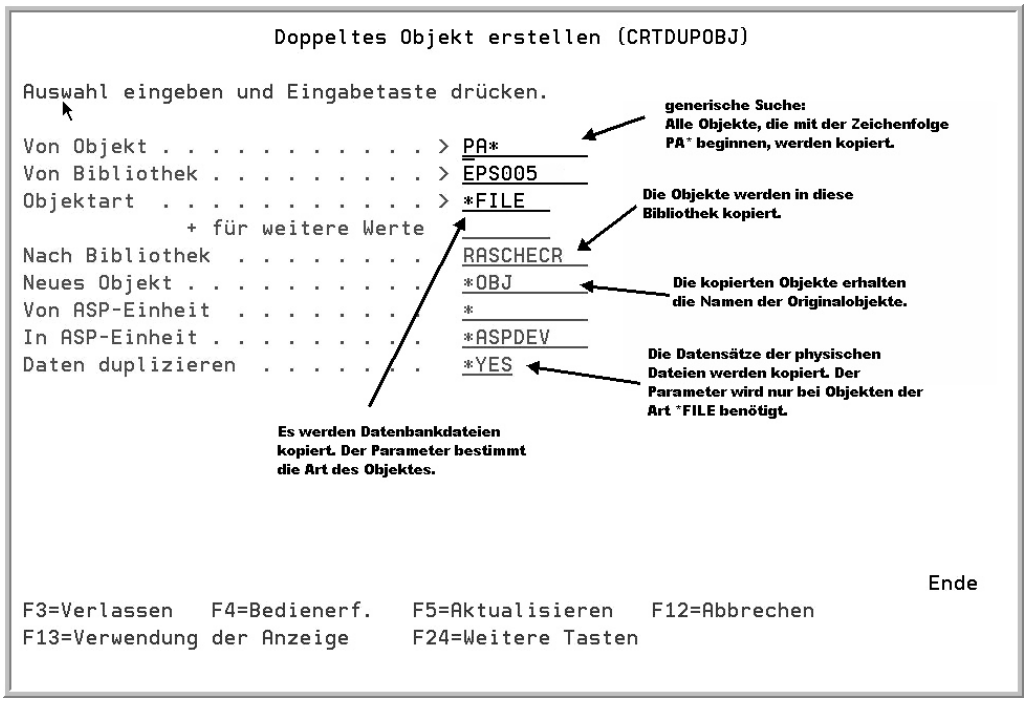
Suche in einer bestimmten Bibliothek

Sie möchten alle Dateien, die mit der Zeichenfolge „PA“ beginnen, bearbeiten:

Suche nach einer bestimmten Zeichenfolge

Der Befehl WRKOBJ stellt Ihnen alle Objektbearbeitungsbefehle über entsprechende Auswahlen zur Verfügung. Natürlich können Sie die Befehle auch direkt einsetzen.

WRKOBJ	Mit Objekten arbeiten
CRTDUPOBJ	Objekt kopieren
RNMOBJ	Objekt umbenennen
DSPOBJD	Objektbeschreibung anzeigen
MOVOBJ	Objekt verschieben



Duplizieren eines Objektes

Beachten Sie bitte, dass für einige Objektattribute kein 100-prozentiges Duplikat erzeugt wird. Das doppelte Objekt verfügt zwar über identische Berechtigungen, aber Eigentümer des neuen Objekts wird der Benutzer, der das Objekt kopiert. Das neu erstellte Objekt muss umbenannt werden, wenn es in derselben Bibliothek wie das Originalobjekt gespeichert werden soll.

2.4.2.1

Seite 4

Objektorganisation in Bibliotheken

Sie wissen bereits, dass alle iSeries-Objekte in Bibliotheken gespeichert werden. Bibliotheken sind Objekte der Art *LIB.

WRKLIB	Bibliotheken verwalten
DSPLIB	Bibliotheken anzeigen
DLTLIB	Bibliotheken löschen
CHGLIB	Bibliotheksattribute verändern
CRTLIB	Bibliotheken erstellen
CLRLIB	Bibliotheksinhalt löschen

Anders als Windows-Verzeichnisse können Bibliotheken nicht hierarchisch gruppiert werden – mit Ausnahme der Systembibliothek QSYS. In der Bibliothek QSYS werden sowohl alle Systembibliotheken als auch alle Benutzerbibliotheken als Eintrag geführt. Systembibliotheken sind dadurch gekennzeichnet, dass Ihr Name mit einem Q beginnt. Als Anwender sollten Sie Ihre Objekte nicht wahllos in Bibliotheken speichern, sondern bei der Erstellung und Benutzung von Bibliotheken bestimmte Regeln beachten und für Ihr Unternehmen dokumentieren.

- Alle Objekte einer Anwendung (Programme, Bildschirme etc.) verwalten Sie in einer Bibliothek.
- Alle Datenobjekte der Anwendung werden in einer separaten Bibliothek verwaltet. Eventuell werden Sie sich zu einem späteren Zeitpunkt entscheiden, für Ihre Dateien weitere Bibliotheken einzusetzen. Achten Sie deshalb darauf, dass Ihre Programme und Querys die Daten nicht mit dem Bibliotheksnamen adressieren, sondern über die Suchliste zugreifen. So erhalten Sie sich die nötige Flexibilität.
- Alle zusätzlichen Objekte (selbst erstellte Querys, eigene Befehle, geänderte Betriebssystemfunktionen usw.) sollten ebenfalls in separaten Bibliotheken gespeichert werden. Achten Sie auch hier auf die unqualifizierte Adressierung. Außerdem müssen Sie dafür sorgen, dass die erstellten Objekte nach der Testphase in offizielle Produktionsbibliotheken gespeichert werden. Nur so behalten Sie den Überblick über Ihre produktiven Objekte.

Kommen wir jetzt noch einmal auf die aktuelle Bibliothek des SIGNON-Bildschirms zurück. Die aktuelle Bibliothek (Current Library) ist die private Bibliothek eines Benutzers. Ein Benutzer, der keine eigenen Objekte erstellt, benötigt keine eigene Bibliothek. Nur Systemadministratoren, Programmierer und manchmal auch einige Anwender entwickeln und testen Objekte zunächst in privaten Bibliotheken. Aber achten Sie darauf, dass die erstellten

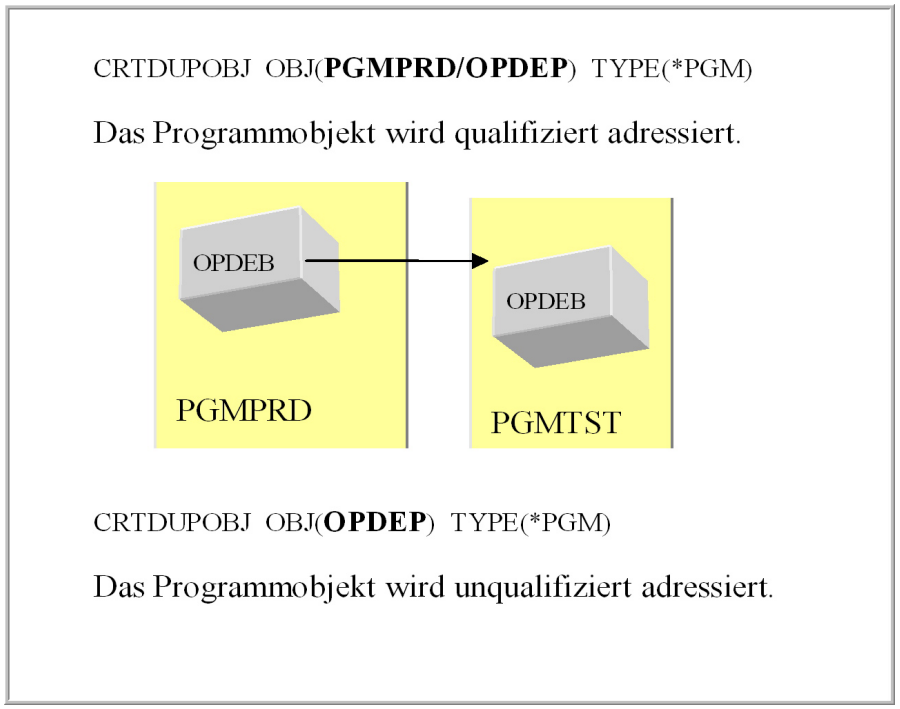
2.4.2.2
Seite 2

Objekte nach Abschluss der Testphase in die allgemeinen Bibliotheken übertragen werden. Denn sonst haben Sie bald eine Vielzahl von Bibliotheken, deren Inhalte Sie nicht mehr kennen und schlimmer noch: Objekte werden in Ihrem System redundant gespeichert und Sie wissen nicht mehr, welches der Objekte aktuell ist. Wenn Sie Ihre Bibliotheken deutlich strukturieren, verhindern Sie nicht nur ein kompliziertes Datensicherungs- und Berechtigungskonzept, sondern Sie ersparen sich auch viel Zeit und Arbeit bei der Pflege Ihres Systems.

Objektadressierung

Sie haben bereits etwas von der Bibliothekssuchliste gehört. Es stellt sich jetzt die Frage, was darunter zu verstehen ist. Stellen Sie sich folgendes vor:

Es gibt zwei Bibliotheken: PGMTST und PGMPRD. In der Bibliothek PGMTST werden Programme entwickelt und getestet. Die Bibliothek PGMPRD enthält alle freigegebenen produktiven Programme. In der Bibliothek PGMPRD befindet sich das Programm OPDEB. Einer der Programmierer kopiert das Programm zu Testzwecken in die Bibliothek PGMTST.

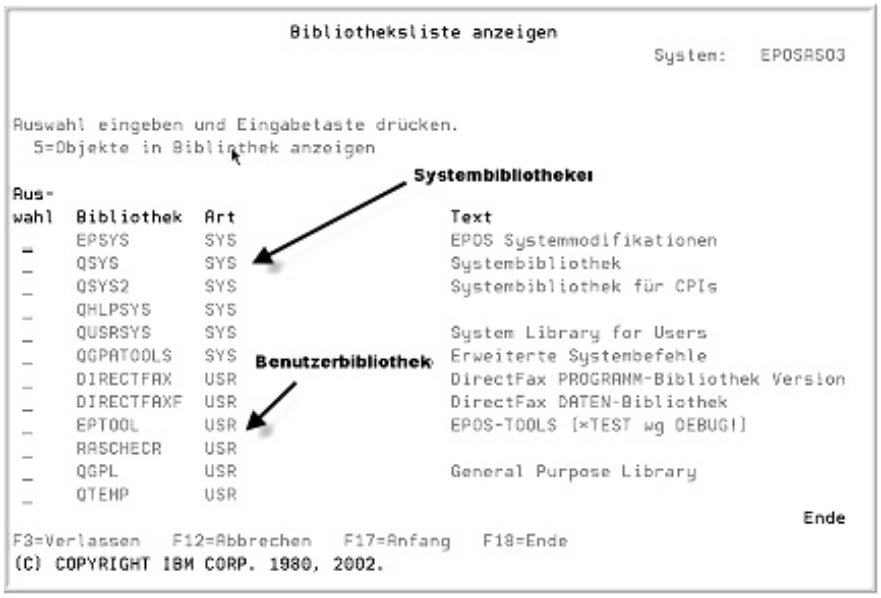


Qualifizierte und unqualifizierte Adressierung

Der Mitarbeiter hat zwei Möglichkeiten, um auf das Objekt zuzugreifen: mit Angabe des Bibliotheksnamens (qualifizierte Adressierung) oder ohne Angabe des Bibliotheksnamens (unqualifizierte Adressierung). Wenn Sie das Objekt ohne Bibliotheksnamen ansprechen, wird für die Suche des Objekts die Bibliotheksliste verwendet. Als Synonym für die Bibliotheksliste nutzt das System den Parameterwert *LIBL.

Die Bibliotheksliste wird jedem Benutzer automatisch vom System zugeordnet. Sie würden das Fehlen der Suchliste in Ihrer täglichen Arbeit schnell bemerken, denn Sie müssten jeden Betriebssystembefehl qualifizieren: QSYS/SNDMSG oder QSYS/SIGNOFF.

2.4.2.3
Seite 2



Bibliotheksliste eines Anwenders

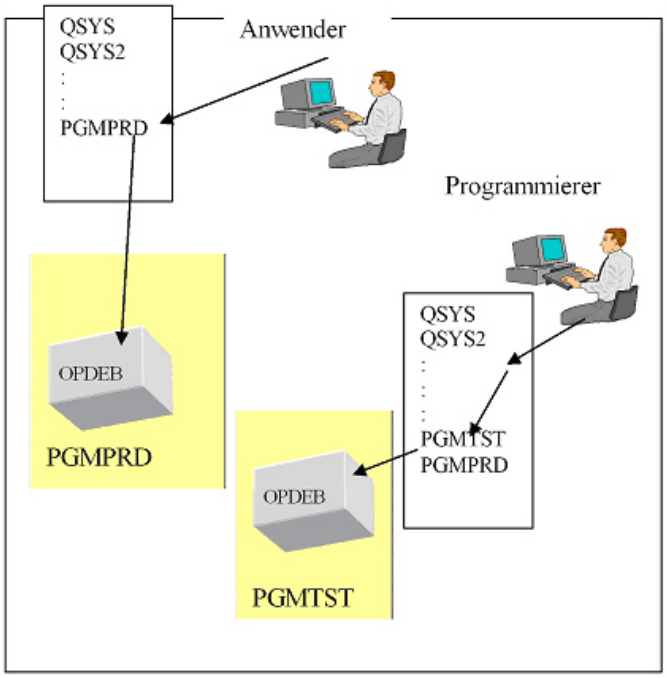
Die Suchliste beinhaltet sowohl System- als auch Benutzerbibliotheken. Welche Bibliotheken dem Job zugeordnet werden, wird durch die Systemwerte QSYSLIBL und QUSRLIBL festgelegt. Im Systemteil der Suchliste sind die Bibliotheken QSYS und QSYS2 eingetragen. Beide Bibliotheken enthalten das Betriebssystem. In der Bibliothek QHLPSYS sind umfangreiche Hilfetexte gespeichert. In der Bibliothek QUSRSYS werden u. a. Sicherheitsdaten gespeichert. Der Benutzerteil der Suchliste wird individuell aufgebaut und sollte Ihre Produktionsbibliotheken mit den freigegebenen Programm- und Query-Objekten sowie den Daten enthalten. Zusätzlich enthält der Benutzerteil zwei wichtige Systembibliotheken: QGPL und QTEMP. QGPL ist eine Bibliothek, die aufgrund der Voreinstellungen häufig als Zielbibliothek verwendet wird, wenn der Anwender vergisst, eine eigene Zielbibliothek zu benennen. Daher befinden sich in der Bibliothek QGPL häufig „verloren gegangene“ Objekte. QTEMP ist eine temporäre Bibliothek – d.h.: Alle Objekte, die Sie in dieser Bibliothek speichern, werden gelöscht, wenn Sie sich vom System abmelden. Jeder Benutzer hat eine eigene temporäre Bibliothek.

Obwohl die Suchliste normalerweise automatisch dem jeweiligen Benutzer zugeordnet wird, stehen verschieden Befehle zur Verfügung, die es gestatten, die Suchliste temporär – für die Dauer der Anmeldung – zu verändern.

DSPLIBL	Bibliotheksliste anzeigen
EDTLIBL	Bibliotheksliste editieren
ADDLIBLE	Bibliothekslisteneintrag hinzufügen
RMVLIBLE	Bibliothekslisteneintrag entfernen
CHGCURLIB	Aktuelle Bibliothek ändern
CHGUSRLIBL	Benutzerteil der Suchliste ändern

Was passiert jetzt in unserem Beispiel?

Die Anwender können mit den alten Objekten ungestört arbeiten, da Sie die Bibliothek PGMPRD in der Suchliste haben. Der Programmierer fügt die Bibliothek PGMTST in seine Suchliste vor die Bibliothek PGMPRD ein. Er testet und modifiziert das Programmobjekt, ohne den produktiven Betrieb zu stören, da das System die Suchliste sequentiell abarbeitet. Nach der Testphase wird das Objekt in die Bibliothek PGMPRD zurückkopiert – und die Anwender arbeiten mit den Programmmodifikationen.



Testen und Modifizieren eines Programmobjektes

Nutzen Sie dieses Wissen auch, um z.B. das Betriebssystem Ihren Anforderungen entsprechend anzupassen. Alle Systembefehle sind Objekte der Art *CMD. Sie könnten z.B. häufig genutzte Betriebsystemfunktionen umbenennen und dadurch den Aufruf der Funktion erleichtern. Um beispielsweise die Suchliste mit dem Befehl „DL“ anzuzeigen, müssen Sie nur den Befehl DSPLIBL (display Librarylist) kopieren (CRTDUPOBJ) und ihn dabei umbenennen. Ab sofort wird die Bibliothekssuchliste angezeigt, wenn Sie Ihren neuen Befehl „DL“ ausführen. Stellen Sie die Objektkopien aber bitte nicht in die Bibliothek QSYS, denn dann vermischt sich das Betriebssystem mit Ihren Modifikationen und Sie können bald nicht mehr Original und Kopie voneinander unterscheiden. Hinzu kommt, dass Ihre Modifikationen sich in einer separaten Bibliothek nicht störend bei einem Release-Wechsel auswirken können, da die Bibliothek QSYS im Originalzustand erhalten bleibt. Erstellen Sie sich also eine neue Bibliothek, die ausschließlich Ihre Modifikationen beinhaltet. Diese Bibliothek hinterlegen Sie im Systemwert QSYSLIBL vor die Bibliothek QSYS, denn es könnte auch mal erforderlich sein, den Originalna-

2.4.2.3
Seite 4

men beizubehalten und lediglich voreingestellte Parameterwerte zu verändern. Parameterwerte der Betriebssystemfunktionen überschreiben Sie mit dem Befehl CHGCMDDFT. In diesem Fall muss Ihre Funktion aber vor dem Original stehen, ansonsten werden Ihre Modifikationen nicht wirksam oder Sie müssten ständig qualifiziert zugreifen.

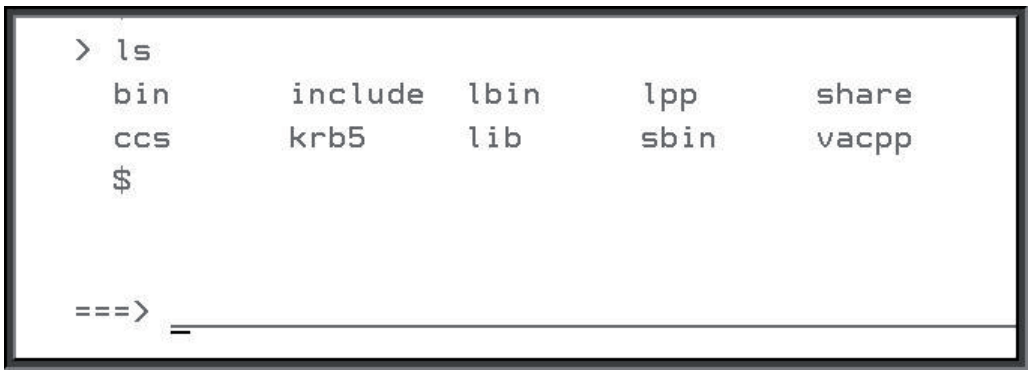
Integrated File System (IFS)

Wenn Sie die Kapitel 2.4 ff gelesen haben, wissen Sie, dass alle i5/OS-Objekte in Bibliotheken organisiert werden. Dabei könnte leicht der Eindruck entstehen, dass dieses Bibliothekskonzept das alleinige Ordnungssystem auf einem i5-Server ist.

Doch auf einem i5-Server müssen nicht nur klassische i5/OS-Daten gespeichert werden; auch Videos, Bilddateien u. v. m. sollten auf einem Datenbankserver abgelegt werden. Auch muss es möglich sein, i5/OS-Daten anderen Systemen zur Verfügung zu stellen. Andere Systeme haben aber andere Dateisysteme – und jedes dieser Systeme hat eine individuelle logische Struktur und vor allem eigene Regeln, wenn es um den Datenzugriff geht. Aus diesem Grund unterstützt das Betriebssystem i5/OS bereits seit einigen Jahren unterschiedliche Dateisysteme, die alle unter dem Begriff Integrated File System (IFS) vereint wurden. Dabei unterstützt das IFS sowohl das Bibliothekskonzept eines i5-Servers, kennt aber auch die Eigenschaften eines Windows- oder Unix-Dateisystems. Nur dadurch ist es möglich, dass Sie beispielsweise auf dem i5-Server eine Unix-Shell öffnen können – ohne eine Unix- oder Linux-Partition einzurichten. Geben Sie hierzu einfach den folgenden Befehl auf einer i5/OS-Befehlszeile ein:

```
CALL QP2TERM
```

Daraufhin öffnet sich ein neues Fenster, das die Eingabe diverser Unix-Befehle erlaubt:

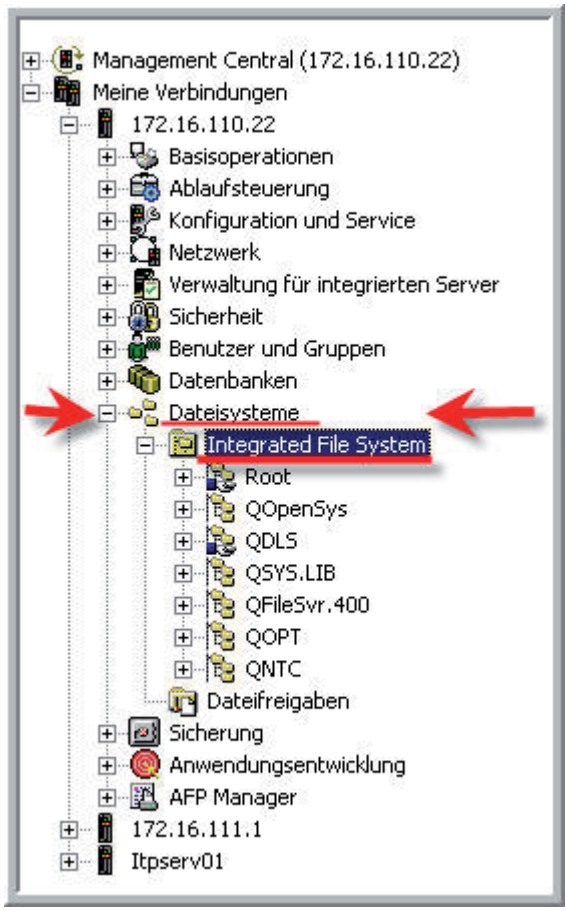


Unix-Shell

Das IFS sehen Sie im 5250-Umfeld immer nur partiell; auch die Administration dieser Dateisysteme ist in diesem 5250-Umfeld recht aufwendig.

2.4.2.4
Seite 2

IBM hat aber die Dateisysteme als eigenständigen Bereich in den iSeries Navigator integriert. Öffnen Sie im iSeries Navigator die Verbindung zu Ihrem Server und erweitern Sie anschließend den Eintrag: „Dateisysteme“.



iSeries Navigator – Dateisysteme

Der Bereich „Dateisysteme“ im iSeries Navigator dient der Verwaltung des „Integrated File Systems“(IFS) im i5/OS. Die gezeigte Struktur ist vergleichbar mit einem PC- oder Unix-System und sollte daher schnell zu erlernen sein. Durch eine hierarchische Struktur werden die Daten logisch angeordnet, so dass – eine vernünftige Struktur vorausgesetzt – auf sie optimal zugegriffen werden kann.

Wenn Sie – wie ich es bereits getan habe – den Eintrag „Integrated Files System“ erweitern, sehen Sie eine Liste aller verfügbaren Dateisysteme für die Sie berechtigt sind. Schauen wir uns zunächst die einzelnen Verzeichnisse im Überblick an:

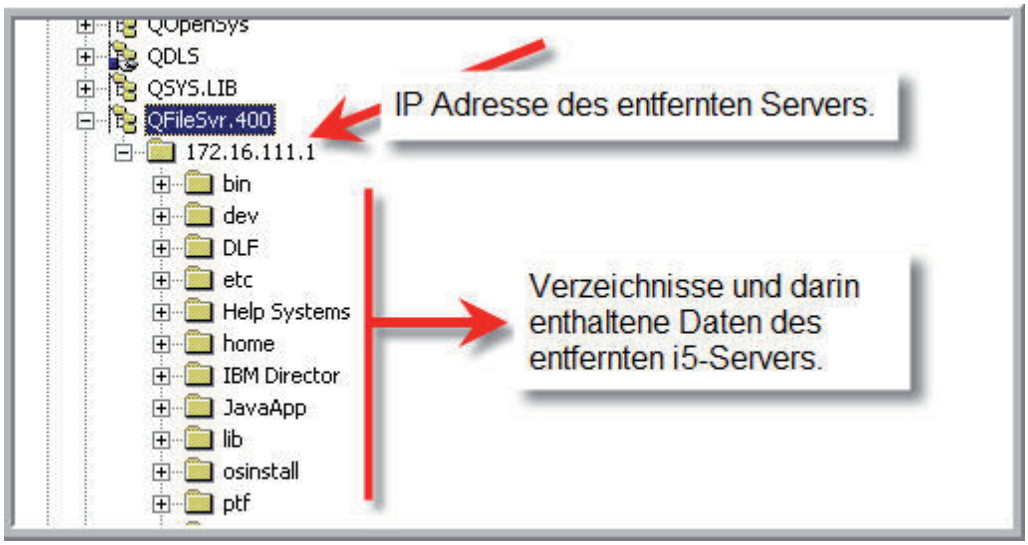
Verzeichnis „ROOT“
Root ist das sogenannte Windows- oder OS/2-kompatible Dateisystem. Wenn Sie hier Daten speichern, werden die Daten im ASCII-Format als „Streamfiles“ abgelegt. Die Namenskonventionen und Eigenschaften dieses Dateisystems entsprechen denen der Windows-Welt.

Verzeichnis „QOpenSys“
Dieses Dateisystem ist Unix-kompatibel. Daten, die hier gespeichert werden, sind ebenfalls im ASCII-Format als sogenannte Streamfiles vorhanden. Allerdings entsprechen die Namenskonventionen denen der Unix-Welt.

Verzeichnis „QDLS“
Das „Document Library System“(QDLS) beinhaltet heute vorwiegend PC-Dateien. Früher wurden in diesen Verzeichnissen auch Office-Vision-Dokumente gespeichert. Der Unterschied zum Dateisystem „ROOT“ besteht vor allem im verwendeten Zeichensatz: Daten in „QDLS“ werden nämlich im EBCDIC-Format als Streamfiles abgelegt. Die Namenskonventionen in diesem Bereich entsprechen denen der MS-DOS-Welt.

Verzeichnis „QSYS.LIB“
Auch das Bibliothekskonzept der i5 ist nichts anderes als ein Verzeichnissystem. Dieses klassische Ordnungssystem finden Sie im Eintrag „QSYS.LIB“. Daten, die hier gespeichert sind, werden im EBCDIC-Format als sogenannte Record-Files abgelegt. Die Namenskonventionen entsprechen denen der i5/OS-Welt.

Verzeichnis „QfileSvr.400“
„QfilsSrv ist ein Brückenverzeichnis. Wenn Sie unterhalb von QfileSvr.400 ein Verzeichnis erzeugen, das den Host-Namen oder die IP-Adresse eines entfernten iSeries-Servers trägt, gelangen Sie automatisch in das „Integrated File System“ des entfernten Systems.



Wirkungsweise des Verzeichnisses „QFileSvr.400“

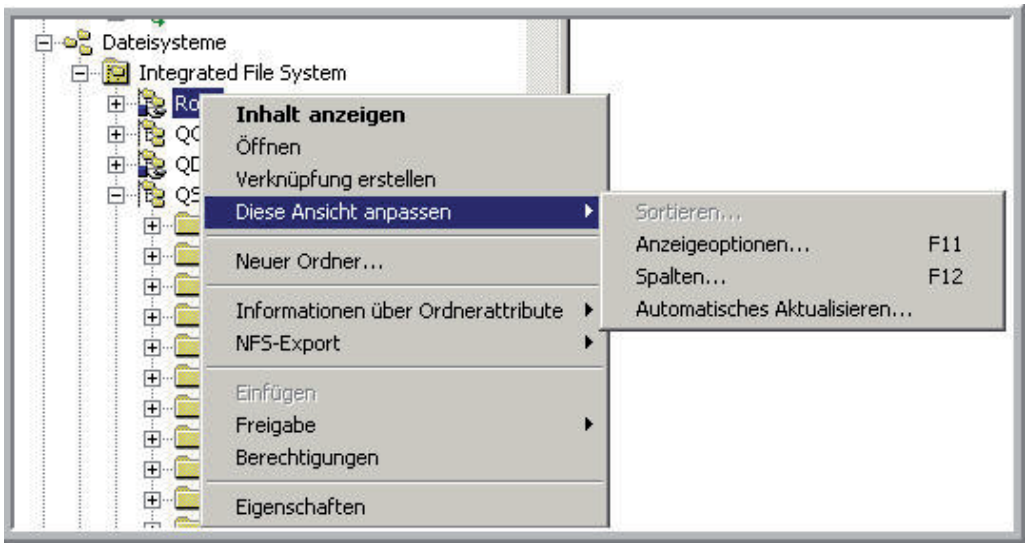
Verzeichnis „QOPT“
„QOPT“ erlaubt den Zugriff auf optische Speichermedien. So erhalten Sie Zugriff auf das CD-Rom-Laufwerk Ihres Servers.

Verzeichnis „QNTC“
Auch das Verzeichnis „QNTC“ ist ein Brückenverzeichnis zu anderen Systemen. Es ermöglicht den Zugriff auf andere Windows-Systeme. Wenn Sie unterhalb von „QNTC“ ein Verzeichnis mit dem Computernamen eines im Netzwerk befindlichen Windows-Systems erzeugen, so erhalten Sie hierüber einen Zugriff auf die freigegebenen Ressourcen dieses Systems. Voraussetzung dafür ist allerdings eine lauffähige Konfiguration des iSeries-Net Servers.

Nachdem wir nun die einzelnen Verzeichnisse im Überblick kennen, ist es an der Zeit, herauszufinden, welche Funktionen das IFS bereitstellt.

Grundfunktionen – „Integrated File System“

Ich werde sicherlich im Folgenden nicht jedes Kontextmenü und jede Funktion detailliert erläutern können. Dies würde einerseits den Rahmen dieses Buchs sprengen, andererseits würden sich viele Erklärungen wiederholen, da häufig die Kontextmenüs der verschiedenen Verzeichnissysteme sowie der in den Verzeichnissen gespeicherten Objekte identisch aufgebaut sind. Schauen wir uns zunächst stellvertretend das Kontextmenü des Dateisystems „ROOT“ an:

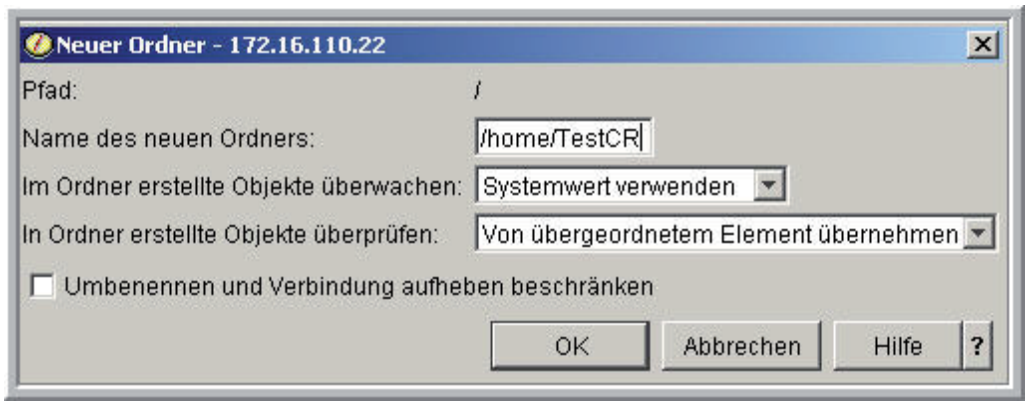


Kontextmenü des Dateisystems „ROOT“

Der Kontextmenüaufbau ist für alle Dateisysteme identisch. Wie in den anderen Kontextmenüs des iSeries Navigators finden wir zunächst die **Standardfunktionen: „Inhalte anzeigen“, „Verknüpfungen auf dem Desktop erstellen“** usw. Auch die Ansicht lässt sich wie in anderen Anzeigen individuell anpassen. Sie können Filter setzen („Anzeigeoptionen“), um das angezeigte Datenvolumen einzuschränken, aber auch die Spaltenreihenfolgen im Ausgabebildschirm individuell bestimmen.

2.4.2.4.1
Seite 2

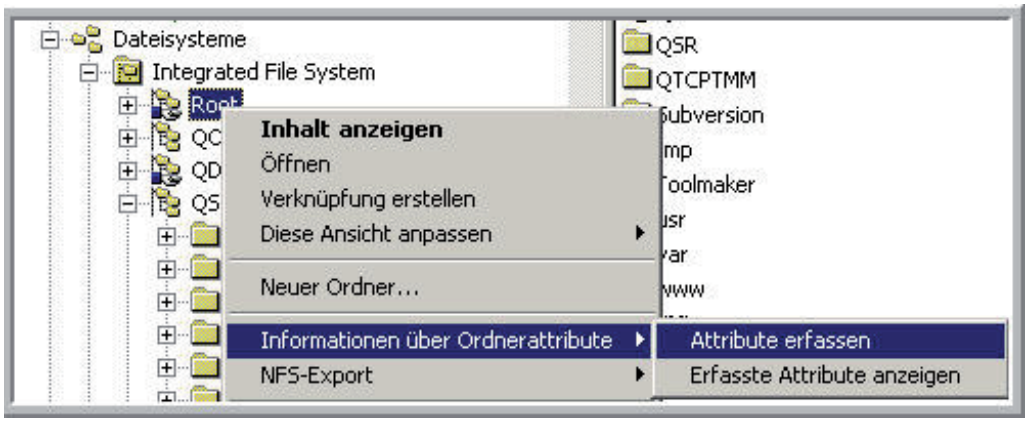
Auch der **Eintrag „Neuer Ordner“** spricht für sich. Wählen Sie diese Funktion, so öffnet sich folgender Dialog:



Neuer Ordner erstellen

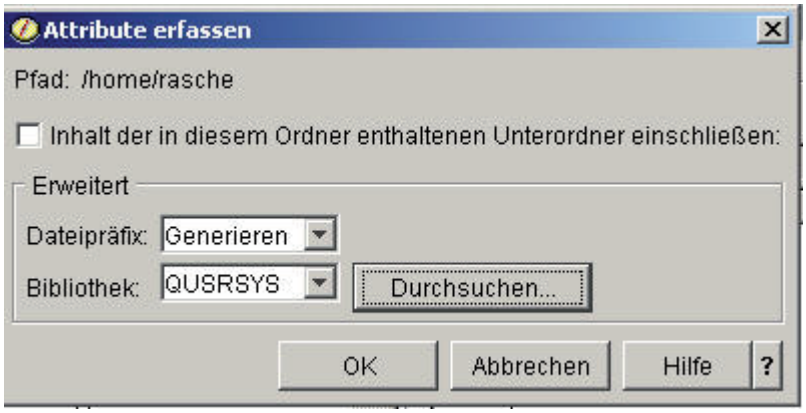
In das Eingabefeld tragen Sie den Namen des neuen Ordners ein. In diesem Fall wird der Ordner „TESTCR“ im „ROOT“-Verzeichnis unterhalb des Ordners „HOME“ erstellt. Beabsichtigen Sie – so wie ich es getan habe –, den Ordner unterhalb von „/Home“ zu erstellen, dann müssen Sie entweder im oben gezeigten Dialog den kompletten Pfad eintragen oder das Kontextmenü des entsprechenden Unterordners verwenden. Ob i5/OS einen Überwachungseintrag an das Systemprotokolljournal (Objekt QAUDJRN in der Bibliothek QSYS) sendet, wenn das Objekt verwendet oder geändert wird, bestimmt die Option „Im Ordner erstellte Objekte überwachen“. Weiterführende Informationen hierzu enthält das Kapitel 2.3.2.4. Die Option „In Ordner erstellte Objekte überprüfen“ gibt an, ob die im Ordner erstellten Objekte überprüft werden, wenn Exit-Programme entsprechend registriert sind. Diese Option kann nur für Dateien in den Dateisystemen „Root“ (/), „QOpenSys“ sowie in benutzerdefinierten Dateisystemen angegeben werden. Obwohl dieses Attribut für Objekte in i5/OS-Bibliotheken (Ordnertyp 1) definiert werden kann, werden nur Objekte in Ordnern vom Typ 2 überprüft.

Kommen wir wieder zurück zum ursprünglichen Kontextmenü:



„Informationen über Ordnerattribute“

Sie können über die im Ordner gespeicherten Objekte **„Attribute erfassen“** lassen. Wenn Sie die Attribute erfassen lassen, öffnet sich zunächst ein neuer Dialog:



„Attribute erfassen“

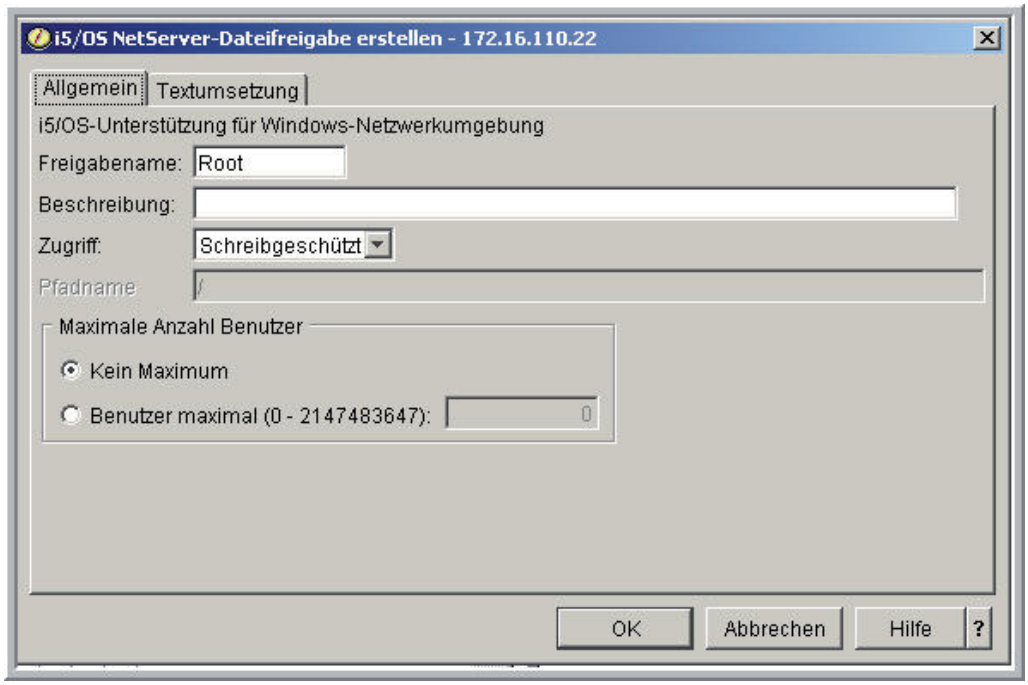
Im Dialog legen Sie ein Dateipräfix und die Bibliothek fest, um die Daten zu speichern. Anschließend erhalten Sie zwei Dateien, die die generierten Daten enthalten. Diese Daten enthalten Angaben über die Größe der gespeicherten Objekte, Datum der Erstellung, Datum der letzten Änderung u.v.m. Probieren Sie es einfach einmal aus. Die erfassten Daten lassen sich im Übrigen sehr einfach über SQL anzeigen und natürlich auch individuell auswerten.

Wenn Sie mit Unix-Servern arbeiten, können Sie das gewählte Dateisystem mit der **Funktion „NFS-Export“** in die Unix-Umgebung exportieren.

Die **Kontextfunktion „Einfügen“** ist nur verfügbar, wenn Sie zuvor ein Objekt in das Windows-Clipboard kopiert haben. Anschließend kann es mithilfe der Funktion „Einfügen“ in einen anderen Ordner des Dateisystems kopiert werden.

2.4.2.4.1
Seite 4

Mit der **Funktion „Freigabe“** sollten wir uns ein wenig ausführlicher beschäftigen. Hier arbeiten Sie mit Objekten und Funktionen des i5/OS Netservers. Der Netserver ist seit OS/400 V4R2 eine Komponente, die die NETBIOS-Schnittstelle im TCP/IP-Protokollstapel aktiviert und damit den i5-Server Microsoft-Netzwerk-kompatibel macht. Wählen Sie im Kontextmenü eines Verzeichnisses die Funktion „Freigabe“, dann öffnet sich ein entsprechender Dialog, der es Ihnen ermöglicht, den Ordner freizugeben.



Dateifreigaben erstellen

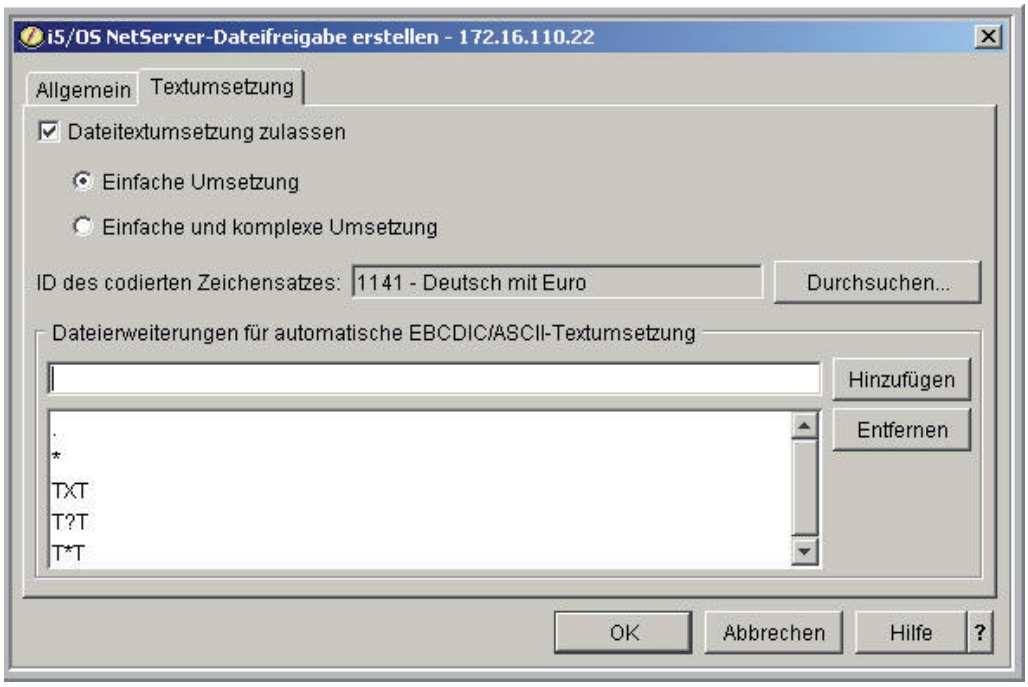
Als Erstes müssen Sie einen Freigabenamen wählen. Vorgeschlagen wird stets der Name der freizugebenden Ressource. Auch einen beschreibenden Text können Sie hinzufügen. Für den Zugriff haben Sie die Wahl zwischen:

– Lesezugriff:
  Nur lesender Zugriff ist über diese Freigabe möglich.

– Schreib-/Lesezugriff:
  Lesender und schreibender Zugriff ist über die Freigabe möglich.

Außerdem können Sie bestimmen, wie viele Benutzer die Ressource gleichzeitig verwenden dürfen.

Im Register „Textumsetzung“ wird die Übersetzung von EBCDIC-Inhalten nach ASCII gesteuert.



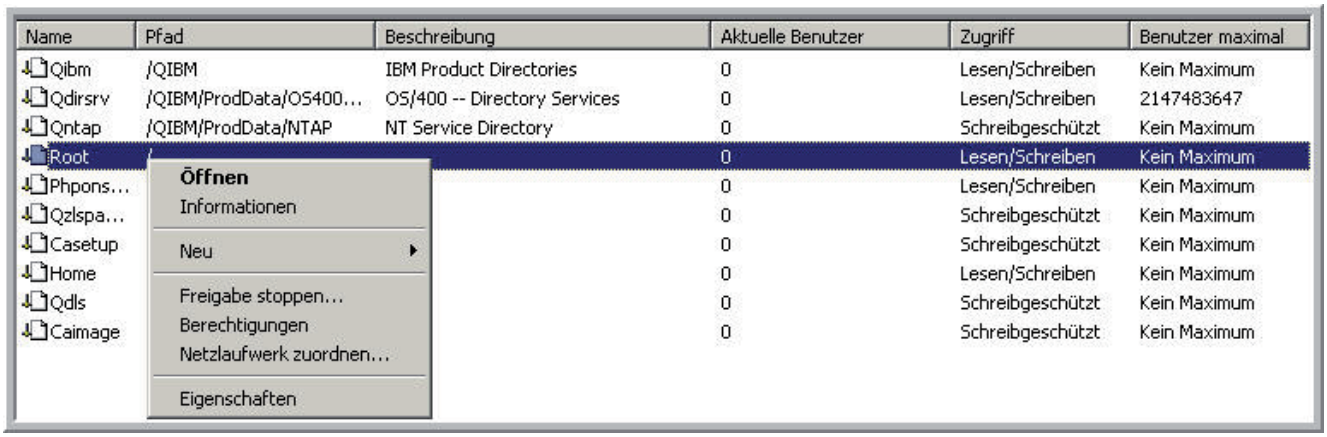
Dateifreigaben – Textumsetzung

Im Vordergrund steht hier die automatische Zeichenumsetzung. Zunächst müssen Sie aber die automatische Textumsetzung aktivieren, indem Sie die Option „Dateitextumsetzung zulassen“ markieren. Wählen Sie diese Option aus, um die Unterstützung für die automatische Textumsetzung für Dateien mit Erweiterungen zu aktivieren. Die betroffenen Dateien geben Sie anschließend im Feld „Dateierweiterungen für automatische EBCDIC-ASCII-Umsetzung“ an. Wenn Sie diese Option nicht zulassen, sind alle anderen Felder deaktiviert. Welche CCSID vom System zum Umsetzen der Objekte von EBCDIC in ASCII verwendet wird, können Sie individuell bestimmen, indem Sie auf die Schaltfläche „Durchsuchen…“ klicken. Abschließend definieren Sie noch, welche Objekte umgesetzt werden sollen. Hierzu erstellen Sie Schablonen für entsprechende Dateierweiterungen. Folgende Möglichkeiten stehen Ihnen zur Verfügung:

2.4.2.4.1
Seite 6

Zeichen	Bedeutung
*	Alle Dateien – unabhängig von deren Endung – werden umgesetzt.
.	Es werden nur Dateien umgesetzt, die keine Dateiendung haben.
Beliebige Zeichenfolge	Alle Dateien mit der hier genannten Dateiendung werden umgesetzt. Wenn Sie beispielsweise die Zeichenfolge „TXT“ wählen, dann würde die Datei „BSP.TXT“ von EBCDIC nach ASCII konvertiert werden.
Platzhalter: ? oder *	Innerhalb der Zeichenfolge können Sie Platzhalter verwenden, um einzelne oder mehrere Zeichen zu maskieren. Sie könnten beispielsweise die Endung „T?T“ für die Umsetzung erfassen. Eine Datei mit der Endung „TBT“, wird dann automatisch umgesetzt werden. Das Fragezeichen („?“) steht damit stellvertretend für ein Zeichen. Wenn mehrere Zeichen maskiert werden sollten, dann müssen Sie einfach anstelle des Fragezeichens das Sternzeichen („*“) verwenden.

Wenn alle Eingaben fertig sind, klicken Sie auf „OK“, um die Freigabe zu erstellen. Alternativ steht Ihnen aber auch der Eintrag „Dateifreigaben“ zur Verfügung. Wenn Sie den Eintrag „Dateifreigaben“ markieren, können Sie auf der rechten Bildschirmseite alle freigegebenen Objekte sehen.



Kontextmenü der Dateifreigaben

Zusätzlich sehen Sie, wie viele Benutzer derzeit mit dem freigegebenen Objekt arbeiten. Außerdem werden die Freigabeeigenschaften dargestellt. Wenn Sie das Kontextmenü einer Freigabe öffnen, dann stehen weitere Möglichkeiten der Bearbeitung zur Verfügung:

Öffnen: Öffnet das mit der Freigabe verbundene Verzeichnis direkt im Microsoft Explorer.

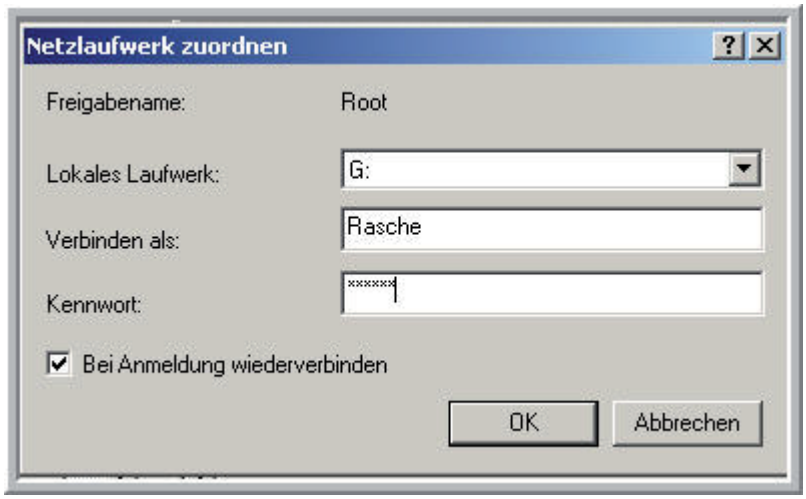
Informationen: Öffnet ebenfalls das mit der Freigabe verbundene Verzeichnis im Microsoft Explorer, jedoch aus der Sicht der Microsoft-Netzwerkumgebung heraus.

Neu: Erstellt eine neue Freigabe aufgrund der Eigenschaften der markierten Freigabe.

Freigabe stoppen: Beendet die Freigabe der Ressource.

Berechtigungen: Es können Berechtigungen für die freigegebene Ressource erteilt werden, um den Zugriff auf einzelne Personen oder Personengruppen zu beschränken. Detaillierte Informationen zum Thema „Berechtigungen“ enthält das Kapitel 2.4.4.

Netzwerklaufwerk zu ordnen: Ordnet einer freigegebenen Ressource ein Laufwerk auf Ihrem Desktop zu.



Netzwerklaufwerk zuordnen

Es werden hier nur Laufwerksbuchstaben zur Verfügung gestellt, die auf Ihrem Desktop noch frei sind. Standardmäßig wird der Windows-Benutzer an die iSeries zur Verbindungsanmeldung übergeben. Sind Windows- und iSeries-Benutzer nicht identisch, dann geben Sie hier einen iSeries-Benutzer und dessen Kennwort ein. Außerdem können Sie festlegen, dass bei jedem Neustart Ihres Windows-PC die Laufwerkszuordnung automatisch erfolgen soll.

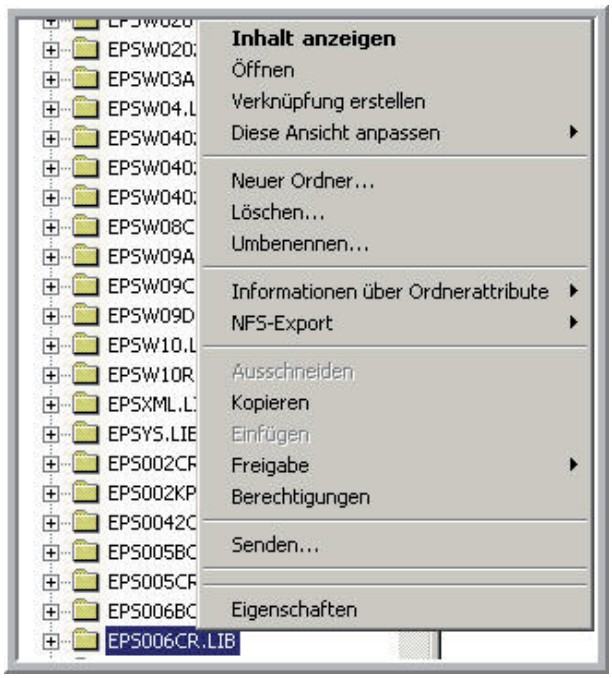
Eigenschaften: Wenn Sie nachträglich die Freigabeoptionen korrigieren wollen, müssen Sie den Dialog „Eigenschaften“ öffnen.

2.4.2.4.1

Seite 8

Mit Ordnern und Objekten arbeiten

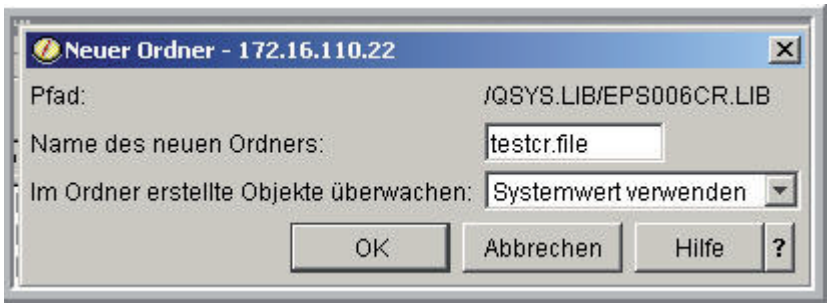
Wenn Sie sich das Kontextmenü eines Ordners ansehen, werden Sie feststellen, dass Sie viele der Funktionen bereits von den Dateiverzeichnissen her kennen:



Kontextmenü eines Ordners

Die Dateifunktionen werden lediglich um einige Funktionen ergänzt:

Neuer Ordner:
Sie können einen neuen Ordner innerhalb des Verzeichnisses erstellen. Mancher mag sich wundern, wenn er die obere Abbildung genau betrachtet. Geöffnet ist das Kontextmenü der Bibliothek „EPS006CR“. Eine Bibliothek kann doch gar keinen weiteren Ordner beinhalten? Wie soll dieser Menüpunkt dann gedeutet werden? Ganz einfach. Wenn Sie über das Kontextmenü einer Bibliothek einen Ordner erstellen, wird eine physische Datei erstellt. Sie müssen daher auch die Namenskonvention „.file“ verwenden – wie folgende Grafik zeigt:

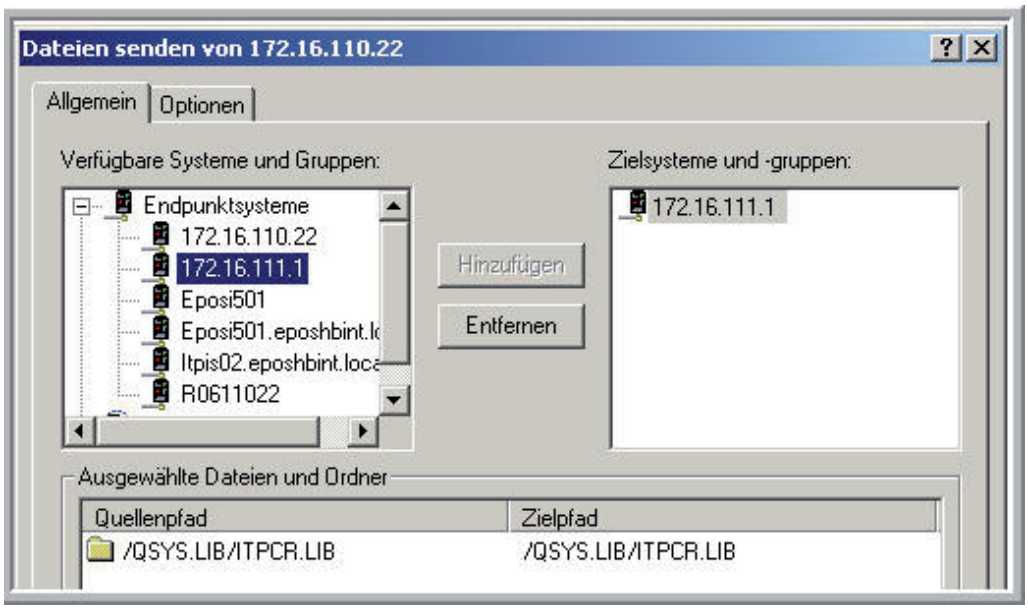


Neuer Ordner

2.4.2.4.2
Seite 2

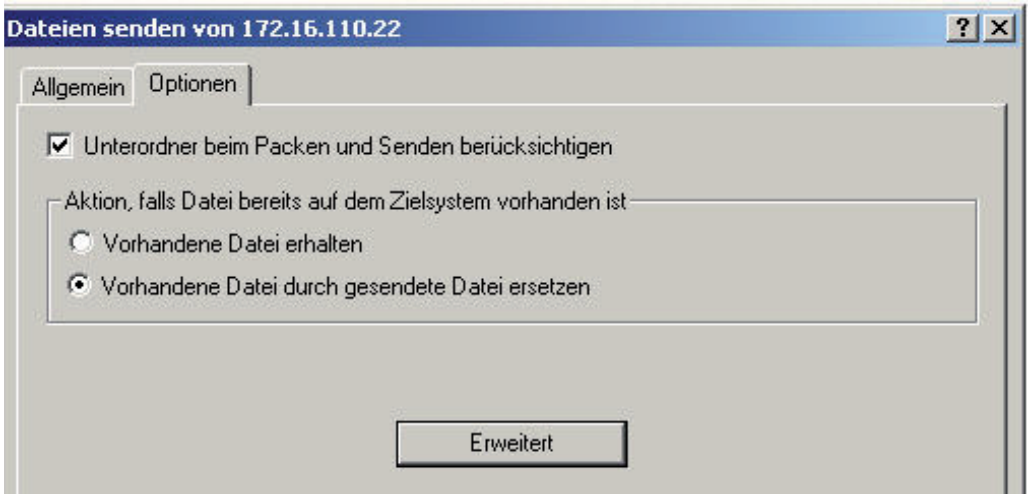
Die **Funktionen „Löschen", „Umbenennen", „Ausschneiden", „Kopieren und „Einfügen"** sollten Ihnen aus der Windows-Welt hinlänglich bekannt sein; sie bergen auch im IFS keine Besonderheiten.

Senden:
Die Funktion „Senden" ist eine „Management-Central-Funktion". Sie erlaubt das entsprechende Verzeichnis an ein oder mehrere entfernte System zu senden. Voraussetzung für das Versenden von Dateien oder Ordnern ist demnach eine Verbindung zum Management-Central-System. Wenn die Verbindung erfolgreich hergestellt werden konnte, erscheint folgender Dialog:



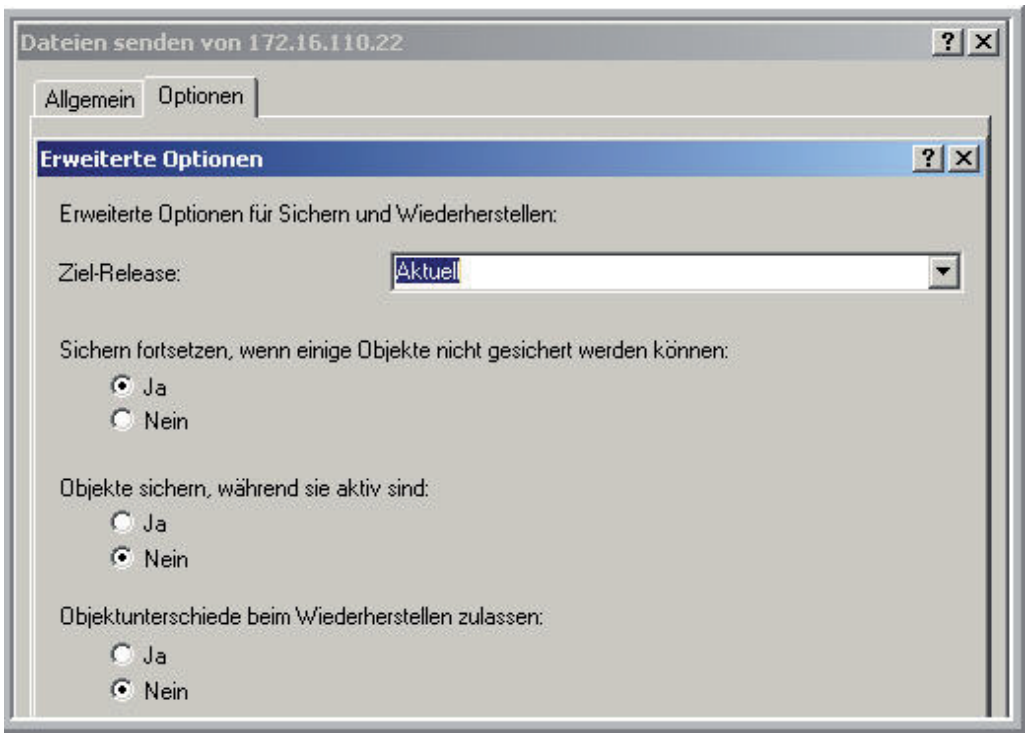
„Dateien senden"

Sie müssen lediglich das oder die Zielsysteme in der Liste der Endpunktsysteme auswählen. Die zu sendenden Objekte stehen bereits im unteren Bildschirmbereich. Das Register „Optionen" bietet weitere Einstellungen:



Dateien senden – Optionen

Über „Optionen“ können Sie Unterordner in die Verteilung miteinbeziehen und entscheiden, was passieren soll, wenn die Daten bereits auf dem empfangenden System vorhanden sind. Zusätzlich können Sie noch den Button „Erweitert“ nutzen, der Ihnen weitere Optionen zur Verfügung stellt:



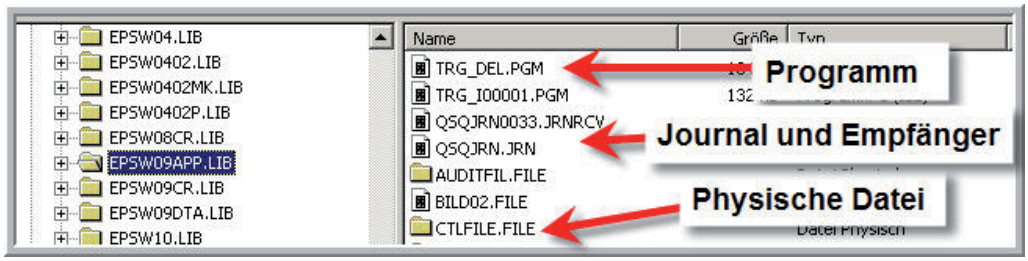
Dateien senden – Erweiterte Optionen

Sobald Sie alle Einstellungen vorgenommen haben, klicken Sie auf den Button „OK“, um die Übertragung zu starten. Den Job finden Sie anschließend im Management Central im Eintrag „Task-Aktivität“ → „Pakete und Produkte“. Hier können Sie auch prüfen, ob die Übertragung erfolgreich verlaufen ist.

Auf Objektebene des iSeries Navigator sind sogar „Drag-and-Drop“-Funktionen aktiviert, um Dateien zu verschieben. Wenn Sie eine Datei markieren und anschließend bei gedrückter Maustaste in einen anderen Ordner verschieben, werden Sie feststellen, dass sich der Mauszeiger entsprechend verändert. Haben Sie den Zielordner erreicht, wird das Objekt in den neuen Zielordner kopiert. Die „Drag-and-Drop“-Funktion ist auch über Systemgrenzen hinweg einsetzbar. Sollten Sie Verbindungen zu zwei oder mehr Systemen unterhalten, dann expandieren Sie einfach das Dateisystem auf dem zweiten System und ziehen das zu versetzende Objekt mit der Maus auf das gewünschte Verzeichnis des zweiten Systems.

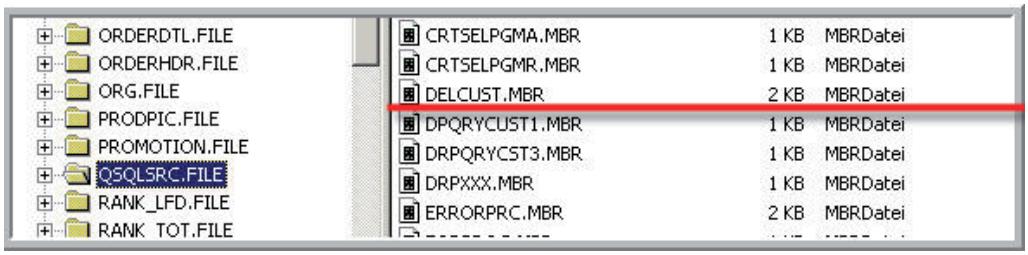
2.4.2.4.2
Seite 4

Auch mit Objekten in den i5/OS-Bibliotheken können Sie so verfahren. Hierfür müssen Sie das Dateisystem „QSYS.LIB“ erweitern. Es stellt die Bibliotheken des i5-Servers dar. Natürlich gelten hier die Namenskonventionen der eigentlichen i5-Umgebung: 10-stellige Namen, ausschließlich Großschreibung usw. Etwas gewöhnungsbedürftig ist die Darstellung der einzelnen Objektarten, die als Namenserweiterungen abgebildet werden.



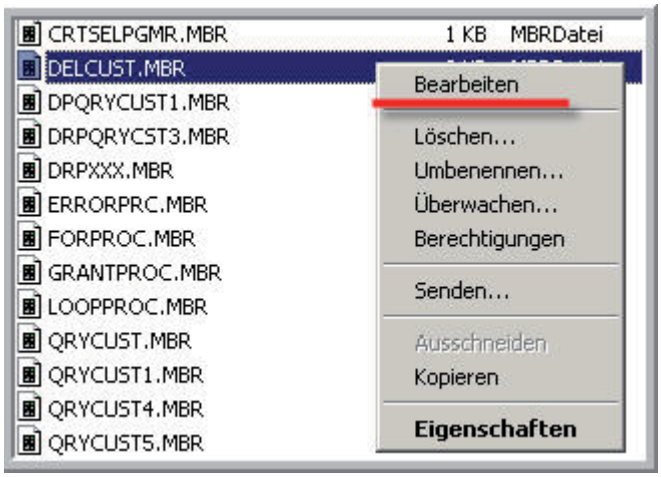
Objekte der Bibliothek „EPSW09APP.LIB“

Die Bibliothek „EPSW09APP“ heißt im IFS „EPSW09APP.LIB“. In der Bibliothek befindet sich das Programm „TRG_DEL“, das im IFS „TRG_DEL.PGM“ heißt. Aus der physischen Datei „CTLFILE“ wird „CTLFILE.FILE“ usw. Innerhalb der Bibliothek „EPSW09APP.LIB“ existiert auch die Quellendatei „QSQLSRC“, in der sich verschiedene SQL-Skriptdateien als „Member“ befinden. Mit diesen Membern will ich jetzt arbeiten:



Mit Membern einer Quellendatei arbeiten

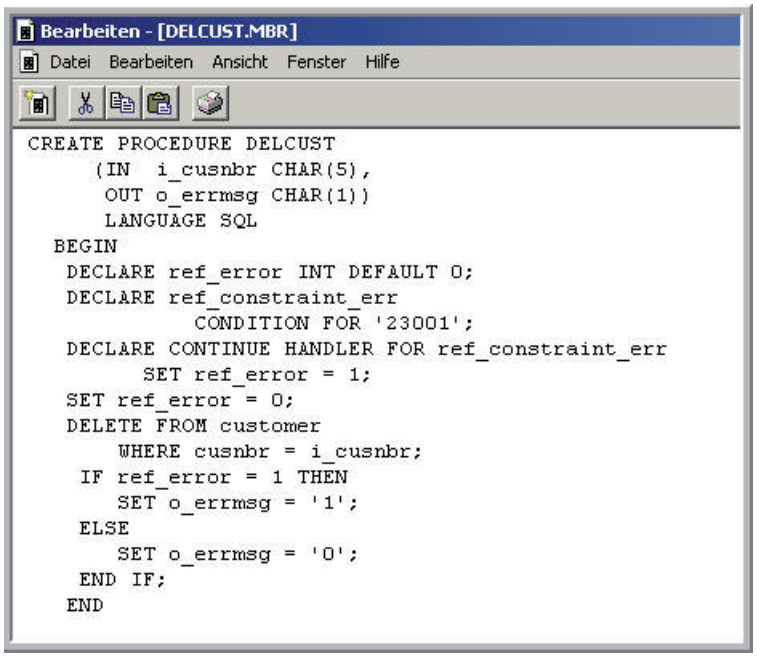
Sie können in der Abbildung sehen, dass es im Verzeichnis „QSQLSRC.FILE“ das Objekt „DELCUST.MBR“ gibt. Hierbei handelt es sich um eine Teildatei, die ich öffnen und editieren möchte. Für die Teildatei fordere ich zunächst das Kontextmenü an.



Kontextmenü der Teildatei

Die einzelnen Menüpunkte sind in den vorherigen Abschnitten hinlänglich erklärt, so dass ich nur auf die Auswahl „Bearbeiten“ eingehen werde.

Wenn ich diese Auswahl treffe, dann öffnet sich der Editor des iSeries Navigator, mit dem Quellentexte bearbeitet werden können.

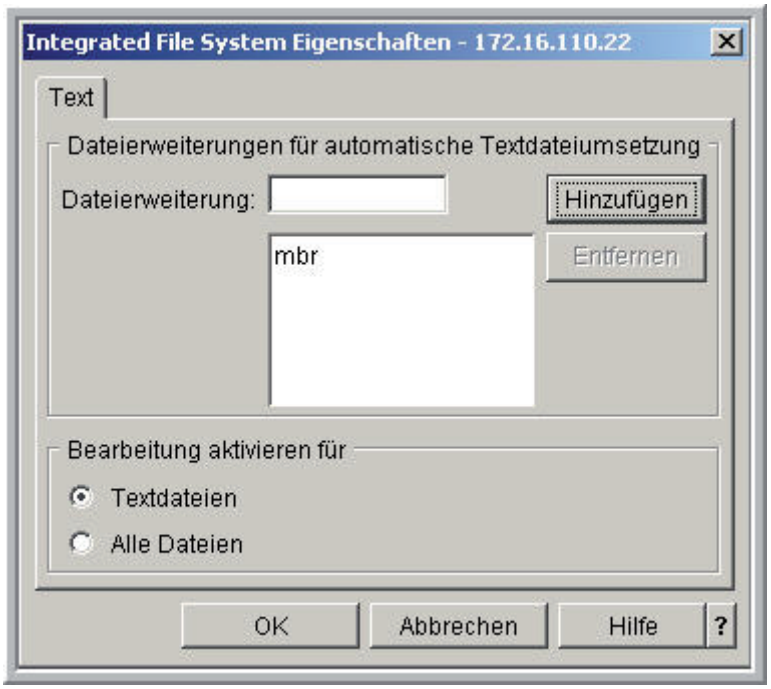


iSeries Navigator-Editor

2.4.2.4.2
Seite 6

Diesen Editor dürfen sie natürlich nicht mit SEU verwechseln. Syntaxprüfungen oder ähnliches sind diesem Editor vollkommen unbekannt. Sollten Sie im Kontextmenü die Auswahl „Bearbeiten“ deaktiviert finden, so kann dies unterschiedliche Ursachen haben:

1. Vielleicht sind Sie nicht berechtigt, mit dem Member zu arbeiten. In diesem Fall müssen zunächst die entsprechenden Rechte vergeben werden.
2. In den Eigenschaften des Integrated File System ist die Namenserweiterung für die Bearbeitung der Textfiles nicht angegeben. Öffnen Sie zur Überprüfung die Eigenschaften des Verzeichnisses „Integrated File System“. Daraufhin sollte der folgende Dialog erscheinen:



Eigenschaften des IFS

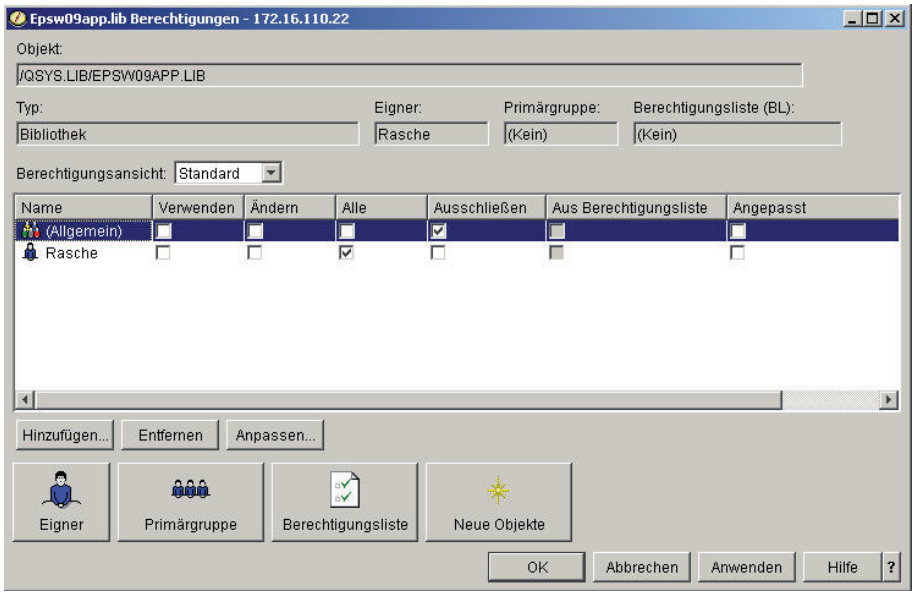
Dieser Dialog wird benötigt, um die Bearbeitung von Text- und Datendateien mit dem Editor des iSeries Navigator im Integrated File System übergreifend zu steuern. Sie sehen in der Abbildung oben, dass Textdateien mit der Namenserweiterung „.mbr“ direkt verarbeitet werden können. Wenn das Eigenschaftenfenster nicht die entsprechende Dateierweiterung enthält, können Sie die Teildatei auch nicht bearbeiten.

Wenn keiner der genannten Gründe zutrifft und dennoch die Funktion „Bearbeiten“ deaktiviert ist, greifen Sie eventuell auf eine Textdatei im Streamfile-Format zu und haben in den Eigenschaften für das Integrated File System die Option „Textdateien“ anstatt „Alle Dateien“ aktiviert. In diesem Fall müssen Sie die Eigenschaften des IFS entsprechend ändern.

Zum Schluss muss noch eine Funktion erwähnt werden, die bislang in allen Kontextmenüs vorhanden war – über die wir aber bisher nicht gesprochen haben.

Funktion „Berechtigungen“

Um im „Integrated File System“ mit Berechtigungen zu arbeiten, werden in den Kontextmenüs der diversen Hierarchiestufen entsprechende Auswahlen zur Verfügung gestellt. Wenn Sie diese Auswahl treffen, wird der nachfolgende Dialog geöffnet:



Dialog „Berechtigungen“

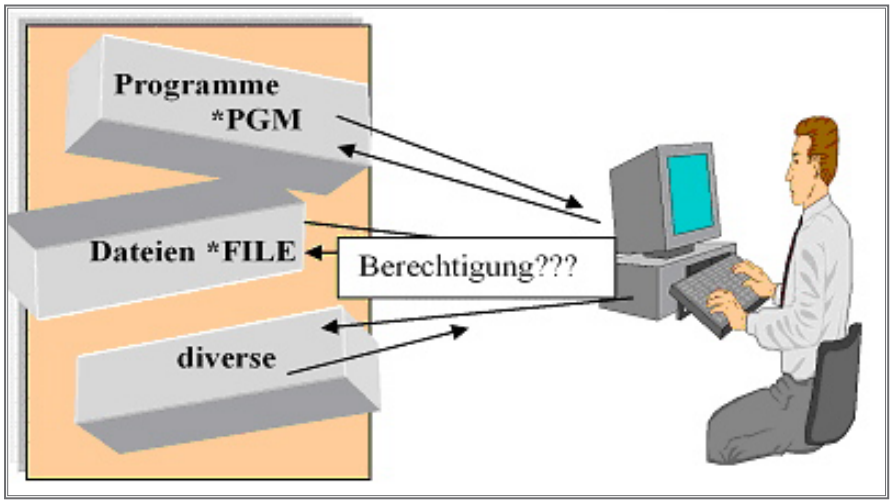
2.4.2.4.2
Seite 8

Dieser Dialog teilt sich in mehrere Bereiche auf. Im oberen Bereich sehen Sie das betroffene Objekt sowie den Objekttyp. Sie können erkennen, wer Eigentümer des Objekts ist und ob das Objekt über eine Berechtigungsliste geschützt wird. Der mittlere Bereich dient der Arbeit mit Benutzern und deren Rechte am Objekt. Hier werden Lese-, Schreib und Ausführungsrechte sowie sogenannte Objektverwaltungsrechte gewährt bzw. verweigert.

Über die Schaltfläche „Hinzufügen…“ werden neue Benutzer, denen Objektzugriffsrechte gewährt werden sollen, interaktiv aus der Benutzerverwaltung übernommen. Mit dem Button „Entfernen“ löscht man dagegen einen Benutzer aus der Liste der berechtigten Anwender. Mit dem Button „Anpassen…“ können Sie mittels eines Detaildialogs die einzelnen Rechte verändern.

Im unteren Bereich des Fensters finden Sie weitere Schaltflächen, mit denen Sie die Eignerschaft eines Objekts ändern. Auch hier greifen Sie wieder direkt auf die Daten der Benutzerverwaltung zu. Mit der Schaltfläche Berechtigungsliste wählen Sie aus den im System gespeicherten Berechtigungslisten diejenige aus, durch die Ihr Objekt geschützt werden soll. Falls Ihnen der Begriff der „Berechtigungsliste“ fremd sein sollte und Sie sich bislang noch nicht mit Berechtigungen auf einem i5-Server auseinandersetzen mussten, finden Sie weiterführende Informationen im Kapitel 2.4.3.

Ressourcenschutz



Wer darf was?

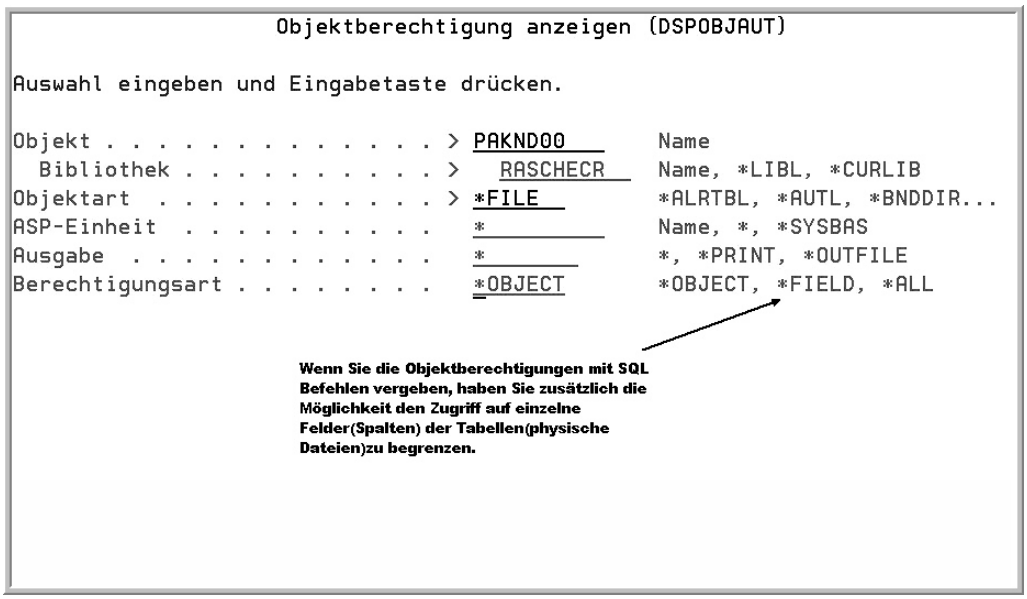
Wir wissen jetzt, wie Objekte organisiert werden, aber wie steuern wir die Zugriffsberechtigungen auf die einzelnen Objekte?

Für jedes iSeries-Objekt gibt es eine eigene Liste, wer was mit dem Objekt tun kann. Das nennt man Ressourcenschutz. Sicherheitsberechtigungen auf Objektebene unterscheiden sich von Berechtigungen auf Benutzerebene. Benutzerberechtigungen benennen für jeden Benutzer Vollmachten auf breiter Ebene, wohingegen Objekte über eine Liste von Benutzern verfügen, die mit dem Objekt arbeiten dürfen. Zunächst ist relevant, wer das Eigentumsrecht an dem Objekt besitzt. Es gibt eine Vielzahl von Benutzern und nur einer kann das Eigentumsrecht besitzen. Anderen Benutzern muss der Zugriff über spezifische Berechtigungen gewährt werden.

2.4.3
Seite 2

Objektberechtigungen verwalten

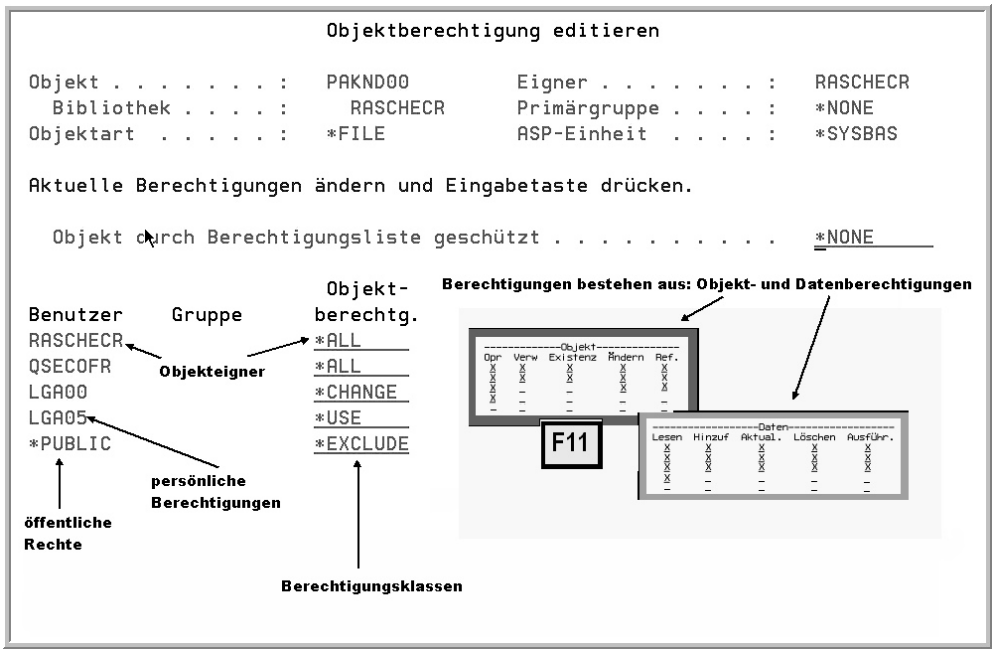
Ob Sie Objekte bearbeiten können, ist somit von Ihren spezifischen Objektberechtigungen abhängig.



Objektberechtigung anzeigen

Die Objektberechtigungen werden in persönliche Berechtigungen und öffentliche Berechtigungen unterteilt. Persönliche Berechtigungen sind Berechtigungen, die an bestimmte Benutzer vergeben werden. Allgemeine Berechtigungen gelten für alle nicht explizit genannten Benutzer und werden mit *PUBLIC gekennzeichnet. Zusätzlich wird zwischen Objektberechtigungen, d.h. Berechtigungen, die das Objekt selbst betreffen und den Datenberechtigungen differenziert. Die Datenberechtigungen beziehen sich auf den Inhalt des Objektes.

2.4.3.1
Seite 2



Berechtigungsklassen für Objekte

Die iSeries verwaltet die Objektberechtigungen in Objektberechtigungsklassen. Man unterscheidet die folgenden Berechtigungsklassen:

***ALL**

Der Benutzer kann alle Operationen am Objekt ausführen. Er kann die Objektexistenz steuern, die Berechtigungen für das Objekt bestimmen und das Objekt verändern. Er hat Zugriff auf alle Daten des Objektes. Es sei denn, die Daten werden durch eigene Objektberechtigungen geschützt.

***CHANGE**

Der Benutzer erhält alle Berechtigungen zum Ausführen des Objektes. Ausgenommen sind die Operationen, die über die Objektexistenzberechtigung entscheiden. Die Berechtigung *CHANGE umfasst die Änderungsberechtigungen, die Objektverwendungsberechtigung und alle Datenberechtigungen.

***USE**

Die Berechtigung *USE besagt, dass der Anwender die Objekte verwenden kann und Leseberechtigung sowie Ausführungsberechtigung für die Daten besitzt.

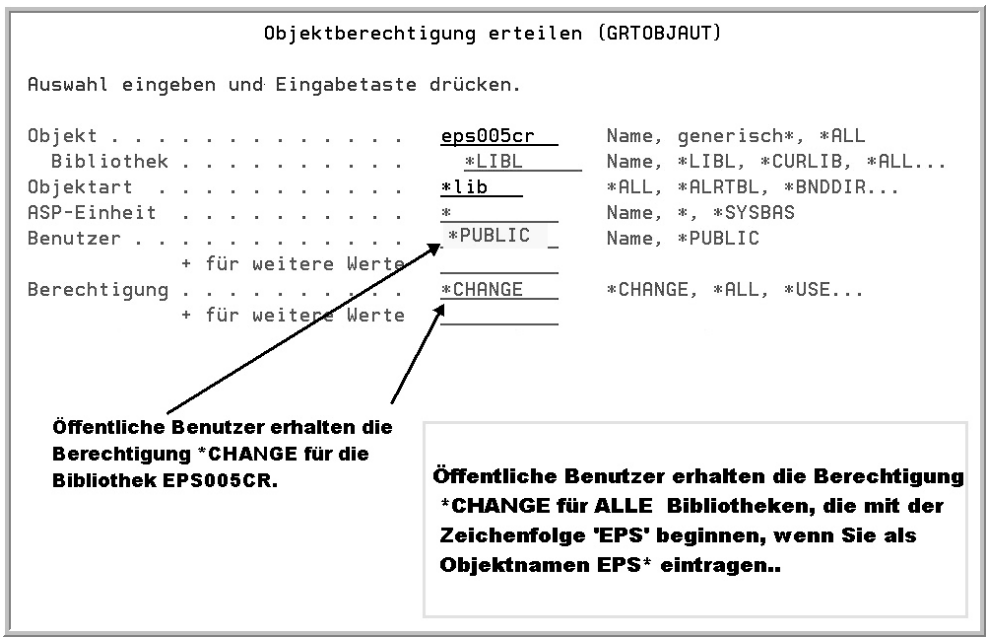
***EXCLUDE**

Der Benutzer kann nicht auf das Objekt zugreifen.

Das Erteilen, Widerrufen und Ändern von Objektberechtigungen geschieht am einzelnen Objekt. Für die Bearbeitung stehen Ihnen verschiedene Befehle zur Verfügung:

EDTOBJAUT	Objektberechtigungen editieren
RVKOBJAUT	Objektberechtigungen entziehen
GRTOBJAUT	Objektberechtigungen gewähren
DSPOBJAUT	Objektberechtigungen anzeigen

Da die Berechtigungsvergabe am einzelnen Objekt ziemlich zeitraubend sein kann, ermöglicht die iSeries die generische Identifikation von Objekten, so dass Sie mit einem Befehl alle Objekte in einer Bibliothek modifizieren können. Mit dieser Methode sparen Sie zwar Zeit, aber der Einsatz sollte sorgfältig geplant werden.



Erteilen einer Objektberechtigung

Viele Unternehmen verzichten bislang auf eine Ressourcen-Schutzstrategie, da der Schutz der Daten durch Menüs gewährleistet werden konnte. Heute sind die iSeries-Systeme aber mit anderen Plattformen vernetzt. Dieser Trend wird sich in Zukunft vermutlich weiter fortsetzen. Lokale und Remote-Benutzer erhalten dadurch neue Methoden, um auf die Daten ihrer iSeries zuzugreifen. Viele der neuen Funktionen, die IBM mit neuen Betriebssystemen ausliefert (wie z.B. FTP, ODBC, IFS oder Operation Navigator), bieten neue Wege für den Zugriff auf die Daten im Netzwerk. Das bedeutet aber nicht, dass Sie sich wieder in die „gute alte Zeit“ zurücksehnen sollten. Verteilte Computer- und Client-Server-Anwendungen haben sich etabliert und genießen zunehmende Popularität. Sie sollten daher den Einfluss der neuen Möglichkeiten bei ihrer Sicherheitsstrategie berücksichtigen.

2.4.3.1
Seite 4

Benutzerprofile verwalten

Wie Sie gesehen haben, müssen Sie aufgrund der eingestellten Sicherheitsstufe für jeden Benutzer ein Benutzerprofil hinterlegen. Das Benutzerprofil enthält:

- grundlegende Sicherheitsinformationen über die Person,
- spezielle Berechtigungen für die Person,
- spezifische Jobverarbeitungsinformationen für die Person.

Auf einer iSeries sind mehrere Benutzerprofile der IBM vorhanden.

QPGMR	Programmierer- und Stapelbenutzerprofil
QSECOFR	Sicherheitsbeauftragter
QSRV	Serviceprofil, das von der Person genutzt wird, die den Service für das System durchführt.
QSYSOPR	Systemadministrator
QUSER	allgemeiner Benutzer

Alle Profile – mit Ausnahme des Profils QSECOFR – haben kein Kennwort. Sie können zwar theoretisch dem Profil QPGMR ein Kennwort zuweisen und sich am System damit anmelden, aber dies wird nicht von IBM empfohlen. Die Profile sollen als Kopiervorlage dienen. Kopieren Sie also einfach die benötigten Profile. Dadurch übernehmen Sie viele Parameter und müssen nur firmenspezifische Ergänzungen vornehmen. Es gibt noch weitere von IBM gelieferte Benutzerprofile, die Sie nicht kopieren sollten, denn hierbei handelt es sich um Benutzerprofile, die das System für interne Aufgaben verwendet.

Grundwerk 10/2004

2.4.4
Seite 2

QAUTPROF	Allgemeines IBM-Berechtigungsprofil
QBRMS	Backup Recovery Media- (BRM-) Profil
QDBSHR	Profil für den Datenbankzugriff
QDFTOWN	Alle Objekte auf einer iSeries benötigen einen Eigentümer. Ist ein Benutzerprofil nicht mehr gültig, geht das Eigentumsrecht an seinen Objekten an das Profil QDFTOWN über.
QDOC	Dokumentenprofil
QLPAUTO	Lizenzierter Programm-Autoinstallationsbenutzer
QLPINSTALL	Lizenzierter Programminstallationsbenutzer
QMSF	Profil für Post-Server
QNETSPLF	Netzspool-Profil
QSPL	Spool-Benutzer
QSYS	Interner Systembenutzer
QTCP	TCP/IP-Benutzer

Diese Profile dienen nicht dazu, sich im System anzumelden. Versuchen Sie nicht, die Profile zu ändern oder zu löschen. Wenn Sie es dennoch versuchen, können interne Funktionen plötzlich nicht mehr funktionieren.

Einzelprofile

Jeder Mitarbeiter Ihres Unternehmens sollte ein eigenes Benutzerprofil haben. Melden sich mehrere Personen unter einem Profilnamen an, ist die Identifizierung des Benutzers in Ihrem System nicht mehr möglich. Sie ersparen sich also unnötige Arbeit bei der Unterstützung Ihrer Anwender in der täglichen Arbeit, wenn Sie Einzelprofile anlegen, und Sie haben es damit sehr viel einfacher, die Benutzerberechtigungen zu bestimmen. Benutzerprofile sind Objekte des Typs *USRPRF.

```
CRTUSRPRF          Benutzerprofil erstellen
DSPUSRPRF          Benutzerprofil anzeigen
CHGUSRPRF          Benutzerprofil ändern
DLTUSRPRF          Benutzerprofil löschen
RSTUSRPRF          Benutzerprofil zurücksichern
```

Wenn Sie neue Benutzerprofile anlegen wollen, müssen Sie neben dem Profilnamen eine Reihe weiterer Parameter festlegen:

```
                    Benutzerprofil erstellen (CRTUSRPRF)

Auswahl eingeben und Eingabetaste drücken.

Benutzerprofil . . . . . .   _________    Name
Benutzerkennwort . . . . . .  *USRPRF     Zeichenwert, *USRPRF, *NONE
Kennwort auf abgelaufen setzen *NO        *NO, *YES
Status . . . . . .            *ENABLED    *ENABLED, *DISABLED
Benutzerklasse . . . . . .    *USER       *USER, *SYSOPR, *PGMR...
Unterstützungsstufe  . . . . .  *SYSVAL   *SYSVAL, *BASIC, *INTERMED...
Aktuelle Bibliothek  . . . . .  *CRTDFT   Name, *CRTDFT
Aufzurufendes Startprogramm  . .  *NONE   Name, *NONE
  Bibliothek . . . . . .        _________  Name, *LIBL, *CURLIB
Anfangsmenü  . . . . . .      MAIN        Name, *SIGNOFF
  Bibliothek . . . . . .        *LIBL     Name, *LIBL, *CURLIB
Möglichkeiten einschränken . . .  *NO     *NO, *PARTIAL, *YES
Text 'Beschreibung'  . . . . . .  *BLANK
_____

                                                                    Ende
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F10=Zusätzl. Parameter
F12=Abbrechen   F13=Verwendung der Anzeige   F24=Weitere Tasten
```

Erstellung eines Benutzerprofils

Benutzerprofil (USRPRF)

Der Name des Profils kann bis zu 10 Zeichen lang sein. Er darf Buchstaben und Ziffern beinhalten, muss aber mit einem Buchstaben beginnen. Wenn Sie eigene Objekte erstellen, sollten Sie nicht den Buchstaben Q als Anfangsbuchstaben verwenden, denn alle IBM-Objekte beginnen mit einem Q. Wenn Sie sich an diese freiwillige Konvention halten, haben Sie es später einfacher,

2.4.4.1
Seite 2

Ihre Objekte von denen der IBM zu unterscheiden. Bevor Sie anfangen Benutzerprofile anzulegen, überlegen Sie sich spezifische Namenskonventionen. Verwenden Sie hierbei möglichst keine Abteilungsnamen, die geändert werden müssten, wenn der Mitarbeiter in eine andere Abteilung versetzt wird. Ihre Profilnamen könnten sich z.B. aus dem Nachnamen und den Initialen des Vornamens zusammensetzen. Sicherlich kosten diese vorbereitenden Überlegungen jetzt etwas Zeit, aber es lohnt sich, zunächst die Konventionen festzulegen. In einem produktiven System mit eventuell mehreren hundert Benutzern wird es fast unmöglich nachträglich Konventionen einzuführen oder zu überarbeiten.

Benutzerkennwort (PASSWORD)

Ab der Sicherheitsstufe 20 sind Kennwörter zwingend erforderlich. Der Aufbau der Kennwörter wird durch verschiedene Systemwerte beeinflusst:

QPWDLMAJC	Benachbarte Zeichen in Kennwörtern begrenzen
QPWDLMTCHR	Ungültige Kennwortzeichen
QPWDMAXLEN	Maximale Kennwortlänge
QPWDMINLEN	Minimale Kennwortlänge
QPWDPOSDIP	Alle Zeichen des Kennwortes müssen neu sein
QPWDRQDDGT	Mindestens eine Zahl muss enthalten sein
QPWDRQDDIF	Das neue Kennwort muss anders sein als das vorhergehende

Trotz dieser Möglichkeiten sollten Sie den Aufbau der Kennwörter nicht zu kompliziert gestalten, ansonsten könnte es passieren, dass Ihre Anwender sich die Passwörter auf der Schreibtischunterlage notieren, um den Anforderungen gerecht zu werden. Da kann man sicherlich nicht mehr von Geheimhaltung sprechen.

Kennwort auf „Abgelaufen“ setzen (PWDEXP)

Sie können mit diesem Parameter beim nächsten Anmeldeversuch des Benutzers eine Kennwortänderung erzwingen. Das ist praktisch, wenn Sie einen neuen Benutzer im System anlegen. Sie verwenden zunächst den Benutzerprofilnamen als Kennwort, setzen den Parameter PWDEXP auf *YES und wissen, dass bei der ersten Anmeldung das Kennwort geändert wird.

Status (STATUS)

Damit Sie ein Benutzerprofil nutzen können, muss das Profil den Status *ENABLED haben. Das System deaktiviert das Profil über diesen Parameter, wenn der Benutzer die Höchstzahl ungültiger Anmeldeversuche überschreitet. Die Höchstzahl unzulässiger Anmeldeversuche ist im Systemwert QMAXSIGN hinterlegt.

Sie können den Status des Profils auch nutzen, um die Profile ausgeschiedener Mitarbeiter zu bearbeiten: Setzen Sie das Benutzerprofil zunächst nur auf den Status *DISABLED. Dadurch schützen Sie das System sofort vor unberechtigten Zugriffen. Aber vergessen Sie nicht, in regelmäßigen Abständen Ihre deaktivierten Profile zu prüfen und zu löschen.

Klasse (USRCLS)

Die Benutzerklasse entscheidet über die sichtbaren Menüpunkte in den iSeries-Menüs. Es werden die folgenden Benutzerklassen unterschieden: *SECOFR, *SECADM, *SYSOPR, *PGMR, *USER. Allerdings bestimmt die Benutzerklasse nicht allein über die zur Verfügung gestellten Betriebssystemfunktionen, denn die Sonderberechtigungen stehen in direktem Zusammenhang mit der Benutzerklasse.

Unterstützungsstufe (ASTLVL)

Die Unterstützungsstufe bestimmt das Aussehen, der iSeries-Benutzersystemoberfläche.

Aktuelle Bibliothek (CURLIB)

Legt die persönliche Bibliothek des Benutzers fest.

Aufzurufendes Startprogramm (INLPGM)

Das hinterlegte Programm wird automatisch aufgerufen, wenn der Benutzer sich anmeldet. Es wird fast immer dann verwendet, wenn dem Benutzerteil der Bibliotheksliste die entsprechenden Programm- und Datenbibliotheken zugeordnet werden.

Anfangsmenü (INLMNU)

Das hinterlegte Menü wird nach der Anmeldung angezeigt.

Möglichkeiten einschränken (LMTCPB)

Mit dem Parameter LMTCPB(*PARTIAL) kann festgelegt werden, ob ein Benutzer in der Anmeldemaske außer Benutzername und Benutzerkennwort weitere Informationen eingeben kann. Wenn für den Parameter LMTCPB der Wert *YES angegeben wird, ist die Befehlszeile gesperrt. Dem Benutzer stehen nur folgende Befehle zur Verfügung:

2.4.4.1
Seite 4

DSPMSG	Nachrichten anzeigen
SNDMSG	Nachrichten senden
DSPJOBLOG	Jobprotokoll anzeigen
SIGNOFF	Abmelden

Text (TEXT)

Sie können Ihren Benutzerprofilen einen beschreibenden Text hinzufügen. Auf diese Möglichkeiten sollten Sie nicht verzichten, denn anhand der Profilnamen fällt es häufig schwer, die einzelnen Mitarbeiter zu identifizieren.

```
                    Benutzerprofil erstellen (CRTUSRPRF)

Auswahl eingeben und Eingabetaste drücken.


                          Zusätzliche Parameter

Sonderberechtigung . . . . . .   *USRCLS      *USRCLS, *NONE, *ALLOBJ...
            + für weitere Werte
Sonderumgebung . . . . . .       *SYSVAL      *SYSVAL, *NONE, *S36
Anmeldeinformationen anzeigen .  *SYSVAL      *SYSVAL, *NO, *YES
Zeitraum für Kennwortablauf . .  *SYSVAL      1-366, *SYSVAL, *NOMAX
Einheitensitzungen begrenzen . .  *SYSVAL     *SYSVAL, *YES, *NO
Tastaturpufferung . . . . . .    *SYSVAL      *SYSVAL, *NO, *TYPEAHEAD...
Maximal zulässiger Speicher . .  *NOMAX       Kilobyte, *NOMAX
Höchste Planungspriorität . . .  3           0-9
Jobbeschreibung . . . . . .      QDFTJOBD     Name
  Bibliothek . . . . . .           *LIBL      Name, *LIBL, *CURLIB
Gruppenprofil . . . . . .        *NONE        Name, *NONE
Eigner . . . . . .               *USRPRF      *USRPRF, *GRPPRF
                                                                Weitere ...
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Parameter für die Erstellung eines Benutzerprofils 1

Sonderberechtigung (SPCAUT)

Sonderberechtigungen sind zum Ausführen bestimmter Funktionen im System erforderlich. Folgende Sonderberechtigungen werden unterschieden:

***ALLOBJ**

Der Benutzer hat Zugriff auf alle Objekte des Systems.

***SECADM**

Der Benutzer kann Benutzerprofile verwalten. Dabei ist der Benutzer nicht berechtigt, Sonderberechtigungen zu erteilen, über die er selbst nicht verfügt. Um die Berechtigung *SECADM zu erteilen, braucht der Benutzer zusätzlich die Berechtigung *ALLOBJ.

***SAVSYS**

Der Benutzer kann das gesamte System sichern und zurücksichern. Auch wenn der Benutzer nicht über entsprechende Objektberechtigungen verfügt.

***JOBCTL**

Der Benutzer kann alle Jobs, Ausgabewarteschlangen und Subsysteme steuern.

***SERVICE**

Die Sonderberechtigung „Service" wird häufig eher dem Servicetechniker der IBM als einem Mitarbeiter im Haus übertragen.

***SPLCTL**

Der Benutzer hat das Recht, Ausgabewarteschlangen zu verwalten und zu steuern. Die Berechtigung wird normalerweise allen Benutzern übertragen, die Ihre Berichte direkt in den Ausgabewarteschlangen lesen müssen. Das Problem dabei ist, dass die Benutzer dann auch die Ausgaben anderer Benutzer lesen können – auch vertrauliche Daten. Eine Lösung für dieses Problem wären berechtigte Ausgabewarteschlangen. Dann können Benutzer nur die Ausgabewarteschlangen nutzen, in denen sie keinen Einschränkungen unterliegen.

***AUDIT**

Der Benutzer kann Systemwerte ändern, die die Protokollierung steuern.

***IOSYSCFG**

Der Benutzer kann Systemkonfigurationspunkte ändern – einschließlich aller Netzwerk- und Kommunikationseinheiten.

Abhängig von der Sicherheitsstufe Ihres Systems werden den Benutzerklassen folgende Sonderberechtigungen zugeordnet, wenn Sie im Parameter SPCAUT (*USRCLS) eintragen:

	*SECOFR	*SECADM	*PGMR	*SYSOPR	*USER
ALLOBJ	A, B	A	A	A	A
SECADM	A, B	A, B			
SAVSYS	A, B		A	A, B	A
JOBCTL	A, B		A	A, B	
SERVICE	A, B				
SPLCTL	A, B				
AUDIT	A, B				
IOSYSCFG	A, B				

A = Sicherheitsstufe 20 B = Sicherheitsstufe 30 bis 50

Sonderberechtigungen der Benutzerklassen

2.4.4.1
Seite 6

Kennwortablaufintervall (PWDEXPITV)

Dieser Wert bestimmt, wie viele Tage ein Kennwort gültig ist, bevor das System den Benutzer zur Änderung seines Kennwortes zwingt. In vielen Unternehmen wird hier die Standardeinstellung *SYSVAL übernommen. Das Kennwortintervall wird dann im Systemwert hinterlegt. Wenn Sie jetzt wissen wollen, wie der zugehörige Systemwert heißt, dann ist dies sehr einfach. Verwenden Sie den entsprechenden Parameternamen PWDEXPITV und stellen Sie dem Parameternamen ein Q voran. In unserem Beispiel erhalten Sie dann den Namen QPWDEXPITV. Jetzt wissen Sie, in welchem Systemwert der entsprechende Wert gespeichert ist. Diese Regel können Sie immer anwenden, wenn Sie als Parameterwert den Verweis *SYSVAL finden. Im Systemwert QPWDEXPITV steht standardmäßig der Wert *NOMAX, d.h.: eine Kennwortänderung wird nie erforderlich.

Sie sollten mindestens für Profile, die über die Sonderberechtigungen *ALLOBJ, *SERVICE und *SAVSYS verfügen, kürzere Änderungsintervalle einstellen.

Einheitensitzungen beschränken (LMTDEVSSN)

Dieser Parameter verhindert, dass sich ein Benutzer von mehr als einem Bildschirm am System iSeries anmeldet.

Maximal zulässiger Speicher (MAXSTG)

Durch diesen Parameter begrenzen Sie den Speicherplatz einzelner Benutzer.

Höchste Ausführungspriorität (PTYLMT)

Dieser Wert begrenzt für den Benutzer die Ausführungspriorität seines Jobs in den Jobwarteschlangen.

Jobbeschreibung (JOBD)

Wenn Sie einen Arbeitsauftrag an das System übertragen, benötigt die iSeries verschiedene Informationen für die Abarbeitung dieses Jobs. Die benötigten Informationen findet das System in einer Jobbeschreibung. Die im Benutzerprofil eingetragene Jobbeschreibung ist die Standardjobbeschreibung des Benutzers.

Gruppenprofil (GRPPRF)

Der Benutzer wird Mitglied der hier hinterlegten Gruppe.

Eigner (OWNER)

Wenn Sie auf einer iSeries eigene Objekte erstellen, werden Sie der Eigentümer des Objekts. Durch den Parameter OWNER ist es möglich, das Eigentumsrecht an Ihren Objekten auf die Gruppe zu übertragen, in der Sie Mitglied sind.

```
                    Benutzerprofil erstellen (CRTUSRPRF)

Auswahl eingeben und Eingabetaste drücken.

Gruppenberechtigung  . . . . . *NONE       *NONE, *ALL, *CHANGE, *USE...
Art der Gruppenberechtigung  . . *PRIVATE    *PRIVATE, *PGP
Zusätzliche Gruppenprofile . . . *NONE       Name, *NONE
            + für weitere Werte
Berechnungscode  . . . . . *BLANK
Dokumentkennwort . . . . . *NONE       Name, *NONE
Nachrichtenwarteschlange . . . *USRPRF     Name, *USRPRF
  Bibliothek . . . . . .           Name, *LIBL, *CURLIB
Zustellung . . . . . *NOTIFY     *NOTIFY, *BREAK, *HOLD, *DFT
Bewertungscodefilter . . . . . 0          0-99
Druckeinheit . . . . . *WRKSTN     Name, *WRKSTN, *SYSVAL
Ausgabewarteschlange . . . . . *WRKSTN     Name, *WRKSTN, *DEV
  Bibliothek . . . . . .           Name, *LIBL, *CURLIB
Abrufprogramm  . . . . . *SYSVAL     Name, *NONE, *SYSVAL, *ASSIST
  Bibliothek . . . . . .           Name, *LIBL, *CURLIB

                                                              Weitere ...
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Parameter für die Erstellung eines Benutzerprofils 2

Gruppenberechtigung (GRPAUT)

Wenn das Gruppenprofil Mitglied einer Gruppe ist und das Eigentumsrecht nicht auf die Gruppe übertragen wurde, können Sie mit diesem Parameter bestimmen, welche Berechtigungen andere Gruppenmitglieder an Ihren Objekten erhalten sollen.

Sie können folgende Werte nutzen:

*ALL	Die Gruppenmitglieder haben alle Rechte an den erstellten Objekten.
*CHANGE	Gruppenmitglieder können die Objekte ändern.
*USE	Gruppenmitglieder können die erstellten Objekte nutzen.
*EXCLUDE	Die Gruppenmitglieder haben keinen Zugriff auf die erstellten Objekte

Gruppenberechtigungsart (SUPGRPPRF)

Ein Benutzerprofil kann Mitglied einer Gruppe sein und damit Berechtigungen für Objekte anderer Gruppenmitglieder erhalten. Zusätzlich ist es durch den Parameter SUPGRPPRF möglich, bis zu 15 weitere Gruppen aufzunehmen, aus denen weitere Berechtigungen abgeleitet werden.

2.4.4.1
Seite 8

Nachrichtenwarteschlange (MSGQ)

Alle Nachrichten werden im System iSeries über Nachrichtenwarteschlangen ausgetauscht. Wenn in dem Parameter MSGQ der Standardwert *USRPRF beibehalten wird, erstellt das System für den Benutzer automatisch eine eigene Nachrichtenwarteschlange. Andernfalls müssen Sie in diesem Parameter den Namen einer anderen Nachrichtenwarteschlange eintragen.

Zustellung (DLVRY)

Der Parameter bestimmt, wie Nachrichten für diesen Benutzer zugestellt werden.

Mögliche Werte:

*NOTIFY	Am Arbeitsplatz ertönt ein akustisches Signal und in der Statuszeile des Bildschirms erscheint das Nachrichtensymbol. Der Benutzer muss die Nachricht mit dem Befehl DSPMSG anzeigen.
*BREAK	Die aktuelle Sitzung des Benutzers wird unterbrochen und die Nachricht wird angezeigt.
*HOLD	Nachrichten werden in der Nachrichtenwarteschlange gehalten. Der Benutzer erhält keine Hinweise bei eintreffenden Nachrichten.
*DFT	Nachrichten, die beantwortet werden müssen, werden mit dem Standardwert beantwortet. Informationsnachrichten werden ignoriert.

Bewertung (SEV)

Wird die Nachrichtenzustellung auf den Wert *NOTIFY oder *BREAK eingestellt, haben Sie damit die niedrigste Nachrichtenstufe festgelegt. Der Standardwert ist 00. Sie erhalten also alle Nachrichten. Wenn Sie einen höheren Bewertungscode einstellen, erhalten Sie nur noch Nachrichten mit einem entsprechend höheren Bewertungscode.

Druckeinheit (PRTDEV)

Bestimmt den Namen des Druckers, der für diesen Benutzer verwendet wird, wenn im Druckjob keine anderen Angaben gemacht werden.

Weitere mögliche Werte:

*WRKSTN	Es wird der Drucker verwendet, der in der Beschreibung des Bildschirms verwendet wird.
*SYSVAL	Wie war das noch gleich mit dem Namen der zugehörigen Systemwerte? Der Systemwert heißt: QPRTDEV. Alle Druckausgaben werden an den dort hinterlegten Systemdrucker weitergeleitet.
*CURRENT	Es wird der Drucker des aktuellen Jobs verwendet.

Ausgabewarteschlange (OUTQ)

Durch den Parameter kann die Ausgabewarteschlange für die Druckausgaben benannt werden.

*WRKSTN	Es wird die Ausgabewarteschlange genutzt, die in der Einheitenbeschreibung des Bildschirms verwendet wird.
*CURRENT	Es wird die Ausgabewarteschlange des aktuellen Jobs verwendet.
*DEV	Es wird die Ausgabewarteschlange verwendet, die zu dem im Parameter DEV hinterlegten Drucker gehört.

```
                    Benutzerprofil erstellen (CRTUSRPRF)

Auswahl eingeben und Eingabetaste drücken.

Sortierfolge . . . . . .   *SYSVAL      Name, *SYSVAL, *HEX...
  Bibliothek . . . . . .     _________   Name, *LIBL, *CURLIB
Sprachen-ID  . . . . . .   *SYSVAL      *SYSVAL...
Landes- oder Regions-ID  . . . .  *SYSVAL      *SYSVAL...
Zeichensatz-ID . . . . . .   *SYSVAL      *SYSVAL, *HEX...
Steuerung für Zeichen-ID . . . .  *SYSVAL      *SYSVAL, *DEVD, *JOBCCSID
Jobattr. länderspez. Angaben . .  *SYSVAL      *SYSVAL, *NONE, *CCSID...
            + für weitere Werte   _______
Länderspezifische Angaben  . . .  *SYSVAL
_____
Benutzerangaben  . . . . . .   *NONE       *NONE, *CLKWD, *EXPERT...
            + für weitere Werte   _______
Benutzernummer . . . . . .   *GEN         1-4294967294, *GEN
Gruppennummer  . . . . . .   *NONE        1-4294967294, *NONE, *GEN
Benutzerverzeichnis  . . . . . .  *USRPRF
_____
                                                                Weitere ...
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Parameter für die Erstellung eines Benutzerprofils 3

Grundwerk 10/2004

2.4.4.1
Seite 10

Sortierfolge (SRTSEQ)

Mit diesem Parameter legen Sie fest, ob Daten nach den Sortierkriterien der Muttersprache oder nach dem Hexadezimalsystem sortiert werden.

Sprachen-ID (LANGID)

Dieser Benutzerprofilparameter legt die für den Benutzer verwendete Sprache fest.

ID des codierten Zeichensatzes (CCSID)

Legt eine Zeichensatz-ID für diesen Benutzer fest. Die Liste der gültigen IDs erhalten Sie, wenn Sie die Taste F4 drücken.

Benutzer ID (UID)

Die iSeries vereint alle Dateiverzeichnissysteme in einem integrierten Dateisystem (IFS = Integrated File System). Benutzer, die das System anwenden, müssen eine UID-Nummer haben. Anhand dieser Nummer legt die iSeries die Berechtigungen fest. Jeder Benutzer erhält automatisch eine eindeutige ID vom System.

Gruppen-ID (GID)

Gruppenprofile erhalten eine GID. Alle Benutzerprofile, die Mitglied der Gruppe sind, erhalten die entsprechende GID zugeordnet.

Benutzerverzeichnis (HOMEDIR)

Dieser Parameter legt das Home-Directory im IFS für den Benutzer fest.

Berechtigung (AUT)

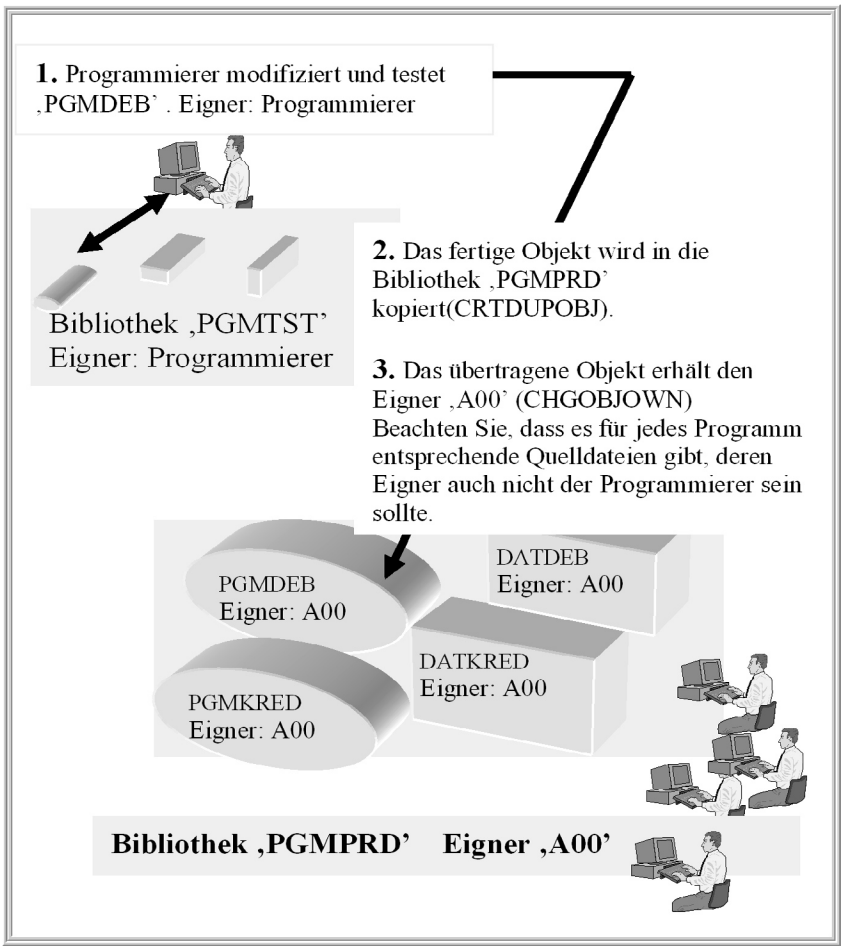
Dieser Parameter bestimmt die Rechte der Öffentlichkeit an diesem Objekt. Alle CRT-Befehle der iSeries haben den Parameter AUT, um die Öffentlichkeitsrechte zu bestimmen.

Gruppenprofile

Einige Mitarbeiter erstellen Objekte, z.B. Query-Abfragen, Programme oder Dateien. Für jedes der Objekte wird das entsprechende Benutzerprofil als Eigner eingetragen. Der Eigner verfügt über alle Rechte am Objekt. Sie können mit dem Befehl CHGOBJOWN den Eigentümer nachträglich ändern. Dabei haben Sie die Möglichkeit, die Berechtigungen des vorhergehenden Eigners zu widerrufen oder beizubehalten. Sie werden diesen Befehl sicherlich häufig verwenden, denn es ist nicht wünschenswert, dass ein Programmierer oder Anwender Eigentümer seiner erstellten Objekte bleibt, wenn er das Programm oder die Dateien für die Produktion frei gibt. Daher sollte das Objekteigentumsrecht bei der Objektfreigabe auf einen Eigner geändert werden, der keine Person ist. Das kann z.B. ein Benutzerprofil ohne Kennwort sein. Werden die Daten oder Programme zu Pflegezwecken überprüft, wird das Eigentumsrecht wieder auf das Profil des Programmierers übertragen.

Erinnern Sie sich an unser Beispiel?

Unser Programmierer, der das Programm OPDEB modifiziert, muss also das Eigentumsrecht an einen „unpersönlichen" Eigner übertragen.



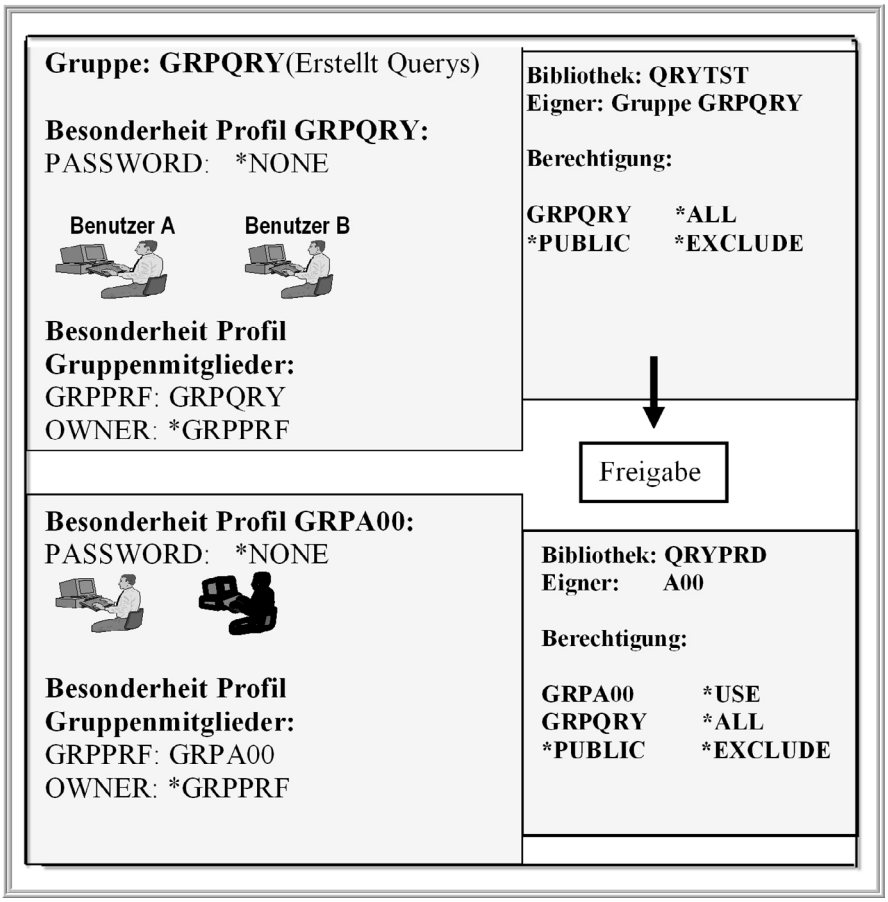
Eigentumsrechte übertragen

2.4.4.2
Seite 2

Dieser Vorgang erscheint zunächst etwas komplex, aber Sie können für diesen Änderungsdienst ein entsprechendes Verwaltungsprogramm schreiben, dass zukünftig diese Aufgabe wahrnimmt. Der Aufwand lohnt sich, denn:

- Wenn Sie das Schaubild betrachten, erkennen Sie, dass viele Objekte redundant im System gespeichert werden. Stellen Sie sich jetzt vor: Sie haben nicht nur einen Benutzer, der Objekte erstellt, sondern mehrere Personen. Dann können Sie wertvollen Plattenplatz sparen, wenn Sie Ihre Programmierer und Anwender zwingen, Sicherungskopien, alte Programme etc. regelmäßig zu löschen. Das erreichen Sie, indem Sie den maximalen Speicherbereich Ihrer Programmierer und/oder Anwender im Benutzerprofil (MAXSTG) begrenzen.
- Sie können Ihre freigegebenen, aktuellen Objekte jederzeit anhand des Eigners „A00“ identifizieren.
- Sie können aktive Benutzerprofile problemlos löschen.

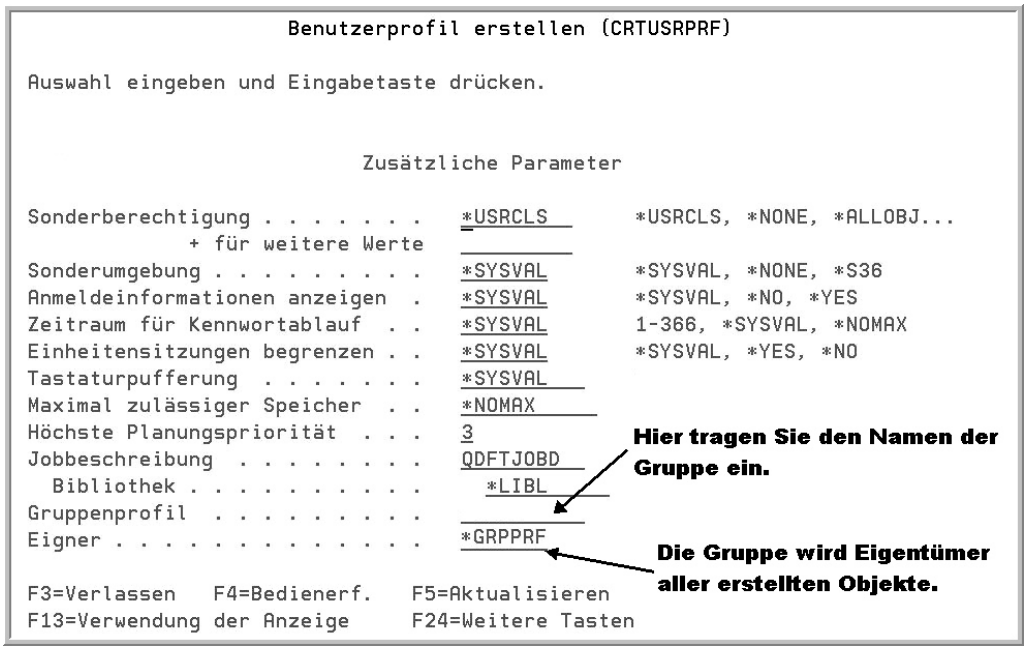
Neben dem Eigentumsrecht haben die Objekte öffentliche Berechtigungen. Wie Sie wissen, handelt es sich um die Berechtigungen, die gewährt werden, wenn keine spezifischen Rechte am Objekt vergeben werden. Das kann in der täglichen Praxis unter Umständen zu Berechtigungskonflikten führen. Eine Möglichkeit, derartige Probleme zu lösen, sind individuelle Berechtigungen einzelner Personen an den Objekten. Bei großen Anwenderkreisen haben Sie allerdings einen beträchtlichen Aufwand und sehr schnell werden Sie die Konflikte lösen, indem Sie *ALLOBJ-Berechtigungen vergeben. Es stellt sich daher die Frage, ob es nicht einfachere und weniger arbeitsintensive Methoden gibt, um Berechtigungsprobleme schon bei der Objekterstellung zu reduzieren. Das Problem kann durch Gruppenprofile gelöst werden. Gruppenprofile sind spezielle Benutzerprofile, denen die Eignerschaft an den Objekten übertragen wird. Die Gruppenmitglieder erhalten entsprechende Rechte an den Objekten der Gruppe.



Gruppenprofile erstellen

Wenn Sie Gruppenprofile benötigen, müssen Sie zunächst das Profil für die Gruppe erzeugen. Kennzeichnend für ein Gruppenprofil ist, dass kein Kennwort eingetragen wird. Dadurch können Sie sich mit dem Profil nicht anmelden. Es wird nur zur Steuerung der Objektberechtigungen verwendet. Für dieses Profil wird eine beliebige Benutzerklasse eingetragen, z.B. der Wert *USER. Die Gruppenmitglieder können die Sonderberechtigungen des Gruppenprofils übernehmen, wenn Ihre eigenen Berechtigungen nicht ausreichen. Nachdem das Profil GRPQRY und GRPA00 erstellt wurde, werden die Gruppenmitglieder der Gruppe zugeordnet.

2.4.4.2
Seite 4



Gruppenprofil zuordnen

Der Parameter „Gruppenprofil“ ordnet die einzelnen Profile der Gruppe zu. Der Parameter „Eigner“ bestimmt das Gruppenprofil zum späteren Objekteigner. Wenn Sie nicht wollen, dass der Objekteigner auf die Gruppe abgeändert wird, haben Sie eine weitere Möglichkeit: Tragen Sie als Gruppe wiederum das Gruppenprofil ein. OWNER (*USRPRF) gibt an, dass das Benutzerprofil – nicht das Gruppenprofil – Eigner der Objekte bleibt, die vom Benutzer erstellt werden. Jetzt müssen Sie über den Parameter GRPAUT spezifizieren, welche Berechtigungen die Gruppenmitglieder für die erstellten Objekte bekommen. Beachten Sie aber, dass die Eignerschaft und die Berechtigungen nur für zukünftige Objekte gelten. Alle bereits bestehenden Objekte sind von den Benutzerprofiländerungen nicht betroffen. Hier müssen Sie manuell die Eignerschaft auf die Gruppe übertragen oder die entsprechenden Objektberechtigungen vergeben.

Bei all Ihren Überlegungen bezüglich der Objekteigner, sollten Sie berücksichtigen, dass Sie Benutzerprofile nicht löschen können, wenn das Profil Eigentümer von Objekten ist. Der Befehl DLTUSRPRF stellt Ihnen drei Möglichkeiten zur Verfügung:

- Alle Objekte des Benutzers werden zusammen mit dem Profil gelöscht. Diese Möglichkeit setzt allerdings voraus, dass reale Benutzer nicht das Eigentumsrecht an produktiven Objekten besitzen.
- Alle Objekte des Benutzers werden an einen anderen Benutzer im System übertragen. Wenn Sie diese Möglichkeiten nutzen, behalten Sie natürlich alle Objekte des Benutzers. Das können beliebige Sicherungskopien, Testobjekte, vorangegangene Objektvarianten etc. sein. Sie werden

zukünftig ein Problem haben, die produktiven Objekte von den unproduktiven Objekten zu unterscheiden und nehmen den „Müll“ ehemaliger Mitarbeiter mit.

- Das Benutzerprofil wird nicht gelöscht, wenn der Benutzer Eigentümer von Objekten ist.

Und noch eine Anmerkung zum Schluss:

Die IBM unterstützt nicht den Befehl RNMOBJ (rename objekt) für Benutzerprofile, weil andere Objekte auf dieses Objekt verweisen. Wenn Sie ein Profil umbenennen wollen, gehen Sie folgendermaßen vor:

- Kopieren Sie das alte Benutzerprofil PROFALT in das neue Profil PROFNEU.
- Übertragen Sie an das neue Benutzerprofil alle persönlichen Berechtigungen des alten Profils:
  GRTUSRAUT GRTUSRAUT USER(PROF_NEU)
  REFUSER(PROF_ALT)
- Löschen Sie jetzt das alte Profil und übertragen Sie die Eigentumsrechte an alle anderen Objekten auf das neue Profil.
  DLTUSRPRF USRPRF(PROF_ALT)
  OWNOPJOPT(*CHGOWN PROF_NEU)

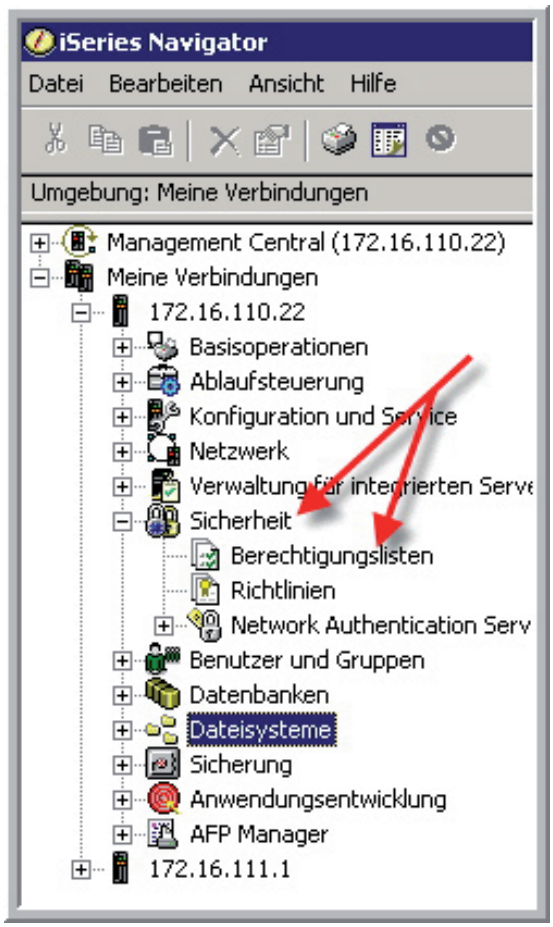
2.4.4.2
Seite 6

Berechtigungslisten

Sie können in Berechtigungslisten die Zugriffsberechtigungen auf Objekte zentral in einer Liste hinterlegen und verwalten. Der Vorteil von Berechtigungslisten ist ein deutlich reduzierter Verwaltungsaufwand. Für die Erstellung und Verwaltung solcher Listen stehen verschiedene 5250-Befehle zur Verfügung:

CRTAUTL	Berechtigungsliste erstellen
CHGAUTL	Berechtigungsliste ändern
DLTAUTL	Berechtigungsliste löschen
DSPAUTL	Berechtigungsliste anzeigen
ADDAUTLE	Berechtigungslisteneintrag hinzufügen
RMVAUTLE	Berechtigungslisteneintrag entfernen

Doch auch der iSeries Navigator bietet Ihnen die Möglichkeit mit Berechtigungslisten zu arbeiten:

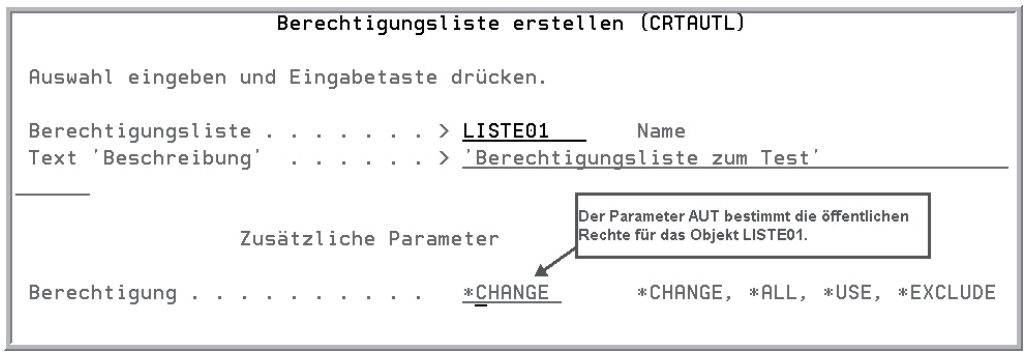


Berechtigungslisten im iSeries Navigator

2.4.4.3
Seite 2

Berechtigungslisten erstellen

Um Berechtigungslisten zu erstellen, nutzen Sie in einer 5250-Emulation den Befehl CRTAUTL (Create authority list).



Eine Berechtigungsliste mit dem Befehl CRTAUTL erstellen

Sie sehen, dass der Befehl nur über wenige Parameter verfügt. Auffällig ist auch, dass Sie keine Bibliothek benennen können. Objekte der Art *AUTL werden grundsätzlich in der Bibliothek QSYS gespeichert. Sie sollten daher die Objektnamen sorgfältig planen.

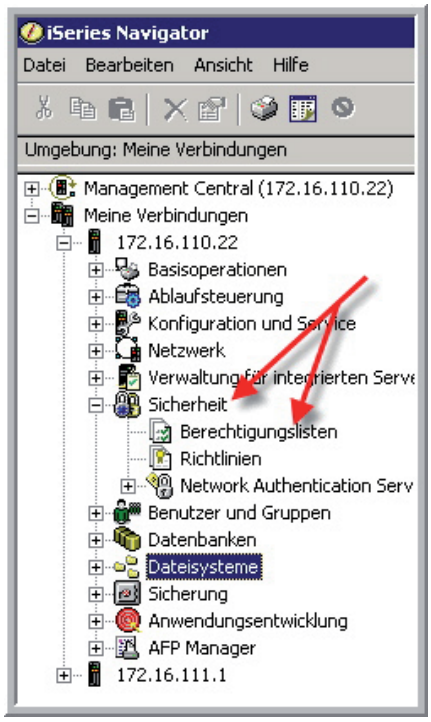
Mit dem Parameter AUT bestimmen Sie die *PUBLIC-Rechte für Objekte, die durch die Berechtigungsliste berechtigt sind. Allerdings muss bei der Rechtevergabe ausdrücklich gesagt werden, dass die *PUBLIC-Rechte über die Berechtigungsliste und nicht durch die persönlichen Objektberechtigungen gesteuert werden. Dazu später mehr.

Um Berechtigungslisten mit dem iSeries Navigator zu erstellen, gehen Sie wie folgt vor:

1. Stellen Sie zunächst die Verbindung zum entsprechenden iSeries Server her.
2. Erweitern Sie anschließend den Punkt „Sicherheit“.

12. Ergänzung 09/2007

2.4.4.3.1
Seite 2

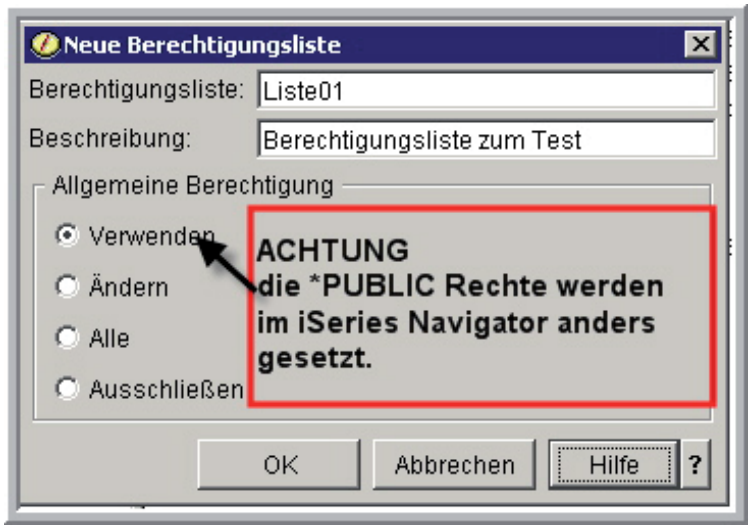


Berechtigungslisten erstellen

3. Markieren Sie jetzt den Eintrag „Berechtigungslisten“ mit der rechten Maustaste und wählen Sie „Neue Berechtigungsliste“.

Berechtigungsliste erstellen

Anschließend erscheint ein weiteres Fenster:



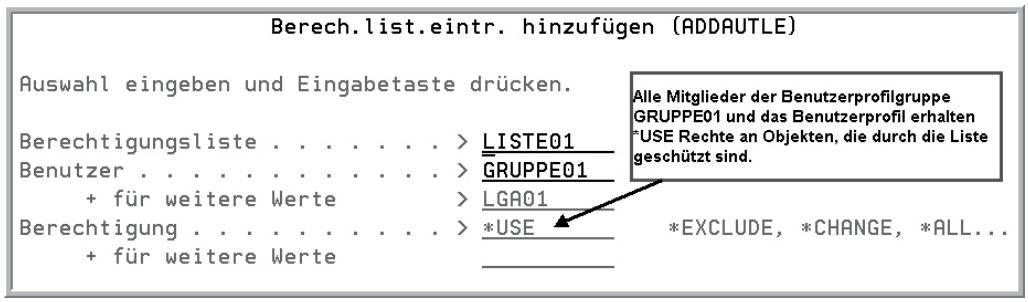
Berechtigungslisten erstellen

Sie sehen, auch im iSeries Navigator sind die Parameter eher spärlich. Eines fällt allerdings auf: der Parameter AUT hat im iSeries Navigator eine andere Default-Einstellung. Während die 5250-Emulation für die *PUBLIC-Rechte den Wert *CHANGE verwendet, erhalten die Mitglieder der Gruppe *PUBLIC nur *USE-Rechte, wenn die Liste im iSeries Navigator erstellt wird.

2.4.4.3.1
Seite 4

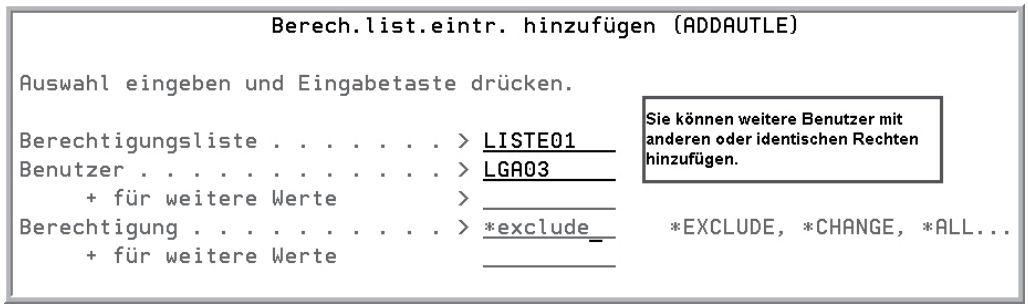
Benutzer hinzufügen und verwalten

Im nächsten Schritt müssen in der Berechtigungsliste Benutzer und/oder Benutzergruppen hinzugefügt werden. Auch hierfür stehen zwei Interfaces zur Verfügung. In einer 5250-Emulation verwenden Sie die Befehle ADDAUTLE (add authority list entry), um Benutzer hinzuzufügen und den Befehl RMVAUTLE (remove authority list entry), um Benutzer zu entfernen.



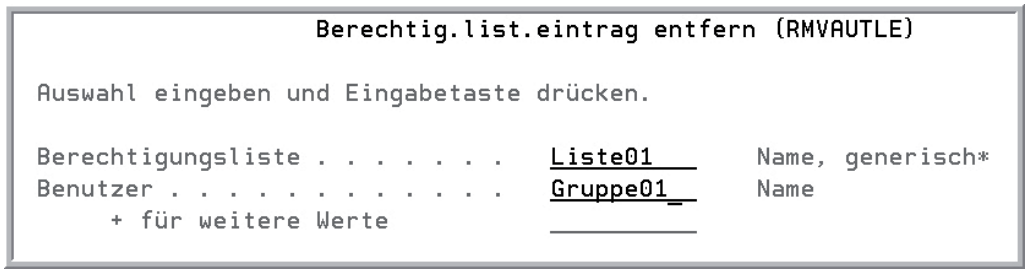
Berechtigungslisteneinträge hinzufügen

Es ist durchaus möglich, mehrere Benutzergruppen mit unterschiedlichen Zugriffsberechtigungen über eine Liste zu verwalten.



Weitere Einträge hinzufügen

Genauso einfach ist es, Benutzer wieder aus der Liste zu entfernen:



Einträge entfernen

Sobald Sie einen Eintrag entfernen, sind die entsprechenden Objektberechtigungen entzogen und Sie können die Liste sogar bearbeiten, während die Benutzer angemeldet sind und die gesicherten Objekte in Gebrauch sind.

2.4.4.3.2
Seite 2

Zur Prüfung der Listeneinträge bietet sich der Befehl DSPAUTL (display authority list) an:

```
                        Berechtigungsliste anzeigen

Objekt . . . . . :   LISTE01          Eigner . . . . . :   RASCHE
  Bibliothek . . . :     QSYS         Primärgruppe . . . :   *NONE

             Objekt-   List
Benutzer     berechtg. verw
*PUBLIC      *CHANGE
RASCHE       *ALL       X
LGA01        *USE
LGA03        *EXCLUDE
```

Die Funktionstaste F15 zeigt die durch die Berechtigungsliste gesicherten Objekte!

Berechtigungslisteneinträge anzeigen

In der Anzeige können Sie jederzeit die eingetragenen Benutzer und die zugeordneten Rechte anzeigen. Wenn Sie nun auch noch die gesicherten Objekte prüfen wollen, können Sie die Funktionstaste F15 nutzen.

```
                  Berechtigungslistenobjekte anzeigen

Berechtigungsliste . . . . . :   LISTE01
  Bibliothek . . . . . :     QSYS
Eigner . . . . . :   RASCHE
Primärgruppe . . . . . :   *NONE

                                                Primär-
Objekt      Bibliothek  Art       Eigner        gruppe      Text

  (Durch diese Berechtigungsliste werden keine Objekte gesichert)
```

Gesicherte Objekte anzeigen

Sie sehen, dass derzeit noch keine Objekte durch die Liste geschützt sind, d.h. bislang wird die Liste nicht verwendet.

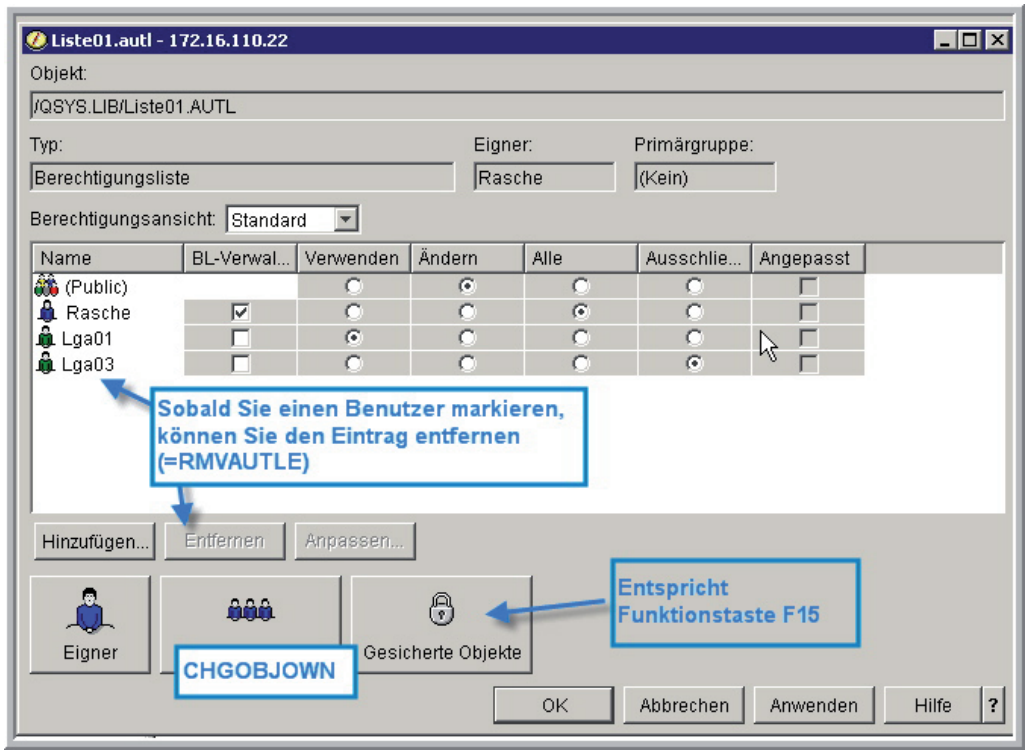
Falls Sie Einträge verändern müssen, steht der Befehl CHGAUTLE (change authority list entry) zur Verfügung:

```
                  Berecht.listeneintrag ändern (CHGAUTLE)

Auswahl eingeben und Eingabetaste drücken.

Berechtigungsliste . . . . . .   Liste01      Name, generisch*
Benutzer . . . . . . . . . . .   LGA01        Name, *PUBLIC
     + für weitere Werte                      
Berechtigung . . . . . . . . .   *USE         *EXCLUDE, *CHANGE,
     + für weitere Werte                      
```

Listeneinträge verändern

Bevor wir nun Objekte mit der Berechtigungsliste verbinden, schauen wir uns kurz an, welche Möglichkeiten Sie im iSeries Navigator haben:

Wenn Sie eine der Berechtigungslisten mit einem Doppelklick öffnen, erhalten Sie die folgende Anzeige:



Listeneinträge mit dem iSeries Navigator verwalten

Der Navigator bietet bei der Verwaltung einige entscheidende Vorteile. Sie können die Einträge sehen und gleichzeitig ändern. Für die Änderung stehen Ihnen zwei Möglichkeiten zur Verfügung:

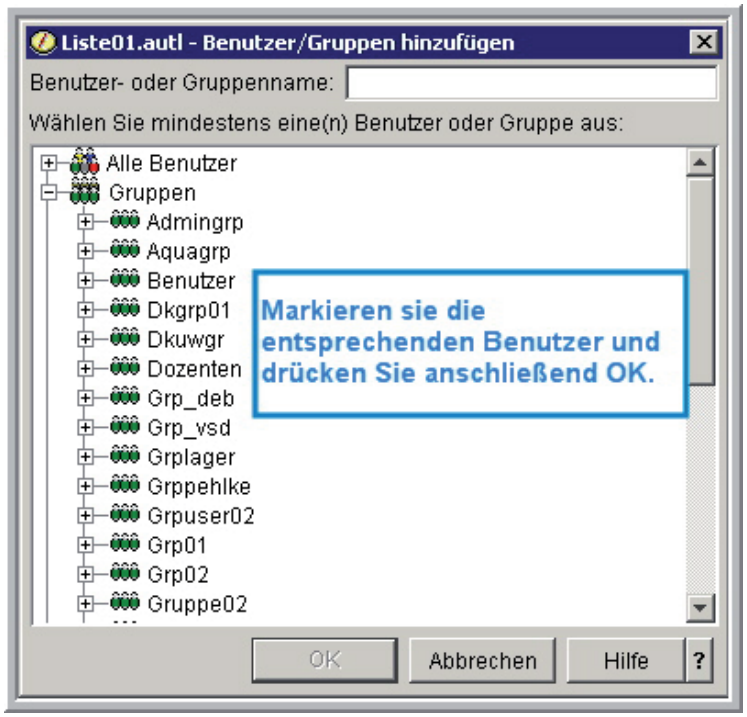
Wenn Sie einen Eintrag markieren und anschließend den Button „ANPASSEN“ nutzen, erscheint ein weiteres Fenster, in dem Sie die Veränderungen vornehmen. Sobald Sie Ihre Änderungen in dem zusätzlichen Fenster bestätigen, werden Sie übernommen.

Es geht aber auch einfacher. Klicken Sie einfach mit der Maus auf die Berechtigungen, um sie zu gewähren oder zu entziehen.

Falls Sie weitere Benutzer in die Liste aufnehmen wollen, wählen Sie „Hinzufügen“. Daraufhin erscheint ein Fenster, das Ihnen komfortabel die Auswahl weiterer Benutzer ermöglicht.

2.4.4.3.2

Seite 4

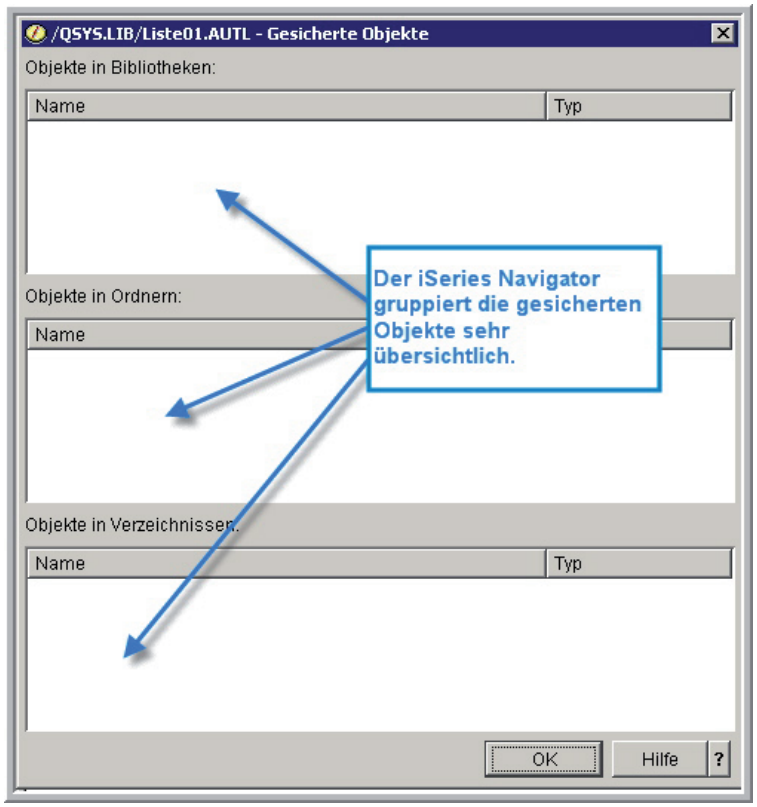


Weitere Benutzer auswählen

Sie können in dem Fenster Gruppenprofile und/oder einzelne Benutzerprofile markieren. Um mehrere Einträge zu markieren, stehen Ihnen im iSeries Navigator die SHIFT- oder die STRG-Taste zur Verfügung. Die Wirkung ist analog der Arbeit mit Windows. Sobald Sie Ihre Arbeit mit „OK“ bestätigen, werden die Einträge übernommen. Die neuen Benutzer erhalten automatisch die Berechtigung *USE. Dies lässt sich aber problemlos ändern, wie wir bereits festgestellt haben.

Natürlich können Sie sich im iSeries Navigator auch die gesicherten Objekte anzeigen lassen. Ich finde, dass die Darstellung im Navigator so sehr viel übersichtlicher ist, da hier die Objekte in drei Gruppen unterteilt werden:

- iSeries-Objekte, die in Bibliotheken gespeichert sind,
- iSeries-Dokumente, die in iSeries Ordnern gesichert wurden,
- und Objekte, die im IFS abgelegt sind.



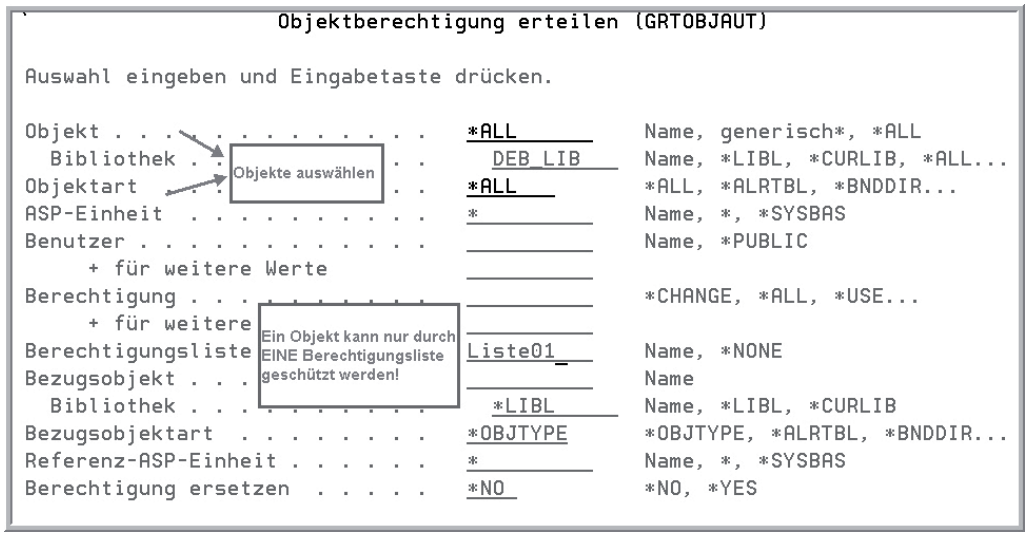
Gesicherte Objekte

2.4.4.3.2

Seite 6

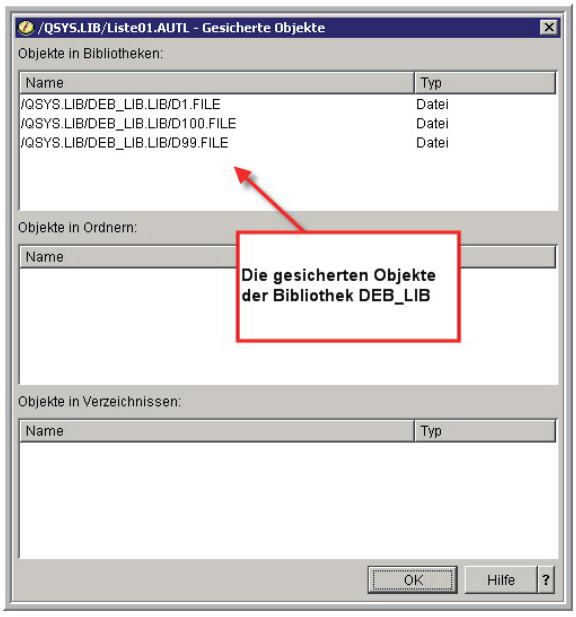
Objekte durch Berechtigungslisten schützen

Alles ist vorbereitet. Jetzt müssen Sie nur noch die Berechtigungslisten den entsprechenden Objekten zuordnen. Natürlich können Sie die Berechtigungslisten jedem Objekt einzeln zuordnen, indem Sie den Befehl EDTOBJAUT (Edit object authority) nutzen. Nur kann dieses Vorgehen ziemlich zeitaufwendig sein und empfiehlt sich daher nur, wenn Sie wenige Objekte ansprechen müssen. Besser ist es, Sie nutzen den Befehl GRTOBJAUT (Grant object authority).



Der Befehl GRTOBJAUT

Sie sehen, dass ich mit einem Befehl alle Objekte der Bibliothek DEB_LIB durch unsere Berechtigungsliste schütze. Das Ergebnis sieht nach Ausführung des Befehls folgendermaßen aus:

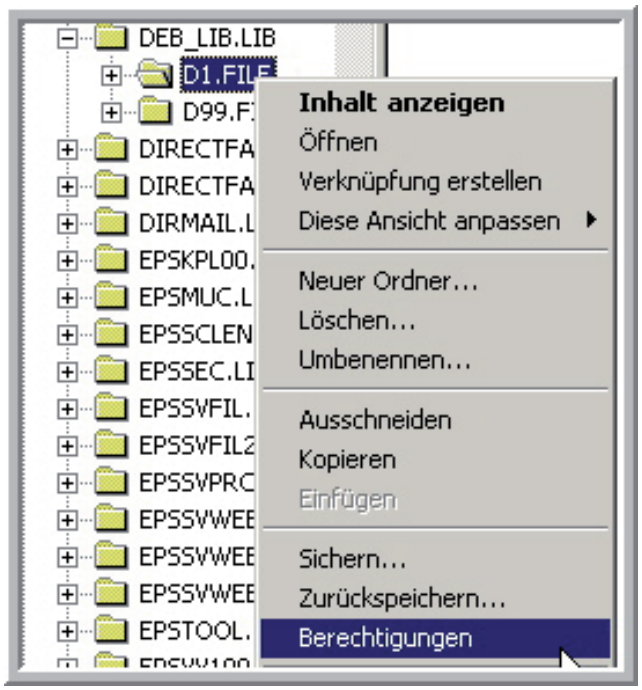


Das Ergebnis

2.4.4.3.3
Seite 2

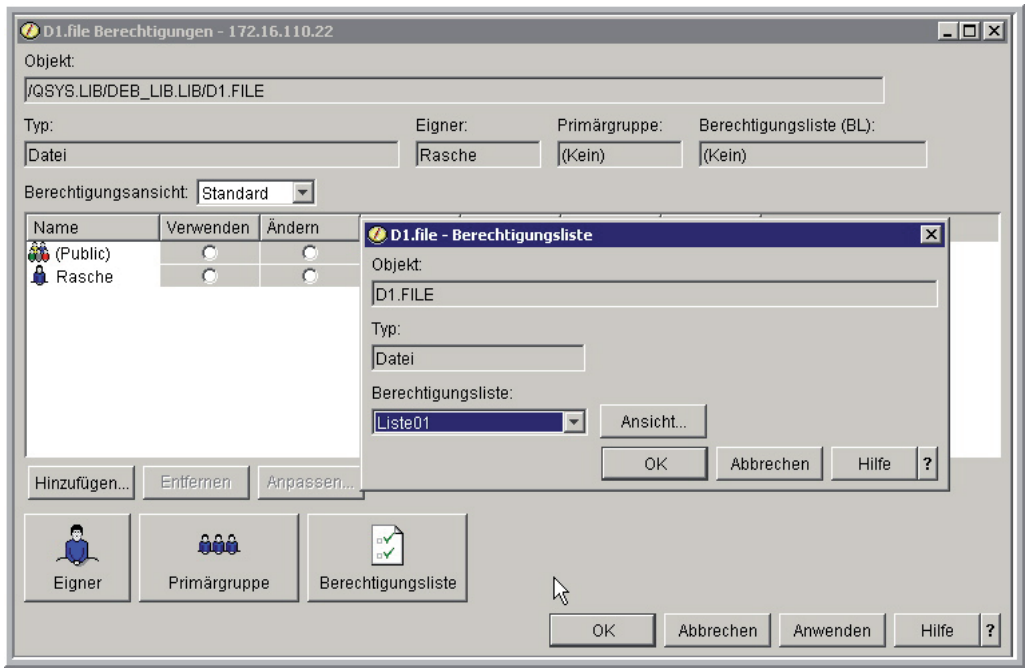
Jetzt erhalten die in der Berechtigungsliste hinterlegten Benutzer die hinterlegten Berechtigungen an den Objekten. Zukünftig reicht es aus, weitere Benutzer in die Listen aufzunehmen oder von der Liste zu entfernen, um die Berechtigungen zu entziehen oder zu gewähren. Genauso einfach ist es, weitere Objekte durch die Liste zu schützen: Sie müssen lediglich die Berechtigungsliste den Objekten zuordnen oder durch eine andere Liste ersetzen. Beachten Sie dabei, dass ein Objekt nur durch eine Berechtigungsliste geschützt werden kann.

Auch der iSeries Navigator ermöglicht es, neue Objekte einer Berechtigungsliste zuzuordnen. Dazu müssen Sie das IFS öffnen, einzelne Objekte mit der rechten Maustaste markieren und anschließend den Eintrag „Berechtigungen“ wählen. Daraufhin erscheint das folgende Fenster:



Objekte einer Berechtigungsliste zuordnen

Nun erscheint das folgende Fenster:



Einzelne Objekte einer Berechtigungsliste zuordnen

Leider ist es bislang nicht möglich, mehrere Objekte zu markieren und anschließend in einem Vorgang zuzuordnen, so dass der Navigator wenig geeignet erscheint, wenn Sie viele Objekte berücksichtigen müssen.

Neben der einfachen Handhabung bieten Berechtigungslisten **weitere Vorteile:**

- Sie können die Listen bearbeiten, während die Benutzer mit den gesicherten Objekten arbeiten.
- Falls Sie die gesicherten Objekte wieder herstellen müssen, ist eine erneute Zuordnung der Berechtigungslisten nicht erforderlich. Dies geschieht automatisch.
- Berechtigungslisten können nicht versehentlich gelöscht werden, denn das Betriebssystem stellt sicher, dass nur Listen gelöscht werden, die nicht verwendet werden.

Natürlich gibt es auch **Nachteile**.

Die Planung und Konfiguration der Berechtigungslisten erfordert etwas Zeit. Sie müssen Berechtigungslisten sorgfältig planen, um unnötigen System-Overhead zu vermeiden und um den Überblick über die bestehenden Listen und die gesicherten Objekte zu behalten. Ein Beispiel für die Planung und den Einsatz finden Sie im Kapitel 2.5.2 „Ressourcenschutz durch Berechtigungslisten“.

2.4.4.3.3

Seite 4

Sicherheitsprotokollierung

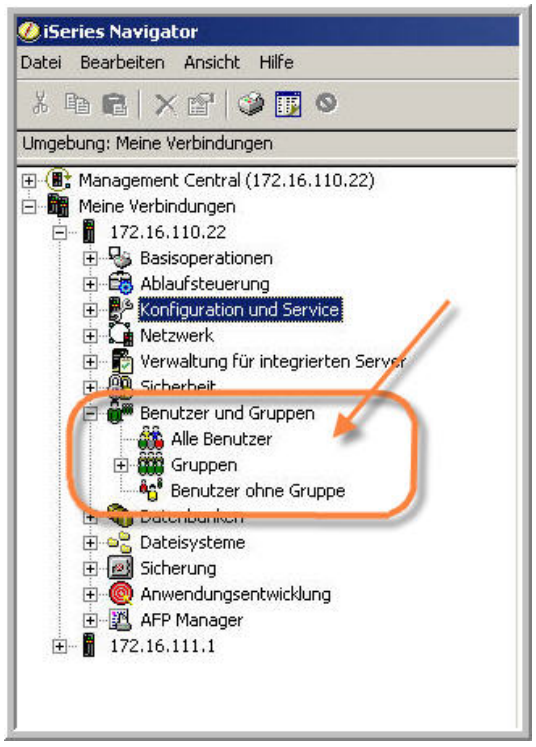
Sie haben die Möglichkeit, die Zugriffe auf Objekte oder Funktionen einzelner Benutzer zu überwachen. Im Gegensatz zu System- und Jobprotokollen, die sofort zur Verfügung stehen, muss das iSeries-Sicherheitsprotokoll allerdings erst aktiviert werden. Dieses Sicherheitsprotokoll kann nicht im Profil des Benutzers aktiviert werden. Der verantwortliche Parameter AUDLVL (audit level) wird mit dem Befehl CHGUSRAUD (change user audit) eingestellt oder verändert. Mit diesem Wert legen Sie die zu überwachenden Aktionen des Benutzers fest. Die Handhabung des Journals und den Journal-Receiver entnehmen Sie bitte dem Cl-Referenzhandbuch. Die Aktivierung des Sicherheitsprotokolls ist meist nur für einen kurzen Zeitraum erforderlich. Nach einem Zeitraum von einigen Tagen deaktivieren Sie die Prüfung und werten die Aufzeichnungen aus.

2.4.4.4

Seite 2

Benutzerverwaltung mit dem iSeries Navigator

Die Administration einer iSeries war lange Zeit die Domäne von 5250-Bildschirmen und -Dialogen. Mit Erscheinen der Client Access-Version 3.2 kam das Produkt iSeries Navigator auf den Windows Desktop. Während früher alle Funktionen einer iSeries aus dem 5250-Interface heraus installier-, konfigurier- und administrierbar waren, hat sich dieses Bild heute drastisch verändert. Heute existieren viele Funktionen, die ausschließlich dem iSeries Navigator vorbehalten sind. Die Dominanz der 5250-Bildschirme ist damit beendet. Ob dies gut oder schlecht ist, möchte ich nicht beurteilen. Fakt ist allerdings, dass der Navigator an vielen Stellen die tägliche Arbeit vereinfacht, so auch die Verwaltung der Benutzerprofile. Mit der Komponente „Benutzer und Gruppen“ können Sie sich eine vollständige Liste aller Benutzerprofile und Gruppenprofile auf einer iSeries anzeigen lassen. Ausgehend von dieser Anzeige können Sie neue Profile erstellen oder bestehende verwalten.



Mit Benutzerprofilen arbeiten

Sie sehen, dass Sie sich „Alle Benutzer“ anzeigen lassen können. Hierbei spielt die Gruppenzugehörigkeit einzelner Profile keine Rolle. Außerdem können „Gruppen“ mit den entsprechenden Gruppenmitgliedern dargestellt werden. Eine Variante, die auf dem 5250-Interface nicht existiert. Und Sie haben zudem die Möglichkeit mit „Benutzern ohne Gruppe“ zu arbeiten. So erhalten Sie einen umfassenden Überblick über die Organisation Ihrer Benutzerprofile.

2.4.5
Seite 2

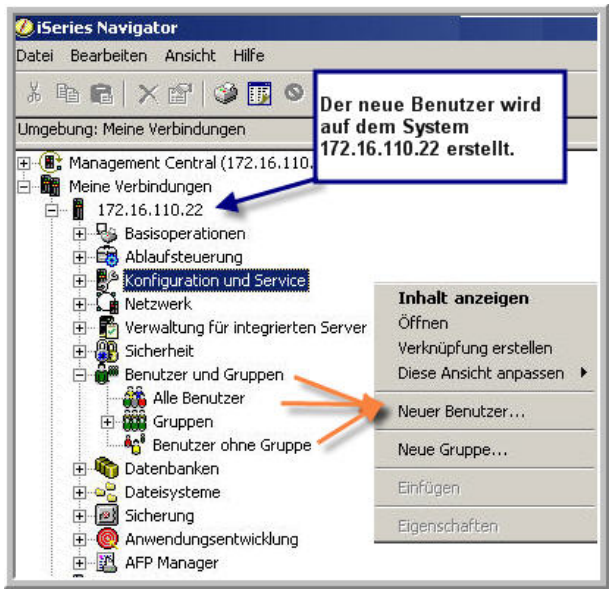
Das Leistungsspektrum der Komponente „Alle Benutzer“ beinhaltet die folgenden Möglichkeiten:
- Benutzer zu vorhandener Gruppe hinzufügen,
- Kennwörter ändern,
- Benutzer kopieren,
- Benutzer erstellen,
- Benutzer löschen,
- Benutzer bearbeiten,
- Zertifikate für einen Benutzer öffnen.

Verwenden Sie die Komponente „Gruppen“, um
- Gruppen zu erstellen,
- Gruppen basierend auf einer anderen Gruppe zu erstellen,
- Gruppen zu löschen,
- Gruppen zu bearbeiten,
- Benutzer aus Gruppen zu entfernen.

Die Komponente „Benutzer ohne Gruppe“ bietet Ihnen die folgenden Möglichkeiten:
- Benutzer zu vorhandener Gruppe hinzufügen,
- Kennwörter ändern,
- Benutzer erstellen,
- Benutzer kopieren,
- Benutzer löschen,
- Benutzer bearbeiten,
- Zertifikate für einen Benutzer öffnen.

Einen neuen Benutzer erstellen

Generell besteht die Möglichkeit mit Management Central einen neuen Benutzer auf einem oder mehreren Endpunktsystemen oder auf allen Systemen in einer Systemverwaltungsgruppe zu erstellen. Sie können aber auch nur einen Benutzer auf einem einzelnen System erstellen. Zunächst erstelle ich lediglich einen neuen Benutzer auf einem System in unserer Liste der Verbindungen. Dazu erweitere ich das System in der Liste der Verbindungen auf dem ich den neuen Benutzer erstellen will und klicke jetzt mit der rechten Maustaste auf „Benutzer und Gruppen“ und wähle „Neuer Benutzer“. Alternativ können Sie auch auf „Alle Benutzer“ oder auf die Komponente „Benutzer ohne Gruppen“ klicken.

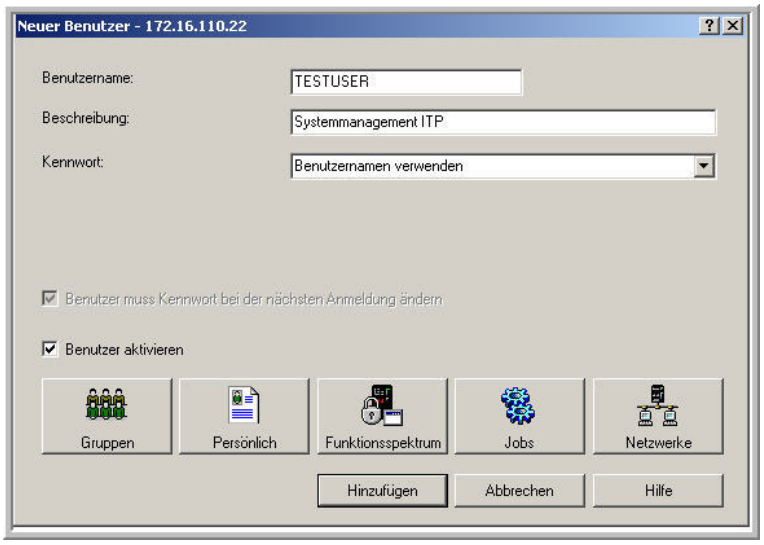


Einen neuen Benutzer anlegen

Anschließend öffnet sich ein Dialog, der sich mit dem CL-Befehl CRTUSRPRF vergleichen lässt. Da viele Parameter bereits detailliert im Kapitel 2.4.4.1 besprochen wurden, verzichte ich an dieser Stelle auf eine erneute Erläuterung und nenne aus Gründen der Vergleichbarkeit lediglich den entsprechenden Parameter des Befehls CRTUSRPRF. Allerdings enthält der iSeries Navigator-Dialog an dieser Stelle auch Möglichkeiten, die das 5250-Interface bzw. der CRTUSRPRF-Befehl so nicht bietet. Diese Möglichkeiten werde ich zu gegebener Zeit entsprechend erläutern.

2.4.5.1
Seite 2

Beginnen wir aber zunächst mit dem Startfenster.

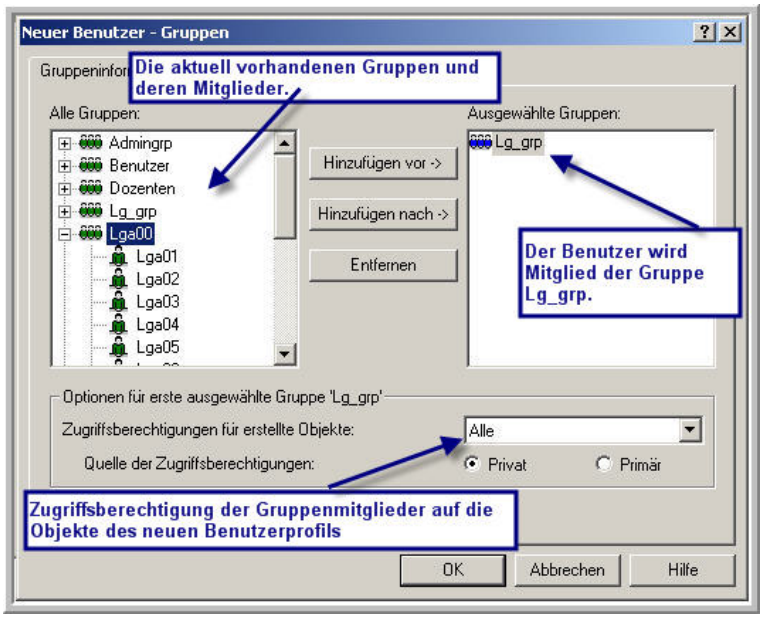


Der Startdialog „Neuer Benutzer“

Zunächst geben Sie den Benutzernamen (USRPRF), den beschreibenden Text (TEXT) und das Kennwort (PASSWORD) ein. Bei dem Kennwort lassen sich die Sonderwerte „Benutzername verwenden“ und „Kein Kennwort“ über ein Pulldown-Menü auswählen. Der Sonderwert „Kein Kennwort (*NONE) verwenden“ wird genutzt, um ein internes Benutzerprofil zu erzeugen, das nicht für die Anmeldung am System verwendet wird. Außerdem aktivieren bzw. deaktivieren Sie auf der ersten Seite das Profil und können damit einen Kennwortwechsel bei der nächsten Anmeldung (PWDEXP) erzwingen. Nachdem Sie die Grundparameter eingegeben haben, stehen am unteren Bildschirmrand weitere Schaltflächen zur Verfügung.

Der Dialog „Gruppen“

Ein Benutzer kann Mitglied einer Gruppe sein. Alle Mitglieder einer Gruppe verfügen über identische Berechtigungen. Normalerweise besteht eine Gruppe aus einem Personenkreis, der in derselben Unternehmensabteilung arbeitet, einen ähnlichen Aufgabenbereich hat und dieselben Anwendungen auf identische Weise nutzen muss.



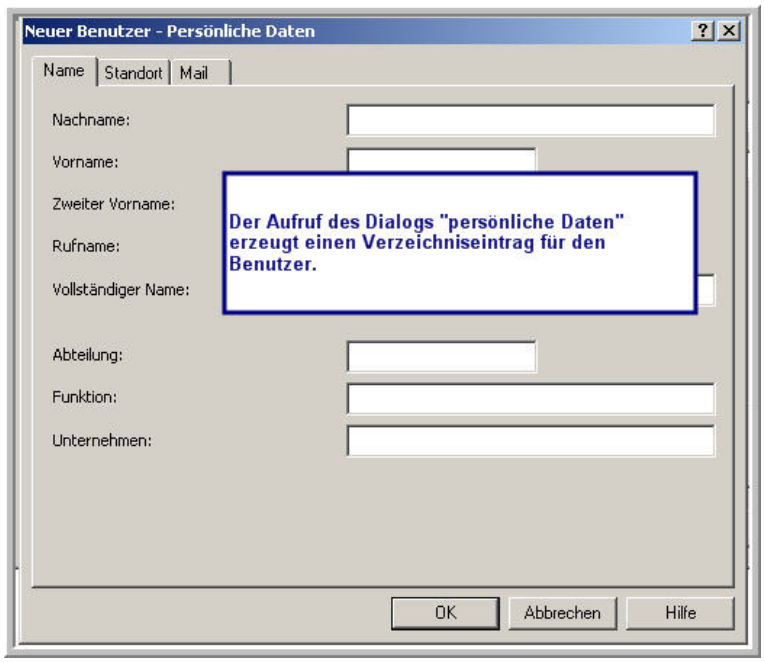
Zuordnung zu einer Gruppe

Auf der linken Seite sehen Sie alle existierenden Gruppenprofile und die aktuell zugeordneten Benutzer. Um das neu erstellte Profil einer Gruppe zuzuordnen, markiere ich auf der linken Seite eine Gruppe und drücke anschließend „hinzufügen vor“ oder „hinzufügen nach“. Die Gruppe erscheint daraufhin im rechten Bildschirmteil. Zusätzlich können Sie für die erste ausgewählte Gruppe des Benutzers weitere Berechtigungsoptionen angeben. Die Optionen (GRPAUT) bestimmen die Zugriffsberechtigungen der Gruppenmitglieder auf Objekte des neuen Benutzers. Mit dem iSeries Navigator können Sie die Objekteigenschaften nicht mehr an die Gruppe übertragen, d.h. der Benutzer „TESTUSER“ behält die Eignerschaft an allen Objekten, die er erstellt. Erstellt der Benutzer TESTUSER z. B. eine Query-Abfrage, ist er trotz der Gruppenmitgliedschaft Eigentümer des Objektes. Aufgrund der hinterlegten Option „ALLE“ dürfen allerdings alle Gruppenmitglieder auf das Objekt zugreifen als seien auch sie Objekteigentümer.

2.4.5.1.1
Seite 2

Der Dialog „Persönlich“

Der Dialog „Persönlich“ erzeugt für den Benutzer einen Verzeichniseintrag. Dies entspricht dem CL-Befehl ADDDIRE (add directory entry). Analog dem Befehl ADDDIRE können Sie weitreichende Benutzerdaten erfassen. Geben Sie hier den Namen, den Standort und die Mail-Optionen des Benutzers an.

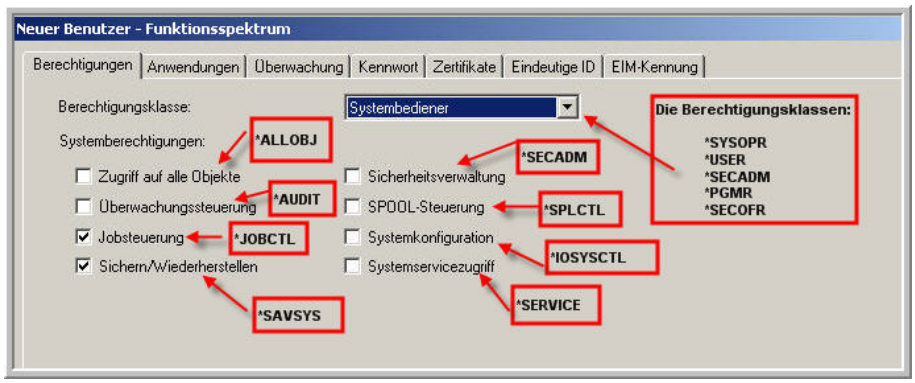


Der Dialog „Persönliche Daten“

2.4.5.1.2
Seite 2

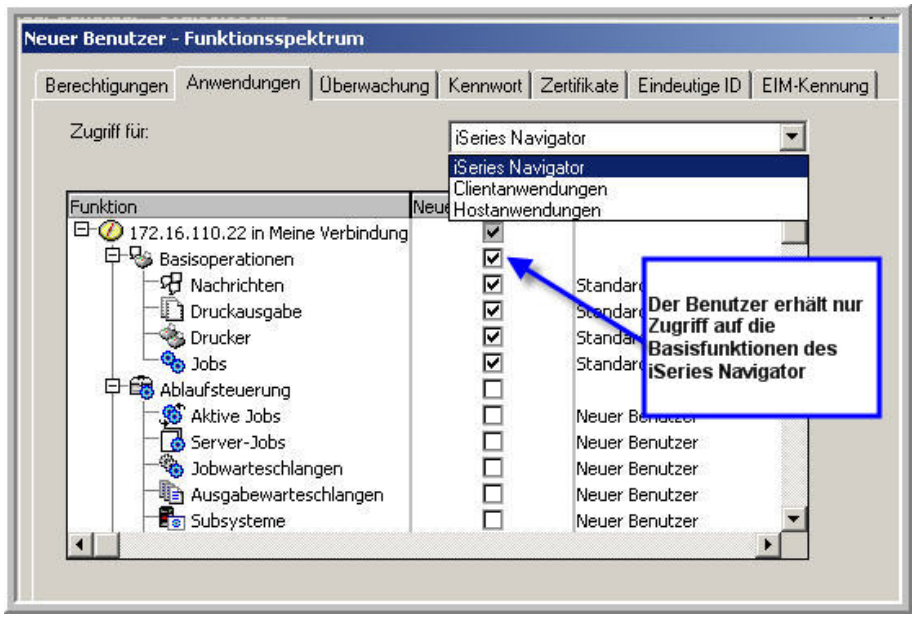
Der Dialog „Funktionsspektrum“

Im Dialog „Funktionsspektrum“ finden Sie wieder viele Parallelen zum CL-Befehl CRTUSRPRF. Im Register „Berechtigungen“ legen Sie die Benutzerklasse (USRCLS) fest und die entsprechenden Sonderberechtigungen (SPCAUT). Anders als im CL-Befehl können Sie allerdings sofort sehen, welche Sonderberechtigungen sich hinter den Benutzerklassen verbergen.



Benutzerberechtigungen hinterlegen

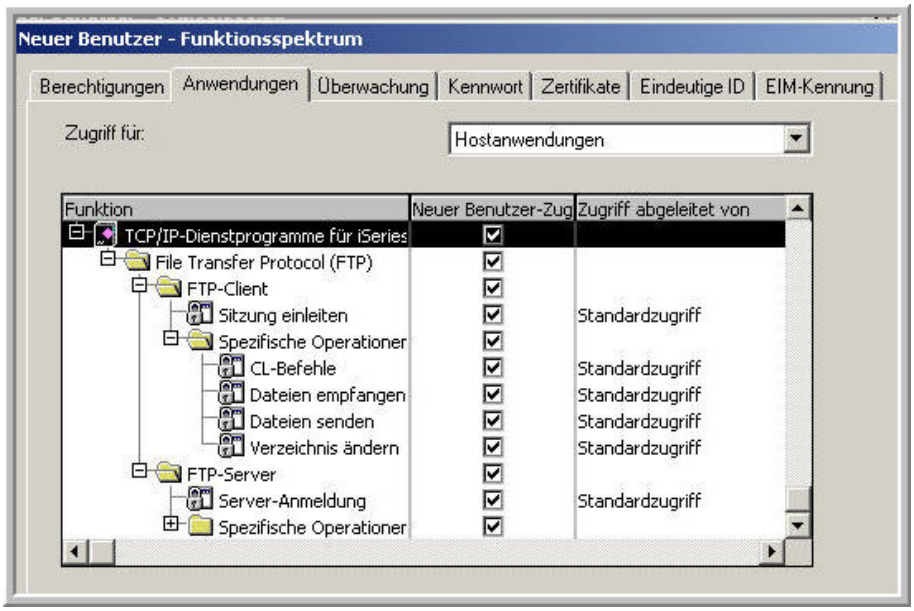
Das Register „Anwendungen“ ist Teil der Management-Zentralfunktion „Anwendungsverwaltung“ und auf dem 5250-Interface nicht verfügbar. Mit Hilfe der „Anwendungsverwaltung“ können einem Benutzer iSeries Navigator-Funktionen, Client Access-Funktionen und i5/OS-Dienstfunktionen gewährt bzw. entzogen werden.



Zugriffsberechtigungen auf iSeries Navigator Funktionen

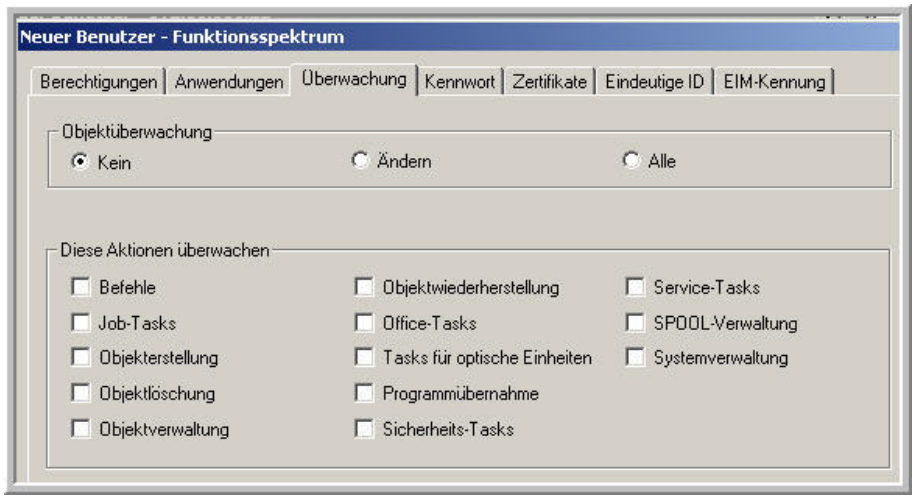
2.4.5.1.3
Seite 2

Oben sehen Sie die Darstellung aller iSeries Navigator-Funktionen. Hinter jeder Funktion steht ein Häkchen, welches bestimmt, ob der Benutzer die entsprechende Funktion nutzen darf. Entfernen Sie das Häkchen, so wird dem Benutzer die Nutzung der betreffenden Funktion entzogen – auch dann, wenn der Navigator in vollem Umfang auf dem entsprechenden PC installiert wurde. Diese Anwendungsverwaltung existiert auch für alle Client Access-Funktionen und i5/OS-Dienste. Sinnvoll ist im Bereich der Host-Anwendungen insbesondere die Möglichkeit, die Nutzung des FTP-Servers und -Clients im i5/OS einzuschränken.



Die Host-Anwendungen einschränken

Das Register „Überwachung“ entspricht dem CL-Befehl CHGUSRAUD. In diesem Register bestimmen Sie, welche Aktionen des Benutzers überwacht werden sollen. Die Aktivierung des Sicherheitsprotokolls ist meistens nur für einen kurzen Zeitraum erforderlich und sollte sehr sorgfältig geplant werden.



Benutzerüberwachung aktivieren

Im nächsten Register finden Sie wieder typische, den Kennwortverfall (PWD-EXPITV) steuernde Parameterwerte.

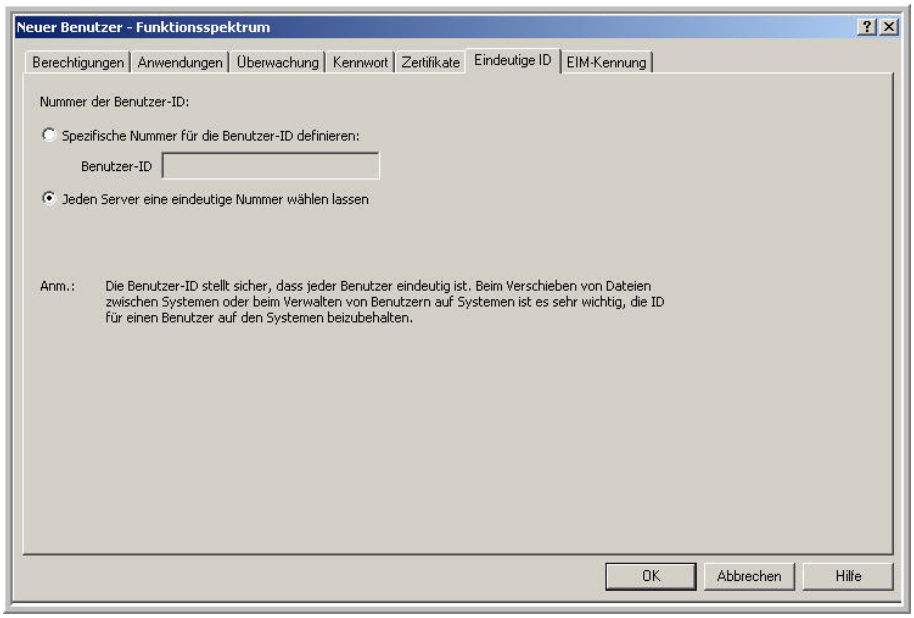
Kennworteinstellungen

Seit dem Release V5R4 können Sie entscheiden, ob die Kennwortverwaltung lokal auf dem iSeries System oder auf einer anderen Plattform verwaltet wird. Die Option entspricht dem Parameter LCLPWDMGT der Befehle CRTUSRPRF und CHGUSRPRF. Bei der Auswahl „Kennwort über eine andere Plattform verwalten“ wird das lokale i5/OS-Kennwort auf den Wert *NONE gesetzt. Der Benutzer ist dann nicht mehr in der Lage, sein eigenes Kennwort mit dem Befehl CHGPWD (Kennwort ändern) zu ändern. Außerdem kann er sich nicht direkt am System anmelden. Sie sollten diesen Wert nur verwenden, wenn der Benutzer ausschließlich über eine fremde Plattform wie Windows auf das System zugreifen muss.

2.4.5.1.3
Seite 4

Im Dialog „Funktionsspektrum – Zertifikate“ können Sie bestehende Zertifikate verwalten, die an einen Benutzer ausgegeben wurden. Ein Zertifikat bindet einen allgemeinen Schlüssel an den Benutzer, dem das Zertifikat gehört und ermöglicht somit die Authentifizierung des Zertifikateigners. Die Liste mit den Zertifikaten zeigt den Herausgeber des Zertifikats, die Seriennummer des Zertifikats und das Verfallsdatum des Zertifikats.

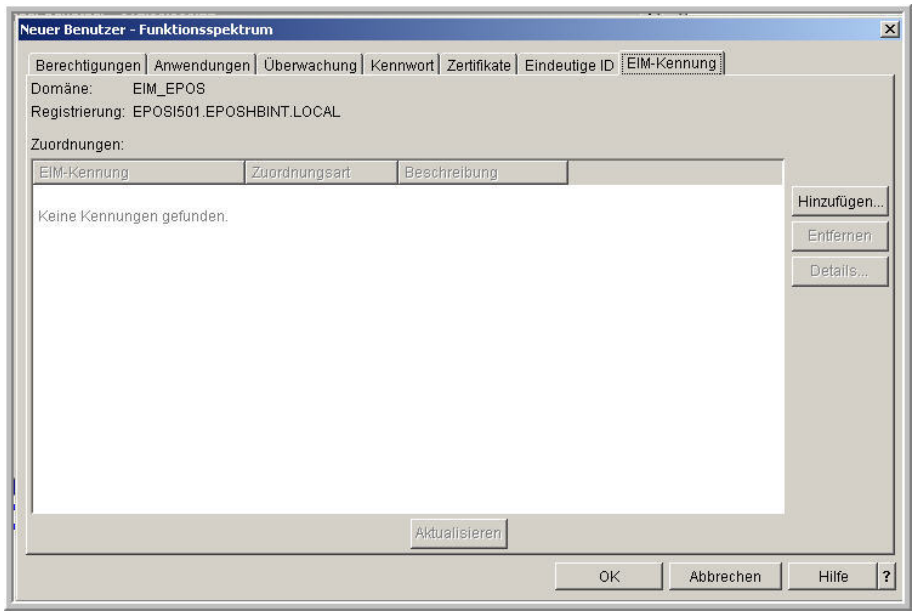
Das System verwendet eine interne eindeutige Benutzer-ID, um ein Benutzerprofil innerhalb eines Netzwerkes zu identifizieren und seine Berechtigungen zu überprüfen.



Die Benutzer-ID

In den meisten Fällen ist es am besten, wenn das System die Benutzer-ID für den neuen Benutzer wählt. Wenn Ihr iSeries-Server jedoch in ein Netzwerk integriert ist, müssen Sie möglicherweise selber die Werte für die Nummer der Benutzer-ID zuordnen, die mit denen übereinstimmt, die bereits auf den anderen Servern zugeordnet wurde. Nur so können Sie sicherstellen, dass die Benutzer-ID dieses Profils mit den anderen Systemen im Netzwerk übereinstimmt. Verwenden Sie in diesen Fällen einen Wert zwischen 1 und 4.294.967.294. Wenn Sie mit Management Central einen neuen Benutzer zu einem oder zu mehreren Endpunktsystemen hinzufügen, können Sie das zentrale System anhand des Inventars auf allen ausgewählten Systemen eine eindeutige Nummer festlegen lassen.

Der Dialog „EIM-Kennung“ ist nur für EIM-Benutzer gültig. Ein EIM-Benutzer ist Mitglied einer vordefinierten LDAP-Benutzergruppe (LDAP = Lightweight Directory Access Protocol) für eine bestimmte Domäne. Die in der LDAP-Benutzergruppe hinterlegten EIM-Zugriffsrechte bestimmen, welche Dienste der Benutzer innerhalb einer bestimmten Domäne nutzen darf. Damit aber auch Ihr iSeries-Server den Benutzer autorisieren kann, muss eine Verbindung zwischen dem iSeries Benutzerprofil und der entsprechenden EIM-Kennung hergestellt werden. Dies geschieht mit dem Dialog „EIM-Kennung“.



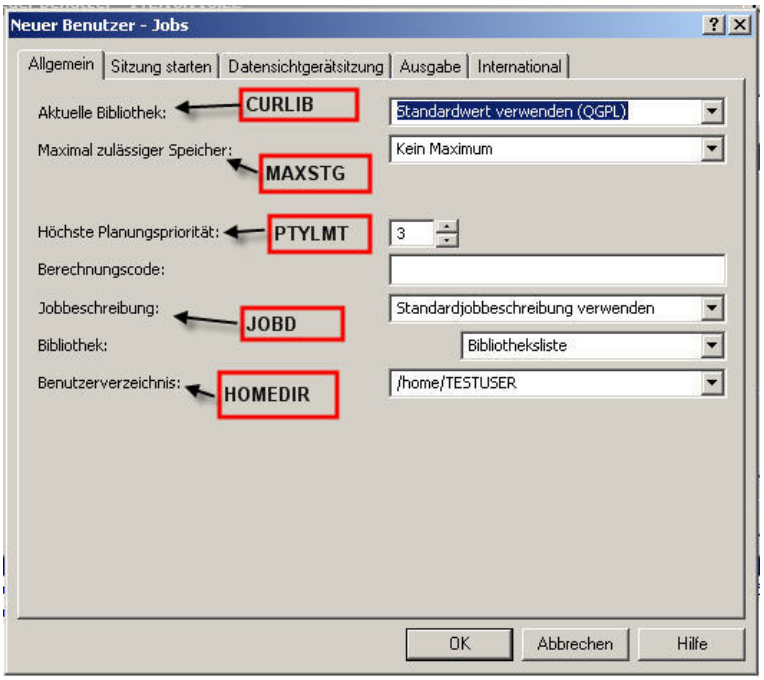
Zuordnung der EIM-Kennungen

2.4.5.1.3
Seite 6

Der Dialog „Job“

2.4.5.1.4
Seite 1

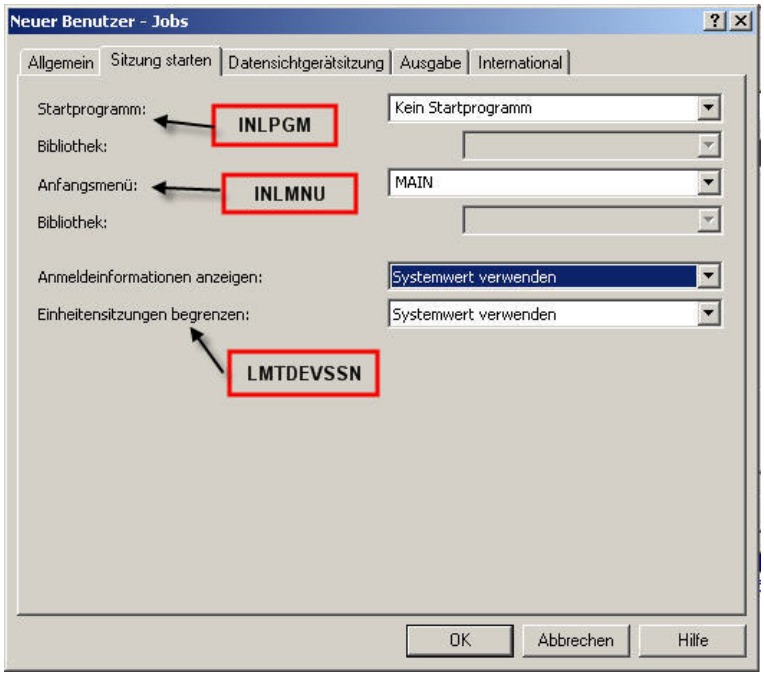
Klicken Sie auf die Schaltfläche „Jobs“, um die Seite mit den Eigenschaften für Jobs anzuzeigen. Hier geben Sie die Joboptionen des Benutzers an. Die Parameter kennen Sie bereits aus dem Kapitel 2.4.4.1 und sollen daher an dieser Stelle nicht erneut erläutert werden. Zur Übersicht zeige ich Ihnen im Folgenden lediglich die entsprechenden Register mit dem Verweis auf die Parameternamen der CL-Befehle CRTUSRPRF und CHGUSRPRF.



Der Dialog Jobs-Allgemein

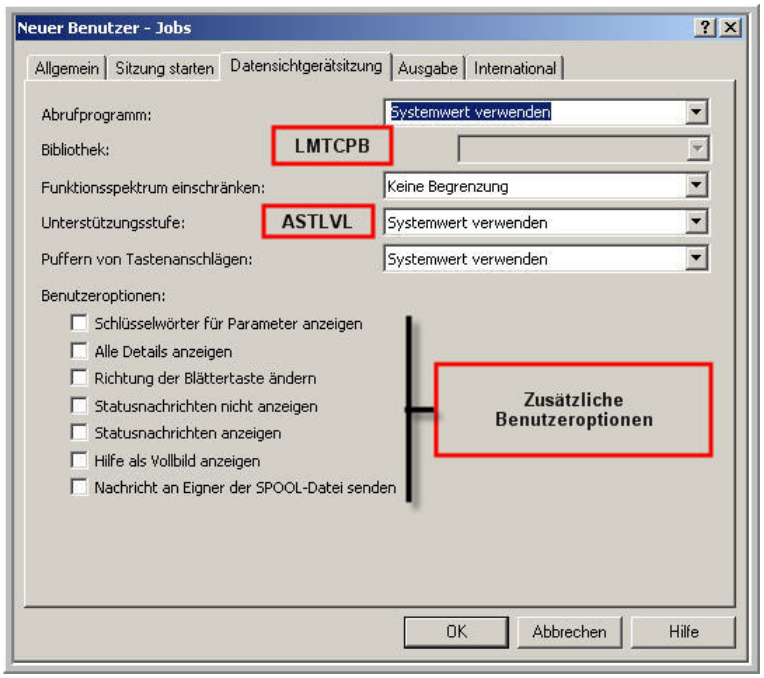
Im nächsten Register sind alle Profilparameter zusammengefasst, die die Session eines Benutzers bestimmen. Hier hinterlegen Sie das Anfangsmenü und ggf. das benötigte Startprogramm.

2.4.5.1.4
Seite 2



Der Dialog „Jobs – Sitzung starten“

Der Dialog Jobs-Funktionsspektrum beinhaltet u.a. den Parameter LMT-CPB, der das Funktionsspektrum, das dem Benutzer auf dem iSeries-Server zur Verfügung steht, nicht, teilweise oder vollständig einschränkt.

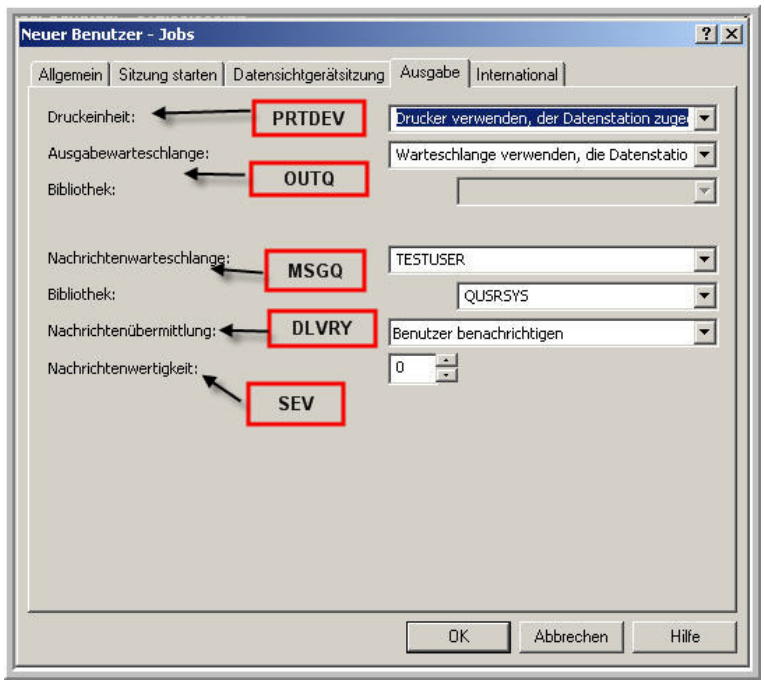


Der Dialog „Jobs – Datensichtgerätsitzung“

In der oberen Abbildung können Sie erkennen, dass dieser Dialog weitere Benutzeroptionen enthält, die den Befehlen CRTUSRPRF und CHGUSRPRF fehlen.

Wählen Sie die Option „Schlüsselwörter für Parameter anzeigen“, wenn anstelle der möglichen Parameterwerte die Schlüsselwörter für die Parameter angezeigt werden sollen. Die Option „Richtung der Blättertaste ändern“ bestimmt die Wirkungsweise der Bild-Auf und Bild-Up-Tasten, d.h. die Wirkungsweise der Taste zum Vorblättern und der Taste zum Zurückblättern werden umgekehrt. Falls Sie verhindern wollen, dass der Benutzer Meldungen in der Statuszeile erhält, müssen Sie die Option „Statusnachrichten nicht anzeigen“ aktivieren; andernfalls wählen Sie die Option „Statusnachrichten anzeigen“. Durch Auswahl der Option „Hilfe als Vollbild anzeigen“ werden stets alle Hilfetexte im Vollbildschirm und nicht in einem Fenster angezeigt. Wenn die Option „Nachricht an Eigner der SPOOL-Datei senden“ gewählt wird, erhält der Eigner der Spooldatei eine Nachricht, wenn ein Druckausgabeprogramm eine Spooldatei erstellt.

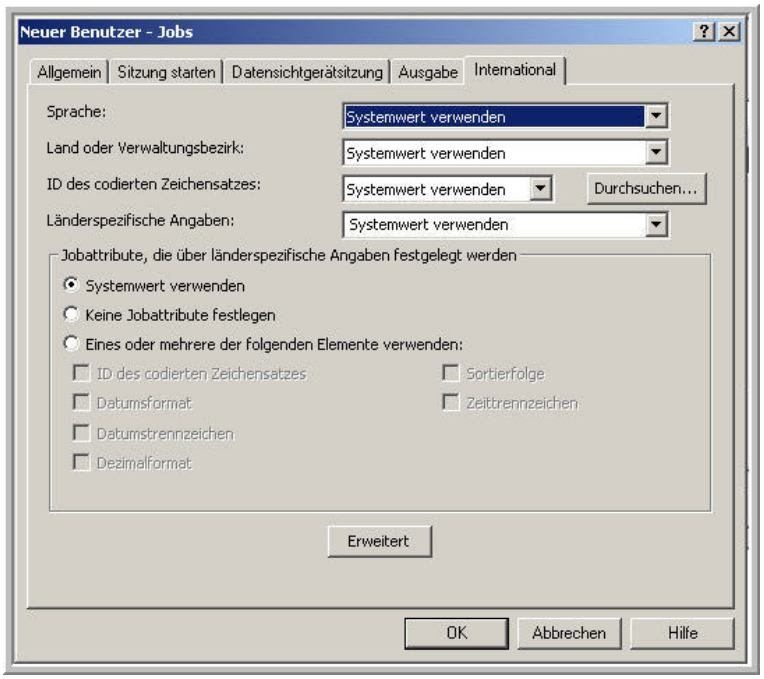
Im Dialog JOBS-Ausgabe ordnen Sie dem Benutzer einen spezifischen Drucker zu, der stets als Standarddrucker verwendet wird, wenn innerhalb des entsprechenden Jobs keine anderen Einstellungen vorgenommen wurden. An dieser Stelle wird auch die Nachrichtenwarteschlange zugeordnet.



Der Dialog „Jobs – Ausgabe“

2.4.5.1.4
Seite 4

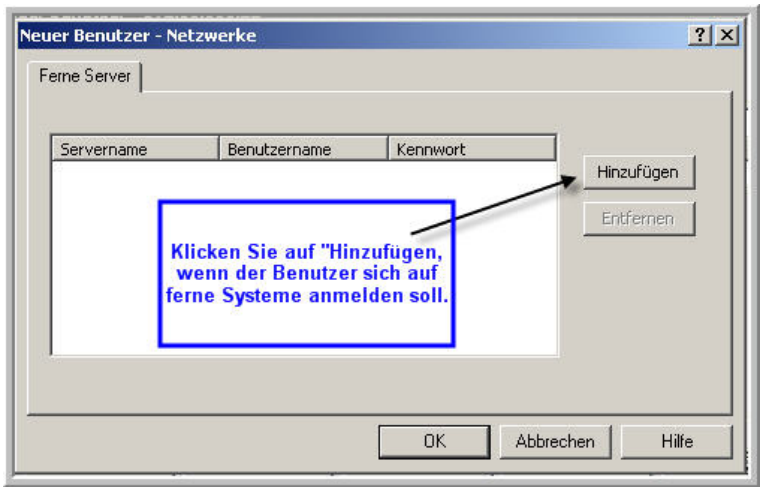
Als letztes enthält der Dialog „Jobs – International“ die Profileinstellungen für die Sprache, Sortierfolgen und sonstige landestypischen Einstellungen.



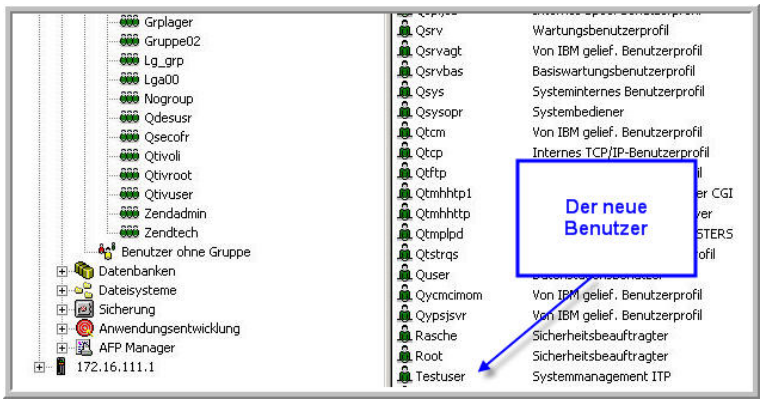
Der Dialog „Jobs-International“

Der Dialog „Netzwerke“

Im Dialog „Netzwerke“ finden Sie eine Liste ferner Server. Diese Liste enthält den Namen des fernen Servers. Jeder Servereintrag muss eindeutig sein. Zusätzlich wird der Benutzername für den Server und das zugehörige Kennwort erfasst. Falls der Benutzer auf dem Server bereits existiert müssen Sie seine eindeutige Benutzer-ID verwenden. Sie können die Liste beliebig sortieren, indem Sie auf die entsprechende Überschriftenzeile klicken.



Wenn Sie alle Benutzereinstellungen erfasst haben, klicken Sie auf „OK“. Anschließend erscheint das Profil in Ihrer Benutzerliste.



Das neue Benutzerprofil

2.4.5.1.5
Seite 2

Benutzer verwalten

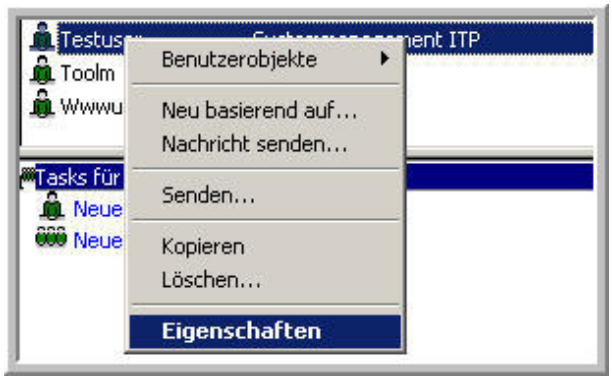
Natürlich können Sie mit dem iSeries Navigator auch Ihre bestehenden Benutzerprofile verwalten, d.h. Benutzerprofile können verändert, kopiert oder auch gelöscht werden. Doch der iSeries Navigator kann mehr! Sie können Ihre Benutzergruppen sehr einfach und komfortabel verwalten und es ist sogar möglich, bestehende Benutzerprofile an andere iSeries-Server zu senden.

2.4.5.2

Seite 2

Benutzer ändern

Wenn Sie einen Benutzer ändern wollen, erweitern Sie zunächst wieder im iSeries Navigator das entsprechende System in Ihrer Liste der Verbindungen. Erweitern Sie dann den Eintrag „Benutzer und Gruppen“, und klicken auf „Alle Benutzer“, um alle Benutzer im System anzuzeigen. Wenn Sie jetzt mit der rechten Maustaste auf den zu bearbeitenden Benutzer klicken, können Sie anschließend die Eigenschaften auswählen und entsprechende Veränderungen vornehmen.



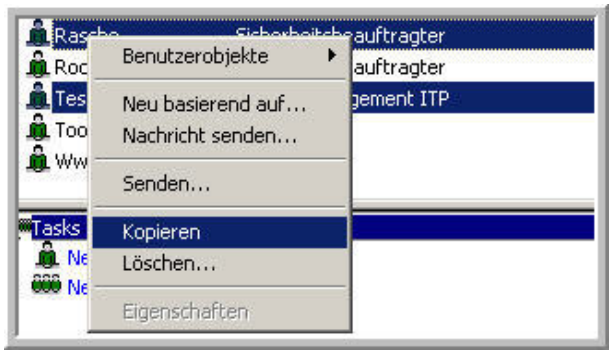
Die Benutzereigenschaften

Der anschließend erscheinende Dialog ist identisch mit der Neuanlage von Benutzern, so dass er hier nicht erneut gezeigt werden muss. Der Weg über die Eigenschaften ist die Standardaktion. Sie können aber auch einfach auf den Benutzer doppelklicken, den Sie ändern möchten. Auch dann erscheint der Dialog zur Änderung der Benutzereigenschaften.

2.4.5.2.1
Seite 2

Benutzer kopieren

Wenn Sie einen oder mehrere bestehende Benutzer kopieren wollen, um einen neuen Benutzer mit denselben Eigenschaften auf einem anderen System oder auf demselben System unter einem anderen Namen zu erstellen, selektieren Sie hierfür den oder die zu kopierenden Benutzer und klicken Sie wiederum mit der rechten Maustaste auf einen der Benutzer.



Benutzer kopieren

Sie sehen, dass ich mehrere Benutzer markiert habe und dann die Funktion „Kopieren“ ausgewählt habe. Wenn der Benutzer auf ein anderes Zielsystem erstellt werden soll, müssen Sie jetzt das Zielsystem in Ihrer Liste der Verbindungen erweitern. Klicken Sie anschließend mit der rechten Maustaste auf „Alle Benutzer“, „Benutzer ohne Gruppe“ oder auf eine bestimmte Gruppe, und wählen Sie den Eintrag „Einfügen“ aus, um den Benutzer zu kopieren.

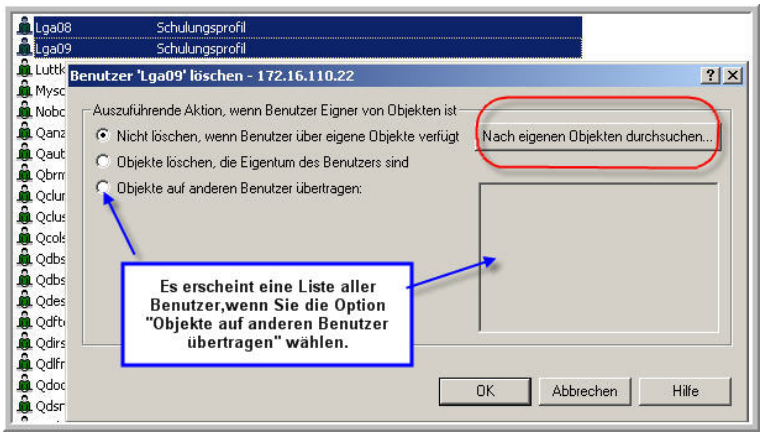
Benutzer einfügen

Danach öffnet sich der Dialog „Neuer Benutzer“. Sie müssen jetzt nur noch die leeren Felder ausfüllen und ein neues Kennwort angeben. Für weitere Änderungen klicken Sie auf die entsprechenden Schaltflächen. Mit „OK“ erstellen Sie schließlich die Benutzerkopie. Der oben geschilderte Weg ist das Standardvorgehen. Sie können einen Benutzer auch einfach durch Drag & Drop in eine andere Gruppe oder auf ein anderes System kopieren.

2.4.5.2.2
Seite 2

Benutzer löschen

Ähnlich gehen Sie vor, wenn Sie ein oder mehrere Benutzerprofile löschen wollen. Wählen Sie den oder die Benutzer aus, der bzw. die gelöscht werden soll(en). Klicken Sie auch jetzt wieder mit der rechten Maustaste auf einen ausgewählten Benutzer, und wählen Sie „Löschen“. Anschließend öffnet sich der folgende Dialog:



Der Dialog „Benutzer löschen“

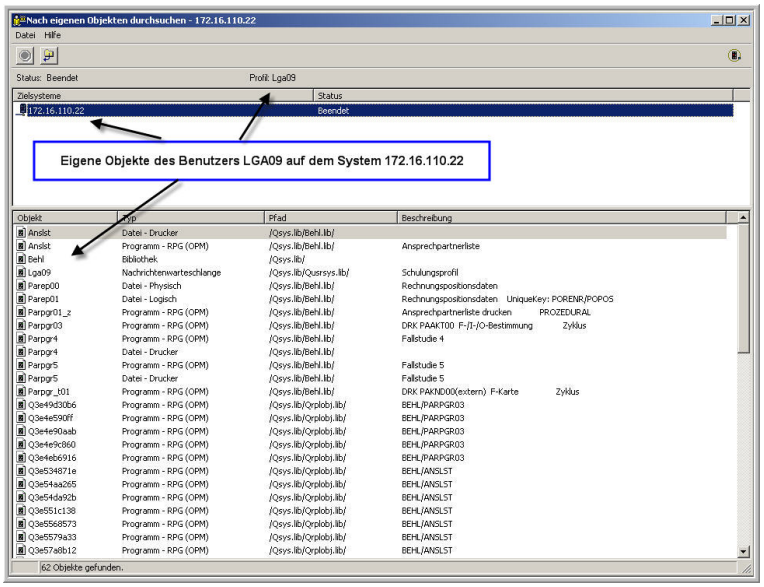
Sie sehen, dass der Benutzer nicht gelöscht wird, wenn er Eigentümer von Objekten ist. Dies entspricht der Standardeinstellung des 5250-Dialoges.

Falls Sie prüfen wollen, welche Objekte dem Benutzer gehören, klicken Sie auf „Nach eigenen Objekten durchsuchen“, um die Objekte anzuzeigen, die dem Benutzer auf dem System gehören.

Suche nach eigenen Objekten

Die Suchergebnisse werden anschließend für jeden Benutzer in separaten Fenstern dargestellt.

2.4.5.2.3
Seite 2



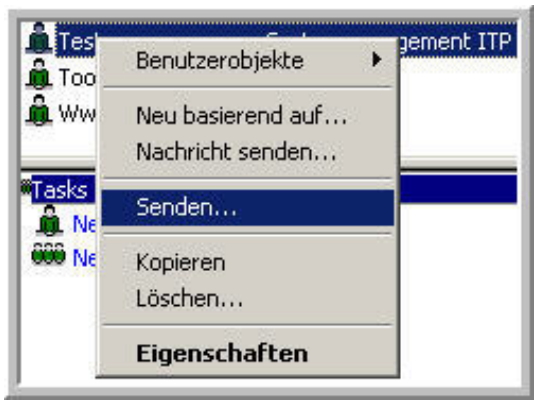
Das Suchergebnis für den Benutzer LGA09

Ebenso wie auf dem 5250-Interface können Sie die Objekte gemeinsam mit dem Profil löschen oder die Objekte auf einen neuen Eigner übertragen. Falls Sie das Eigentumsrecht der Objekte auf einen anderen Benutzer übertragen möchten, wählen Sie den gewünschten Benutzernamen aus der Liste der verfügbaren Benutzer aus. Die Objekte werden anschließend auf den neuen Benutzer übertragen, und zwar unabhängig von der aktuellen Berechtigung, die der Benutzer für die Objekte besitzt.

Benutzer an entfernte Systeme senden

Sie können einen oder mehrere Benutzer an ein anderes System oder an alle Systeme einer Systemverwaltungsgruppe senden. Dabei werden so viele Benutzereigenschaften wie möglich auf das neue System kopiert. Die Informationen, die für jeden Benutzer gesendet werden, enthalten das Benutzerprofil, die persönlichen Berechtigungen, evtl. IDs für Enterprise Identity Mapping (EIM) sowie die Kennwörter. Für jeden gesendeten Benutzer wird automatisch ein Eintrag im Systemverteilerverzeichnis erstellt bzw. aktualisiert. Sie können mit Ausnahme einiger systemdefinierter Benutzer jedes Profil an entfernte Systeme senden. Die Namen der systemdefinierten Benutzerprofile beginnen normalerweise mit einem Q wie QSECOFR.

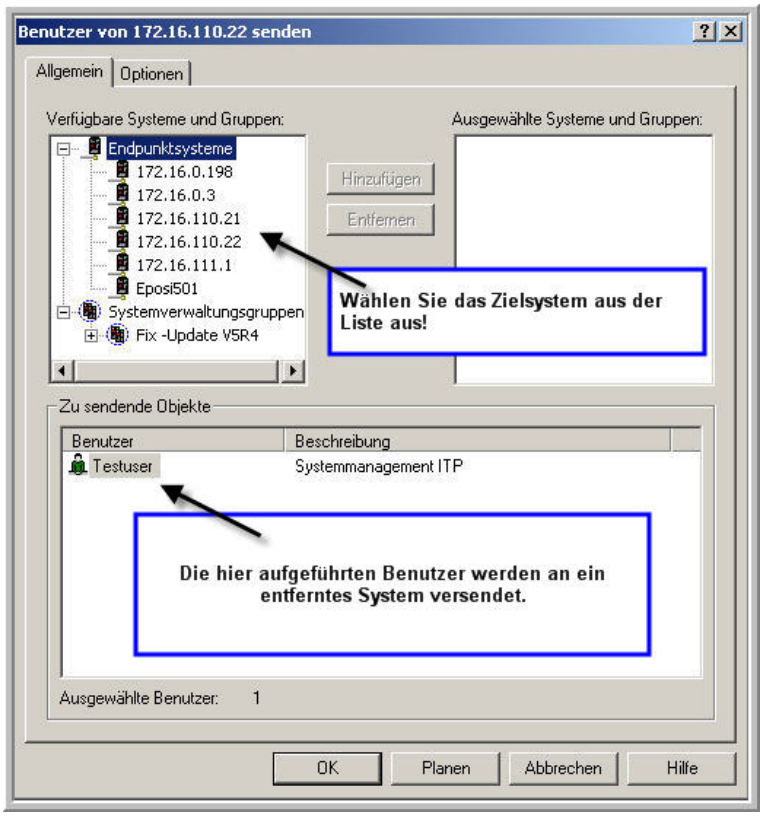
Wenn ich unseren TESTUSER an ein entferntes System senden will, muss ich wieder die Liste meiner Endpunktsysteme erweitern und den Benutzer markieren, der versendet werden soll.



Benutzer senden

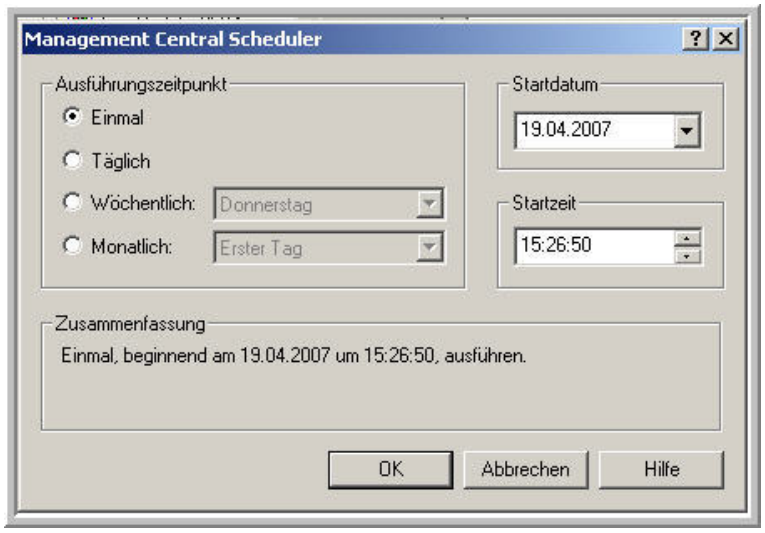
2.4.5.2.4
Seite 2

Anschließend öffnet sich ein weiterer Dialog.



Zielsysteme auswählen

Im rechten Fenster erscheint eine Liste mit Endpunktsystemen und Systemverwaltungsgruppen, an die eine Liste mit Benutzern oder Gruppen gesendet werden kann. Sie können ein System auswählen, indem Sie den entsprechenden Eintrag in der Liste der verfügbaren Systeme und Gruppen auswählen und dann auf „Hinzufügen“ klicken. Am unteren Bildschirmrand sehen Sie die Benutzer und Gruppen, die an die ausgewählten Endpunktsysteme und Systemverwaltungsgruppen gesendet werden. Klicken Sie auf den Reiter „Optionen“, um die Aktion anzugeben, die ausgeführt werden soll, wenn ein selektierter Benutzer bereits auf dem Zielsystem vorhanden ist. Sie können wählen, ob der bereits vorhandene Benutzer nicht geändert oder mit den Einstellungen des gesendeten Benutzers aktualisiert werden soll. Klicken Sie anschließend auf „OK“, um die Übertragung sofort zu starten, oder klicken Sie auf „Planen“, um festzulegen, wann der Task gestartet werden soll.



Den Sende-Task planen

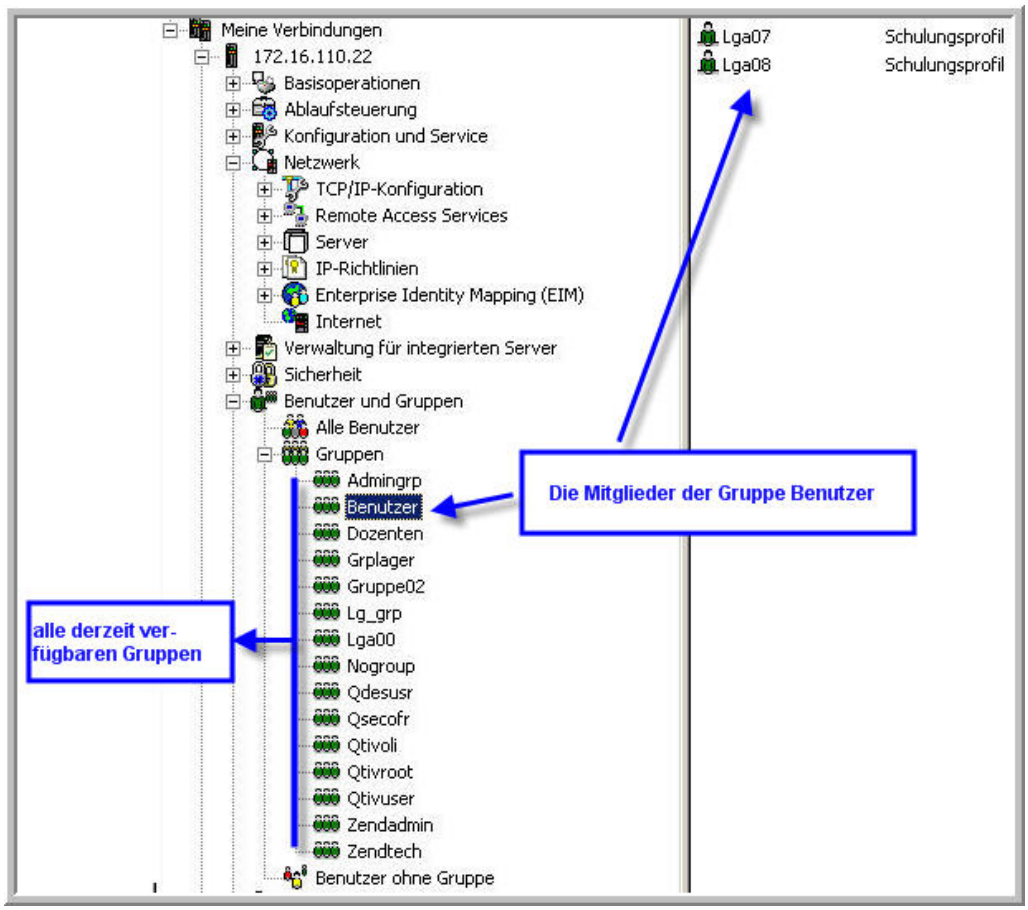
Mit dem Management Central Scheduler können Sie bestimmen, wann die Management Central-Tasks ausgeführt werden sollen. Soll der Task nur einmalig ausgeführt werden, wählen Sie die Option „Einmal“. Der Task wird ein einziges Mal ausgeführt. Der Beginn wird durch das angegebene Datum und die Uhrzeit festgelegt. Sie können die Übertragung auch täglich, wöchentlich oder monatlich ausführen. Diese Optionen sind allerdings im Zusammenhang mit Benutzerprofilen wenig sinnvoll.

2.4.5.2.4

Seite 4

Benutzergruppen

Besonders einfach gestaltet sich die Arbeit mit Benutzergruppen mit Hilfe des iSeries Navigators. Im Gegensatz zum i5/OS, wo es den Objekttypus des Gruppenprofils nicht gibt, ist hier sehr wohl eine separate Bearbeitung von Benutzerprofilen, die den Status einer Gruppe haben, möglich. Erweitern Sie einfach „Benutzer und Gruppen“. Im Verzeichniseintrag „Gruppen“ werden alle Gruppenprofile Ihres Systems übersichtlich dargestellt.



Mit Benutzergruppen arbeiten

Sobald Sie eine Gruppe markieren, sehen Sie in der Detailanzeige des iSeries Navigator die Gruppenmitglieder. Ein Doppelklick auf eines der Member-Profile ermöglicht Ihnen die Änderung des Benutzers.

2.4.5.3

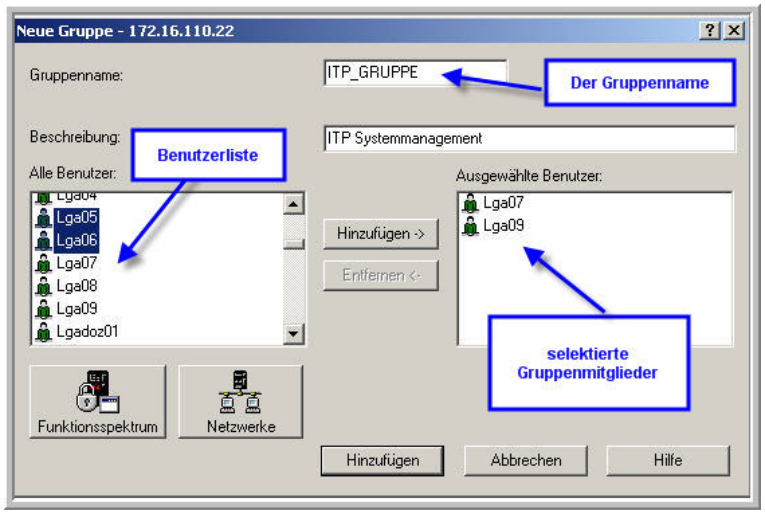
Seite 2

Neue Benutzergruppe

Interessant am iSeries Navigator ist insbesondere die Funktion „Neue Gruppe“. Dies würde einem CL-Befehl CRTGRPPRF entsprechen – nur als erfahrener iSeries Operator wissen Sie, dass IBM diesen Befehl niemals im 5250-Interface implementiert hat, aber im Navigator gibt es diese Funktion:

Eine neue Gruppe erstellen

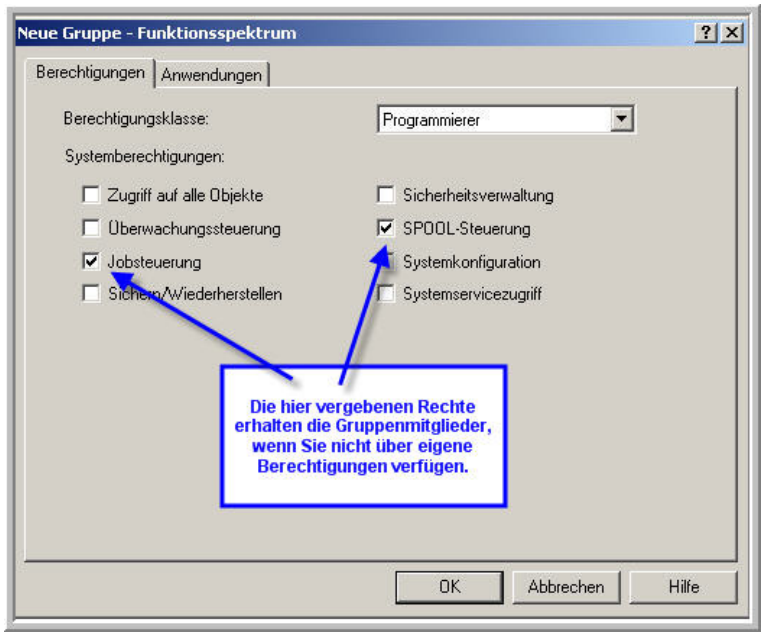
Um eine neue Gruppe zu erstellen, öffnen Sie entweder das Kontextmenü „Gruppen“ mit der rechten Maustaste und wählen „Neue Gruppe“ oder Sie klicken mit der rechten Maustaste auf den Verzeichniseintrag „Benutzer und Gruppen“. In beiden Fällen gelangen Sie in den zentralen Dialog, der es Ihnen ermöglicht, eine neue Benutzergruppe zu erstellen und dieser Gruppe Benutzerrechte und Funktionen zuzuordnen.



Gruppenmitglieder zuordnen

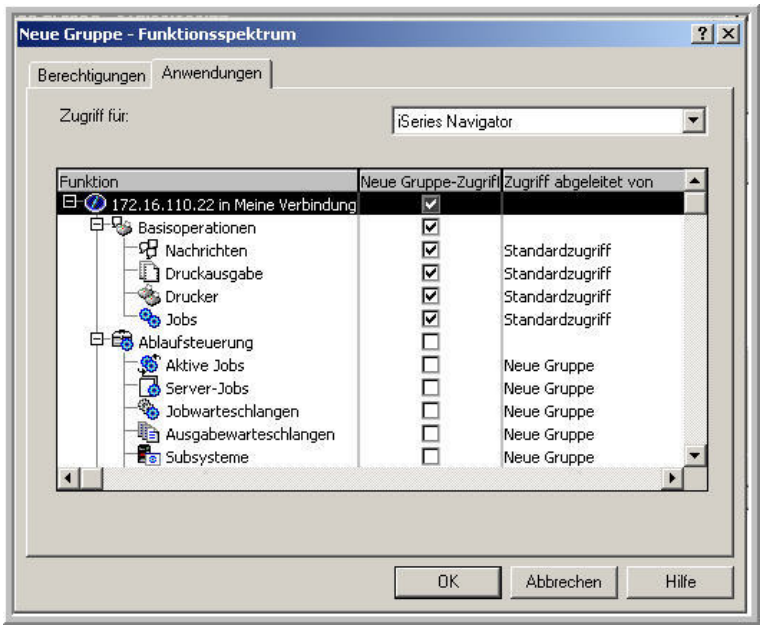
2.4.5.3.1
Seite 2

Sie sehen, dass ich der Gruppe zunächst einen sprechenden Namen gebe. In der Liste „Alle Benutzer“ finden Sie die Benutzerprofile, die der Gruppe zugeordnet werden können. Ich habe die User LGA07 und LGA09 selektiert und anschließend „Hinzufügen“ gewählt. Daraufhin erscheinen die Benutzer in der Liste der „ausgewählten Benutzer“. Falls weitere Benutzer hinzugefügt werden sollen, kann dieser Vorgang beliebig oft wiederholt werden – natürlich auch bei bereits bestehenden Gruppen. Da es möglich ist, die Sonderberechtigungen der Benutzer über die Gruppe zu steuern, steht uns jetzt noch der Dialog „Funktionsspektrum“ zur Verfügung, den Sie bereits kennen.



Die Sonderberechtigungen der Gruppe

Und auch die Anwendungsverwaltung können Sie zentral über die Gruppe steuern.



Die Anwendungsverwaltung

Wenn Sie alle Eingaben gemacht haben, klicken Sie auf „OK“. Die Gruppe ist erstellt und erscheint anschließend in der Liste Ihrer Gruppenprofile.

2.4.5.3.1

Seite 4

Benutzergruppenverwaltung

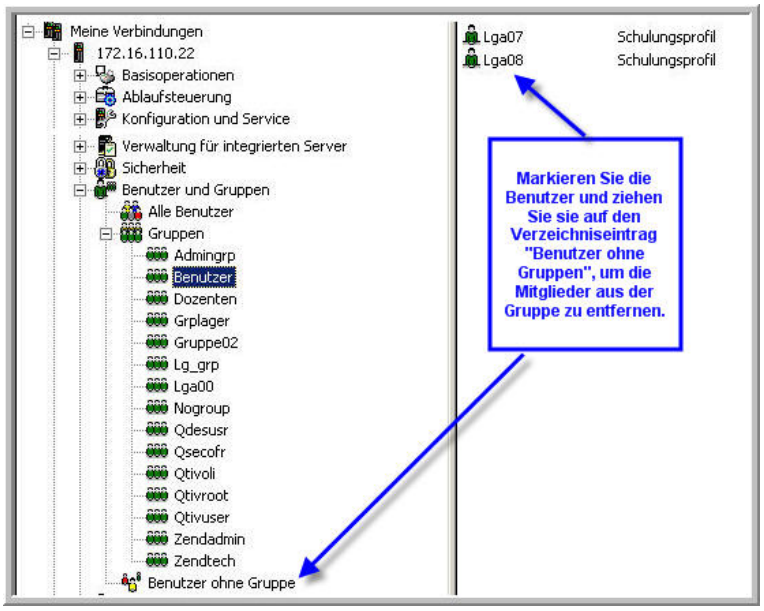
Die Verwaltung der Gruppen, d.h. das Ändern, Löschen und Kopieren ist identisch mit der Bearbeitung eines einzelnen Benutzerprofils und muss daher nicht erneut vorgestellt werden. Gruppenmitglieder können Sie wie erläutert über die Standarddialoge zuordnen und entfernen oder einfach mittels Drag & Drop zuordnen. Um Benutzerprofile einer Gruppe zuzuordnen, erweitern Sie den Eintrag „Alle Benutzer“ oder „Benutzer ohne Gruppe“, markieren die entsprechenden Benutzer und ziehen die Benutzer anschließend auf die entsprechende Gruppe. Fertig!

Ähnlich gehen Sie vor, um Benutzer aus einer Gruppe zu entfernen.

Öffnen Sie hierfür zunächst den Verzeichniseintrag „Gruppen“. In der Detailanzeige des iSeries Navigators werden die Gruppenmitglieder angezeigt. Markieren Sie die zu entfernenden Member. Mit der Kontexttaste steht Ihnen die Funktion „Aus Gruppe entfernen“ zur Verfügung.

2.4.5.3.2
Seite 2

Sie können aber auch hier die Drag & Drop-Funktion nutzen, indem Sie die Gruppenmitglieder markieren und anschließend auf den Verzeichniseintrag „Benutzer ohne Gruppen“ ziehen.



Benutzer aus einer Gruppe entfernen

Eine letzte Anmerkung zu dem Verzeichniseintrag „Benutzer ohne Gruppen“: Verwenden Sie „Benutzer ohne Gruppe“ um einzelne Benutzer, die sich in keiner Gruppe befinden, auf dem verwendeten iSeries-Server anzuzeigen und/oder um mit solchen Benutzern zu arbeiten. So kann es nicht passieren, dass Sie vergessen, einen Benutzer einer Gruppe zuzuordnen. Eine sehr sinnvolle Darstellung wie ich finde, die auch in dieser Form nicht auf dem 5250-Bildschirm existiert.

Sie sehen, die Verwaltung der Benutzerprofile im iSeries Navigator hat einiges zu bieten und besitzt viele Vorteile. Zudem ist die Bedienung denkbar einfach und komfortabel. Probieren Sie es aus!

Sicherheitsüberlegungen

Es wird viel über Computer-Sicherheit geschrieben. Was aber bedeutet Computer-Sicherheit? Ganz allgemein versteht man darunter die Einführung verschiedener Maßnahmen, die Ihre iSeries und deren Umgebung vor Sabotage, Spionage und alle Arten von zufälligen und weniger zufälligen Schäden bewahrt. Die Angst vor Maschinenausfall und Datenverlusten nutzen viele Hersteller und entwickeln geeignete Tools. Aber niemand – kein Auditor, kein Fremdanbieter – kann Ihnen sagen, welche Sicherheitsrichtlinien für Sie geeignet sind. Denn um entsprechende Maßnahmen zu ergreifen, müssen Sie zunächst einmal Ihre individuellen Risiken analysieren. Sie entscheiden, welche Systeme, welche Daten und welche Prozesse von Bedeutung für Ihr Unternehmen sind und daher in besonderem Maße geschützt werden sollten und welche Risiken Sie bereit sind zu akzeptieren. Erst dann sind Sie in der Lage, Ihre spezifischen Security-Richtlinien im Unternehmen zu implementieren. Auf einer iSeries können Sie u.a. Benutzerprofile, Objektberechtigungen, Objektsignierung, Auditjournale u.v.m. verwenden, um Ihr System zu schützen. Doch leider werden diese Bordmittel wenig oder gar nicht genutzt. In den folgenden Beiträgen soll daher die Bedeutung dieser Features für Ihre Sicherheitsstrategie erörtert werden.

Einige werden sich jetzt fragen, was soll das? Die iSeries war und ist eines der sichersten und zuverlässigsten Systeme am Markt, denn von Beginn an wurde bei der Entwicklung des Betriebssystems der Fokus auf integrierte Sicherheitsmechanismen gelegt. Das Betriebssystem i5/OS bietet eine integrierte Architektur kombiniert mit hoher Verfügbarkeit und einfach zu handhabenden Sicherheitsfeatures basierend auf der Mainframe-Technologie. Am wichtigsten ist hierbei sicherlich die Objekt-basierte Architektur des Servers. Nach wie vor ist es nicht möglich ein Datenobjekt in einen ausführbaren Code zu wandeln und umgekehrt. Warum dann aber die scheinbar nicht enden wollende Diskussion um das Thema Sicherheit?

Nun, im Gegensatz zu früher sind die iSeries-Server heute Bestandteil eines heterogenen Netzes und stehen nicht mehr isoliert. Noch vor einigen Jahren liefen die Applikationen ausschließlich auf der iSeries. Heute erfolgen die Zugriffe aus Web-Applikationen, über FTP-, ODBC- und JDBC-Schnittstellen. Die Zeiten haben sich geändert. Hat sich Ihre Sicherheitsstrategie auch entsprechend weiterentwickelt? Eine gute Strategie muss neben der eigentlichen Anwendungssicherheit insbesondere die Möglichkeiten des Betriebssystems und des Netzwerks nutzen, um die Verfügbarkeit des Servers und dessen Daten zu gewährleisten.

2.5
Seite 2

Verwendung von Benutzergruppen

Eine besondere Bedeutung in punkto Sicherheit haben die Benutzerprofile. Im Folgenden sollen aber nicht erneut die einzelnen Profilparameter detailliert besprochen werden. Dies ist bereits im Kapitel 2.4.4 geschehen. Vielmehr geht es jetzt um den Einsatz und die Verwaltung von Benutzergruppen.

Gehen wir von folgendem Szenario aus: Mehrere Entwickler arbeiten in verschiedenen Projekten. Ohne den Einsatz von Gruppenprofilen könnte sehr schnell die folgende Situation auftreten: Der Programmierer A, nennen wir ihn „Mueller“, kompiliert ein Programm. Er wird Eigentümer des ausführbaren Programmobjekts.

```
Objekt . . . . . :   STRQUERY        Eigner . . . . . :   MUELLER
  Bibliothek . . . :     EPS002CR    Primärgruppe . . . :   *NONE
Objektart  . . . . :   *PGM          ASP-Einheit  . . . :   *SYSBAS

Aktuelle Berechtigungen ändern und Eingabetaste drücken.

  Objekt durch Berechtigungsliste geschützt . . . . . .   *NONE

                        Objekt-
Benutzer    Gruppe      berechtg.
*PUBLIC                 *CHANGE
MUELLER                 *ALL
```

Die Eignerberechtigungen

Der Objekteigner erhält stets alle Berechtigungen am Objekt, während alle anderen Personengruppen aufgrund der öffentlichen Berechtigungseinstellungen auf das Objekt zugreifen.

In unserem Beispiel bedeutet dies, dass jeder das Objekt nutzen aber nicht löschen darf. Will nun ein anderes Projektmitglied das Programmobjekt modifizieren, sendet das Betriebssystem folgende Fehlermeldung:

```
3>> CRTCLPGM PGM(EPS002CR/STRQUERY) SRCFILE(EPS002CR/QCLSRC)
    Nicht berechtigt für STRQUERY in Bibliothek EPS002CR Art *PGM.
    Programm STRQUERY nicht erstellt.
```

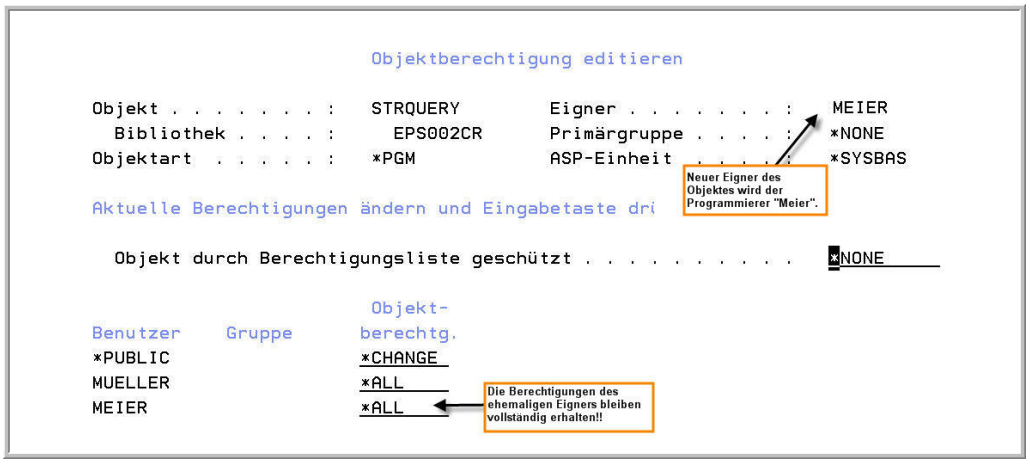
Unberechtigter Objektzugriff

Sie könnten selbstverständlich dem entsprechenden Programmierer die persönlichen Berechtigungen am Programmobjekt erteilen. Nur wer will für jedes Objekt und jeden Programmierer persönliche Berechtigungen verwalten?

2.5.1
Seite 2

Also lautet die Lösung häufig: *ALLOBJ-Berechtigung. Durch die Sonderberechtigung *ALLOBJ setzen Sie aber den Objektschutz des Betriebssystems außer Kraft. Genauso gut könnten Sie die Berechtigungsstufe von 30 auf 20 ändern. Sie müssen sich darüber im Klaren sein, dass Sie den Objektschutz für ALLE Objekte aufheben. Ihre Programmierer erhalten somit uneingeschränkten Zugriff auf jedes Objekt, auch auf alle Tabellen und die sich darin befindenden Daten. Ist das wirklich gewollt?

Und es passiert noch etwas:



Die Eignerberechtigungen

Anhand der Abbildung erkennen Sie, dass zwar der Eigentümer wechselt, nur bleiben die alten Eignerberechtigungen des Programmierers „Mueller“ erhalten. Er hat weiterhin vollen Zugriff auf das Objekt unabhängig davon, ob dieser Programmierer noch im Projekt arbeitet oder nicht. Stellen Sie sich vor, mehr als zwei Programmierer sind beteiligt und kompilieren die entsprechenden Programme. Sie haben keinen Überblick darüber, wer welche persönlichen Berechtigungen an den Objekten hat. Die Lösung für dieses Problem ist denkbar einfach:

Durch den Einsatz von Gruppenprofilen können Sie mit minimalem Verwaltungsaufwand den Zugriff auf die jeweiligen Objekte steuern. Gruppenprofile sind eine spezielle Ausprägung von Benutzerprofilen, d.h. Sie können mehrere Profile zu einer Gruppe zusammenfassen, um gemeinsame Berechtigungen für diese Personengruppe festzulegen. Gruppenprofile sind immer dann sinnvoll, wenn ein bestimmter Personenkreis, z. B. Ihre Programmierer identische Rechte und gemeinsamen Zugriff auf bestimmte Ressourcen benötigt.

Planen wir einfach mal ein Gruppenprofil für unsere Entwickler, die im Projekt „A07“ tätig sind. Zunächst müssen wir die Gruppe in Form eines Profils definieren.

```
                    Benutzerprofil erstellen (CRTUSRPRF)

Auswahl eingeben und Eingabetaste drücken.

Benutzerprofil . . . . . .   GP_PJ_A07      Name
Benutzerkennwort . . . . . .  *NONE         Zeichenwert, *USRPRF, *NONE
Benutzerklasse . . . . . .    *PGMR         *USER, *SYSOPR, *PGMR...
Sonderberechtigung . . . . . .  *JOBCTL     *USRCLS, *NONE, *ALLOBJ...
            + für weitere Werte
```

Die Gruppe

Hierbei sollten Sie folgende Regeln beachten:

- Vereinbaren Sie Namenskonventionen für Ihre Gruppenprofile. Dadurch fällt es Ihnen später leichter, die Gruppen zu identifizieren. Ich verwende in meinem Beispiel die Projektbezeichnung „GP_PJ_A07“ als Gruppennamen, wobei das Prefix „GP_PJ“ anzeigt, dass es sich um ein Gruppenprofil für den Projektbereich handelt.
- Beachten Sie, dass eine Anmeldung unter dem Gruppennamen nie erforderlich ist und daher von vornherein unterbunden wird, indem das Kennwort auf den Wert *NONE gesetzt ist (PASSWORD is *NONE).
- Als letztes vergeben Sie die erforderlichen Sonderberechtigungen für die Gruppe. Die Gruppenmitglieder übernehmen die Berechtigungen der Gruppe, solange Sie keine persönlichen Berechtigungen erteilen.

In einem zweiten Schritt kann ich die Entwickler dem Projekt „A07“ zuordnen, indem ich Sie zu Membern der Gruppe „GP_PJ_A07“ mache.

```
Auswahl eingeben und Eingabetaste drücken.

Benutzerprofil . . . . . > MEIER          Name
Benutzerkennwort . . . . . .  *SAME       Zeichenwert, *SAME, *NONE
Kennwort auf abgelaufen setzen  *NO       *SAME, *NO, *YES
Status . . . . . .  *ENABLED              *SAME, *ENABLED, *DISABLED
Benutzerklasse . . . . . .  *PGMR         *SAME, *USER, *SYSOPR...

                Zusätzliche Parameter
Sonderberechtigung . . . . . .  *NONE      <-- Die Member erhalten keine persönlichen Sonderberechtigungen.
     + für weitere Werte
Gruppenprofil . . . . . .  GP_PJ_A07      <-- Der Parameter GRPPRF bestimmt die erste Gruppe.
Eigner . . . . . .  *GRPPRF               <-- Eigner zukünftiger Objekte wird die Gruppe.
```

Die Gruppenmember

2.5.1
Seite 4

Das erste Gruppenprofil wird über den Parameter GRPPRF im Benutzerprofil spezifiziert. Weitere Gruppen benennen Sie im Parameter SUPGRPPRF. Ein Benutzer kann insgesamt Mitglied in 16 unterschiedlichen Gruppen sein. Planen Sie die Gruppen und die Gruppenzugehörigkeiten aber vorsichtig. Die erste Gruppe (Parameter GRPPRF) sollte stets die Gruppe sein, deren Objekte der User am häufigsten verwendet. Wenn Sie dennoch mehrere Gruppen für einen User einsetzen müssen, beachten Sie das Verfahren, das die iSeries verwendet, um die Zugriffsberechtigung für ein Objekt zu prüfen.

Außerdem übertrage ich die Eignerschaft neuer Objekte an die Gruppe „GP_PJ_A07“. Dies kann ich mit dem Parameter OWNER festlegen. Wenn die Gruppe Eigentümer wird (OWNER is *GRPPRF), erhalten Gruppenmitglieder, die Objekte erzeugen, keine spezifischen Berechtigungen am Objekt. Ihre Zugriffsberechtigung wird nur durch die Gruppenzugehörigkeit bestimmt.

```
                        Objektberechtigung editieren

Objekt . . . . . :   STRQUERY        Eigner . . . . . :   GP_PJ_A07
  Bibliothek . . . :     EPS002CR    Primärgruppe . . . :   *NONE
Objektart  . . . . :   *PGM          ASP-Einheit  . . . :   *SYSBAS

Aktuelle Berechtigungen ändern und Eingabetaste drücken.

  Objekt durch Berechtigungsliste geschützt . . . . . .   *NONE

                        Objekt-
Benutzer    Gruppe      berechtg.
*PUBLIC                 *CHANGE
GP_PJ_A07               *ALL
```

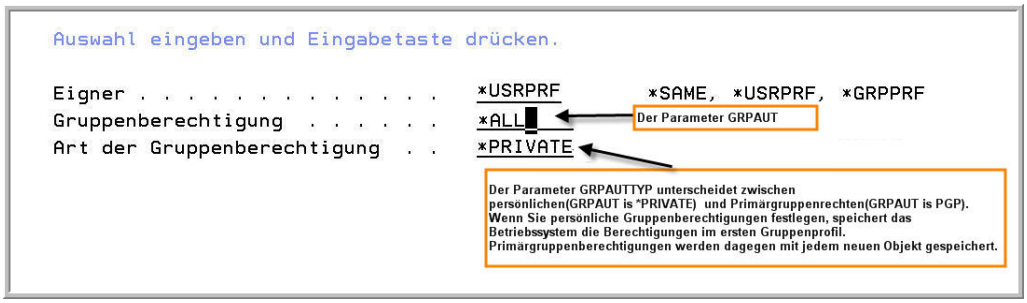
Die Eignerberechtigung

In Entwicklungs- oder Testumgebungen kann es sinnvoll sein, die Eignerschaft an die Gruppe zu übertragen (OWNER is *GRPPRF) wie in unserem Beispiel. Bedenken Sie aber, dass sich bestehende Objekteignerschaften nicht automatisch anpassen. Die Eignerschaft bestehender Objekte müssen Sie manuell mit dem Befehl CHGOBJOWN verändern.

```
                    Objekteigner ändern (CHGOBJOWN)

Auswahl eingeben und Eingabetaste drücken.

Objekt . . . . . .   strquery    Name
  Bibliothek . . . .   eps002cr  Name, *LIBL, *CURLIB
Objektart  . . . . .   *pgm      *ALRTBL, *AUTL, *BNDDIR...
ASP-Einheit  . . . .   *         Name, *, *SYSBAS
Neuer Eigner . . . .   GP_PJ_A07   Neuer Eigentümer des Objektes wird die Gruppe
                                   "GP_PJ_A07"
Aktuelle Eignerberechtigung  . .   *REVOKE   Aktuelle Eignerberechtigungen werden entzogen.
```

Bestehende Objekteignerschaften an die Gruppe übertragen

Falls die Gruppe nicht Eigentümer der Projektobjekte werden soll, gibt es eine andere Möglichkeit, die Gruppe zu berechtigen.



Persönliche und Primärgruppenrechte

In diesem Beispiel behält der Entwickler zwar die Objekteignerschaft, die Gruppenberechtigung wird aber zusätzlich durch die Parameter „GRPAUT“ und „GRPAUTTYP“ bestimmt, d.h. jedes Gruppenmitglied kann uneingeschränkt aufgrund seiner Gruppenzugehörigkeit und der Gruppenberechtigung *ALL auf das Programm zugreifen. Selbst wenn der Eigner des Objektes einer anderen Gruppe angehört, bleibt die Gruppenberechtigung bestehen und jeder, der zum Zeitpunkt des Zugriffs Mitglied der Gruppe ist, darf das Programmobjekt uneingeschränkt nutzen.

Und hier zeigt sich auch der Vorteil von Gruppenprofilen: Wenn neue Projektmitarbeiter hinzukommen, Mitarbeiter das Unternehmen verlassen oder die Projektzugehörigkeit wechselt, müssen Sie keine persönlichen Objektberechtigungen mehr erteilen, sondern nur die Gruppenzugehörigkeit durch Austausch der Gruppenprofilnamen ändern. Zusätzliche Sonderberechtigungen erteilen oder entziehen Sie der Gruppe anstatt einzelnen Benutzern. Dieses Vorgehen spart Zeit und ist sehr viel übersichtlicher als die Steuerung am einzelnen Profil.

Doch leider bietet die 5250-Oberfläche keine geeigneten Befehle, um schnell und einfach Gruppen und deren Mitglieder zu verwalten. Dieses Problem behebt die grafische Oberfläche des iSeries Navigator. Diese grafische Benutzeroberfläche ermöglicht einen schnellen Überblick über bestehende Gruppen und erlaubt es, mit einfachen Mitteln die Gruppenstruktur zu verändern. Leider ist es bislang nicht möglich mit iSeries Navigator Gruppen zu erstellen, die die Objekteignerschaften übernehmen. Der iSeries Navigator arbeitet grundsätzlich mit den Parametern „GRPAUT“ und „GRPAUTTYP“, um die Gruppenberechtigung festzulegen. Die Verwaltung solcher Gruppen ist allerdings problemlos möglich. Detaillierte Informationen zur Verwaltung von Benutzerprofilen mit iSeries Navigator entnehmen Sie bitte dem Kapitel 2.4.5.

2.5.1
Seite 6

Trotz dieser kleinen Einschränkung ist der Einsatz der grafischen Oberfläche für die Benutzerverwaltung sinnvoll und selbst wenn die Planung der Gruppen und die Einrichtung zunächst etwas Zeit kostet, werden Sie schnell feststellen: Es lohnt sich!

Ressourcenschutz durch Berechtigungslisten

Der Zugriff auf die iSeries-Daten wird in der Regel mit drei Methoden realisiert:

Menüsicherheit, Bibliothekssicherheit und Objektschutz. Die Menüsicherheit beschränkt den Anwender auf bestimmte Menüpunkte und ist im Rahmen einer Sicherheitsstrategie sicherlich auch weiterhin sinnvoll, um die Sicherheit der iSeries-Anwendungen zu gewährleisten. Doch Anwendungssicherheit allein ist heute nicht mehr ausreichend, denn DDM, TCP/IP, ODBC, JDBC und IFS bieten neue Wege für den Zugang zu den iSeries-Daten. Natürlich können Sie diese neuen Zugriffsmethoden blockieren, um den Zugriff auf Ihre Daten zu verhindern. Sie können die „Türen" zur iSeries verschließen, so dass ein widerrechtlicher Zutritt unmöglich ist. Durch Exit-Programme könnten Sie z.B. versuchen, PC-Anfragen auf die iSeries-Daten zu kontrollieren. Der Nachteil dieser Strategie?

Für jede Zugriffsart müssen verschiedene Lösungen gefunden werden und was passiert, wenn Sie vergessen, eine der Türen abzuschließen? Dann ist der Zutritt doch möglich! Haben Sie alle Türen abgeschlossen? Und was ist, wenn eine der Türen sich gar nicht verschließen lässt?

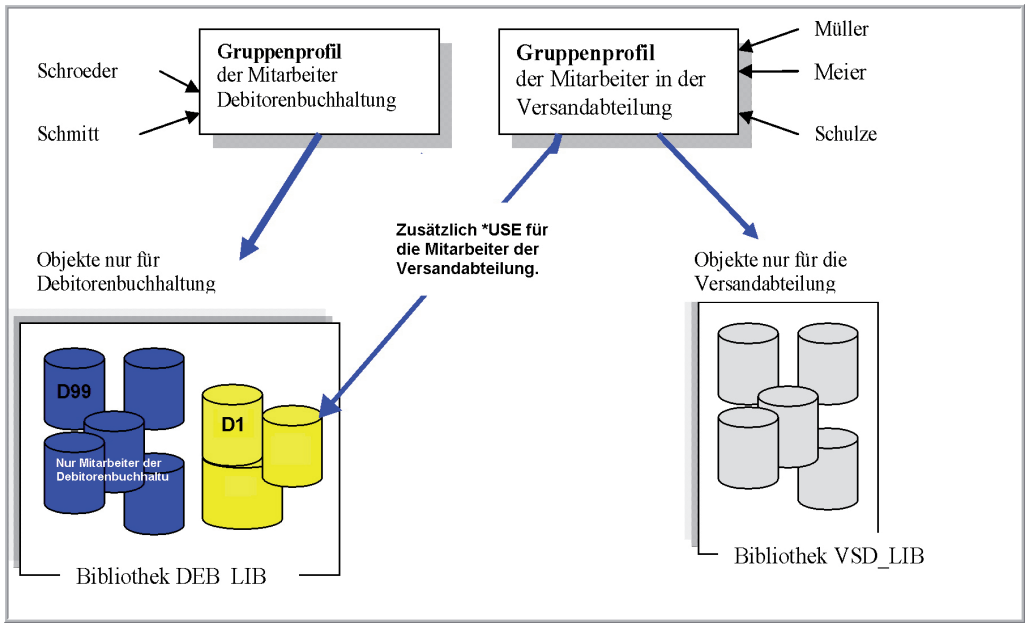
Besser ist es auf der iSeries selbst, die Ressourcen zu schützen. Ressourcensicherheit beinhaltet Bibliotheks- und Objektschutz gleichermaßen. Sie bestimmen über das Datenobjekt, wer in welchem Umfang mit den Daten arbeiten darf.

Das Problem dieser Sicherheitsstrategie ist die ständige Pflege der Zugriffsberechtigungen und der damit verbundene Verwaltungsaufwand. Jedes Mal, wenn ein neues Objekt erstellt wird, muss seine Berechtigung für jeden Benutzer korrekt gesetzt werden. Gleichzeitig müssen Sie darauf achten, dass die Mechanismen nicht zu streng sind, um nicht die Mitarbeiter bei der Arbeit zu behindern.

Sie können die Pflege der Berechtigungszuweisungen vereinfachen, wenn Sie die Berechtigungen organisieren, indem Sie die Objekte und die Benutzer zunächst klassifizieren. Gruppenprofile in Kombination mit Berechtigungslisten sind hervorragend dazu geeignet.

Stellen Sie sich folgendes Szenario vor: Jeder Mitarbeiter der Debitorenbuchhaltung gehört zum Gruppenprofil „GRPDEB". Alle Mitarbeiter der Versandabteilung gehören in die Gruppe „GRPVSD" usw. Aber was geschieht mit der Kundendatei? Beide Gruppen benötigen den Zugriff! Mitarbeiter der Debitorenabteilung dürfen die Kundendaten verändern, während die Versandmit-

arbeiter lediglich die Daten lesen. Ich muss also die Datenobjekte klassifizieren. Das hätte zur Folge, dass die Dateien der Debitorenbuchhaltung, die reinen Versanddateien und die gemeinsam genutzten Dateien je eine Datenklasse darstellen.



Die Ausgangssituation

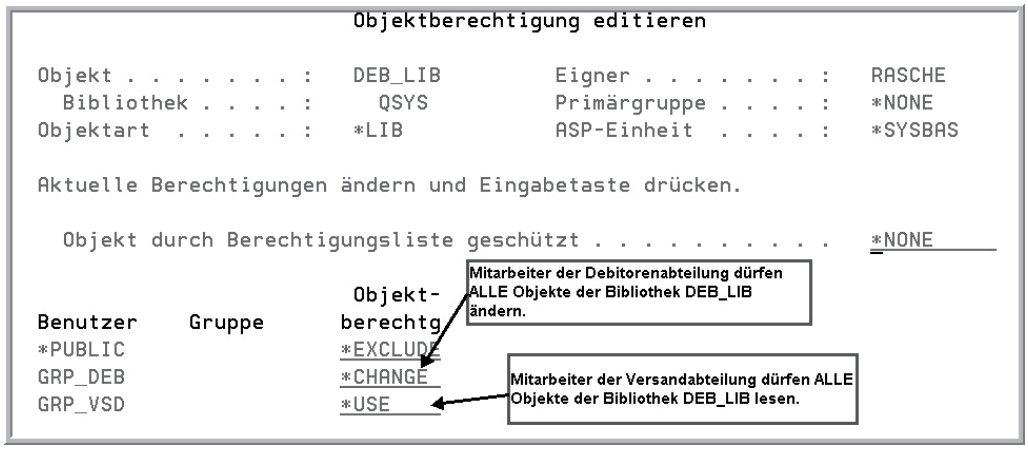
Jetzt kann ich sehr einfach bestimmen, welche Berechtigungsgruppe auf welche Datenklasse zugreifen darf und welche Operationen erlaubt sind. Wird ein neuer Mitarbeiter in der Versandabteilung eingestellt, muss ich den Benutzer lediglich der Benutzergruppe „Versand“ zuordnen und neue Datenobjekte ordne ich der entsprechenden Objektklasse zu.

Nun wissen Sie vielleicht, dass es die eben erfundenen Datenklassen auf einem iSeries-Server nicht gibt. Allerdings können Sie Objekte kategorisieren, indem Sie unterschiedliche Bibliotheken verwenden. Wenn dies nicht ausreichend oder nicht möglich ist, wie in unserem Beispiel, dann können Sie Objektberechtigungsklassen durch Berechtigungslisten auf Objektebene implementieren. Bevor Sie allerdings beginnen mit Berechtigungslisten zu arbeiten, sind ein paar Vorüberlegungen notwendig:

- Welche Objekte gehören zu welcher Berechtigungsliste? Objekte können nur einer Berechtigungsliste zugeordnet werden. Daher ist es notwendig, zunächst die Datenobjekte sorgfältig zu klassifizieren!
- Wie viele Berechtigungslisten sind erforderlich? Hier gibt es keine globale Antwort. Wenn Sie eine höhere Flexibilität benötigen, ist es empfehlenswert, mehrere Berechtigungslisten zu erstellen, mit denen Sie unterschiedliche Berechtigungsebenen implementieren können. Um die Datenpflege zu erleichtern, sollten Sie hingegen nicht zu viele Berechtigungslisten nutzen.

- Welche Namen sollen die Listen erhalten? Anders als die meisten iSeries-Objekte können die Berechtigungslisten nur in der Bibliothek QSYS gespeichert werden und daher müssen die Namen eindeutig und aussagekräftig gewählt werden. Sie sollten vielleicht die Namen mit einer entsprechenden Buchstabenkombination kennzeichnen, so kann problemlos festgestellt werden, zu welcher Anwendung oder „Objektkategorie" die Liste gehört. Ich verwende für mein Beispiel als Listennamen „L_VSD_DEB" für die Objekte beider Gruppen und „L_DEB" als Listennamen für Objekte der Debitorenbuchhaltung.

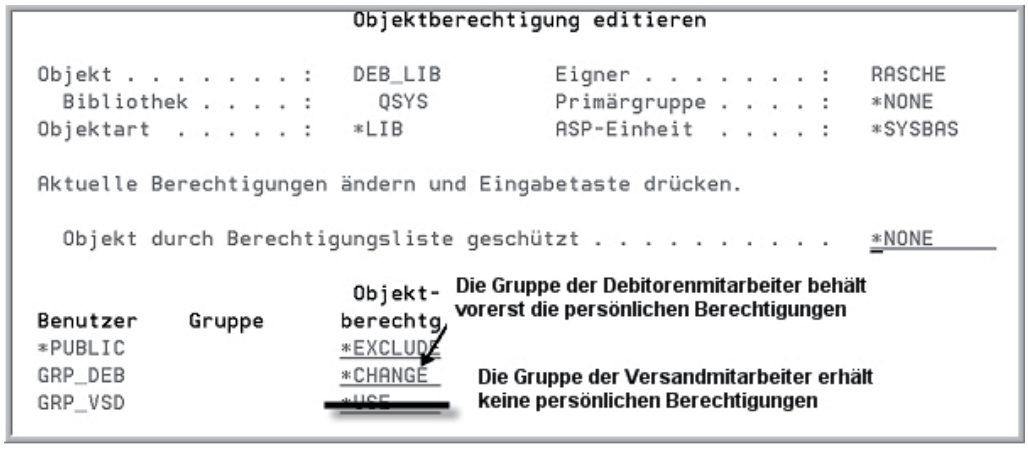
Zunächst erstellen wir die persönlichen Berechtigungen für die Bibliotheken:



Die persönlichen Berechtigungen an der Bibliothek DEB_LIB

Jetzt dürfte aber jedes Mitglied der Gruppe „GRP_VSD" ALLE Objekte in der Bibliothek „DEB_LIB" nutzen.

Besser ist es, wir arbeiten direkt mit einer Berechtigungsliste. Ich entferne also zunächst einmal wieder die persönliche Berechtigung für die Versandmitarbeiter.



Persönliche Berechtigungen für die Versandgruppe entfernen

2.5.2
Seite 4

Stattdessen erstelle ich eine Berechtigungsliste, um die Versandmitarbeiter an den Objekten in der Bibliothek „DEB_LIB“ zu berechtigen:

```
                    Berechtigungsliste erstellen (CRTAUTL)

Auswahl eingeben und Eingabetaste drücken.

Berechtigungsliste . . . . . .   L_VSD_DEB     Name
Text 'Beschreibung'  . . . . . .   Berechtigungsliste Versand für Debitorendate
```

Die Berechtigungsliste L_VSD_DEB erstellen

In einem zweiten Schritt hinterlege ich in der Liste das Gruppenprofil „GRP_VSD“ mit der Berechtigung *USE.

```
                  Berech.list.eintr. hinzufügen (ADDAUTLE)

Auswahl eingeben und Eingabetaste drücken.

Berechtigungsliste . . . . . .   L_VSD_DEB     Name, generisch*
Benutzer . . . . . .   GRP_VSD     Name
     + für weitere Werte
Berechtigung . . . . . .   *USE
     + für weitere Werte

                                                Es können Einzel- oder Gruppenprofile
                                                eingesetzt werden.
```

Berechtigungslisteneintrag für die Gruppe der Versandmitarbeiter hinzufügen

Idealerweise verwenden Sie hier bereits Gruppenprofile, aber auch individuelle Profile sind möglich. Wenn unser Benutzer „Müller“ Mitglied der Gruppe „GRP_VSD“ ist, erhält er die *USE-Berechtigung für alle Objekte, die durch die Liste „DEB_VSD“ geschützt sind.

Soll der Benutzer „Müller“ dagegen *CHANGE-Rechte erhalten, können Sie das Profil als Individualprofil mit der Berechtigung *CHANGE in die Berechtigungsliste aufnehmen. Der Mitarbeiter „Müller“ erhielte *CHANGE-Rechte, während alle anderen Gruppenmitglieder nur *USE-Rechte besitzen.

Im letzten Schritt ordnen wir jetzt die Objekte zu, die durch die Liste geschützt werden sollen.

```
                    Objektberechtigung erteilen (GRTOBJAUT)

Auswahl eingeben und Eingabetaste drücken.

Objekt . . . . . . . . . . . . . > D1          Name, generisch*, *ALL
  Bibliothek . . . . . . . . . . >   DEB_LIB   Name, *LIBL, *CURLIB, *ALL...
Objektart  . . . . . . . . . . . > *FILE       *ALL, *ALRTBL, *BNDDIR...
ASP-Einheit  . . . . . . . . . .   *           Name, *, *SYSBAS
Benutzer . . . . . . . . . . . .               Name, *PUBLIC
     + für weitere Werte
Berechtigung . . . . . . . . . .   *CHANGE     *CHANGE, *ALL, *USE...
     + für weitere Werte
Berechtigungsliste . . . . . . . > L_VSD_DEB   Name, *NONE
Bezugsobjekt . . . . . . . . . .               Name
  Bibliothek . . . . . . . . . .     *LIBL     Name, *LIBL, *CURLIB
Bezugsobjektart  . . . . . . . .   *OBJTYPE    *OBJTYPE, *ALRTBL, *BNDDIR...
Referenz-ASP-Einheit . . . . . .   *           Name, *, *SYSBAS
Berechtigung ersetzen  . . . . . > *no         *NO, *YES
```

Objekte schützen

Für unser Beispiel müsste ich den Befehl für die Bibliothek „DEB_LIB“ und alle betroffenen Dateien wiederholen. Einfacher wäre es, wenn ich die Dateien generisch ansprechen könnte.

Sicherlich haben Sie bemerkt, dass ich auf die einzelnen Befehlsparameter nicht sehr detailliert eingehe und auch nicht den iSeries Navigator verwende. Beides wurde bereits ausführlich im Kapitel 2.4.4.3 getan. Jetzt soll es lediglich um den Einsatz und die Planung der Berechtigungslisten gehen, um eine in sich geschlossene Sicherheitsstrategie zu implementieren.

Es ist also an der Zeit zu testen, ob alles so funktioniert wie wir es uns vorstellen. Ich melde mich dafür mit einem Versandprofil an und lasse mir den Inhalt der Datei D1 anzeigen:

```
                    Physische Teildatei anzeigen
Datei  . . . . . :   D1                 Bibliothek . . . :   DEB_LIB
Teildatei  . . . :   D1


*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...

  (Angegebene Teildatei enthält keine Datensätze)
```

Die Daten lesen

Kein Problem! Versuche ich dagegen den Inhalt der Datei zu löschen, erhalte ich die folgende Fehlermeldung:

```
Nachricht . . . :   Keine Berechtigung zum Löschen, Initialisieren oder
  Kopieren von Teildatei *N.
Ursache  . . . . :  Es liegt keine Berechtigung vor, Teildatei *N in Datei D1
  in der Bibliothek DEB_LIB zu löschen, zu initialisieren oder zu kopieren.
Fehlerbeseitigung:  Die erforderliche Berechtigung beim
  Sicherheitsbeauftragten oder beim Eigner von Datei D1 in der Bibliothek
  DEB_LIB einholen. Anschließend die Anforderung wiederholen.
```

Keine Berechtigung zum Löschen oder Verändern der Daten

Dies ist so ja auch gewollt. Alles scheint prima zu funktionieren. Doch was passiert, wenn wir auf eine Datei zugreifen, für die der Versandmitarbeiter keine Berechtigung erhalten soll? Löschen oder Ändern der Daten ist nicht möglich, aber wir erhalten problemlos lesenden Zugriff:

```
                    Physische Teildatei anzeigen
Datei  . . . . . :   D99                Bibliothek . . . :   DEB_LIB
Teildatei  . . . :   D99

*...+....1....+....2....+....3                                .+....7....+...

  (Angegebene Teildatei enthält keine Datensätze)
```

Unser Berechtigungskonzept erlaubt den lesenden Zugriff auf alle Dateien in der Bibliothek DEB_LIB.

Unser Berechtigungskonzept ist fehlerhaft

Was ist passiert? Nun, wir haben zwar keine Berechtigungen über die Berechtigungsliste erteilt, aber die Versandmitarbeiter gehören natürlich auch zur Gruppe *PUBLIC und die Öffentlichkeit hat derzeit an den Objekten der Bibliothek DEB_LIB ein *CHANGE-Recht:

```
                        Objektberechtigung editieren

Objekt . . . . . . . :   D99            Eigner . . . . . . . :   RASCHE
  Bibliothek . . . . :     DEB_LIB      Primärgruppe . . . . :   *NONE
Objektart  . . . . . :   *FILE          ASP-Einheit  . . . . :   *SYSBAS

Aktuelle Berechtigungen ändern und Eingabetaste drücken.

  Objekt durch Berechtigungsliste geschützt . . . . . . . . . .   *NONE

                         Objekt-
Benutzer     Gruppe      berechtg.
*PUBLIC                  *CHANGE
RASCHE                   *ALL
```

Die öffentliche Berechtigung erlaubt den Zugriff auf das Objekt

Die Objektberechtigung für das Objekt DEB_LIB/D99

Normalerweise interessiert diese Berechtigung nicht, da die Öffentlichkeit die Bibliothek nicht nutzen darf, aber in unserem Fall gewähren wir den Zutritt zur Bibliothek für die Versandmitarbeiter, die dann natürlich auf die Objekte der Bibliothek über genau diese öffentlichen Rechte zugreifen. Wir müssen also immer dann, wenn wir mit Objektrechten innerhalb einer Bibli-

othek arbeiten auch die Öffentlichkeitsrechte aller Objekte innerhalb der Bibliothek auf *EXCLUDE setzen:

```
GRTOBJAUT  OBJ(DEB_LIB/*ALL)  OBJTYPE(*ALL)  *USER(*PUBLIC)
AUT(*EXCLUDE)
```

Jetzt sind alle *PUBLIC-Rechte korrekt gesetzt und die Versandmitarbeiter haben keine Möglichkeit mehr auf die Datei D99 zuzugreifen.

```
Nachricht . . . :   Keine Berechtigung für Datei D99 in Bibliothek DEB_LIB.
Ursache  . . . . :  Es besteht keine Berechtigung, die Datei D99 in Bibliothek
  DEB_LIB zu benutzen.
Fehlerbeseitigung:  Die Benutzungsberechtigung vom Sicherheitsbeauftragten
  oder Dateieigner besorgen. Die Anforderung anschließend wiederholen.
```

Keine Berechtigung für Datei D99 in DEB_LIB

Ein Problem bleibt allerdings: Sobald Sie neue Objekte in der Bibliothek erstellen, müssten die *PUBLIC-Rechte immer manuell auf *EXCLUDE gesetzt werden, da das Betriebssystem in vielen Fällen der Öffentlichkeit zunächst das Recht einräumt, Objekte zu ändern. Hier könnten sich schnell Fehler einschleichen.

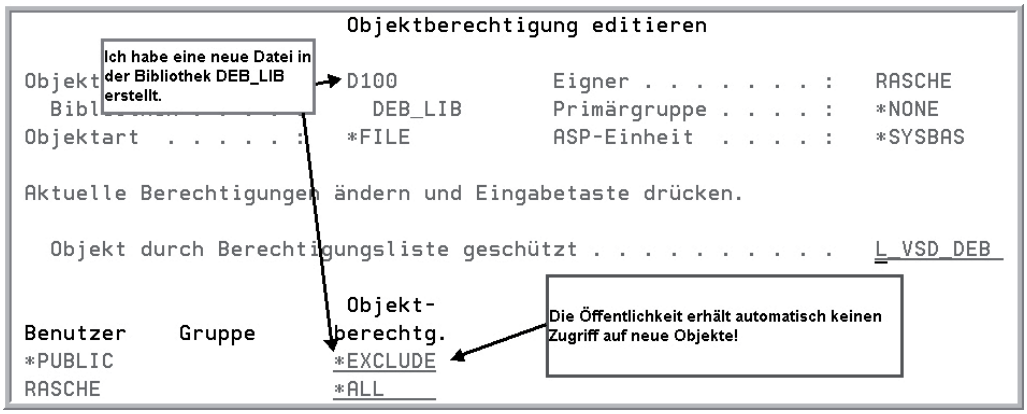
Es ist möglich, die Berechtigung *PUBLIC *EXCLUDE automatisch zu vergeben. Hierfür müssen Sie nur die Attribute der Bibliothek DEB_LIB folgendermaßen ändern:

```
                        Bibliothek ändern (CHGLIB)

Auswahl eingeben und Eingabetaste drücken.

Bibliothek . . . . . . . . . . . > DEB_LIB      Name, *CURLIB
Bibliotheksart . . . . . . . . . > *PROD        *SAME, *PROD, *TEST
Text 'Beschreibung'  . . . . . .   *BLANK
_____

                Zusätzliche Parameter

Berechtigung für neue Objekte  . > *EXCLUDE     Name, *SAME, *SYSVAL...
Objektprotokollierung  . . . . .   *SYSVAL      *SAME, *SYSVAL, *NONE...
```

Neue Objekte dieser Bibliothek erhalten automatisch die Berechtigung *PUBLIC *EXCLUDE

Bibliotheksattribute ändern

Die Öffentlichkeitsrechte neuer Objekte können durch den Parameter CRTAUT gesteuert werden. Für mein Beispiel ist das genau die richtige Lösung, denn so muss ich auf die *PUBLIC-Berechtigungen der Objekte nicht mehr achten.

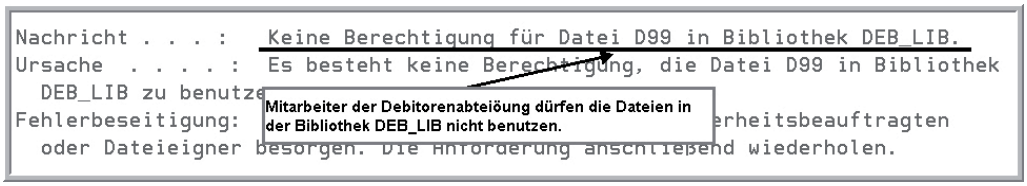


Die öffentliche Berechtigung für neue Objekte

ACHTUNG:

Der Parameter AUT der jeweiligen Bibliothek wird nur beachtet, wenn Objekte direkt erstellt werden, d.h. Objekte, die Sie z.B. mit dem CPYF-Befehl oder mit dem Befehl CRTDUPOBJ erstellen, ignorieren den AUT-Parameter der jeweiligen Bibliothek. Sie leiten Ihre Rechte vom kopierten Objekt ab.

Als letztes bleibt zu prüfen, ob die Mitarbeiter der Debitorenabteilung die Zugriffsrechte korrekt erhalten haben und hier stehen wir auch schon wieder vor einem erneuten Problem: Keiner der Debitorenmitarbeiter darf auf Dateien in der Bibliothek DEB_LIB zugreifen. Das kann nicht gewollt sein!!



Mitarbeiter der Debitorenbuchhaltung haben keinen Zugriff auf die Daten

Unsere Strategie funktioniert nur, wenn wir eine zweite Berechtigungsliste einsetzen oder persönliche Berechtigung für die Objekte der Bibliothek DEB_LIB erteilen. Ich entscheide mich für die Berechtigungsliste:

```
CRTAUTL  AUTL(L_DEB)
```

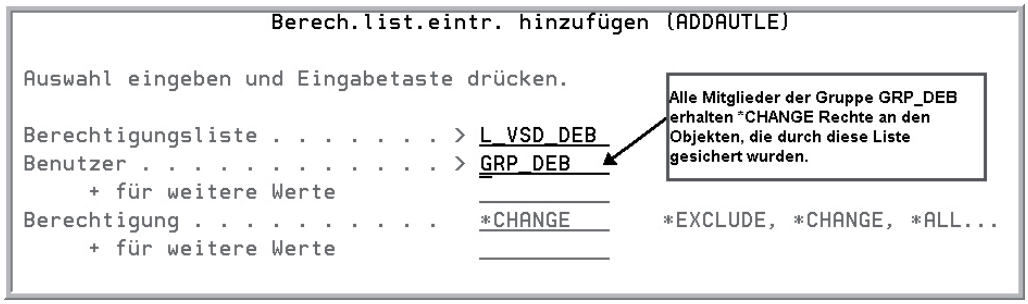
Dieser Liste ordne ich die Mitarbeiter der Debitorenbuchhaltung zu.

```
ADDAUTLE  AUTL(L_DEB)  USER(GRP_DEB)
```

In einem weiteren Schritt verwende ich die Liste L_DEB für alle Objekte der Bibliothek DEB_LIB die bislang nicht durch eine Berechtigungsliste geschützt sind.

Nun dürfen wir nur nicht die Objekte vergessen, die bereits durch die Liste L_VSD_DEB geschützt sind. Objekte können generell nur durch eine einzige Berechtigungsliste geschützt werden. Aber auch hier ist die Lösung einfach: Wir nehmen die Debitorenmitarbeiter in die Liste L_VSD_DEB auf und erteilen *CHANGE-Rechte. Es ist durchaus möglich, über eine Liste mehrere Persongruppen mit unterschiedlichen Zugriffsrechten zu verwalten.

Um den Mitarbeitern Zugriff auf die Objekte zu gewähren, die bereits durch die Berechtigungsliste L_VSD_DEB geschützt sind, ist eine Anpassung der Objekte selbst nicht mehr nötig. Die Zuordnung der Mitarbeiter zur Liste ist ausreichend.



Debitorenmitarbeiter berechtigen

Falls neue Mitarbeiter für die Debitorenabteilung eingestellt werden, ist es sogar ausreichend, den neuen Mitarbeiter nur in die Gruppe GRP_DEB aufzunehmen. Scheidet ein Mitarbeiter aus, entfernen Sie ihn aus der Gruppe. Mit neuen Objekten verfahren Sie ähnlich: Neue Objekte schützen Sie entweder durch die eine oder andere Berechtigungsliste, mehr Arbeit ist nicht erforderlich. Und genau hier ist auch der entscheidende Vorteil zu sehen:

- Mit wenigen Befehlen können Sie neue Benutzer oder neue Objekte hinzufügen und von Anfang an ist dafür gesorgt, dass die richtige Berechtigung zugewiesen wird. Indem Sie Benutzer in die Listen aufnehmen oder von der Liste entfernen, gewähren bzw. entziehen Sie die entsprechenden Objektberechtigungen für alle Objekte, die die Berechtigungsliste nutzen.
- Wenn Sie ein Benutzerprofil löschen, wird das Profil automatisch aus allen Berechtigungslisten entfernt.
- Sie können die Zugriffsberechtigungen über die Berechtigungsliste jederzeit ändern – auch während sich die betroffenen Objekte im Zugriff befinden. Sollen die Mitarbeiter der Debitorenabteilung z.B. *ALL-Rechte an den Dateiobjekten erhalten, müssen Sie nur die Berechtigung in den Listen abändern, und die veränderten Berechtigungen sind sofort wirksam.

2.5.2
Seite 10

- Hinzu kommt, dass die Verbindung zwischen dem Objekt und der Liste automatisch wiederhergestellt wird, wenn Sie Objekte wiederherstellen, die über eine Berechtigungsliste geschützt sind, d.h. Sie müssen die Objektberechtigungen nicht neu zuweisen.

Sie sehen, auch wenn Ihnen im ersten Moment der Einsatz der Berechtigungslisten kompliziert erscheint, lohnt sich der Aufwand. Die anfängliche Investition wird sich später auszahlen! Wichtig ist eine sorgfältige Planung:

Schauen Sie sich hierfür noch einmal die Ausgangssituation an. Anfänglich haben wir die Objekte in drei Gruppen unterteilt, die sich in zwei unterschiedlichen Bibliotheken befanden. Wir haben die Objekte klassifiziert! Wenn Sie zwei Objektklassen innerhalb einer Bibliothek erhalten, benötigen Sie eine entsprechende Anzahl Berechtigungslisten.

Abschließend noch ein paar Bemerkungen, denn falsch platzierte Berechtigungslisten können auch unnötigen System-Overhead produzieren:

Gruppieren Sie immer zunächst die Objekte in Bibliotheken und schränken Sie – wenn möglich – nur den Zugriff auf die Bibliothek ein. Den Zugriff auf die Objekte innerhalb der Bibliothek steuern Sie über die *PUBLIC-Berechtigungen. Diese Technik wird als Bibliothekssicherheit bezeichnet. Sie positionieren also nur eine Wache an der „Haustür“ und erlauben innerhalb des „Hauses“ den Zugang zu allen „Räumen“.

Nur wenn die Bibliothekssicherheit nicht ausreichend ist, da verschiedene Personengruppen mit unterschiedlichen Berechtigungen auf die Objekte zugreifen, setzen Sie Berechtigungslisten oder persönliche Berechtigungen ein. Um bei unserem Beispiel mit dem Haus zu bleiben: Sie gestatten den Zutritt an der „Haustür“, unterbinden aber den Zugang zu einzelnen „Räumen“. Vermeiden Sie hierbei aber Überschneidungen, d.h. gewähren Sie Ihren Mitarbeitern nicht den Zugriff über Einzelberechtigungen und gleichzeitig über eine Berechtigungsliste. Die Prüfung einer Berechtigungsliste bei gleichzeitiger persönlicher Berechtigung führt zu unnötigem System-Overhead. Es ist als ob Sie eine Wache an der „Tür“ positionieren, die sowohl den Mitarbeiterausweis (=persönliche Berechtigung) als auch eine Kontrollliste (=Berechtigungsliste) prüft. Fehlt dagegen die persönliche Berechtigung, muss der Benutzer nur anhand der „Kontrollliste“ identifiziert werden. Dies kann den Prozess beschleunigen. Auch mein kleines Beispiel hat bereits solch eine Überschneidung:

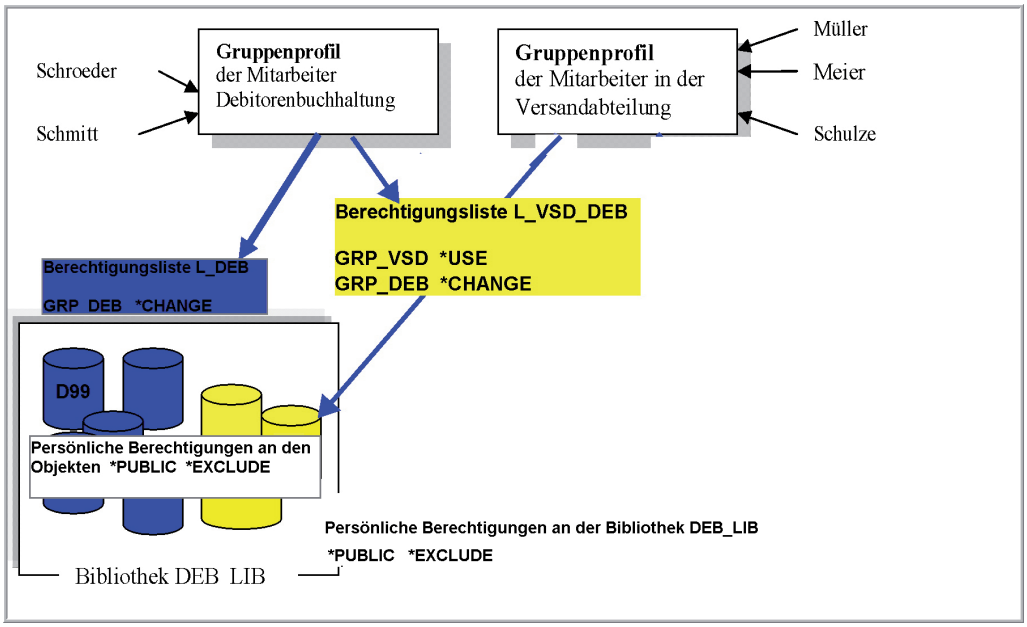
Die Mitarbeiter der Debitorenabteilung werden für die Bibliothek DEB_LIB sowohl über persönliche Berechtigungen als auch durch die Berechtigungsliste L_VSD_DEB berechtigt. Dies ist unnötig, wir können die persönlichen Berechtigungen entfernen ohne unser Berechtigungskonzept zu gefährden.

```
                        Objektberechtigung editieren

Objekt . . . . . :   DEB_LIB          Eigner . . . . . :   RASCHE
  Bibliothek . . . . :     QSYS       Primärgruppe . . . . :   *NONE
Objektart  . . . . . :   *LIB          ASP-Einheit  . . . . :   *SYSBAS

Aktuelle Berechtigungen ändern und Eingabetaste drücken.

  Objekt durch Berechtigungsliste geschützt . . . . . .  L_VSD_DEB

                          Objekt-
Benutzer     Gruppe       berechtg.
*PUBLIC                   *EXCLUDE
```

Persönliche Berechtigungen sind nicht mehr erforderlich, da der Zugriff vollständig über die Berechtigungsliste gesteuert werden kann.

Die Berechtigungen für die Bibliothek DEB_LIB

Jetzt ist unser Berechtigungskonzept fertig und kann folgendermaßen dargestellt werden:



Das fertige Konzept

2.5.2
Seite 12

Berechtigungsübernahmen

Kennen Sie die folgenden Situationen? Ihre Vertriebsmitarbeiter sollen im Rahmen der Anwendung Kundendaten pflegen. Dazu gehören natürlich auch Adressänderungen. Die gleichen Mitarbeiter dürfen die Kundeninformationen aber nur lesen, wenn sie SQL oder andere Abfragetools verwenden.

Eine Mitarbeiterin der Lohn- und Gehaltsabteilung muss im Rahmen der Applikation auf die Lohn- und Gehaltsdaten zugreifen. Gleichzeitig wollen Sie dieser Mitarbeiterin aber auch Query zur Verfügung stellen, ohne den Zugriff auf einige Lohn- und Gehaltstabellen zu erlauben.

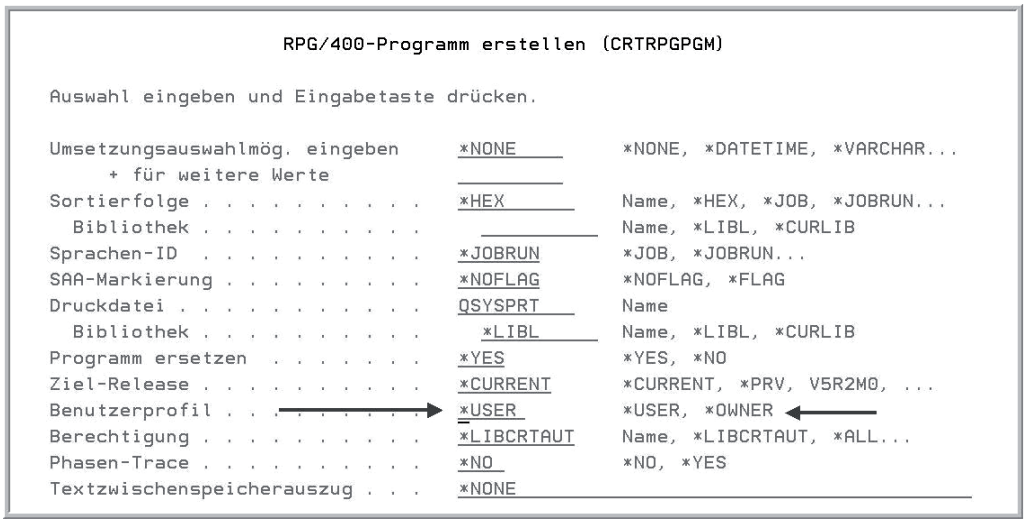
Die genannten Probleme tauchen auf, weil das Betriebssystem i5/OS es nicht zulässt, dass ein Benutzer in unterschiedlichen Situationen unterschiedliche Objektberechtigungen für ein Objekt besitzt. Diese Einschränkung lässt sich mit Berechtigungsübernahmen umgehen. Bei einer Berechtigungsübernahme erhält der Benutzer während der Programmausführung – also temporär – zusätzlich die Berechtigungen einer anderen Person oder besser gesagt eines anderen Benutzerprofils. Berechtigungsübernahmen sind hilfreich aber gleichzeitig mit Vorsicht einzusetzen. Das nachfolgende Kapitel soll daher nicht nur die Funktionsweise und Vorteile sondern auch die Risiken von Berechtigungsübernahmen verdeutlichen.

12. Ergänzung 09/2007

2.5.3
Seite 2

Berechtigungsübernahmen – wie funktioniert das?

Programme (*PGM), Serviceprogramme (*SRVPGM), SQL-Packages (*SQLPKG) und Java-Programme können die Rechte des Programmeigners adoptieren. Das ausführbare Objekt erbt somit zur Ausführungszeit die Objekt- und Sonderberechtigungen (*JOBCTL, *ALLOBJ, usw.) des Eigentümers. Es ist sehr einfach einem Programm die Berechtigungen zu übertragen.



Der Parameter USRPRF im Befehl CRTxxxPGM

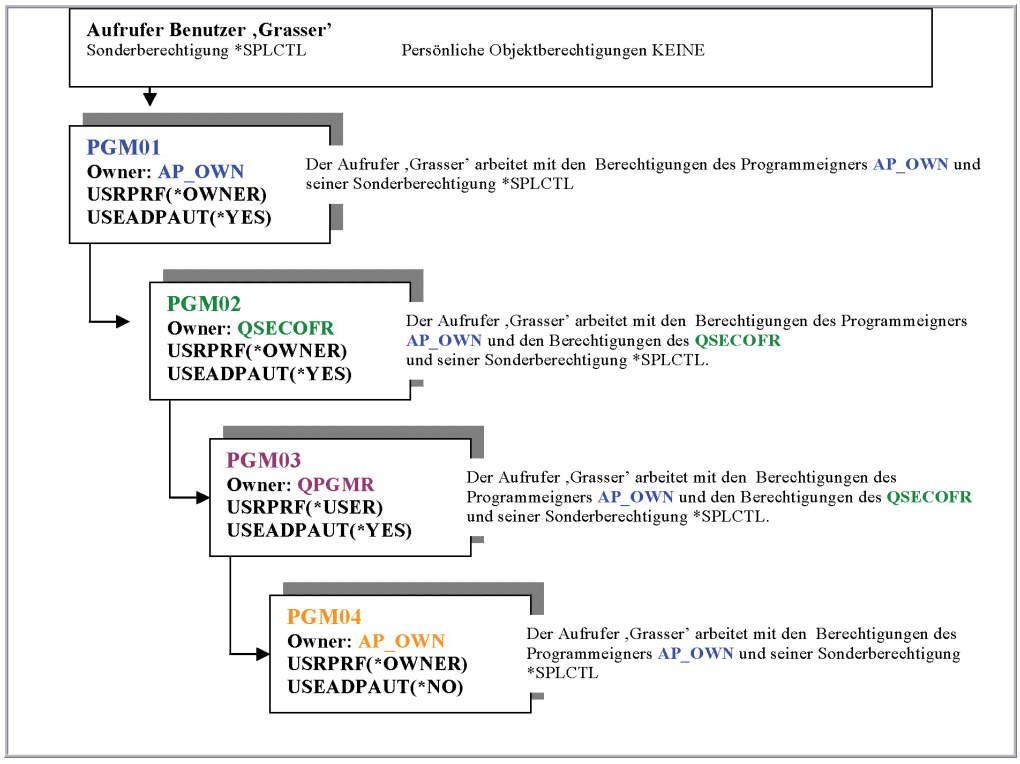
Jeder CRTxxxPGM-Befehl enthält den Parameter „Benutzerprofil“ (USRPRF). Der Standardwert USRPRF (*USER) führt dazu, dass nur die Zugriffsberechtigungen des Benutzers während der Programmausführung gültig sind. Kompilieren Sie hingegen das Programm mit der Einstellung USRPRF (*OWNER), berücksichtigt das Betriebssystem bei der Programmausführung sowohl die Benutzer- als auch die Programmeignerberechtigungen. Der ausführende Benutzer verfügt somit über die Berechtigungen des Programmeigners solange sich das Programmobjekt im Aufrufstapel (Stack) befindet. Wird das Programm beendet, hat der Benutzer wieder seine individuellen Objekt- und Sonderberechtigungen.

Das klingt zunächst sehr einfach. Dennoch verbirgt sich hier ein Risiko. Sie könnten einem Benutzer unbeabsichtigt erlauben, nicht zulässige Aktionen auszuführen. Alles was der Benutzer benötigt ist eine Kommandozeile oder ein geeignetes Tool, um die übernommenen Berechtigungen für seine Zwecke zu verwenden. Diese Sicherheitslücke tritt insbesondere dann auf, wenn Sie CL-Programme einsetzen, die Menüs zur Verfügung stellen. Übernimmt das CL-Programm die Berechtigungen seines Eigners, ist alles, was der Benutzer tut während sich das CL-Menüprogramm im Stapel befindet, ein Sicherheitsrisiko.

2.5.3.1
Seite 2

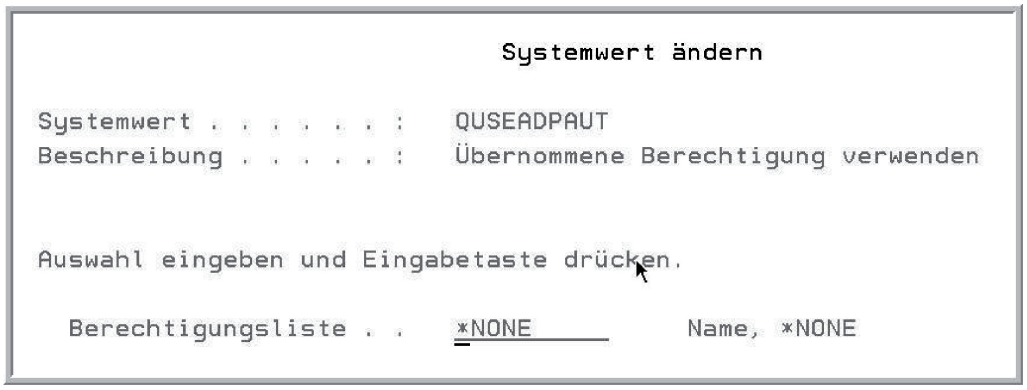
Warum? Menüs enthalten häufig Betriebssystembefehle, wie z.B. den Befehl WRKSPLF – „Mit Druckausgaben arbeiten“ und diese Betriebssystembefehle stellen eine Befehlszeile zur Verfügung! Dieses Risiko wird größer, wenn das Programm *JOBCTL- oder gar *ALLOBJ-Berechtigungen vom Programmeigner erbt. Mit der Sonderberechtigung *JOBCTL und einer Befehlszeile könnte der Benutzer jeden Job im System modifizieren. Ein Benutzer, der *ALLOBJ-Berechtigungen erbt, darf auf jedes Objekt im System zugreifen und hätte somit Zugang zu allen Firmendaten. Achten Sie also darauf, dass Ihre Programme, wenn Sie Berechtigungen übernehmen, keine Schnittstellen zum Betriebssystem beinhalten. Dies gilt jedoch nicht für Systemanfragen, für die Bearbeitung von Abbruchnachrichten oder Debug-Operationen. In diesen Fällen wird die Berechtigungsübernahme automatisch unterbrochen.

Ein weiteres Problem sind unbeabsichtigte Berechtigungsübernahmen: Programme, die mit der Einstellung USRPRF (*USER) kompiliert werden, können Berechtigungen anderer Programme übernehmen. Schuld daran ist der Parameter „übernommene Berechtigungen nutzen“(USEADPAUT). Dieser Parameter hat bei der Programmumwandlung immer die Einstellung *YES. Dadurch übernimmt ein Programm die Berechtigungen von Programmen, die weiter oben im Aufrufstapel stehen (siehe folgende Grafik).



Wirkungsweise der Berechtigungsübernahmen

Programme können somit auf zwei unterschiedliche Arten Berechtigungen übernehmen. Zum einen ermöglicht der Parameter USRPRF (*OWNER) es dem Programm, die Berechtigungen des Programmeigners zu erben. Zum anderen kann ein Programm die Berechtigungen vorangegangener Programme im Aufrufstapel nutzen – USEADPAUT (*YES). Die zweite Möglichkeit lässt sich über den Systemwert QUSEADPAUT auf einen bestimmten Personenkreis einschränken.

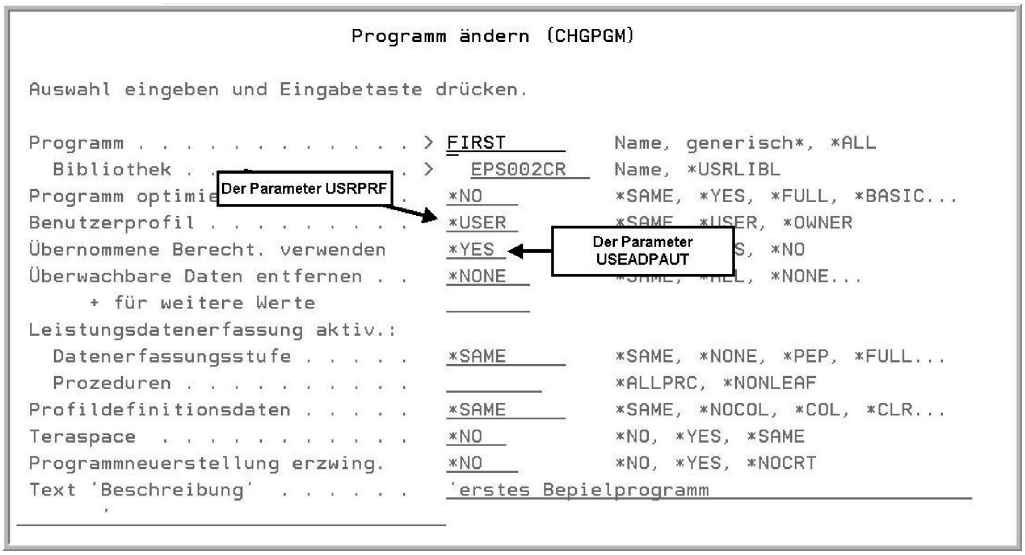


Der Systemwert QUSEADPAUT

Dieser Systemwert definiert, welche Benutzer-Programme mit dem Wert USEADPAUT (*YES) erstellen dürfen. Die Defaulteinstellung *NONE erlaubt es zunächst allen Benutzern. Sie können aber den Namen einer Berechtigungsliste hinterlegen. Dies führt dazu, dass zunächst anhand dieser Liste die Benutzerberechtigungen überprüft werden und nur wenn der Benutzer mindestens die Berechtigung *USE hat, darf er Programme kompilieren, die die Berechtigung anderer im Aufrufstapel vorhandener Programme übernehmen.

Und noch etwas müssen Sie beachten: die Parameter USRPRF und USEADPAUT werden ignoriert, wenn Sie ein Programm erneut kompilieren. Das neue Programm wird stets über die gleichen Werte verfügen wie das ersetzte Programm. Um die Parameter zu ändern, müssen Sie entweder das alte Programmobjekt vor der Umwandlung explizit löschen oder die Werte mit dem Befehl „Programm ändern“(CHGPGM) manuell ändern.

2.5.3.1
Seite 4



Der Befehl CHGPGM

Zum Schluss noch einige kritische Anmerkungen:

- Versuchen Sie, Ihre Objekte zu kategorisieren, d.h. ordnen Sie Ihre Objekte in Kategorien ein. Programme, Bildschirm- und Druckdateien speichern keine Daten. Datenbereiche und Tabellen dagegen enthalten Daten. Im Allgemeinen sollten Sie für die Datenobjekte die öffentlichen Berechtigungen auf *EXCLUDE setzen. Objekte, die keine Daten spezifizieren, können auch mit der *PUBLIC-Berechtigung *USE gespeichert werden. Eine gute Möglichkeit Objekte zu kategorisieren, besteht darin, sie in unterschiedlichen Bibliotheken zu organisieren. Wenn Sie eine Bibliothek erstellen, die ausschließlich Datenobjekte enthält, können Sie den Parameter „Berechtigungen erstellen“(CRTAUT) des Befehls „Bibliothek erstellen“(CRTLIB) auf *EXCLUDE setzen. Dadurch bestimmen Sie bereits beim Erstellen der Bibliothek, dass alle Objekte, die Sie in der Bibliothek mit einem entsprechenden CRT-Befehl erstellen, die öffentliche Berechtigung *EXCLUDE erhalten. Ein konkretes Beispiel finden Sie im Kapitel 2.5.2. Analog nutzen Sie natürlich die Berechtigung *USE für Ihre Programmbibliotheken. Nachteilig an dieser Methode ist, dass indirekt erzeugte Objekte unberücksichtigt bleiben, d.h. Objekte, die dupliziert oder durch den Befehl „Datei kopieren“(CPYF) erstellt werden, erhalten die öffentlichen Berechtigungen in Abhängigkeit der verwendeten Befehle und nicht aufgrund der Bibliothekseinstellung.
- Achten Sie auch darauf, dass Sie die Möglichkeiten Ihrer Benutzer, mit der Befehlszeile zu arbeiten, bereits im Benutzerprofil unterbinden. Wenn Sie den Parameter „Möglichkeiten einschränken“ (LMTCPB) auf *YES setzen, können Ihre Mitarbeiter keine Befehle nutzen – auch nicht unter übernommenen Berechtigungen. Falls es Befehle gibt, die Sie allgemein-

gültig zur Verfügung stellen wollen, ändern Sie mit dem Befehl „Befehl ändern“(CHGCMD) deren Parameter ALWLMTUSR auf *YES.

- Vermeiden Sie den Benutzer QSECOFR als Programmeigner bei Berechtigungsübernahmen. Auch auf die Sonderberechtigung *ALLOBJ sollten Sie im Rahmen von Berechtigungsübernahmen verzichten. Konfigurieren Sie lieber einige Pseudobenutzerprofile. Erteilen Sie diesen Profilen die nötigen Zugriffsberechtigungen für die Objekte, verzichten Sie aber – wenn möglich – auf Sonderberechtigungen und setzen Sie das Kennwort für die Pseudoprofile auf *NONE, so dass sich niemand mit diesen Profilen anmelden kann.
- Startprogramme, die im Benutzerprofil hinterlegt sind, sollten keine Berechtigungen übernehmen, denn diese Programme stehen im Aufrufstapel (Stack) solange der Benutzer interaktiv arbeitet und es kann schnell zu unbeabsichtigten Berechtigungsübernahmen kommen.
- Versuchen Sie – wenn möglich – Stapeljobs für die Berechtigungsübernahmen einzusetzen. Einerseits wirken sich die Batchjobs positiv auf die Systemleistung aus, andererseits können Batchjobs von übergeordneten Programmen keine Berechtigungen erben.

2.5.3.1

Seite 6

Berechtigungsübernahmen umsetzen

Erinnern Sie sich noch an das anfangs geschilderte Problem? Eine Mitarbeiterin der Lohn- und Gehaltsabteilung muss im Rahmen der Applikation auf alle Lohn- und Gehaltsdaten zugreifen. Außerdem soll dieser Mitarbeiterin auch Query zur Verfügung gestellt werden, ohne dabei den Zugriff auf einige Lohn- und Gehaltstabellen zu erlauben. Wir müssen dieser Mitarbeiterin also einerseits den vollen Zugriff auf die Lohn- und Gehaltstabellen ermöglichen und gleichzeitig sicherstellen, dass sie mit Query die Lohn- und Gehaltsdaten nicht lesen kann. Im Folgenden werden wir versuchen, dieses Problem beispielhaft zu lösen.

Zunächst benötigen wir einige Benutzerprofile:

Es ist empfehlenswert, die Objekte einer Anwendung stets einem Eigner zu übertragen, dadurch ersparen Sie sich viel Verwaltungsaufwand. In meinem Beispiel ist das Benutzerprofil „LG_APOWN“ für die Lohn- und Gehaltsbuchführung. Ein weiteres Benutzerprofil, „LG_QRY“ steuert den Zugriff auf die Bibliotheken und Tabellen mit Query. Mit dem dritten Profil „LG_USR“ soll sich die Mitarbeiterin der Lohn- und Gehaltsabteilung später anmelden. Ein viertes Profil kann Eigentümer verschiedener Objekte werden, die der Problemlösung dienen.

Wir haben demzufolge vier Profile, die folgendermaßen definiert sind:

Profilname	Beschreibung	Passwort	Sonder-Berechtigungen	Anfangs-Menü	Möglichkeiten einschränken
LG_QRY	Für Zugriff auf Bibliotheken/Objekte mit Query	*NONE	*NONE	Nicht erforderlich	Nicht erforderlich
LG_APOWN	Eigentümer der Anwendung Lohn- und Gehalt. Verfügt über alle Berechtigungen für die entsprechenden Lohn- und Gehaltsobjekte.	*NONE	Entsprechend der Anwendung	Nicht erforderlich	Nicht erforderlich
LG_USR	Beispielprofil für Mitarbeiter der Lohn- und Gehaltsabteilung	*YES	*NONE	LG_STR	*YES
LG_DIV	Pseudoprofil für diverse Objekte	*NONE	*NONE	Nicht erforderlich	Nicht erforderlich

2.5.3.2
Seite 2

Sie könnten dem Profil LG_USR zusätzlich eine aktuelle Bibliothek zuweisen, so dass die Query-Abfragen immer in dieser persönlichen Bibliothek gespeichert werden. Eigentümer der Bibliothek sollte dann das Profil „LG_USR“ sein, damit es alle Berechtigungen an der Bibliothek erhält.

Und jetzt benötigen wir noch ein Anfangsmenü „LG_MNU01“ für unsere Mitarbeiterin der Lohn- und Gehaltsabrechnung.

```
LG_MNU01                Lohn- und Gehaltsbuchführung            22.01.07
                                                                13:33:00
Auswahlmöglichkeiten:

              1. Lohnabrechnung
              2. Gehaltsabrechnung
              3. Personalstammdaten


             20. Queryabfragen erstellen




                                                      99. Abmelden




Auswahl oder Befehl
     ===
```

Anfangsmenü

Wenn die Mitarbeiterin den Menüpunkt 1 wählt, dann wird sie im Rahmen der Anwendung in der Lohn- und Gehaltsabrechnung arbeiten. Dafür schreiben wir ein kleines CL-Programm (LG_STR), das zunächst die entsprechenden Bibliotheken in die Suchliste der Mitarbeiterin aufnimmt. Außerdem übernimmt es die Berechtigungen des Applikation-Owners „LG_APOWN“, um auf die entsprechenden Objekte zugreifen zu können. Das Programm wird mit der Parametereinstellung USRPRF(*OWNER) kompiliert.

Mein CL-Programm „LG_STR“ sieht wie folgt aus:

```
PGM
ADDLIBLE LG_DATA /* Zuordnung der erforderlichen Bibliotheken */
ADDLIBLE LG_PGM
GO LG_MNU02      /* Menü der Lohn- und Gehaltsabrechnung */
RMVLIBLE LG_DATA /* Entfernen der Lohn- und Gehaltsbibliotheken
                    aus der Suchliste */
RMVLIBLE LG_PGM
ENDPGM
```

Eigentümer des Programms „LG_STR“, das die Lohn- und Gehaltsabrechnung startet, muss das Benutzerprofil „LG_ApOWN“ sein. Ich übertrage daher die Eignerschaft am ausführbaren Programmobjekt auf das entsprechende Profil.

```
                    Objekteigner ändern (CHGOBJOWN)

Auswahl eingeben und Eingabetaste drücken.

Objekt . . . . . > LG_STR      Name
  Bibliothek . . . . . *LIBL     Name, *LIBL, *CURLIB
Objektart . . . . . > *PGM      *ALRTBL, *AUTL, *BNDDIR...
ASP-Einheit . . . . . *          Name, *, *SYSBAS
Neuer Eigner . . . . . > LG_APOWN  Name
Aktuelle Eignerberechtigung . .  *REVOKE     *REVOKE, *SAME
```

Objekt-Eignerschaft übertragen

Mit einem zweiten CL-Programm „QRY_STR“ bestimmen wir die Query-Umgebung:

```
PGM
ADDLIBLE LG_DATA     /* Zuordnung der erforderlichen Bibliotheken */
STRQRY               /* Aufruf Query */
RMVLIBLE LG_DATA     /* Entfernen der Lohn- und Gehaltsbibliotheken
                     aus der Suchliste */
ENDPGM
```

Auch hier soll das Programm zur Ausführungszeit die Berechtigungen des Eigners erben (USRPRF*OWNER). Das Benutzerprofil „LG_QRY“ wird Eigentümer dieses Objekts.

Unsere Lösung ist fertig!

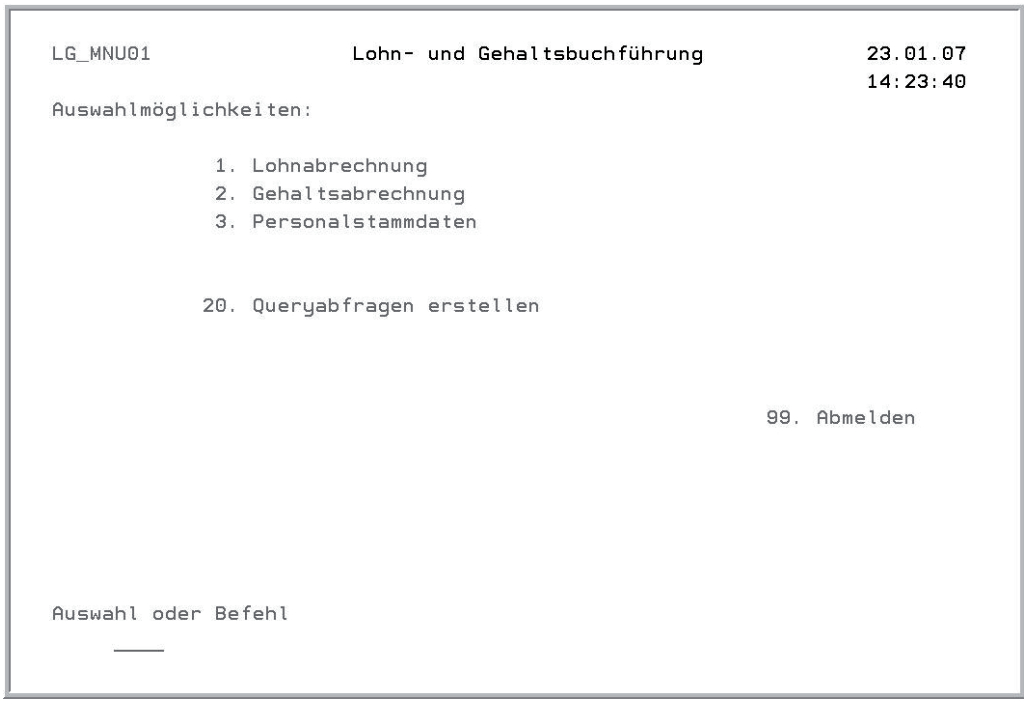
Schauen wir uns jetzt die Zugriffsberechtigungen der beteiligten Objekte an:

Objekt	Beschreibung	Eigentümer	*PUBLIC Berechtigung	Private Berechtigungen
Bibliothek LG_PGM	Lohn- u. Gehalts-Programmbibliothek	LG_ApOWN	*EXCLUDE	
Bibliothek LG_DATA	Lohn- und Gehalts-Datenbibliothek	LG_ApOWN	*EXCLUDE	Das Benutzerprofil LG_QRY erhält *USE-Berechtigung

2.5.3.2
Seite 4

Bibliothek DIV_OBJ	Diverse Objekte für unterschiedliche Applikationen	LG_DIV	*EXCLUDE	*USE-Berechtigung für alle User, die die entsprechenden Objekte nutzen (z. B. der Benutzer LG_USR)
Menü LG_MNU01 in DIV_OBJ	Anfangsmenü der Mitarbeiter Lohn- und Gehalt, die Query nutzen sollen.	LG_DIV	*EXCLUDE	*USE Berechtigung für die Benutzer der Lohn- und Gehaltsabteilung.
Programm LG_STR in DIV_OBJ	Startprogramm Lohn- und Gehalt	LG_ApOWN	*EXCLUDE	*USE Berechtigung für die Benutzer der Lohn- und Gehaltsabteilung.
Programm QRY_STR in DIV_OBJ	Startprogramm Query	LG_QRY	*EXCLUDE	*USE Berechtigung für die Benutzer der Lohn- und Gehaltsabteilung.
Lohn- und Gehaltstabellen ohne Query	Tabelle mit den Lohn- und Gehaltsdaten ohne Query	LG_ApOWN	*EXCLUDE	
Lohn- und Gehaltstabellen mit Query	Tabelle mit den Lohn- und Gehaltsdaten mit Query Zugriff	LG_ApOWN	*USE	
Programme Lohn- und Gehaltsabrechnung	Alle Programme	LG_ApOWN	*USE	

Wenn die Mitarbeiterin der Lohn- und Gehaltabteilung sich jetzt anmeldet, erhält sie zunächst das entsprechende Anfangsmenü in der Bibliothek „DIV_OBJ“ – gesteuert durch ihr Benutzerprofil.



Anfangsmenü

Die Mitarbeiterin braucht sowohl für die Bibliothek „DIV_OBJ“ als auch für das Menüobjekt „LG_MNU01“ die *USE-Berechtigung. Anstelle einer separaten Bibliothek könnte sie auch QGPL für die diversen Objekte nutzen. Ich vermeide es allerdings, eigene Objekte in IBM-Bibliotheken zu speichern. Entscheidet sich die Mitarbeiterin, anschließend in der Lohn- und Gehaltsanwendung zu arbeiten, wird das Programm „LG_STR“ aufgerufen. Für den Programmaufruf sind ebenfalls *USE-Rechte erforderlich. Das Programm „LG_STR“ adoptiert die Rechte des Programmeigners – also die Rechte des Benutzerprofils „LG_ApOWN“, so dass die Mitarbeiterin jetzt zusätzlich über die Rechte des Programmeigners verfügt. Innerhalb der Applikation erhält sie uneingeschränkten Zugriff auf alle Daten- und Programmobjekte. Kehrt die Mitarbeiterin wieder zum Ausgangsmenü zurück, wird das Programm „LG_Start“ aus dem Programmstapel entfernt; die Rechte des Programmeigners sind nicht mehr wirksam.

Falls die Mitarbeiterin anschließend den Menüpunkt 20 wählt, startet das Programm „QRY_STR“. Auch hierfür sind *USE-Berechtigungen erforderlich. Dieses Programm adoptiert die Rechte des Eigners „LG_QRY“. Das Profil „LG_QRY“ hat die *USE-Berechtigung für die Datenbibliothek und darf aufgrund der *PUBLIC-Berechtigung *USE auf ausgewählte Tabellen der Lohn- und Gehaltsanwendung zugreifen. Dies würde aber nicht für die Mitarbeiterin ausreichen, denn die *PUBLIC-Berechtigungen können nicht vererbt werden. Sie wurde aber auch nicht explizit von der Nutzung der Datenobjekte ausgeschlossen; somit erhält auch sie Zugang zu den Objekten. Die übernommene Berechtigung benötigt sie lediglich für die Datenbibliothek.

2.5.3.2
Seite 6

Diese Berechtigung verfällt, sobald die Anwenderin zum Ausgangsmenü zurückkehrt.

Falls sie Query aus der Lohn- und Gehaltsanwendung heraus aufrufen möchte, ist auch das kein Problem. In diesem Fall muss sie darauf achten, dass das Startprogramm „QRY_STR“ keine Rechte im Stack übergeordneter Programme übernimmt – das heißt: Der Parameter USEADPAUT des Programms „QRY_STR“ muss auf *NO gesetzt werden.

Auch wenn die Lösung im ersten Moment vielleicht etwas komplex erscheint, werden Sie sehr schnell feststellen, dass der anschließende Verwaltungsaufwand minimal ist. Sie könnten den Verwaltungsaufwand weiter minimieren, indem Sie für die erforderlichen *USE-Berechtigungen der diversen Objekte ein Gruppenprofil oder eine Berechtigungsliste einsetzen. Zukünftig ist es dann ausreichend, die Mitarbeiter der Lohn- und Gehaltsabteilung mit Query-Zugriff über das entsprechende Anfangsmenü zur Verfügung stellen – alles andere funktioniert automatisch!

Berechtigungsübernahmen kontrollieren

Wenn Sie sich jetzt entschließen, mit Berechtigungsübernahmen zu arbeiten, sollten Sie nicht vergessen, die Übernahmen von Zeit zu Zeit zu kontrollieren, indem Sie sich die Berechtigungsübernahmen anzeigen lassen. Der Befehl „Übernommene Programmberechtigungen anzeigen“ (DSPPGMADP) zeigt alle Objekte, die die Berechtigungen eines speziellen Benutzerprofils übernehmen.

```
                    Programmübernahme anzeigen (DSPPGMADP)

Auswahl eingeben und Eingabetaste drücken.

Benutzerprofil . . . . . > LG_APOWN     Name
Objektart  . . . . . .     *ALL         *ALL, *PGM, *SQLPKG, *SRVPGM
     + für weitere Werte
Ausgabe  . . . . . .       *           *, *PRINT, *OUTFILE
```

DSPPGMADP

Das Ergebnis

```
                          Programmübernahme anzeigen

Benutzerprofil . . . . . :   LG_APOWN

                                    ASP-
Objekt      Bibliothek  Art         Einheit     Attribut    Text
LG_STR      DIV_OBJ     *PGM        *SYSBAS     CLP         Startprogramm LG
```

Ergebnis des Befehls DSPPGMADP

Mir erscheint zur Überprüfung der Befehl „Objekte mit Berechtigungsübernahmen drucken“ (PRTADPOBJ) am geeignetsten, da hier Berichte aller Objekte gedruckt werden können. Sie erhalten zwei Berichte je Benutzerprofil. Der erste Bericht ist ein Gesamtbericht, der sämtliche Objekte beinhaltet, die die Berechtigungen des Benutzerprofils übernehmen. Der zweite Bericht enthält Änderungen – das heißt: Sie erhalten einen Überblick über die Objekte, die bei der letzten Ausführung des Befehls „PRTADPOBJ“ noch nicht zu den Objekten mit Berechtigungsübernahme für dieses Benutzerprofil gehörten. Der zweite Bericht erscheint nicht, wenn Sie das erste Mal den Befehl nutzen. Außerdem sollten Sie bedenken, dass die überprüften Benutzerprofile für die Dauer der Befehlsausführung gesperrt sind.

2.5.3.3
Seite 2

```
                    Obj. mit Ber.-Übernahme druck. (PRTADPOBJ)

Auswahl eingeben und Eingabetaste drücken.

Benutzerprofil . . . . . . . . .   *all        Name, generisch*, *ALL
Nur Änderungsbericht . . . . . .   *NO         *NO, *YES
```

Befehl PRTADPOBJ

```
Benutzerprofil . . . . . . . . :   LG_APOWN
Sonderberechtigungen . . . . . :   *NONE
--------------Objekt---------------    --------------Bibliothek--------------
                          Allgem.                                Allgem.     Persönl.
Name          Art         Berecht.    Name          ASP-Einh.    Berecht.    Berecht.
LG_STR        *PGM        *EXCLUDE    DIV_OBJ       *SYSBAS      *EXCLUDE       N
```

Ergebnis des Befehls PRTADPOBJ

Diese Betriebssystembefehle sind im Übrigen auch im Menü „SECBATCH“ hinterlegt.

```
SECBATCH    Sicherheitsberichte zur Stapelverarb. übergeben/planen
                                                          System:   EPOSI501
Auswahlmöglichkeiten:

  Berichte zur Stapelverarbeitung übergeben
     1. Objekte mit Berechtigungsübernahme
     2. Protokolljournaleinträge
     3. Berechtigungen für Berechtigungsliste
     4. Befehlsberechtigung
     5. Persönliche Befehlsberechtigung
     6. DFV-Datenschutz
     7. Verzeichnisberechtigung
     8. Persönliche Verzeichnisberechtigung
     9. Dokumentberechtigung
    10. Persönliche Dokumentberechtigung
    11. Dateiberechtigung
    12. Persönliche Dateiberechtigung
                                                                Weitere ...
Auswahl oder Befehl
===> _____________________________________________________________________
F1=Hilfetext   F3=Verlassen   F4=Bedienerführung   F9=Auffinden
F12=Abbrechen
```

Menü SecBatch

Nutzen Sie doch einfach diese Möglichkeiten, um potentielle Sicherheitsrisiken aufgrund von Berechtigungsübernahmen auf Ihrem System aufzuspüren!

Workmanagement

Workmanagement ist ein grundlegender Baustein des i5/OS-Betriebssystems. Workmanagement-Funktionen sind die Basis für die Ausführung aller Aufgaben, die im System eintreffen, ausgeführt und abgeschlossen werden. Dabei ist es egal, ob es sich um einen Batch-Job handelt, der nur einmal wöchentlich benötigt wird, oder ob es sich um eine Anwendung handelt, die täglich ausgeführt wird. Erst das Workmanagement ermöglicht die Verarbeitung des Arbeitsauftrags, indem die benötigten Ressourcen und die Arbeitsumgebung für den Auftrag zur Verfügung gestellt werden. Ein iSeries Server ist natürlich weitgehend vorkonfiguriert und kann sofort eingesetzt werden, und viele Benutzer müssen die Standardeinstellungen des Servers nie verändern. Doch falls Sie eine maßgeschneiderte Arbeitsumgebung auf Ihrem iSeries Server wünschen, sollten Sie verstehen, wie die Verarbeitung der Arbeitsaufträge auf Ihrem Server abläuft. Dabei ist es notwendig die zugehörigen Terminologien und Konzepte kennen zu lernen und Sie sollten das Zusammenspiel der unterschiedlichen Objekte verstehen. Nur so ist es möglich, die Umgebung individuell zu gestalten, um die optimale Performance bei der Verarbeitung Ihrer iSeries Jobs zu erzielen.

Grundsätzlich lässt sich der Begriff Workmanagement in fünf grundlegende Bereiche gliedern:

- Jobs
- Jobwarteschlangen
- Subsysteme
- Speicherpools
- und Ausgabewarteschlangen

Jeder der Bereiche hat eigene Konzepte und eine eigene Terminologie, und jeder Bereich für sich betrachtet produziert eigene Ergebnisse. Erst im Zusammenspiel aller Komponenten erhalten Sie eine leistungsfähige Arbeitsumgebung. Die nachfolgenden Kapitel sollen helfen, die einzelnen Komponenten kennen zu lernen. Dabei werde ich sowohl das 5250-Interface verwenden als auch iSeries Navigator einsetzen, denn einige Aufgaben sind heute mit der grafischen Benutzeroberfläche sehr viel einfacher und übersichtlicher zu lösen.

Bevor wir allerdings einen Job auf seinem Weg durch das System verfolgen, sollten wir klären, was wir unter dem Begriff JOB zu verstehen haben:

17. Ergänzung 12/2008

Jede anfallende Arbeit wird von der iSeries als Job umgesetzt. Wenn Sie also einen Arbeitsauftrag für den Server haben, wird zunächst ein Job initiiert. Parallel mit der Initiierung des Jobs wird ein initialisierender Thread gestartet, der eine eigenständige Arbeitseinheit innerhalb des Jobs darstellt. Im weiteren Verlauf der Verarbeitung kann der Job zusätzliche Threads aufbauen, die Sekundär-Threads. Manchmal wird der Begriff Thread auch gleichgesetzt mit dem Begriff Prozess. Grundsätzlich lässt sich sagen, dass der Thread als unabhängige Arbeitseinheit innerhalb eines Jobs abläuft und dass ein Job aus mehreren Threads bestehen kann. Threads übernehmen Teilaufgaben, die für die Verarbeitung des Jobs notwendig sind. So kann ein Thread z. B. während der Verarbeitung eines Jobs dafür verantwortlich sein, die benötigten Daten abzurufen, die erforderlich sind, um den Job zu beenden. Letztendlich helfen Threads mehrere Operationen parallel auszuführen während der Job läuft. Das Konzept der Threads verwendet die iSeries für die Jobverarbeitung seit der Betriebssystemversion 4. Sie werden sich zunächst aber nicht weiter mit Threads auseinandersetzen müssen, denn die meisten Jobs werden auf Job-Ebene gesteuert.

Da es viele Arbeitsaufträge auf dem iSeries Server gibt, werden Sie auch viele unterschiedliche Jobtypen auf dem Server finden. Hier eine kurze Übersicht:

Batch-Job
Batch-Jobs laufen typischerweise im Hintergrund mit einer relativ niedrigen Priorität zwischen dem Benutzer, der den Batch-Job startet, und dem laufenden Batch-Job findet keine Interaktion statt, sodass der Benutzer parallel zur Ausführung des Batch-Jobs ungestört weiterarbeiten kann. Ein einfacher Batch-Job wird an eine Jobwarteschlange übergeben und wartet dort auf die Verarbeitung, wobei dessen Jobpriorität und die Eingangsreihenfolge des Jobs in der Jobwarteschlange für die Verarbeitung entscheidend sind. Es gibt aber auch sofort zu verarbeitende Batch-Jobs, die viele Jobattribute des Parent-Jobs übernehmen und im gleichen Subsystem laufen wie der Elternjob. Derartige Batch-Jobs können schneller gestartet werden, da sie nicht über eine Jobwarteschlange initiiert werden.

Autostart-Job
Ein Autostart-Job startet automatisch, wenn das zugehörige Subsystem gestartet wird. Autostart-Jobs werden häufig für wiederkehrende Aufgaben eingesetzt, oder sie stellen zentralisierte Funktionen für die anderen Jobs im gleichen Subsystem zur Verfügung.

Prestart-Jobs
Prestart-Jobs starten, bevor eine Systemanforderung vorhanden ist. Man könnte auch sagen, dass diese Jobs vorsorglich im System initiiert werden, um eventuell eintreffende Anforderungen schnellstmöglich beantworten zu können, denn der anfordernde Job muss nicht auf den Job warten. Außerdem müssen vorgestartete Jobs nur einmal gestartet werden und können anschließend viele Anfragen bearbeiten, ohne erneut gestartet zu werden. Die meisten Client-Server-Applikationen nutzen Prestart-Jobs für die Client-Kommunikation. Derartige Jobs können durch den i5/OS-Befehl STRPJ (Start prestart Job) gestartet werden. In den meisten Fällen allerdings werden diese Jobs beim Start eines Subsystems initiiert.

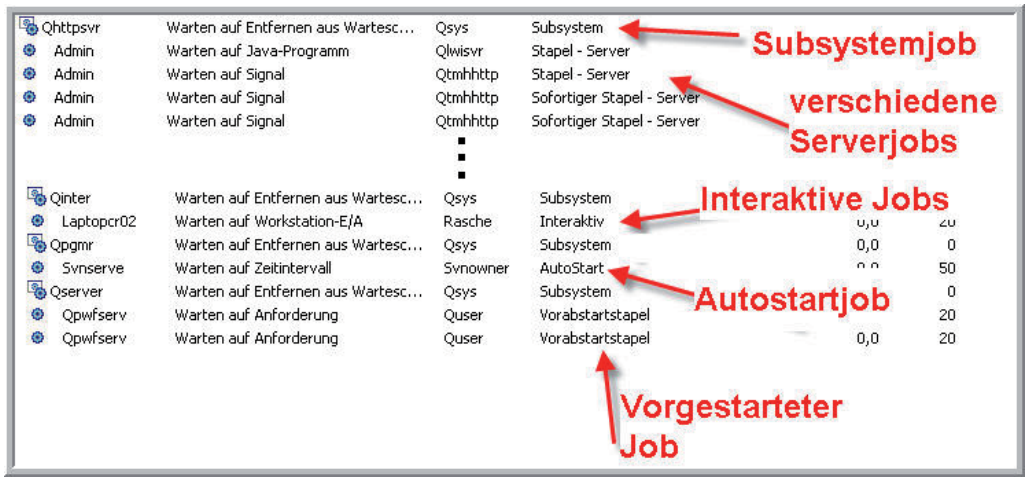
Communication-Jobs
Communication-Jobs werden gestartet, wenn eine Remote-Anforderung eintrifft. Aus Gründen der Performance wird heute häufig ein vorgestarteter Job für die Kommunikation genutzt, um zu verhindern, dass der Kommunikations-Job erst noch gestartet werden muss, wenn die Remote-Anfrage eintrifft

Interactive-Job
Ein interaktiver Job startet, wenn ein Benutzer sich auf dem iSeries Server anmeldet.

17. Ergänzung 12/2008

Subsystem-Job
Ein Subsystem-Job wird durch das Betriebssystem erstellt, um benötigte Ressourcen zu verwalten und um andere Jobs zu starten, auszuführen und zu beenden. Subsystem-Jobs erhalten dafür die Kontrolle über gestartete Subsysteme.

System-Jobs
System-Jobs werden vom Betriebssystem gestartet, um die Systemressourcen zu verwalten und um Systemfunktionen auszuführen. So gibt es Systemjobs, die für die Ausführung zeitgesteuerter Jobs verantwortlich sind. Andere werden zum Start und zum Beenden eines Subsystems benötigt. Eine detaillierte Erläuterung der Systemjobs finden Sie im iSeries Information Center. Systemjobs werden grundsätzlich gestartet, wenn der iSeries Server startet. Benutzereingriffe sind hierbei nicht erforderlich.



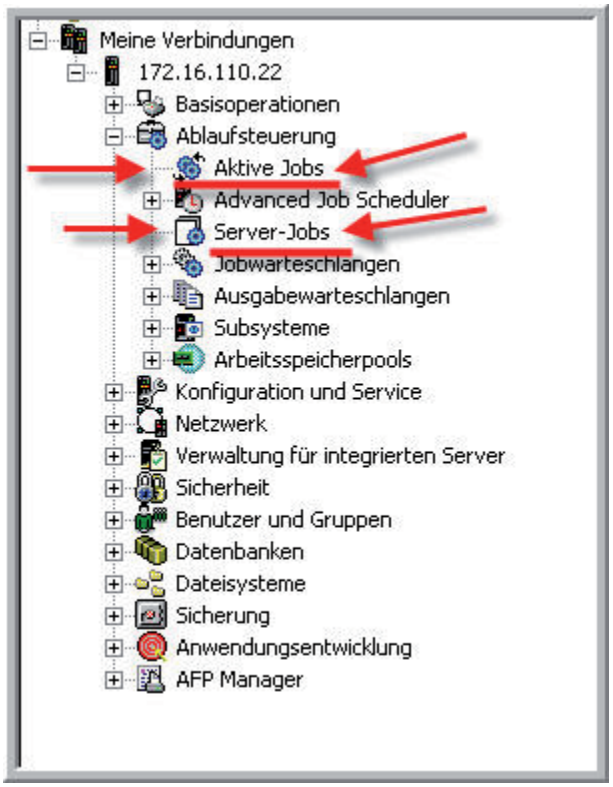
Die verschiedenen Jobtypen

Sie sehen, es gibt eine Vielzahl verschiedener Jobs auf Ihrer iSeries. Jeder Job enthält Informationen darüber, wie er im System ausgeführt werden soll. Diese Eigenschaften werden zugeordnet, wenn der Job erstellt wird. Auf alle Fälle können Sie sich die Jobattribute anzeigen lassen und / oder verändern. Ich verwende an dieser Stelle nicht die 5250-Oberfläche, wie man vermuten könnte, sondern nutze hierfür iSeries Navigator, da die grafische Oberfläche in diesem Bereich sehr viel komfortabler ist als vergleichbare i5/OS-Befehle. Bevor wir uns die Jobeigenschaften ansehen, eines vorweg: Die Jobeigenschaften kann sich grundsätzlich jeder ansehen. Um aber die Eigenschaften zu ändern, benötigen Sie in den meisten Fällen die Sonderberechtigung *JOBCTL. Eigenschaften der Systemjobs können grundsätzlich nicht mit iSeries Navigator verändert werden. Hierzu benötigen Sie weiterhin eine zeichenorientierte Oberfläche und den Befehl CHGSYSJOB (Change System Job).

2.6
Seite 6

Schauen wir uns an, welche Funktionen der iSeries Navigator uns bietet:

Wir öffnen also als erstes die entsprechende Serververbindung und anschließend den Eintrag „Ablaufsteuerung“:



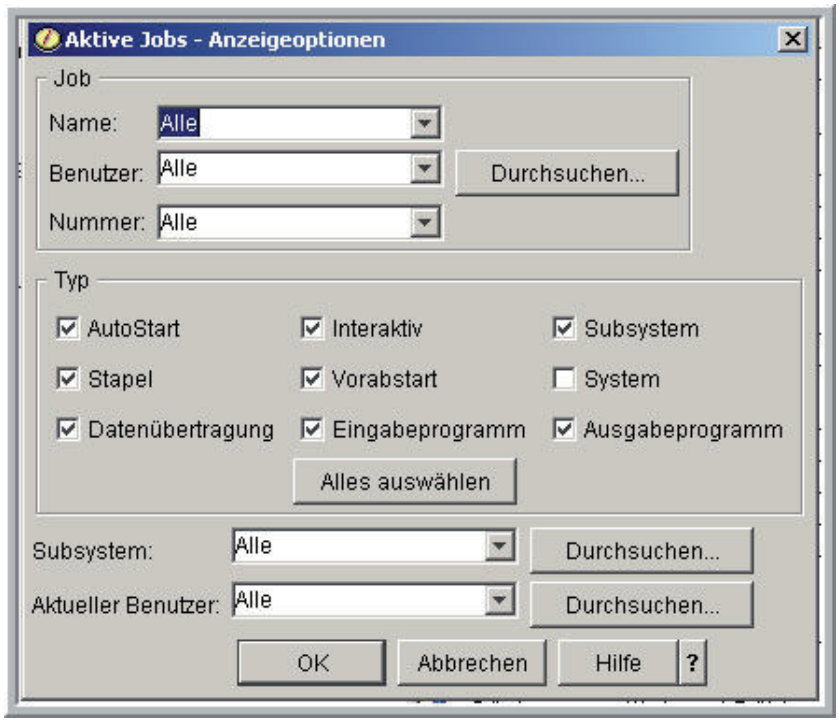
iSeries Navigator „Ablaufsteuerung“ öffnen

Der Begriff „Ablaufsteuerung“ ist in diesem Kontext meiner Meinung nach eher missverständlich. Im amerikanischen Original des iSeries Navigators wird diese Funktion mit dem Begriff „Workmanagement“ dargestellt. Dieser Begriff ist für mich deutlich aussagekräftiger als die deutsche Übersetzung. Durch Doppelklick auf den Punkt „Aktive Jobs“ bzw. „Server-Jobs“ erhalten Sie eine Übersicht der entsprechenden Jobs, je nachdem, welchen Eintrag Sie gewählt haben. Für die Funktion „Server-Jobs“ finden Sie auf der 5250-Befehlsebene der iSeries im Übrigen keine Entsprechung. Es handelt sich hierbei aber auch nicht um eine eigenständige, neue Funktion. Vielmehr ist diese Funktion eine Teilmenge der Funktion „Aktive Jobs“. Da die iSeries Systeme als Batch-orientierte Systeme anzusehen sind, sind z. B. Funktionen des Netzwerks und der Datenbank als Server-Jobs im Betriebssystem implementiert worden. In den letzten Betriebssystemversionen stieg die Zahl der Serverdienste permanent an und das in einem nicht unerheblichen Umfang. Im traditionellen Bildschirm-Interface ist die Darstellung der Serverdienste eher ungenügend, da Sie die Systemjobs lediglich innerhalb der WRKACTJOB-Anzeige betrachten können. Eine separate Darstellung im iSeries Navigator kann von daher nur begrüßt werden. Es mag jetzt den einen oder anderen

Skeptiker geben, der die erweiterten Parameter des Befehls WRKACTJOB überprüft und dabei feststellt, dass man auch dort auf die System-Jobs filtern kann: Probieren Sie es ruhig aus! Sie erhalten im traditionellen Umfeld leider nur die klassischen SYS-Jobs angezeigt. Remote Server-Jobs, vorgestartete Server-Jobs und dergleichen mehr werden mit dem Befehl *WRKACTJOB JOB(*SYS)* nicht angezeigt.

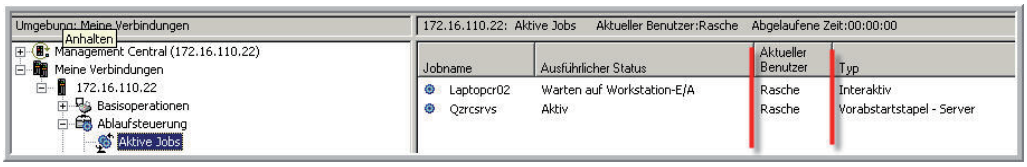
Ich öffne als nächstes die Anzeige „Aktive Jobs". Nach einem Doppelklick auf den Eintrag „Aktive Jobs" baut sich im Detailfenster auf der linken Seite des iSeries Navigators das scheinbar vertraute Bild der Anzeige WRKACTJOB auf. Soweit, so gut! In der Regel wollen wir aber gar nicht mit allen Jobs arbeiten, insbesondere dann nicht, wenn wir die Jobeigenschaften eines Jobs anzeigen und ggf. ändern wollen. Wir benötigen also ganz bestimmte Jobs und kommen somit automatisch auf die Frage: „Wie finde ich einen Job?"

In iSeries Navigator können Sie eine sehr detaillierte Suche starten, damit Sie nicht alle aktiven Jobs auf Ihrem System angezeigt bekommen. Hierzu nutzen Sie – wie in den Basisoperationen Kapitel 2.1.6 bereits erklärt – die Funktionstaste F11 „Anzeigeoptionen".



Die Suchkriterien

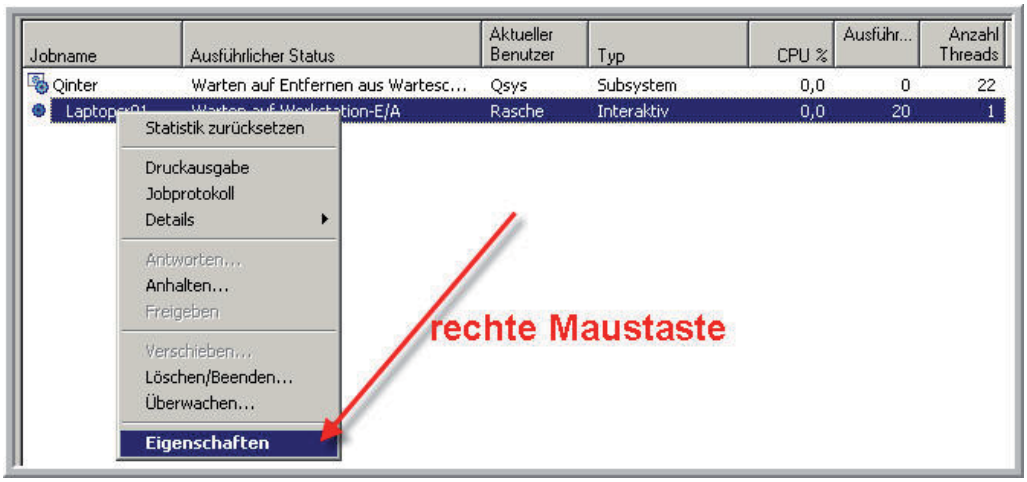
Der oben dargestellte Dialog bietet eine Vielzahl an Selektionskriterien. Schauen wir einmal, ob es funktioniert: Ich möchte mir nur die Jobs des Benutzers RASCHE anzeigen lassen und gebe dafür im Feld „Aktueller Benutzer" den entsprechenden Wert an. Das Ergebnis sieht anschließend folgendermaßen aus:



Das Suchergebnis

Aber Sie können die Anzeige nicht nur auf die Jobs eines bestimmten Benutzers beschränken. Die restriktivsten Suchkriterien sind Name, Benutzer, Nummer und ein einzelnes Subsystem. Die Angabe von Typ, Aktueller Benutzer und mehreren Subsystemen ist nicht so wirkungsvoll, aber immer noch besser, als sich hunderte von aktiven Jobs im System anzeigen zu lassen. Die Felder Name, Benutzer, Nummer, Subsystem und Aktueller Benutzer haben den Standardwert „Alle“. Sie können diesen Standardwert jederzeit wiederherstellen, indem Sie „Alle“ aus der Dropdownliste auswählen. Sie müssen bei der Suche auch nicht zwingend den vollständigen Suchbegriff eingeben, sondern Sie verwenden ein Wildcardzeichen(*) um alle Jobs anzuzeigen, die einen Teil Ihrer Kriterien enthalten. Wenn Sie z. B. Q* im Feld „Benutzer“ eingeben, werden alle Jobs angezeigt, die mit Q beginnen. Solche Platzhalter können Sie allerdings nur am Ende einer Zeichenkette verwenden. Bei den Feldern Benutzer, Subsystem und Aktueller Benutzer können Sie zusätzlich auf die entsprechende Schaltfläche „Durchsuchen“ klicken. Über diese Schaltfläche wird eine Anzeigeliste geöffnet, aus der Sie die gewünschten Kriterien auswählen können. Wenn Sie mehrere Elemente auswählen möchten, halten Sie die rechte Steuertaste (STRG) beim Klicken gedrückt. Wenn Sie dann auf OK klicken, werden die Werte eingefügt. Die angegebenen Anzeigebedingungen bleiben bis zu deren Änderung in Kraft. Verglichen mit den Selektionsmöglichkeiten des Befehls WRKACTJOB können wir hier von echtem Luxus sprechen.

Ich möchte mir jetzt die Eigenschaften meines interaktiven Jobs anschauen. Dazu markiere ich den vorhin gesuchten Job mit der rechten Maustaste und wähle den Eintrag „Eigenschaften“:



Eigenschaften eines Jobs anzeigen

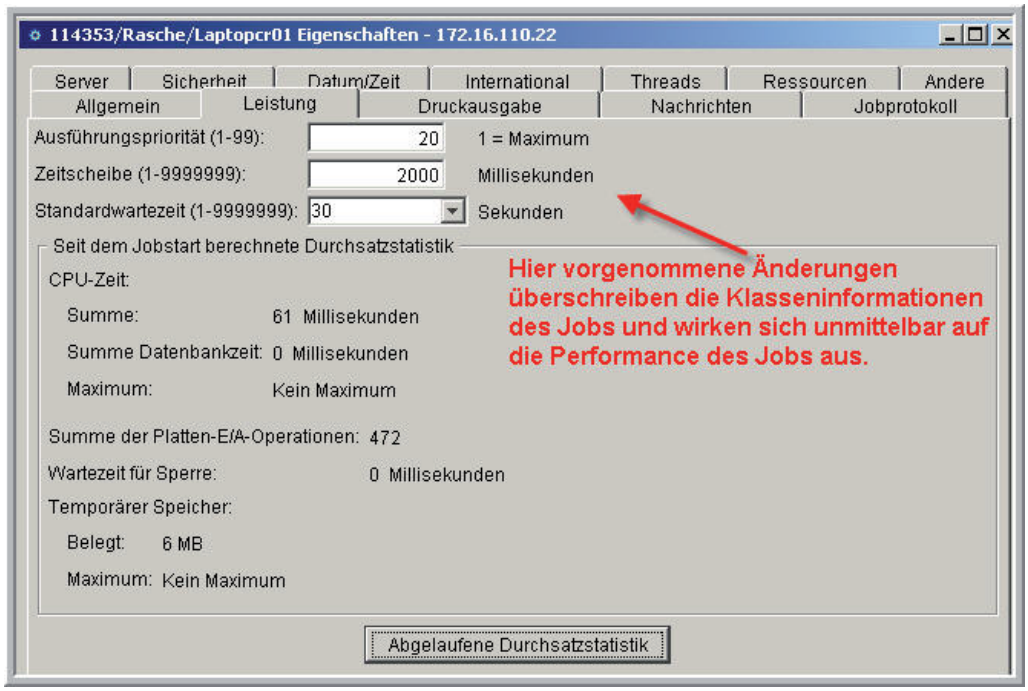
Der Eigenschaftsdialog ist vergleichbar mit dem i5/OS-Befehl DSPJOB (Display Job) und zeigt mir die Kenndaten meines interaktiven Jobs. Anders als bei der 5250-Bildschirmausgabe werden die Eigenschaften aber nach verschiedenen Kriterien gruppiert:

Das Register „Allgemein“
Im Register „Allgemein“ finden Sie die grundlegenden Jobidentifikationsdaten: den Jobnamen, den Jobtyp, den aktuellen Benutzer des Jobs, den derzeitigen Status, das vom Job aufgerufene Programm, den Zeitpunkt des Eintritts in das System und den Zeitpunkt des Jobstarts.

2.6
Seite 10

Das Register „Leistung“

Im Register „Leistung“ werden performancerelevante Daten und Durchsatzstatistikdaten angezeigt. Schauen Sie sich den nachfolgenden Dialog an:

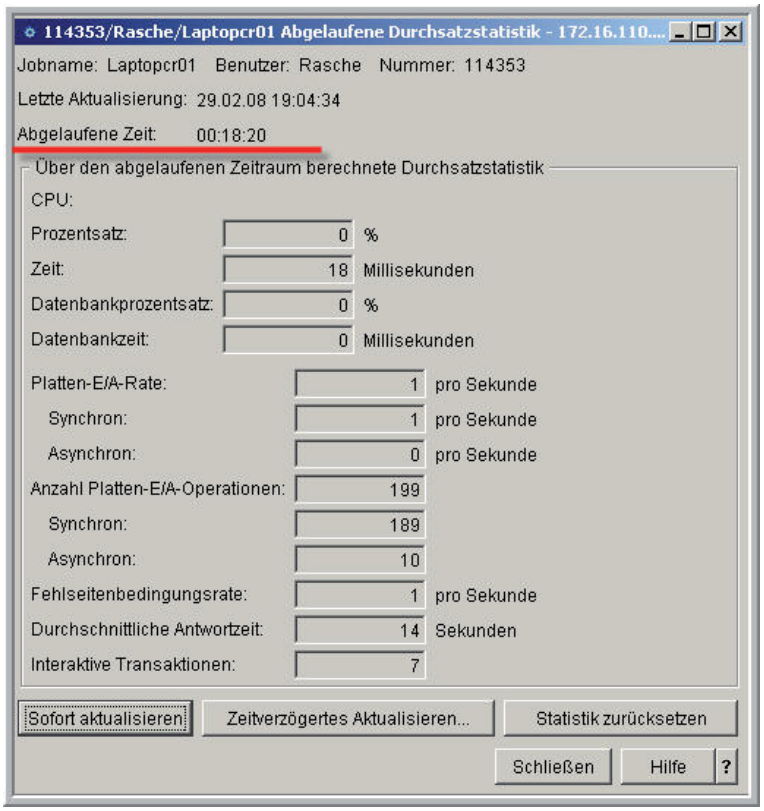


Eigenschaften – Leistung

In diesem Register finden wir also Basisinformationen bezüglich der Jobperformance. Sie sehen, dass mein Job mit der Priorität 20 ausgeführt wird. Maximal erhält jeder Thread meines Jobs 2000 Millisekunden Verarbeitungszeit, bevor andere Threads in diesem Job oder in anderen Jobs die Gelegenheit zur Verarbeitung bekommen. Außerdem müsste ich auf die Zuordnung von gesperrten Ressourcen 30 Sekunden warten. In diesem Dialog sind die ersten drei Parameterwerte änderbar. Ihre Änderung wirkt sich anschließend unmittelbar auf die Ausführung meines Jobs aus.

Im unteren Teil des Dialogs sehen Sie die aktuelle Durchsatzstatistik meines Jobs. Zu den aufgelaufenen Statistikwerten gehört die genutzte CPU-Zeit, die Anzahl der Plattenzugriffe und der belegte temporäre Speicher, seitdem der Job im System gestartet wurde.

Historische Durchsatzstatistikdaten sind hier auch verfügbar. Dafür müssen Sie allerdings auf die Schaltfläche „Abgelaufene Durchsatzstatistik“ klicken. Anschließend erscheint der folgende Dialog:



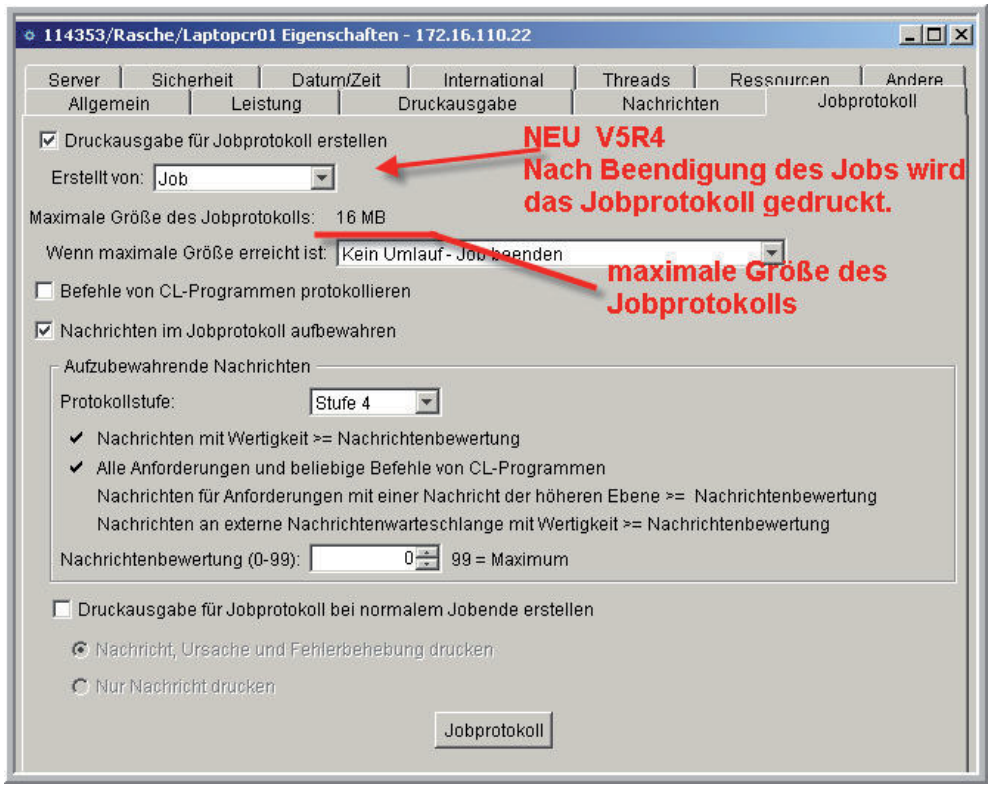
Abgelaufene Durchsatzstatistik

Über die Seite „Abgelaufene Durchsatzstatistik“ können Sie detaillierte Leistungsstatistikdaten eines Jobs oder Threads anzeigen, z. B.: Jobname, CPU-Statistik, Platten-E/A-Geschwindigkeiten und -Zähler sowie die Raten von Fehlseitenbedingungen. Die abgelaufene Durchsatzstatistik wird über ein Zeitintervall berechnet, das im Feld „Abgelaufene Zeit“ angegeben ist. Die abgelaufene Zeit beginnt, wenn das erste Mal innerhalb einer Verbindung diese Funktion aufgerufen wird. Sehen Sie sich in der Abbildung „Abgelaufene Durchsatzstatistik“ die Detailkopfzeile etwas genauer an. Dort finden Sie den Hinweis „Abgelaufene Zeit: 00:18:20“. Alle angezeigten Werte beziehen ihre Aussage relativ zu dieser abgelaufenen Zeit. Ein Job, für den 0% CPU-Leistung angezeigt wird, beansprucht nicht in diesem Moment 0% der CPU, sondern vielmehr hat er in den letzten 18 Minuten keine CPU-Leistung beansprucht. Im Grunde genommen kann man nicht einmal von den letzten 18 Minuten sprechen. Korrekterweise müssen wir von den letzten 18 Minuten vor dem letzten Messzeitpunkt sprechen. Da sich die Anzeige nicht selbstständig aktualisiert, kann die aktuelle Situation vom angezeigten Wert erheblich abweichen. Wollen Sie die Anzeige auf den aktuellen Stand bringen, so klicken Sie doppelt auf „Statistik zurücksetzen“. Die abgelaufene Zeit wird für diesen Job auf 00:00:00 gesetzt. Verwenden Sie die Schaltflächen „Sofort aktualisieren“ und „Zeitverzögertes Aktualisieren“ zum Refresh der Daten, d.h. die Daten werden für die abgelaufene Zeit neu berechnet.

Das Register „Ausgabe“
Das Register „Ausgabe“ erlaubt Ihnen die Druckereinstellungen des entsprechenden Jobs anzuzeigen und / oder zu ändern. Dabei können Sie über entsprechende Pull-Down-Menüs den Drucker und die Ausgabewarteschlange auswählen.

Das Register „Nachrichten“
Im Register „Nachrichten“ können Sie Eigenschaften zur Bearbeitung von Nachrichten für den Job anzeigen und ändern.

Das Register „Jobprotokoll“
Im Register „Jobprotokoll“ bearbeiten Sie Einstellungen des Jobprotokolls.



Eigenschaften – Jobprotokoll

Seit dem Release V5R4 existiert das Feld „Druckausgabe für Jobprotokoll erstellen“. Wenn dieses Feld aktiviert ist, wird bei Beendigung des Jobs ein Jobprotokoll erstellt. Außerdem können Sie Befehle in ausgeführten CL-Programmen protokollieren lassen und Sie bestimmen, was passieren soll, wenn das Jobprotokoll die maximale Größe von 16 MB erreicht. Mein interaktiver Job würde beendet werden, wenn das Jobprotokoll voll ist. Alternativ stehen die folgenden Parameterwerte zur Verfügung:

Umlauf – überschriebene Nachrichten drucken
Wenn das Jobprotokoll voll ist, werden Nachrichten am Anfang des Jobprotokolls hinzugefügt. Bevor neue Nachrichten hinzugefügt werden, werden die zu überschreibenden Nachrichten gedruckt.

Umlauf – überschriebene Nachrichten nicht drucken
Wenn das Jobprotokoll voll ist, werden Nachrichten am Anfang des Jobprotokolls hinzugefügt. Die überschriebenen Nachrichten werden nicht gedruckt.

Das Register „Server“
In diesem Register finden Sie Informationen zum Server-Job, d. h. Sie sehen die Jobbenutzeridentifikation, die für Berechtigungsprüfungen verwendet wird. Außerdem wird die IP-Adresse des Clients angezeigt.

Das Register „Sicherheit“
Das Register erlaubt Ihnen, die Sicherheitseigenschaften des Jobs anzuzeigen. Dies beinhaltet die Anzeige der Jobbenutzeridentifikation, den aktuellen Benutzernamen und die Gruppenprofile, die mit dem initialisierenden Thread assoziiert wurden.

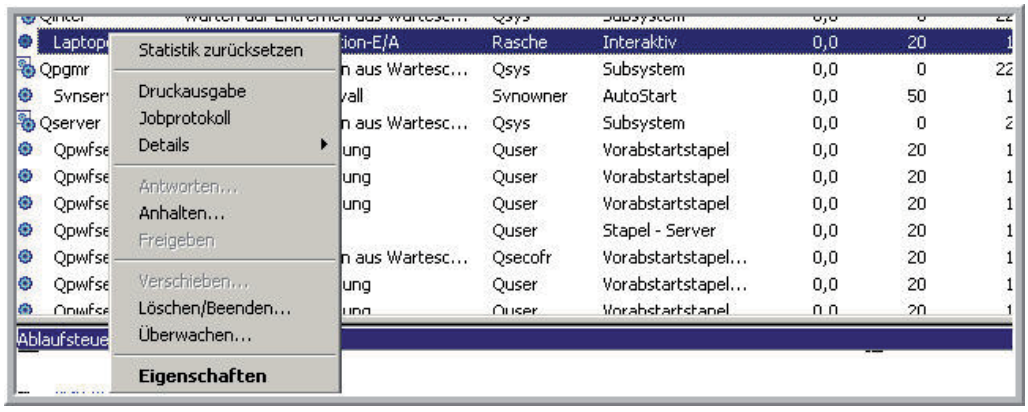
Das Register „Thread“
In diesem Register finden Sie Informationen über die zum aktuellen Job gehörenden Threads.

Eigenschaften – Threads

Wenn Sie jetzt in diesem Dialog auf den Button „Threads“ klicken, können Sie sich die einzelnen Threads dieses Jobs anzeigen lassen.

Sie sehen, dass der iSeries Navigator sehr kompakt alle Informationen über einen Job liefert. Vergleichen Sie die Funktionalität einmal mit dem i5/OS-Befehl CHGJOB (Change Job). Sie werden sehr schnell feststellen, dass einige der angezeigten Informationen identisch sind, aber es sind nicht alle Informationen vorhanden. Im 5250-Modus benötigen Sie eine Vielzahl weiterer Befehle um die gleichen Informationen zu erhalten.

Außerdem können Sie in iSeries Navigator zusätzlich das Kontextmenü nutzen, um Informationen über den Job zu erhalten, d. h. Sie müssen sich für einige Informationen nicht durch die einzelnen Register der Jobeigenschaften klicken. Markieren Sie einfach wahlfrei irgendeinen Job in der Detailanzeige und betätigen Sie dann die rechte Maustaste. Danach sehen Sie folgendes Kontextmenü.



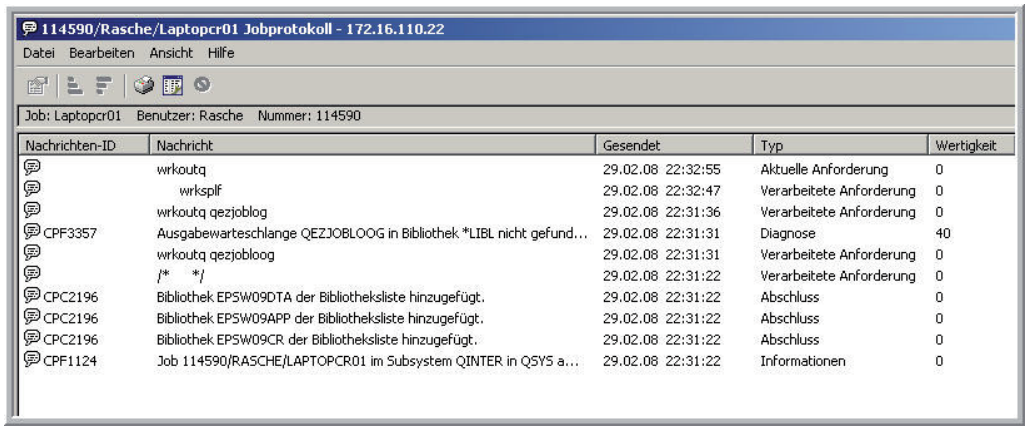
Das Kontextmenü eines Jobs

Folgende Verarbeitungsoptionen werden im Kontextmenü je nach Jobstatus angeboten:

„Statistik zurücksetzen“
Diese Funktion ist mit der soeben erläuterten Funktion identisch. Die „Abgelaufene Zeit“ wird für diesen Job auf Null gesetzt.

„Druckausgabe“
Dieser Punkt spricht für sich selbst. Erzeugt ein Job eine Druckausgabe, so kann sie mit Hilfe dieses Punkts eingesehen werden. Auch hier wird für die Anzeige der Ausgabedatei der AFP Workbench Viewer genutzt. Die Möglichkeiten kennen Sie bereits aus Kapitel 2.1.6

„Jobprotokoll“
Wählen Sie den Menüeintrag „Jobprotokoll“, so werden die jeweiligen Jobprotokolleinträge für den markierten Job angezeigt.



Das Jobprotokoll anzeigen

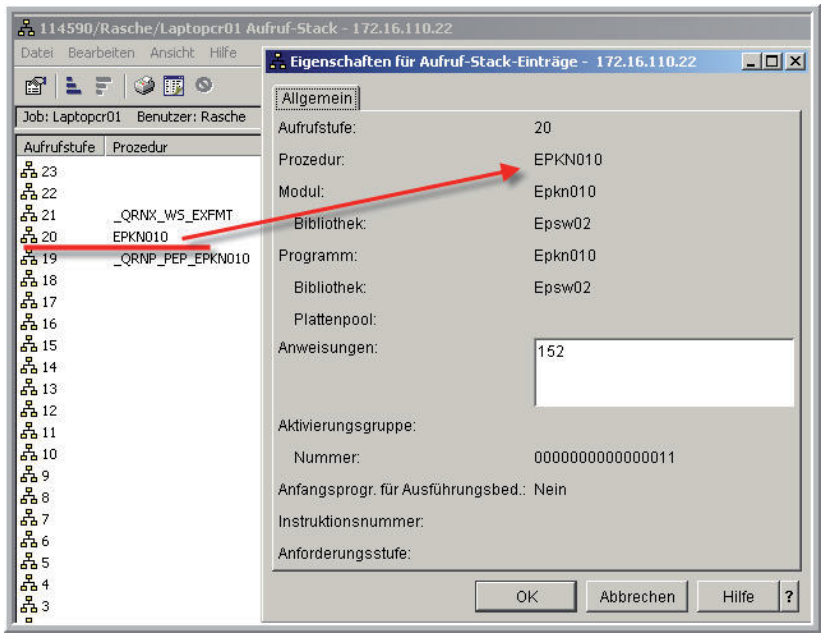
Die Ausgabe entspricht dem i5/OS-Befehl DSPJOBLOG (Display Joblog). Ein Doppelklick auf die jeweilige Nachricht zeigt diese in der Detailsicht.

„Details“

Im Job-Kontextmenü finden Sie die Auswahl „Details“. Der Punkt entspricht vermutlich am ehesten der Option 5 in der 5250-Anzeige des Befehls WRKACTJOB (Work with active Jobs). Wenn Sie diesen Menüpunkt wählen, öffnet sich zunächst ein Submenü:

Das Submenü der Detailanzeige

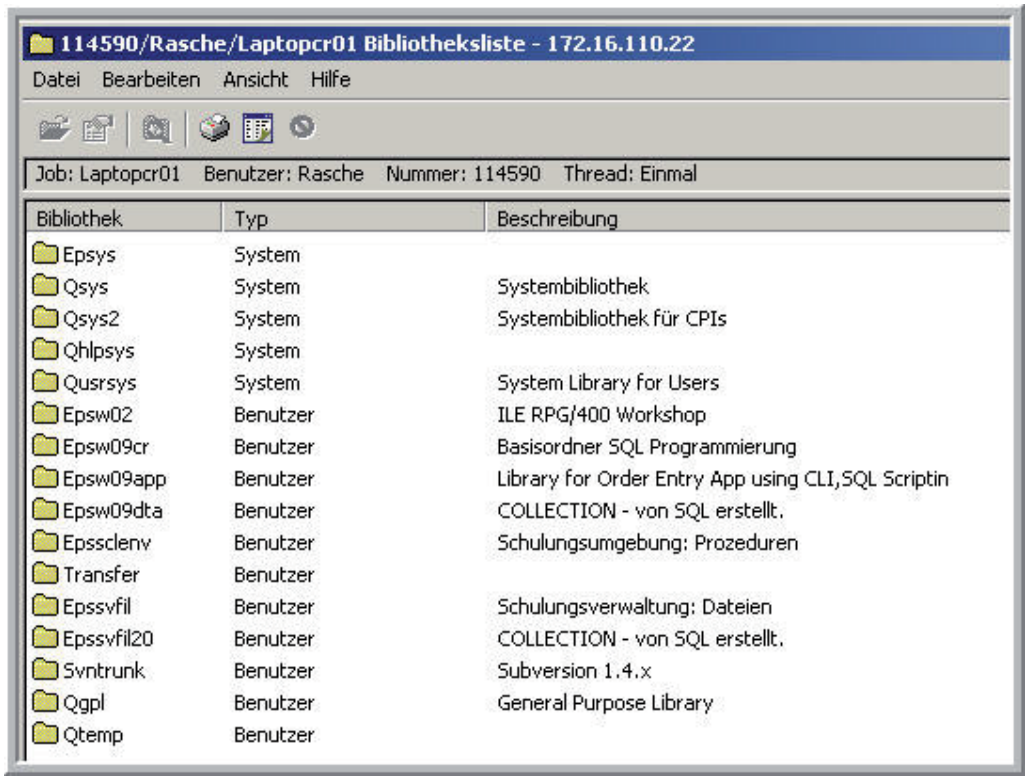
Klicken Sie doppelt auf die Auswahl „Aufruf-Stack“ und Sie sehen den Programmaufrufstapel für den jeweiligen Job. Gerät ein Job in eine Endlosschleife, so können Sie diesen Sachverhalt dieser Anzeige sehr gut entnehmen. Voraussetzung ist jedoch, dass Sie eifrig die F5-Taste zum Aktualisieren der Anzeige drücken. Interessiert Sie im Programmstapel der Zustand eines bestimmten Programms, so führen Sie einfach einen Doppelklick auf das Programm aus und die Eigenschaften des Programms werden angezeigt.



Der Aufruf-Stack

2.6
Seite 16

Schauen wir jetzt einmal, mit welcher Bibliotheksliste mein interaktiver Job arbeitet. Hierfür wählen wir den Menüpunkt „Bibliotheksliste“ in dem geöffneten Submenü „Details“:

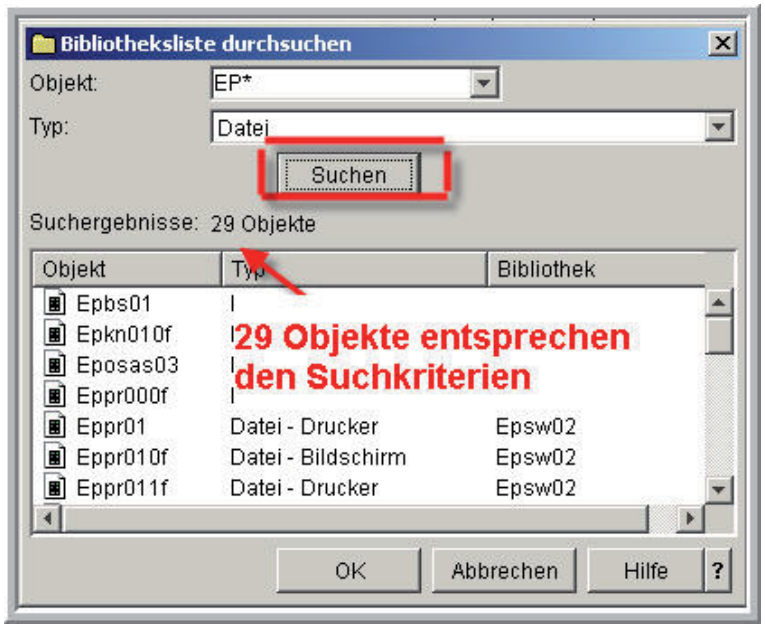


Die Bibliotheksliste des interaktiven Jobs

Sie wollen wissen, welche Objekte in der Bibliothek EPSW02 gespeichert sind? Kein Problem: Mit einem Doppelklick auf den Bibliotheksnamen öffnen Sie die Bibliothek und es werden alle Objekte der Bibliothek angezeigt. Sollte die Objektanzeige zu umfangreich werden, öffnen Sie zunächst das Kontextmenü der Bibliothek und wählen hier den Eintrag „Bibliotheksliste durchsuchen“:

Objekte in der Bibliotheksliste suchen

Mit Hilfe dieser Auswahl durchsuchen Sie die Bibliotheksliste gezielt nach ganz bestimmten Objekten.

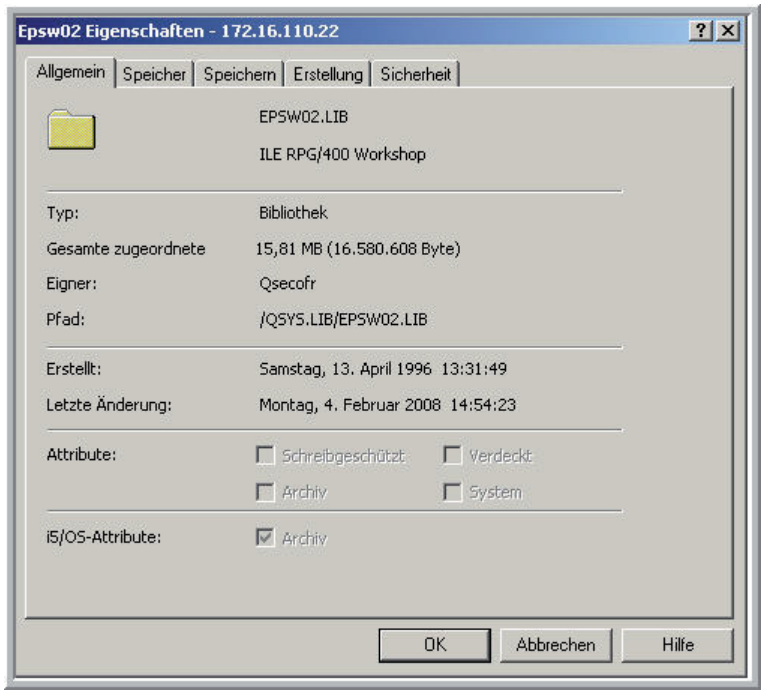


Das Ergebnis der Suche

Das Beste aber ist, dass Sie die angezeigten Objekte über ein weiteres Kontextmenü bearbeiten können.

Objekte bearbeiten

Das Kontextmenü variiert natürlich abhängig vom markierten Objekt, aber letztendlich können Sie sich jetzt immer weiter in die Objektbearbeitung „hineinklicken“. Sie können Berechtigungen neu setzen, das Objekt löschen u. v. m. Und falls Sie nicht mit Objekten einer Bibliothek arbeiten wollen, haben Sie auch die Möglichkeit, über das Kontextmenü mit der Bibliothek selbst zu arbeiten, d. h. die Eigenschaften der Bibliothek anzeigen lassen. So erhalten Sie umfangreiche Informationen über die Größe der Bibliothek, über die Sicherheitseinstellungen u.s.w.



Die Eigenschaften der Bibliothek EPSW02

So komfortabel ist der WRKACTJOB jedenfalls nicht! Bilden Sie sich selbst ein Urteil. Eine kurze Erläuterung des 5250 i5/OS-Befehls WRKACTJOB (Work with active Jobs) finden Sie im Kapitel 2.6.6.

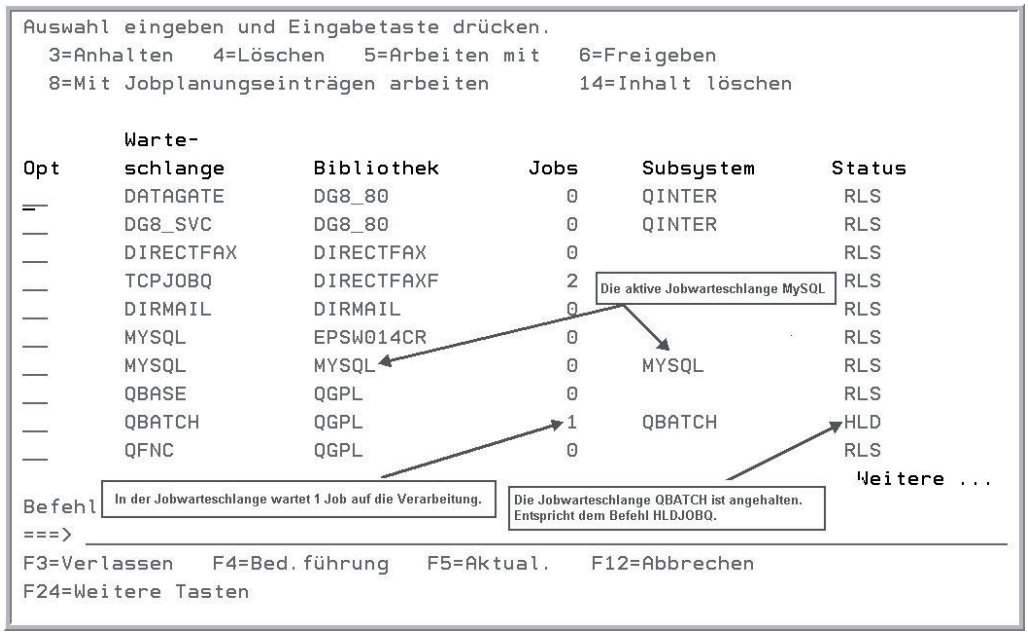
Jobwarteschlangen

Zum Bereich der Ablaufsteuerung im iSeries Navigator gehört ebenfalls die Arbeit mit Jobwarteschlangen. Jobwarteschlangen sind eine weitere zentrale Komponente des Workmanagements. Aber beginnen wir der Reihe nach und klären zunächst wieder die Frage:

„Was sind Jobwarteschlangen und welche Aufgaben erfüllen Sie im System?“

Jobwarteschlangen enthalten Jobs, die darauf warten in einem Subsystem verarbeitet zu werden. Die Jobwarteschlange ist somit für viele Jobs die erste Station, die sie durchlaufen, bevor ihre Verarbeitung beginnt. Zentrale Aufgabe der Jobwartesschlangen ist es, Jobs zu sammeln und in ein bestimmtes Subsystem zur Verarbeitung weiterzuleiten. Man könnte auch sagen, dass die Jobwarteschlange eine Jobsammelstelle ist.

Wenn Sie feststellen möchten, welche Jobwarteschlangen es auf Ihrem System gibt, geben Sie einfach den i5/OS-Befehl: WRKJOBQ an. Daraufhin erhalten Sie die folgende Ausgabe:



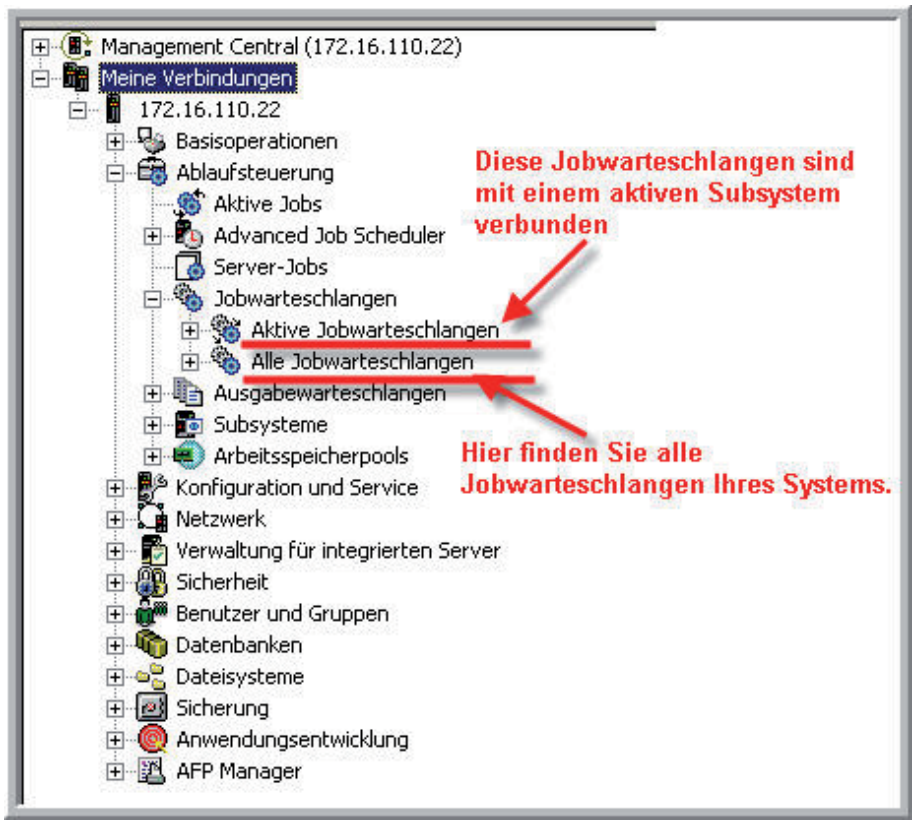
Mit Jobwarteschlangen arbeiten

Sie sehen, es gibt sehr viele Jobwarteschlangen auf einem iSeries System. Alle Jobwarteschlangen sind eigenständige Objekte der Objektart *JOBQ. Wie aber arbeiten Jobwarteschlangen? Wir wissen bereits, dass die Jobwarteschlangen mit den Subsystemen verbunden sind. Allerdings können Sie durchaus Jobs in Jobwarteschlangen stellen, deren Subsystem bislang noch nicht gestartet wurde. Sobald das Subsystem aktiviert wird, werden die Jobs verarbeitet. Wir müssen also zwischen aktivierten und deaktivierten Jobwarteschlangen unterscheiden, denn damit die Jobwarteschlange Jobs an ein Subsystem abgeben kann, muss es einerseits eine „physikalische“ Verbindung

2.6.1
Seite 2

zwischen einem Subsystem und der Jobwarteschlange geben und andererseits muss das Subsystem gestartet sein. In der oberen Abbildung sehen Sie, dass es bei einigen Jobwarteschlangen eine Verbindung zu einem Subsystem gibt, bei anderen nicht.

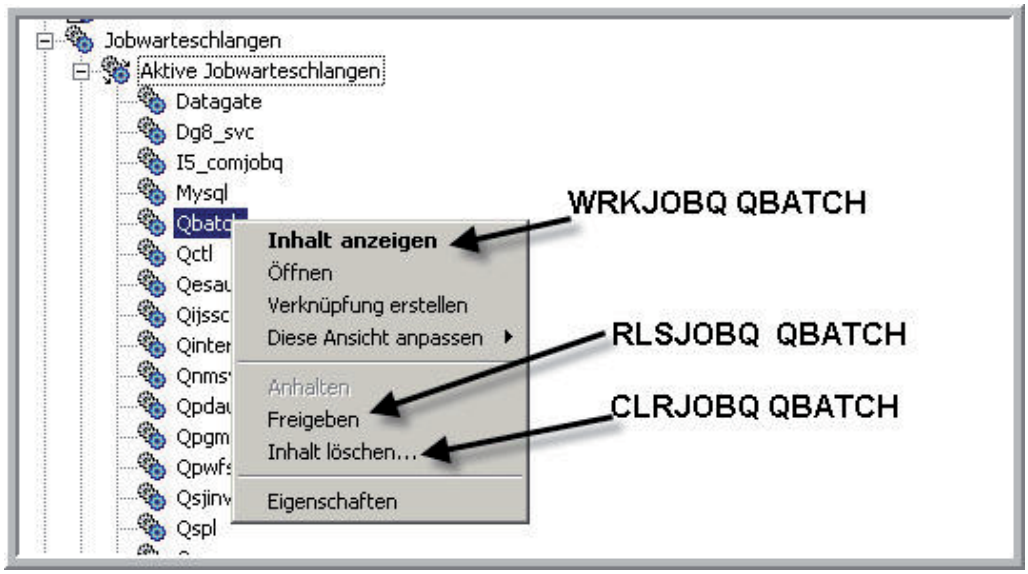
iSeries Navigator unterscheidet aus diesem Grund zwischen dem Eintrag „Jobwarteschlangen“ und „aktiven Jobwarteschlangen“. Öffnen Sie zunächst wieder die Verbindung zu Ihrem System, erweitern Sie anschließend den Eintrag „Ablaufsteuerung“ und klicken dann auf Jobwarteschlangen.



Mit Jobwarteschlangen im iSeries Navigator arbeiten

Unter der iSeries-Navigator-Funktion „Alle Jobwarteschlangen“ finden Sie die aktiven Jobwarteschlangen **plus** die Jobwarteschlangen, die derzeit keine Verbindung zu einem Subsystem unterhalten – Sie sehen somit ALLE Jobwarteschlangen, die es in Ihrem System gibt. Die Funktion entspricht der 5250-Anzeige des Befehls WRKJOBQ. Darüber hinaus verfügt iSeries Navigator über den Eintrag „Aktive Jobwarteschlangen“. „Aktive Jobwarteschlangen“ ist eine Teilmenge der Anzeige „Alle Jobwarteschlangen“ und beinhaltet nur Jobwarteschlangen, die mit einem aktiven Subsystem verbunden sind. Die Zuordnung der Jobwarteschlange zum Subsystem erfolgt im Übrigen beim Start des Subsystems. Nur wenn die Zuordnung zwischen Jobwarteschlange und Subsystem erfolgreich war, kann das Subsystem Jobs der entsprechenden Jobwarteschlange verarbeiten. Dabei kann ein Subsystem durchaus auch mit mehreren Jobwarteschlangen gleichzeitig verbunden sein.

Das Kontextmenü einer Jobwarteschlange bietet Ihnen zahlreiche Möglichkeiten der Verarbeitung. Es entspricht weitestgehend den Verarbeitungsoptionen des Befehls WRKJOBQ in der Abbildung „Mit Jobwarteschlangen arbeiten“ auf Seite 1 dieses Kapitels bzw. diversen i5/OS-Befehlen. Die vergleichbaren i5/OS-Befehle habe ich zum besseren Verständnis den Menüpunkten gegenübergestellt:



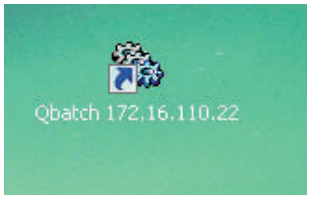
iSeries Navigator – Kontextmenü der Jobwarteschlange QBATCH

Schauen wir uns kurz die einzelnen Verarbeitungsmöglichkeiten an:

„Inhalt anzeigen“
Der Menüpunkt „Inhalt anzeigen“ ist im Grunde genommen überflüssig, da der Inhalt der Jobwarteschlange QBATCH bereits im rechten Fenster – der Detailansicht – angezeigt wird, sobald Sie auf den Eintrag QBATCH klicken.

„Öffnen“
Wählen Sie hingegen „Öffnen“, wird der Jobwarteschlangeninhalt in einem separaten Fenster dargestellt.

„Verknüpfung erstellen“
Sie können eine Verknüpfung auf dem Desktop erstellen, um sich sehr komfortabel den Inhalt der Jobwarteschlange jederzeit anzuzeigen.



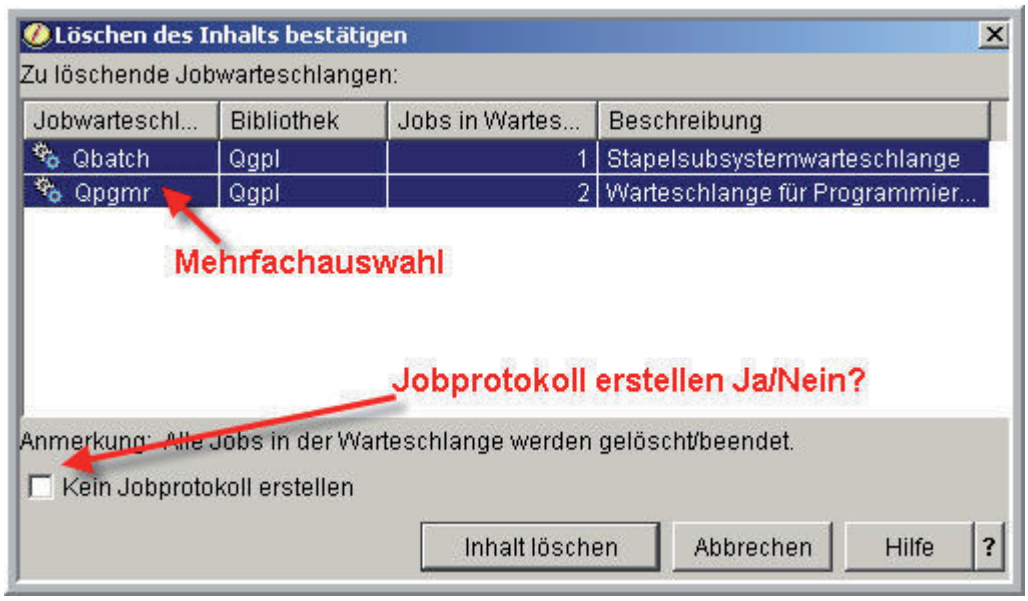
Inhalt der Jobwarteschlange QBATCH

Eine solche Verknüpfung kann im Tagesgeschäft Zeit sparen, da Ihnen sofort der Inhalt der entsprechenden Jobwarteschlange angezeigt wird.

„Anhalten“
Jobwarteschlangen können vorübergehend angehalten werden. Alle Jobs in der Warteschlange sind anschließend von der Verarbeitung ausgeschlossen solange bis die Jobwarteschlange wieder freigegeben wird. Derzeit ist der Eintrag „Anhalten“ deaktiviert, da die Jobwarteschlange bereits angehalten wurde.

„Freigeben“
Wenn eine Jobwarteschlange den Status HLD hat, müssen Sie die Warteschlange freigeben, damit Jobs der Warteschlange verarbeitet werden.

„Inhalt löschen“
Sollen alle Jobs innerhalb einer oder mehrer Jobwarteschlangen gelöscht werden, markieren Sie die Jobwarteschlangen und wählen anschließend über das Kontextmenü den Punkt „Inhalt löschen“. Daraufhin öffnet sich der Dialog „Löschen des Inhalts bestätigen“:



iSeries Navigator – Inhalt einer / mehrerer Jobwarteschlangen löschen

In dem oben dargestellten Dialog sehen Sie, dass ich beabsichtige den Inhalt der Jobwarteschlangen OBATCH und QPGMR zu löschen. Achtung: Die Mehrfachauswahl ist nur dann möglich, wenn die Jobwarteschlangen im Detailfenster angezeigt werden. Innerhalb der Baumstruktur ist keine Mehrfachauswahl möglich! Im Dialog „Löschen des Inhalts bestätigen“ sehen Sie daraufhin alle Jobs, die in den markierten Jobwarteschlangen warten. Sie müssen die Aktion bestätigen, indem Sie auf „Inhalt löschen“ klicken. Außerdem können Sie in diesem Dialog entscheiden, ob für die beendeten Jobs ein Jobprotokoll erforderlich ist. Klicken Sie auf „Kein Jobprotokoll erstellen“, um anzugeben, dass kein Jobprotokoll für jeden Job erstellt werden soll, der aus der Jobwarteschlange entfernt wird.

Wenn dieses Kontrollkästchen nicht ausgewählt ist, erfolgt die normale Verarbeitung jedes Jobs gemäß den Vorgaben in den Jobprotokolleigenschaften.

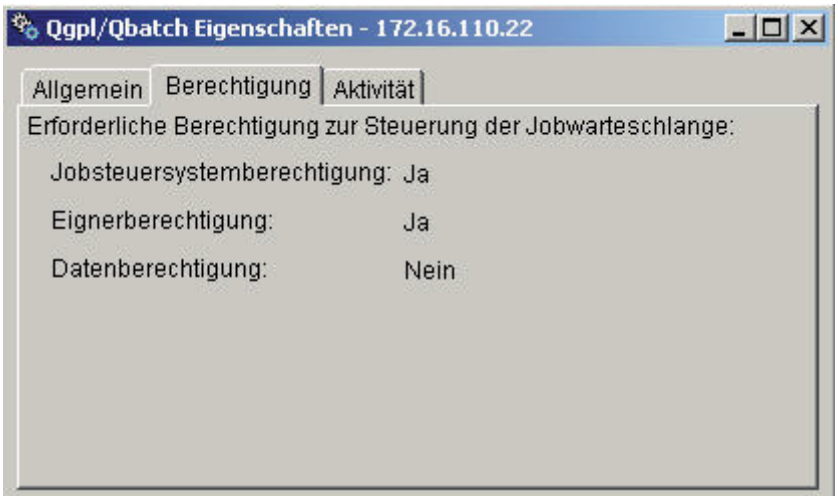
Eigenschaften
Der Punkt „Eigenschaften“ zeigt Ihnen die unterschiedlichen Eigenschaften der jeweiligen Jobwarteschlange.

Register „Allgemein“

Allgemeine Eigenschaften einer Jobwarteschlange

Der wichtigste Parameter im Register „Allgemein“ ist sicherlich die Statusanzeige. Der Status „Aktiv“ bedeutet in diesem Kontext, dass die Jobwarteschlange einem Subsystem zugeordnet ist. Falls Sie dagegen den Status „Nicht zugeordnet“ erhalten, besteht keine Verbindung zwischen der Jobwarteschlange und einem aktiven Subsystem. Alle weiteren Attribute des Registers „Allgemein“ sind meiner Meinung nach selbsterklärend und bedürfen keiner weiteren Erläuterung.

Register „Berechtigung“



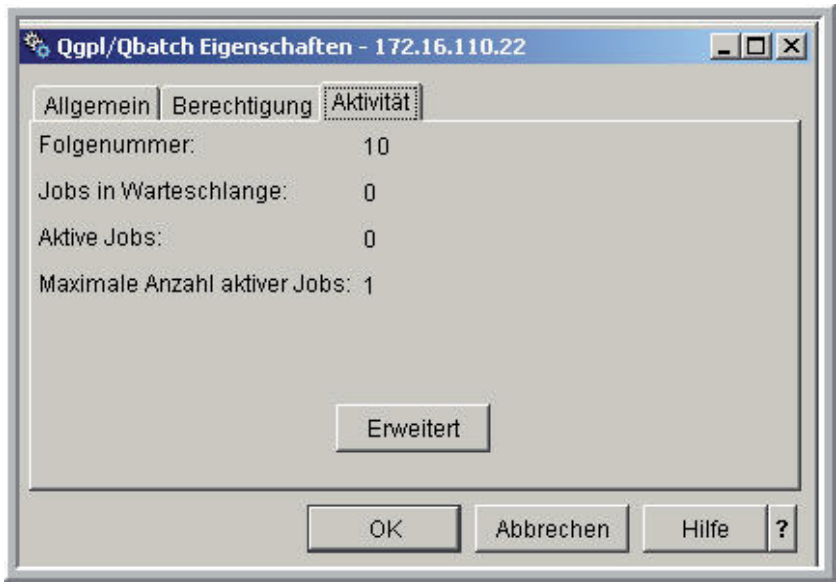
2.6.1
Seite 6

Auf der Registerseite „Berechtigung“ können Sie die Berechtigungen anzeigen, die für die Steuerung und für den Zugriff der Jobwarteschlange erforderlich sind. Schauen wir uns an, wie es sich mit der Jobwarteschlange QBATCH verhält:

Die Jobwarteschlange QBATCH und die Jobs in der Jobwarteschlange darf ein Benutzer mit der Sonderberechtigung *JOBCTL steuern (= Jobsteuerberechtigung). Der Eigner der Jobwarteschlange kann diese durch Anhalten und Freigeben steuern (= Eignerberechtigung). Mit der Eignerberechtigung kann ein Benutzer außerdem die Jobwarteschlange anzeigen, ihren Inhalt löschen sowie die Jobs in der Jobwarteschlange anhalten, freigeben, beenden und ändern.

Die Datenberechtigung definiert letztendlich, ob zur Steuerung der Jobwarteschlange durch Anhalten und Freigeben auch die Berechtigung zum Lesen, Hinzufügen und Löschen für die Jobwarteschlange erforderlich ist.

Das Register „Aktivität“

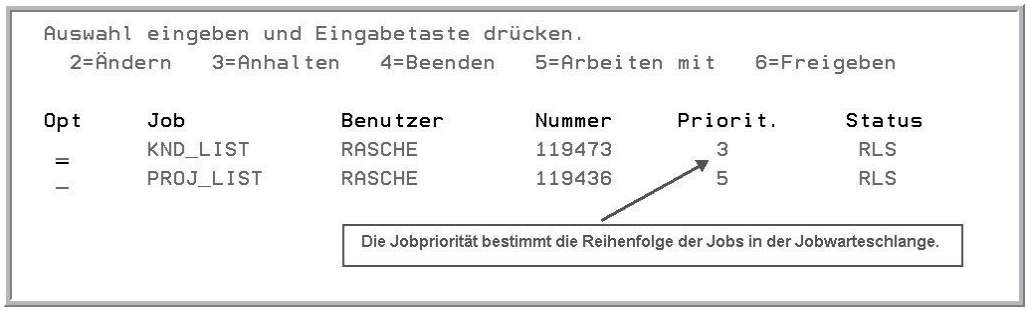


Das Register „Aktivität“

Im Dialog „Aktivität“ können Sie Informationen zur Aktivitätsauslastung der Jobwarteschlange anzeigen. Sie sehen die Folgenummer des Jobwarteschlangeneintrags, mit der das Subsystem die Verarbeitungsreihenfolge in der Jobwarteschlange bestimmt. Stellen Sie sich Folgendes vor: Ein Subsystem ist mit zwei Jobwarteschlangen verbunden. Die erste Jobwarteschlange erhält die Folgenummer 10, die zweite Jobwarteschlange erhält die Nummer 20. Eine niedrige Folgenummer zeigt eine hohe Priorität an. Die Folgenummer 1 zeigt beispielsweise an, dass die Jobwarteschlange zuerst verarbeitet wird. Sobald alle Jobs, die dieser Warteschlange zugeordnet sind, bearbeitet worden sind oder sobald die maximale Aktivitätsauslastung erreicht wurde, wird die Jobwarteschlange mit der Folgenummer 2 verarbeitet usw.

Außerdem können Sie der Anzeige entnehmen:
- die Anzahl der Jobs, die momentan in der Jobwarteschlange warten
- die Anzahl der Jobs, die aus der Jobwarteschlange stammen und momentan im Subsystem aktiv sind
- die maximale Anzahl Jobs aus dieser Jobwarteschlange, die gleichzeitig im Subsystem aktiv sein können

Manchmal ist eine Jobwarteschlange leer – wie in unserem Beispiel – und ein eintreffender Job kann sofort an das Subsystem weitergeleitet werden. Manchmal stehen aber – wie im nächsten Beispiel – bereits Jobs in der Warteschlange, die auf ihre Verarbeitung warten. Die Reihenfolge der Jobs in der Jobwarteschlange wird in solchen Fällen durch die Jobpriorität bestimmt. Wenn der Job gestartet wird, erhält jeder Job eine individuelle Jobpriorität. Als mögliche Werte stehen die Werte 0 bis 9 zur Verfügung, wobei der Wert 0 die höchste Priorität repräsentiert. Diese Jobpriorität bestimmt also anschließend die Reihenfolge der Jobs innerhalb der Jobwarteschlange. Haben hingegen alle Jobs die gleiche Priorität, werden sie nach dem Prinzip First-in-First-out an das Subsystem weitergereicht.

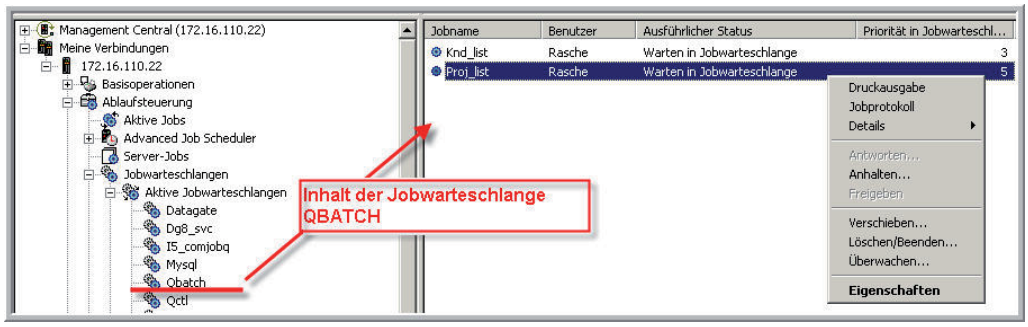


5250-Anzeige der Jobwarteschlange QBATCH

In der Jobwarteschlange QBATCH stehen jetzt zwei Jobs, der erste Job „KND_LIST“ wird als erstes bearbeitet werden, da er eine höhere Priorität hat als der Job „PROJ_LIST“. Sie können die Jobs innerhalb der Warteschlange manipulieren (Option 2), indem Sie den Job in eine andere Jobwarteschlange stellen oder die Jobpriorität ändern. Aber Vorsicht, die maximale Jobpriorität der Jobs wird durch den Parameter ‚Höchste Planungspriorität’ im jeweiligen Benutzerprofil begrenzt. Außerdem haben Sie die Möglichkeit, Jobs innerhalb der Jobwarteschlange anzuhalten (Option 3) oder aus der Jobwarteschlange zu entfernen. Wenn Sie einen Job in der Warteschlange anhalten, wird der Job solange nicht verarbeitet, bis Sie den Job wieder freigeben (Option 6). Der Status des Jobs wechselt in diesem Fall zwischen RLS (= freigegeben) und HLD (= gehalten).

2.6.1
Seite 8

Schauen wir uns zum Vergleich die Ausgabe im iSeries Navigator an:



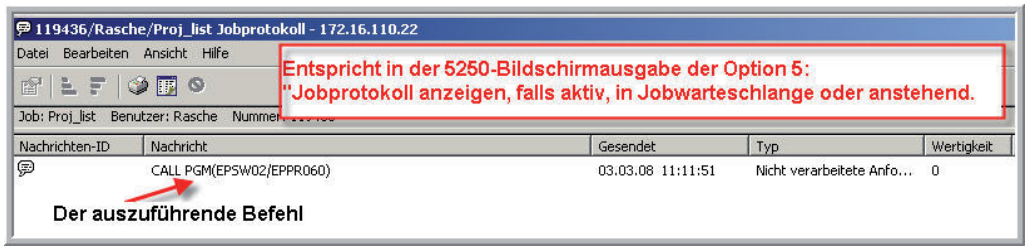
iSeries-Navigator-Anzeige der Jobwarteschlange QBATCH

Auch hier sehen Sie auf der rechten Seite die in dieser Warteschlange auf Verarbeitung wartenden Jobs. Im gezeigten Dialog sehen Sie, dass beide Jobs auf die Verarbeitung warten. Für den Job „PROJ_LIST“ lasse ich zudem die Verarbeitungsmöglichkeiten anzeigen. Im 5250-Bildschirm müssten Sie alternativ die Option 5 „Arbeiten mit“ nutzen. Ich werde mich im Folgenden auf die Erläuterung iSeries-Navigator-Anzeige konzentrieren, benenne aber – wenn möglich – die entsprechenden Befehle im 5250-Umfeld.

Schauen wir uns nun das Kontextmenü im Einzelnen an.

„Druckausgabe“
Da der Job derzeit in der Jobwarteschlange auf Freigabe wartet, konnte er noch keine Ausgabe erstellen. Folglich ist hier derzeit keine sinnvolle Anzeige zu erwarten. Im 5250-Bildschirm können Sie alternativ die Option 5 wählen und müssen dann den Eintrag „Mit Spool-Dateien arbeiten“ wählen. Doch auch in diesem Umfeld ist momentan keine sinnvolle Druckausgabe zu erwarten.

„Jobprotokoll“
In der Auswahl „Jobprotokoll“ sehen wir den zur Verarbeitung anstehenden Befehl.



iSeries Navigator – Jobprotokollanzeige

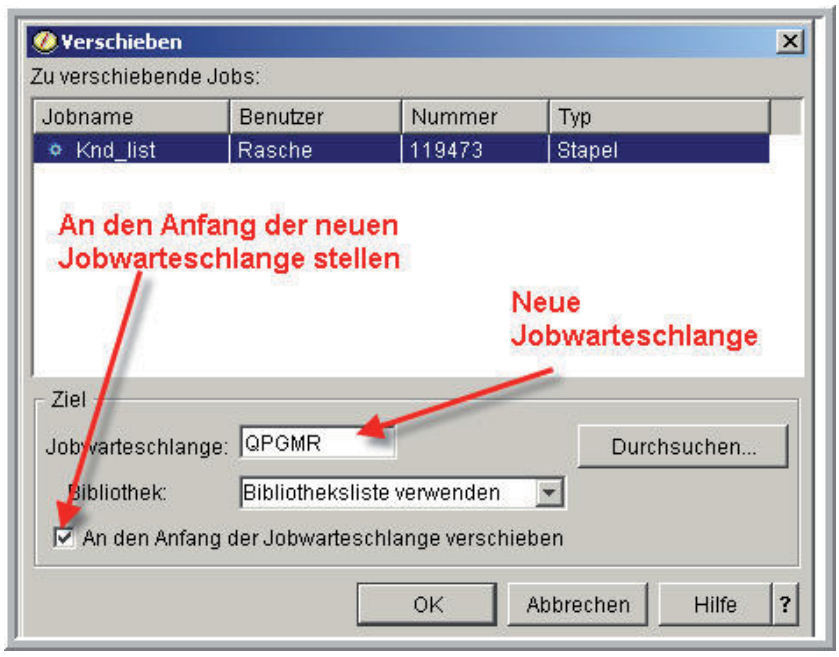
„Details“
Da der Job derzeit nicht aktiv ist, sind alle Punkte im Submenü „Details“ deaktiviert.

„Antworten“
Da der Job derzeit nicht aktiv ist, ist auch dieser Punkt deaktiviert.

„Anhalten“
Halten Sie den Job mit dieser Auswahl an. Wird der Job angehalten, bleibt er in der Jobwarteschlangen stehen bis er wieder freigeben wird. Dies entspricht der Option 3 in der Abbildung auf Seite 7.

„Freigeben“
Geben Sie den Job mit dieser Auswahl frei. Ist der Job freigegeben, wird die Auswahl „Freigeben“ deaktiviert. DAS entspricht der Option 6 in der Abbildung auf Seite 7.

„Verschieben“
Mit dieser Auswahl verschieben Sie den Job in eine andere Jobwarteschlange.



iSeries Navigator – Job verschieben

Im oben gezeigten Dialog verschiebe ich meinen Job „KND_LIST“ in die Jobwarteschlange QPGMR. Ein identisches Ergebnis erziele ich mit einem 5250-Bildschirm, wenn ich die Option 2 „Ändern“ wähle, anschließend die erweiterten Parameter öffne und dann den neuen Jobwarteschlangennamen einfüge. Ein Durchsuchen-Dialog – den der iSeries Navigator zur Verfügung stellt – existiert im 5250-Umfeld allerdings nicht. Außerdem habe ich im obe-

ren Dialog den Eintrag „An den Anfang der Jobwarteschlange verschieben“ markiert. Hierdurch bestimme ich, dass der Job an den Anfang der Zieljobwarteschlange verschoben werden soll. Wenn dieses Feld nicht markiert ist, bleibt der Job an seiner relativen Position in der Zielwarteschlange. Wenn dieses Feld markiert ist, wird der Job an den Anfang der Zielwarteschlange verschoben, wobei die aktuelle Priorität des Jobs entsprechend verändert wird. Ein Beispiel soll diesen Sachverhalt erläutern: Wenn der ausgewählte Job derzeit die Jobpriorität 5 hat und der erste Job in der Zielwarteschlange hat die Priorität 3, wird die Priorität des ausgewählten Jobs auf 3 geändert, und der ausgewählte Job wird vor die anderen Jobs in der Zielwarteschlange gestellt. Falls der ausgewählte Job derzeit angehalten ist, wird er freigegeben und ebenfalls ggf. mit Änderung der Priorität an den Anfang der Zielwarteschlange verschoben. Falls die erforderliche Jobpriorität allerdings die höchste Planungspriorität übersteigt, erhalten Sie eine Fehlermeldung und der Job wird nicht verschoben. Auch zeitgesteuerte Jobs lassen sich nicht auf diese Art verschieben. Auch dann erhalten Sie eine Fehlermeldung.

Eine andere, sehr viel einfachere Möglichkeit einen Jobs zu verschieben besteht darin, den Job per Drag&Drop in eine andere Jobwarteschlange zu ziehen. Dazu markieren Sie den Job und ziehen ihn dann mit gedrückter Maustaste in eine andere Jobwarteschlange. Sie sehen, sowohl der iSeries Navigator als auch das 5250-Umfeld bieten zahlreiche Möglichkeiten, einen Job im System zu verfolgen.

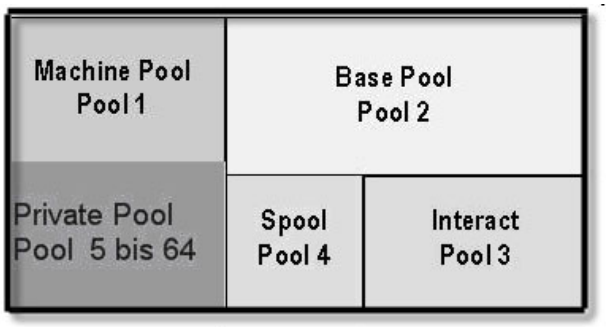
Abschließend einige Überlegungen zu möglichen Fehlersituationen. Wenn Ihre Jobwarteschlange keinen Job aktiviert, kann es dafür verschiedene Gründe geben:

- Wenn eine Warteschlange angehalten wird – Status HLD – kann die Warteschlange Jobs aufnehmen, jedoch nicht an das Subsystem weitergeben. Erst wenn die Jobwarteschlange wieder freigegeben wird – Status RLS –, gelangen die Jobs in das Subsystem zur Verarbeitung.
- Vielleicht ist aber auch nur ein einzelner Job angehalten. Lassen Sie sich in diesem Fall den Inhalt der entsprechenden Jobwarteschlange anzeigen. Wenn der Job den Status HLD zeigt, müssen Sie ihn freigeben.
- Ist das zugehörige Subsystem nicht aktiv (= gestartet), können auch keine Jobs bearbeitet werden. Sie müssen dann das Subsystem mit dem Befehl STRSBS aktivieren.
- Sie können eigene Jobwarteschlangen erstellen, aber diese Jobwarteschlangen aktivieren keine Jobs in einem Subsystem, da die physikalische Verbindung zum Subsystem nicht hergestellt wurde. Diese Verbindung müssen Sie mit dem Befehl ADDJOBQE herstellen. Unerwünschte Jobwarteschlangeneinträge entfernen Sie mit dem Befehl RMVJOBQE oder verändern die Parameter mit dem Befehl CHGJOBQE. Weiterführende Informationen hierzu finden Sie im Kapitel 2.7

Subsystembeschreibungen

Alle Jobs werden in Subsystemen verarbeitet. Jobs, die im Subsystem ausgeführt werden, erhalten einen Speicherbereich im Arbeitsspeicher zugewiesen. Der Arbeitsspeicher einer iSeries ist in mehrere Pools unterteilt. Ein Pool ist also ein Bereich des Hauptspeichers. Je mehr Arbeitsspeicher einem Pool zugeordnet wird, desto schneller können die Jobs in dem jeweiligen Arbeitsspeicherbereich verarbeitet werden. Grundsätzlich unterscheidet das System zwischen: **Shared Pools** und **Private Pools**. Ein Private Pool unterstützt maximal ein Subsystem während Shared Pools viele Subsysteme gleichzeitig unterstützen. Maximal können Sie 60 eigene Pools definieren. Notwendig sind zunächst einmal der Maschinen- und der Base-Pool. Beide Pools sind automatisch vorhanden und können nicht gelöscht werden. Im MACHINE-Pool laufen die System-Tasks. Darüber haben Sie keine Kontrolle. Die Speicherkapazität des Maschinenpools wird im Systemwert QMCHPOOL festgelegt. Die Größe des BASE-Pools können Sie nicht festlegen, denn der BASE-POOL erhält generell den Speicher, der von anderen Pools nicht benötigt wird. Es ist nur möglich im Systemwert QBASPOOL die Mindestgröße zu definieren. Zusätzlich zum Machine-Pool und Base-Pool existieren die Pools:

*INTERACT	Unterstützt interaktive Subsysteme.
*SPOOL	Unterstützt die Spool-Ausgabeprogramme.



Pools einer iSeries/i5

Die Speicherpools werden in den Subsystembeschreibungen den jeweiligen Subsystemen zugeordnet und, falls es sich um Private Pools handelt, auch definiert. Für jedes Subsystem existiert somit eine entsprechende Subsystembeschreibung. Die Subsystembeschreibungen sind Objekte vom Typ *SBSD.

CRTSBSD	Subsystembeschreibung erstellen
CHGSBSD	Subsystembeschreibung ändern
DLTSBSD	Subsystembeschreibung löschen
DSPSBSD	Subsystembeschreibung anzeigen
WRKSBSD	Mit Subsystembeschreibungen arbeiten

iSeries-Jobs laufen ausschließlich in Subsystemen. Subsysteme haben die Aufgabe, Jobs zu gruppieren und ihnen eine optimale Umgebung zu erstellen. Natürlich könnten Sie auch alle Jobs – unabhängig von der Jobart – in einem Subsystem verarbeiten, aber ein gut strukturierter Arbeitsspeicher beschleunigt die Verarbeitung. Denken Sie einmal an ein Freibad. Das Freibad hat nur ein großes Schwimmbecken, indem alle Besucher gleichzeitig aktiv werden wollen: Leistungsschwimmer, Jugendliche, Kleinkinder, alte und junge Menschen. Sie können sich leicht vorstellen, dass das Management des Freibads mit dieser Situation viel Arbeit haben wird. Ähnliche Aufgaben müsste auch das Arbeitsmanagement der iSeries bewältigen. Wenn Sie aber in dem Freibad verschiedene Pools organisieren, die den Anforderungen der Gäste entsprechen, ist der Arbeitsaufwand geringer. So geht es auch der iSeries. Die verschiedenen Subsysteme ermöglichen es, die Ressourcen optimal auf die Jobs zu verteilen. Alle Subsysteme werden erst aktiv, wenn sie gestartet werden. Alle Standardsubsysteme werden beim IPL automatisch gestartet. Verantwortlich ist das Programm QSTRUP in der Bibliothek QSYS. Dieses Programm ist im Systemwert QSTRUP als Startprogramm hinterlegt. Wenn Sie Ihre eigenen Subsysteme auch automatisch beim IPL starten wollen, benötigen Sie ein CL-Programm, das diese Aufgabe automatisiert. Um Ihr Problem zu lösen, schreiben Sie einfach ein eigenes Startprogramm.

Das Programm QSTRUP.
Achtung überschreiben Sie nicht das Originalprogramm!

```
        ****************** Datenanfang ************
0021.00       PGM
0022.00       CALL PGM(QSYS/QSTRUP)
0023.00       QSYS/STRSBS SBSD(QPGMR)
0023.01       MONMSG MSGID(CPF0000)
0023.02       QSYS/STRSBS SBSD(KDSBS01)
0024.00       MONMSG MSGID(CPF0000)
0025.00       QSYS/STRTCP
0026.00       MONMSG MSGID(CPF0000)
0043.00       ENDPGM
        ******************Datenende **************
```

Ihr eigenes Startprogramm

Das Programm erhält auch den Namen QSTRUP, wird aber in einer eigenen Bibliothek gespeichert. Sie sollten alle Betriebssystemmodifikationen – wie bereits besprochen – in einer eigenen „Systembibliothek“ speichern. Jetzt müssen Sie nur noch die Adressierung im Systemwert QSTRUP prüfen und ab sofort werden auch Ihre Subsysteme bei jedem IPL automatisch gestartet.

2.6.2
Seite 4

Jobklassen

Wenn der Job im Subsystem bearbeitet werden soll, benötigt das System die Laufzeitattribute. Jobklassen bestimmen diese Ausführungsattribute. Sie legen die Zuteilung von Systemressourcen – wie Hauptspeicher, Prozessorzuteilung, Auslagerung und Priorität – fest.

```
                        Klasseninformation anzeigen
                                                          System:   EPOSAS03
 Klasse  . . . . . . . . . . . . . . . . . . . . . . :   QBATCH
   Bibliothek  . . . . . . . . . . . . . . . . . . . :     QSYS
 Ausführungspriorität  . . . . . . . . . . . . . . . :   50
 Zeitscheibe in Millisekunden  . . . . . . . . . . . :   5000
 Auslagern möglich . . . . . . . . . . . . . . . . . :   *NO
 Standardwartezeit in Sekunden . . . . . . . . . . . :   120
 Maximale CPU-Zeit in Millisekunden  . . . . . . . . :   *NOMAX
 Maximaler temporärer Speicher in Megabyte . . . . . :   *NOMAX
 Maximale Anzahl Threads . . . . . . . . . . . . . . :   *NOMAX
 Text  . . . . . . . . . . . . . . . . . . . . . . . :   BATCH SUBSYSTEM CLASS
```

Zuteilung der Systemressourcen

Ablaufpriorität

Ablaufprioritäten sind für die Zuteilung von Systemressourcen verantwortlich. Sie entscheiden über die Zuteilung von Systemressourcen.

Zeitscheibe

Die Zeitscheibe beschreibt, wie viel Prozessorzeit einem Job während eines Verarbeitungsschrittes zur Verfügung gestellt wird.

Auslagern möglich

Der Parameter steuert das Verhalten des Jobs im Arbeitsspeicher, wenn die Abarbeitung nicht in einem Arbeitsschritt fertig gestellt werden konnte. Kann der Job nicht fertig gestellt werden, wird der Job auf die Festplatte ausgelagert, wenn dieser Parameter auf *YES steht.

Standardwartezeit

Die Standardwartezeit gibt bekannt, wie lange ein Job maximal auf die benötigten Ressourcen wartet, bevor eine Abbruchnachricht gesendet wird.

Maximale CPU-Zeit

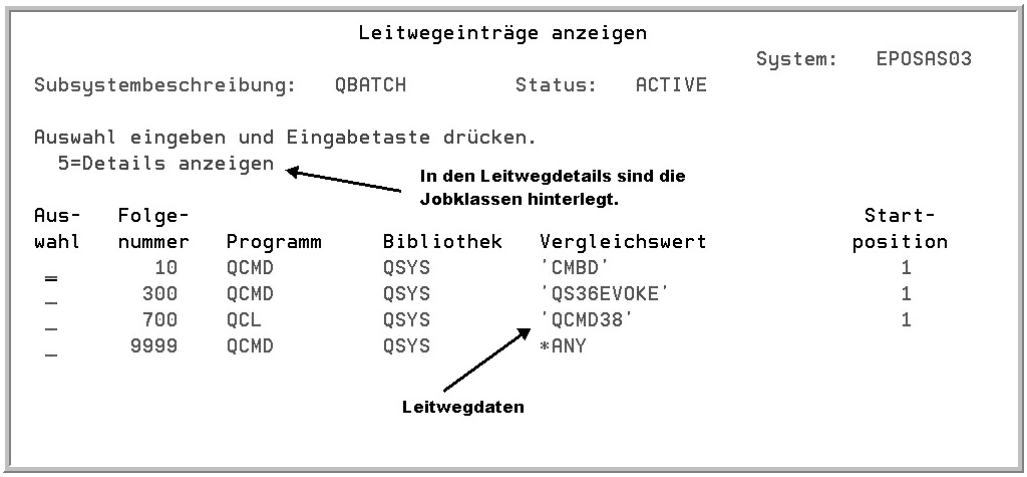
Die maximale CPU-Zeit bestimmt, wie viel Prozessorzeit ein Job maximal nutzen kann.

Die Zuteilung der Jobklasse erfolgt anhand der Leitwegdaten eines Jobs. Jeder Job verfügt über Leitwegdaten und in jeder Subsystembeschreibung sind Leitweg-Einträge hinterlegt. Wenn es eine Übereinstimmung zwischen den Daten

Grundwerk 10/2004

2.6.3
Seite 2

gibt, kann das Subsystem die Ausführungsattribute anhand der Jobklasse ermitteln.



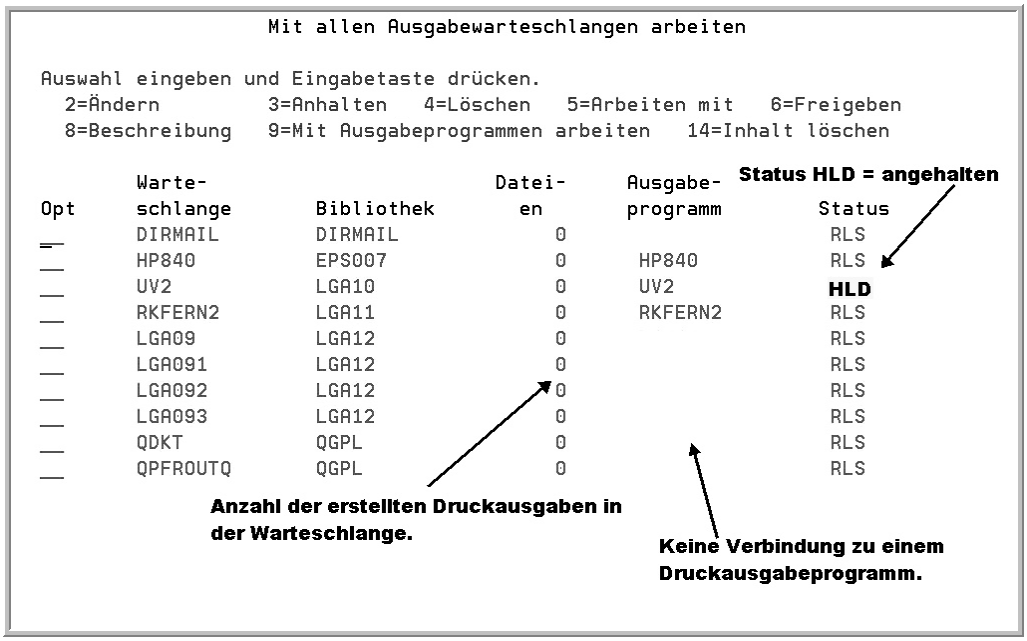
Ansicht der Leitwegeinträge

Ausgabewarteschlangen

Jeder Job, der Ausgaben produziert, muss die Druckausgaben in einer Ausgabewarteschlange ablegen. Die Ausgabe wird als „Spooled Output File“ (Spoolfile) bezeichnet. Ausgabewarteschlangen sind Objekte der Art *OUTQ.

CRTOUTQ	Ausgabewarteschlange erstellen
DLTOUTQ	Ausgabewarteschlange löschen
CHGOUTQ	Ausgabewarteschlange ändern
WRKOUTQ	Mit Ausgabewarteschlangen arbeiten
RLSOUTQ	Ausgabewarteschlangen freigeben
HLDOUTQ	Ausgabewarteschlangen anhalten
CHGSPLFA	Spoolfile-Attribute verändern

Ihre Aufgabe ist es, Druckausgaben zu koordinieren und einem Drucker zuzuleiten. Die Zuordnung der Druckausgabe zum Drucker erfolgt durch ein Druckausgabeprogramm. Druckausgabeprogramme sind Systemprogramme, die fertige Druckausgaben an den Drucker übertragen. Manchmal ist die Warteschlange leer und die Druckausgabe wird sofort an den Drucker weitergeleitet.



Koordination der Druckausgabe

Grundwerk 10/2004

2.6.4
Seite 2

Manchmal stehen aber auch bereits Spoolfiles in der Warteschlange, die auf die Verarbeitung warten. Jeder Spoolfile ist eine Priorität zugeordnet. Die Priorität bestimmt die Reihenfolge. Wenn alle Spoolfiles die gleiche Priorität haben, werden sie nach dem Prinzip „First-in-First-out“ zum Drucker geschickt. Sie können die Spoolfiles in der Warteschlange manipulieren, indem Sie die Ausgabepriorität verändern. Die maximale Ausgabepriorität Ihrer Spoolfiles wird durch den Parameter „Höchste Planungspriorität“ in Ihrem Benutzerprofil begrenzt. Außerdem haben Sie die Möglichkeit, die anderen Spoolfiles anzuhalten. Damit Sie Spoolfiles in der Warteschlange bearbeiten können, benötigen Sie die Sonderberechtigung *SPLCTL in Ihrem Benutzerprofil.

```
                  Spool-Dateiattribute ändern (CHGSPLFA)

Auswahl eingeben und Eingabetaste drücken.

Spool-Datei  . . . . . > QSYSPRT       Name, *SELECT
Jobname  . . . . . > LGA15         Name, *
  Benutzer . . . . . >   RASCHECR    Name
  Nummer . . . . . >   362592      000000-999999
Spool-Dateinummer  . . . . . > 1             1-999999, *ONLY, *LAST, *ANY
Jobsystemname  . . . . . > EPOSAS03      Name, *ONLY, *CURRENT, *ANY
Spool-Datei erstellt:
  Erstellungsdatum . . . . . > '23.09.04'    Datum, *ONLY, *LAST
  Erstellungszeit  . . . . . > '15:17:41'    Zeit, *ONLY, *LAST
Drucker  . . . . . PRT01       Name, *SAME, *OUTQ
Druckreihenfolge . . . . . *SAME         *SAME, *NEXT
Formularart  . . . . . *STD        Form.art *SAME, *STD
Kopien . . . . . 1          1-255, *SAME
Druck neu starten  . . . . . *STRPAGE     Zahl, *SAME, *STRPAGE...


                                                                      Ende
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F10=Zusätzl. Parameter
F12=Abbrechen   F13=Verwendung der Anzeige   F24=Weitere Tasten
```

Spoolfiles in der Warteschlange bearbeiten

Solange die Druckausgaben nicht im Status WTR sind, können Sie viele Spoolfile-Attribute verändern. Die Änderungen gelten nur für die aktuelle Verarbeitung der Datei. Wird die Datei derzeit gedruckt, können Sie nur die Parameter COPIES, RESTART und SAVE ändern. Über die Bedeutung der einzelnen Parameter können Sie sich in der Bedienerhilfe informieren.

Wenn Ihre Ausgabewarteschlange keine Spoolfiles zum Drucker schickt, kann es dafür verschiedene Gründe geben:

- Wenn eine Warteschlange angehalten wird (Status HLD), kann die Warteschlange Spoolfiles aufnehmen, jedoch nicht an den Drucker weitergeben. Erst wenn die Ausgabewarteschlange wieder freigegeben (Status RLS) wird, gelangen die Spoolfiles zum Drucker.

- Ist der zugehörige Drucker nicht aktiv (=gestartet), können auch keine Ausgabedateien verarbeitet werden. Sie müssen dann den Drucker mit dem Befehl STRPRTWTR aktivieren und ihn mit dem Befehl ENDWTR beenden.
- Sie können eigene Ausgabewarteschlangen erstellen, aber diese Ausgabewarteschlangen sind nicht mit einem Drucker verbunden. Solche Warteschlangen können als Aufbewahrungsort für wichtige Listen verstanden werden, die nicht auf Papier ausgedruckt werden sollen. Druckausgabewarteschlangen erstellt die iSeries automatisch, wenn Sie einen Drucker installieren.

2.6.4
Seite 4

Verfolgung eines Batch-Jobs

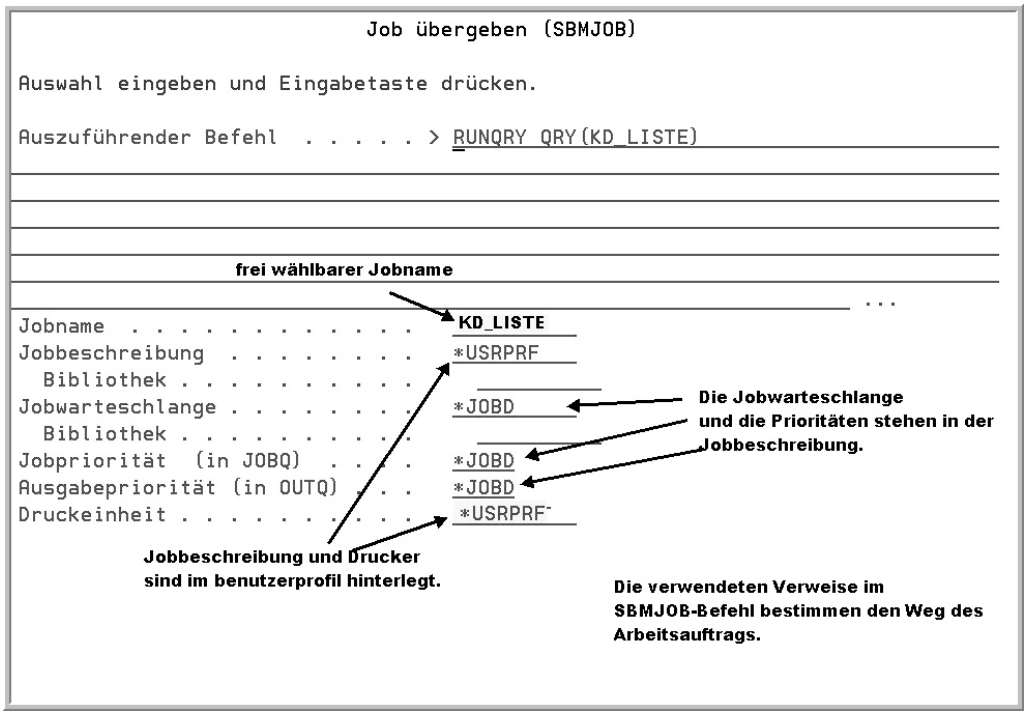
Sie kennen jetzt die Objekte, die an der Verarbeitung eines Jobs beteiligt sind, aber wie findet Ihr Job seinen Weg zu den einzelnen Objekten? Aus diesem Grund verfolgen wir einen Batch-Job auf seinen Weg. Lassen Sie uns vorab nur kurz den Begriff „Batch-Job“ klären. Das System unterscheidet zunächst zwei grundlegende Formen der Verarbeitung: die interaktive Verarbeitung und die Stapelverarbeitung. Synonym werden Begriffe wie Batch-Verarbeitung, Bildschirmverarbeitung, Batch-Job oder Dialogjob verwendet. Ein interaktiver Job ist ein Arbeitsauftrag, der schrittweise verarbeitet wird. Die Aufgaben und die Einzelschritte werden nicht vollständig definiert. Zwischen dem Anwender und dem System findet ein ständiger Dialog statt, um die weiteren Verarbeitungsschritte festzulegen. Das Ergebnis der Verarbeitung wird auf dem Bildschirm angezeigt. Im Gegensatz zum Dialogjob übergibt der Benutzer beim Stapeljob nur die Verarbeitung an das System. Nachdem der Stapeljob gestartet wurde, muss der Benutzer keine weiteren Eingaben machen, damit der Job seine Aufgabe beenden kann.

Grundwerk 10/2004

2.6.5
Seite 2

Den Batch-Job starten

Die Abarbeitung eines Querys ist traditionell ein Stapeljob. Allerdings bestimmt nicht der auszuführende Befehl über die Klassifizierung des Jobs, sondern die Art der Übergabe ist entscheidend. Daher können Sie Query interaktiv starten, wenn Sie den Befehl RUNQRY auf der Befehlszeile verwenden. Oder Sie verwenden den Befehl SBMJOB und tragen im Parameter „Auszuführender Befehl“ den Aufruf des Querys ein, dann läuft Ihr Query im Hintergrund als Batch-Job. Nachdem wir jetzt die einzelnen Objekte kennen, die für die Ausführung unserer Arbeitsaufträge verantwortlich sind, verfolgen wir einen Batch-Job, der ein Query ausführt. Wir starten unseren Job mit dem Befehl SBMJOB.



Jobs überprüfen mit SBMJOB

Jeder Job benötigt für die Ausführung eine Jobbeschreibung. Die Jobbeschreibungen definieren die Jobattribute. Jobbeschreibungen sind iSeries-Objekte der Art *JOBD.

CRTJOBD	Jobbeschreibung erstellen
DLTJOBD	Jobbeschreibung löschen
CHGJOBD	Jobbeschreibung ändern
DSPJOBD	Jobbeschreibung anzeigen
WRKJOBD	Mit Jobbeschreibungen arbeiten

2.6.5.1
Seite 2

Die Jobbeschreibung, die unser Query verwenden soll, steht im Benutzerprofil des Anwenders. Auch der Drucker wird über das Benutzerprofil zugeordnet. Auf der nächsten Seite des SBMJOB-Befehls wird die Ausgabewarteschlange eingetragen, in die der Job ausgegeben wird. Der Parameterwert *DEV bestimmt, dass die dem Drucker zugeordnete Ausgabewarteschlange verwendet werden soll. Auf der dritten Seite Ihres Benutzerprofils finden Sie die relevanten Einträge.

```
                    Benutzerprofil erstellen (CRTUSRPRF)

Auswahl eingeben und Eingabetaste drücken.

Gruppenberechtigung . . . . . *NONE       *NONE, *ALL, *CHANGE, *USE...
Art der Gruppenberechtigung . . *PRIVATE    *PRIVATE, *PGP
Zusätzliche Gruppenprofile . . . *NONE      Name, *NONE
          + für weitere Werte
Berechnungscode . . . . . *BLANK
Dokumentkennwort . . . . . *NONE      Name, *NONE
Nachrichtenwarteschlange . . . . *USRPRF    Name, *USRPRF
  Bibliothek . . . . . .          Name, *LIBL, *CURLIB
Zustellung . . . . . *NOTIFY      *NOTIFY, *BREAK, *HOLD, *DFT
Bewertungscodefilter . . . . . 0          0-99
Druckeinheit . . . . . *WRKSTN    Name, *WRKSTN, *SYSVAL
Ausgabewarteschlange . . . . . *WRKSTN    Name, *WRKSTN, *DEV
  Bibliothek . . . . . .          Name, *LIBL, *CURLIB
Abrufprogramm . . . . . *SYSVAL   Name, *NONE, *SYSVAL, *ASSIST
  Bibliothek . . . . . .          Name, *LIBL, *CURLIB

                                                          Weitere ...
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Parameter für die Erstellung eines Benutzerprofils 3

In diesem Profil wurde kein Drucker hinterlegt, sondern wiederum mit einem Verweis gearbeitet. *SYSVAL bezeichnet den Systemwert QPRTDEV. In dem Systemwert wird der Standardsystemdrucker hinterlegt. In den meisten Fällen handelt es sich hier um den Drucker PRT01. Im Benutzerprofil wird die Standardjobbeschreibung QDFTJOBD verwendet.

```
                          Jobbeschreibung anzeigen
                                                              System:   EPOSAS03
Jobbeschreibung:   QDFTJOBD       Bibliothek:   QGPL

Benutzerprofil . . . . . . . . . . . . . . . . . . :   *RQD
CL-Syntaxprüfung . . . . . . . . . . . . . . . . . :   *NOCHK
In Jobwarteschlange anhalten . . . . . . . . . . . :   *NO
Abbruchbewertung . . . . . . . . . . . . . . . . . :   30
Jobdatum . . . . . . . . . . . . . . . . . . . . . :   *SYSVAL
Jobschalter  . . . . . . . . . . . . . . . . . . . :   00000000
Antwort auf Anfragenachricht . . . . . . . . . . . :   *RQD
Jobpriorität (in Jobwarteschlange) . . . . . . . . :   5
Jobwarteschlange . . . . . . . . . . . . . . . . . :   QBATCH
  Bibliothek . . . . . . . . . . . . . . . . . . . :     QGPL
Ausgabepriorität (in Ausgabewarteschlange) . . . . :   5
Druckereinheit . . . . . . . . . . . . . . . . . . :   *USRPRF
Ausgabewarteschlange . . . . . . . . . . . . . . . :   *USRPRF
  Bibliothek . . . . . . . . . . . . . . . . . . . :

                                                                      Weitere ...
Eingabetaste --> Weiter
```

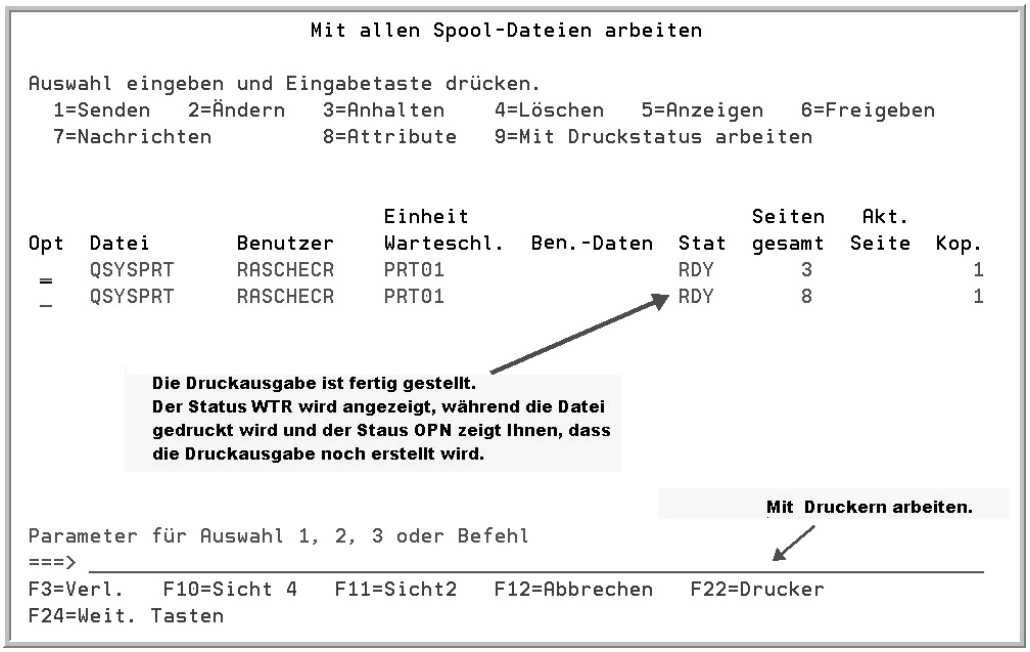
Jobbeschreibung QDFTJOBD

In der Jobbeschreibung QDFTJOBD wird die Jobwarteschlange QBATCH verwendet, d.h.: Alle Jobs, die die Jobbeschreibung QDFTJOBD verwenden, werden zunächst in die Jobwarteschlange QBATCH gestellt. Der Job erhält die Jobpriorität 5. Die Zuordnung des Druckers erfolgt auch hier wieder über das Benutzerprofil. Da im Benutzerprofil auf den Systemwert QPRTDEV verwiesen wird, bestimmt dieser den Drucker. Der Eintrag des Druckers und einer entsprechenden Ausgabewarteschlange bleibt aber unberücksichtigt, da der SBMJOB-Befehl direkt auf das Benutzerprofil verweist. Auf der folgenden Seite der Jobbeschreibung finden Sie die Leitwegdaten. Leitwegdaten werden benötigt, um den Ablauf des Jobs im Arbeitsspeicher zu steuern. Aber auch hier ist zunächst nicht der Eintrag in der Jobbeschreibung wichtig, denn im Befehl SBMJOB werden eigene Leitwegdaten verwendet.

Wenn wir zum Schluss prüfen wollen, ob die Ausgabe in der Ausgabewarteschlange PRT01 eingetroffen ist, verwenden wir am besten den Befehl WRKSPLF. Jetzt finden Sie auch die Druckausgabe Ihres Querys wieder.

Grundwerk 10/2004

2.6.5.1
Seite 4

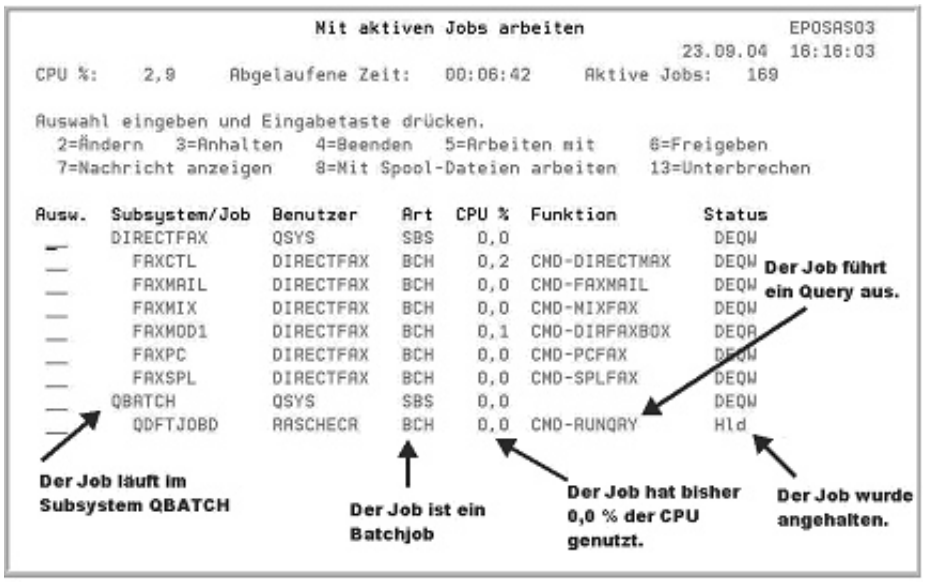


Druckausgabe Ihres Querys

Ihre Liste wurde nicht erstellt? Dann müssen Sie Ihren Job suchen. Aber wie?

Wo ist der Job geblieben?

Zu den Aufgaben des Systemadministrators gehört die Überwachung der System- und Benutzerjobs. Für die Überwachung der Jobs stehen verschiedene Befehle zur Verfügung. Solange Ihr Job aktiv ist, können Sie die Verarbeitung des Jobs mit dem Befehl WRKACTJOB überwachen und manipulieren.



Arbeiten mit dem Befehl WRKACTJOB

Eine weitere nützliche Funktion des Befehls WRKACTJOB ist, dass Sie sehen, wie viel Prozessorleistung die einzelnen Jobs beanspruchen. Setzen Sie den Cursor in die Spalte %CPU und drücken Sie die Taste F16. Alle Jobs sind jetzt aufsteigend sortiert und Sie haben es leichter einen ressourcenintensiven Job, der eventuell Ihr System blockiert, zu identifizieren. Wenn ein Programm oder ein Benutzer Ihr System blockiert, können Sie den Job analysieren (Auswahl 5). Sie können sehen, welches Programm läuft, und können Nachrichten und Dateien prüfen, die von dem Job genutzt werden. Wenn das Programm tatsächlich einen „Stau" verursacht, können Sie das Programm anhalten (Auswahl 3) oder gegebenenfalls beenden (Auswahl 4). Vielleicht ist es aber auch ausreichend, die Ausführungspriorität herabzusetzen (Auswahl 2).

Eine Analyse der Job- und Ausgabewarteschlangen ist mit dieser Funktion jedoch nicht möglich. Wenn Sie die Job- und Ausgabewarteschlangen prüfen wollen, sollten Sie den Befehl WRKUSRJOB einsetzen. Der Befehl WRKUSRJOB ermöglicht eine Selektion nach Benutzern und Jobarten. Außerdem können Sie den Jobstatus als weiteres Selektionskriterium verwenden. Die Ausgabe kann dann angezeigt (*) oder gedruckt (*PRINT) werden.

2.6.6
Seite 2

```
                    Mit Benutzerjobs arbeiten (WRKUSRJOB)

Auswahl eingeben und Eingabetaste drücken.

Jobs für Benutzer anzeigen . . . > RASCHECR     Name, *, *ALL
Status der Benutzerjobs  . . . .   *ALL         *ALL, *ACTIVE, *JOBQ, *OUTQ
Ausgabe  . . . . . . . . . . . .   *            *, *PRINT
Jobart . . . . . . . . . . . . .   *ALL         *ALL, *INTERACT, *BATCH
```

Arbeiten mit dem Befehl WRKUSRJOB

```
                          Mit Benutzerjobs arbeiten                EPOSAS03
                                                                23.09.04  16:18:24
Auswahl eingeben und Eingabetaste drücken.
  2=Ändern   3=Anhalten   4=Beenden   5=Arbeiten mit     6=Freigeben
  7=Nachricht anzeigen    8=Mit Spool-Dateien arbeiten   13=Unterbrechen

Aus-
wahl  Job         Benutzer    Art      -----Status-----  Funktion
 _    LGA15       RASCHECR    INTER    AKTIV             CMD-WRKUSRJOB
 _    QDFTJOBD    RASCHECR    STAPEL   OUTQ
 _    QDFTJOBD    RASCHECR    STAPEL   OUTQ
 _    QDFTJOBD    RASCHECR    STAPEL   OUTQ
 _    QDFTJOBD    RASCHECR    STAPEL   AKTIV    HELD     CMD-RUNQRY

AKTIV = Subsystem
OUTQ = Ausgabewarteschlange
JOBQ = Jobwarteschlangen

ausgeführte Funktion/Programm

Der Job ist angehalten.

                                                                        Ende
Parameter oder Befehl
===> ____________________________________________________________________
F3=Verlassen   F4=Bedienerführung   F5=Aktualisieren      F9=Auffinden
F11=Planungsdaten anzeigen   F12=Abbrechen   F17=Anfang   F24=Weitere Tasten
```

Ausgabe der Daten am Bildschirm

Jobplanung

Jobs werden auf dem System i nach unterschiedlichen Kriterien differenziert. Bekannt sind beispielsweise interaktive Jobs, bei denen der Anwender im Dialog mit dem System steht und über Eingaben bzw. über Befehle Informationen anfordert. Andere Jobs können von einem Anwender für eine Verarbeitung im Hintergrund – ohne einen weiteren Benutzereingriff – gestartet und ausgeführt werden. Diese Formen von Jobs werden als Batchjobs bezeichnet. Mit diesen Jobformen ist gemeint, dass die Jobs durch einen Benutzer gestartet werden können. Bei den zuerst genannten Jobs ist das meistens auch der Fall, bei den zuletzt genannten kann der Job automatisiert und ohne einen Benutzereingriff auf dem System gestartet werden.

Die benutzerlose Ausführung bestimmter Jobs wird mit dem Betriebssystem IBM i geboten.

„Jobplanungseinträge" – so die allgemeine Bezeichnung – erlauben es, eine Vielzahl von Jobs und Aktionen auf dem System automatisiert auszuführen. Ein Beispiel dazu wäre der Tagesabschluss, der auf den meisten Systemen fortwährend ausgeführt werden muss. Meist wird dieser Job am Abend gestartet – d. h., zu einer Zeit, in der kein Administrator mehr im Unternehmen tätig ist. Diese Jobs werden nach Administrationsvorgaben an fest definierten Tagen zu bestimmten Uhrzeiten und mit vorgegebenen Rahmenbedingungen und Einstellungen gestartet. Im realen Normal- und Idealfall enden diese Jobs ohne Fehler.

In diesem Kapitel erfahren Sie mehr über die Möglichkeiten, die sich mit den Jobplanungseinträgen bieten. Neben den Standardoptionen werden Sie auch die zusätzlichen Funktionen des Advanced Job Scheduler kennenlernen.

2.6.7
Seite 2

Advanced Job Scheduler

Automatisierungen sind in der heutigen Wirtschaft unabdingbar – und das natürlich nicht nur im Bereich der IT. Das System IBM i bietet hier Funktionen, mit denen Jobs zeitgesteuert werden können. Mit der Möglichkeit, ein Programm an einem bestimmten Tag zu einer bestimmten Uhrzeit übergeben zu können, lassen sich viele unterschiedliche Anforderungen abdecken.

Die Planungsmöglichkeiten unter dem Betriebssystem IBM i lassen sich in zwei Bereiche aufteilen:

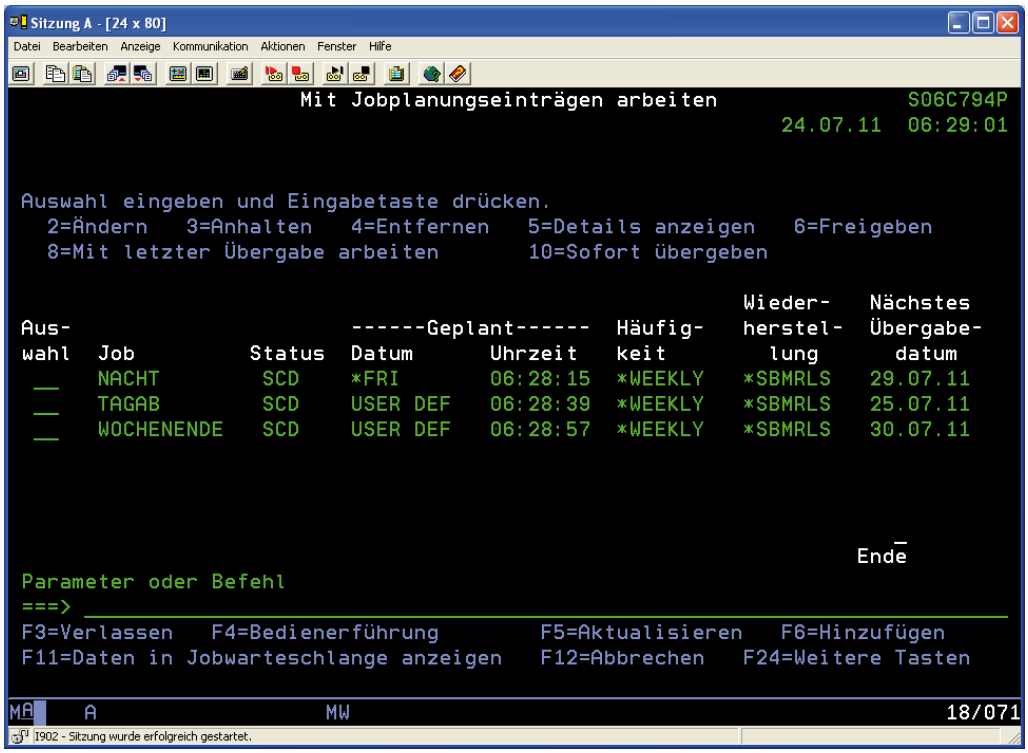
Standard-Jobplanung
- Im Betriebssystem enthalten
- Aufrufbar über den Befehl WRKJOBSCDE

Advanced Job Scheduler
- Zusätzliches (kostenpflichtiges) Lizenzprogramm unter 6.1 5761-JS1
- Zusätzliche Funktionen – wie z. B. Zugriff auf Job-Historie und Überwachung bzw. Einbindung des Jobstatus'.

Dies sind die beiden unter der 5250-Oberfläche verfügbaren Varianten der Jobplanung. Darüber hinaus stehen außerdem mit Management Central und dem System i Navigator grafische Werkzeuge für die Verwaltung der Jobplanung zur Verfügung.

2.6.7.2
Seite 2

Schauen wir uns im Folgenden nun die verschiedenen Möglichkeiten der Jobplanung genauer an:



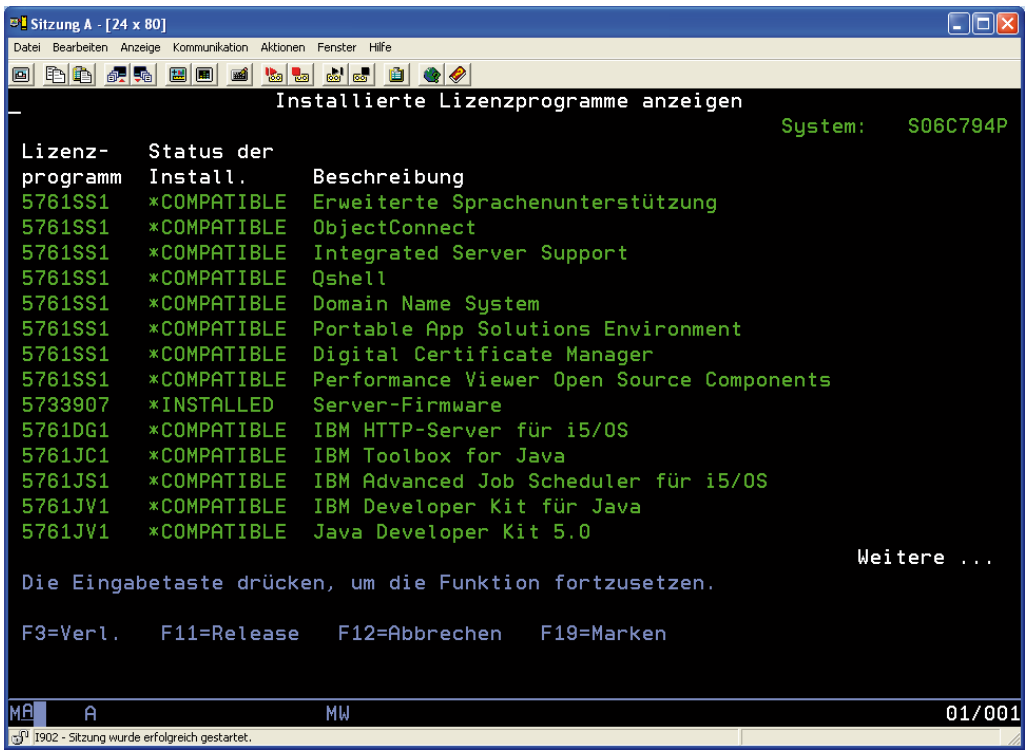
WRKJOBSCDE

Advanced Job Scheduler
Das kostenpflichtige Lizenzprogramm „Advanced Job Scheduler“ (57xx-JS1) verfügt über etliche Funktionen, die bei der Verwaltung der Prozesse unter IBM i durchaus hilfreich sein können:

- Job-Steuerung über System i-Netzwerk
- Verwaltung von AJS-Funktionen über Web-Oberfläche
- Benachrichtigungsfunktionen
  - Normales / Abnormales Job-Ende
  - Verwendung von Nachrichten und Pagern
- Überwachung von Ausgabewarteschlangen (OUTQs)
- Überwachung von Nachrichten (MSGQs)
- Ausführung von bis zu 999 Befehlen innerhalb eines geplanten Jobs

– Verwendung von Kalenderfunktionen.
  Neben der Verwendung von Ferienkalendern können auch individuelle Kalender für den AJS-Einsatz konfiguriert werden.
– Umfassende Steuerungsmöglichkeit eines Jobs
  • Jobsteuerung in Abhängigkeit von anderen Jobs oder Programmaufrufen

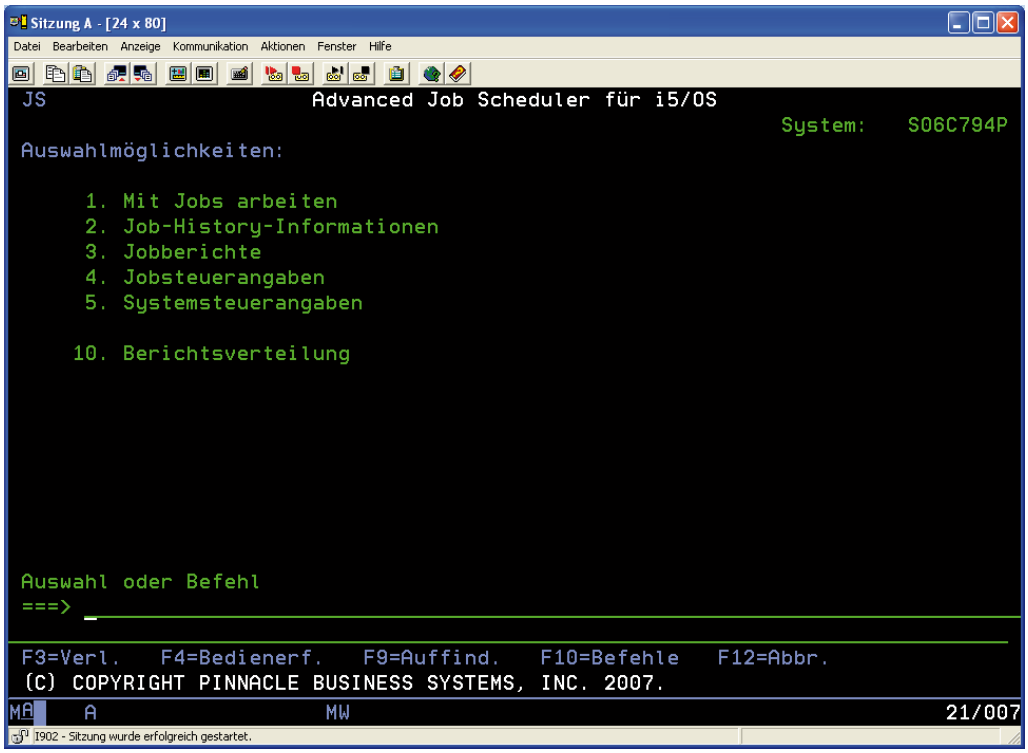
Wie bereits erwähnt, handelt es sich um ein kostenpflichtiges Lizenzprogramm, das auf IBM i installiert sein muss (RSTLICPGM).



LICPGM

2.6.7.2
Seite 4

Man kann recht einfach prüfen, ob der Advanced Job Scheduler (kurz: AJS) installiert worden ist – und zwar durch die Eingabe des Befehls „GO JS“.



GO JS

Die folgende Abbildung zeigt das Menü für den AJS.

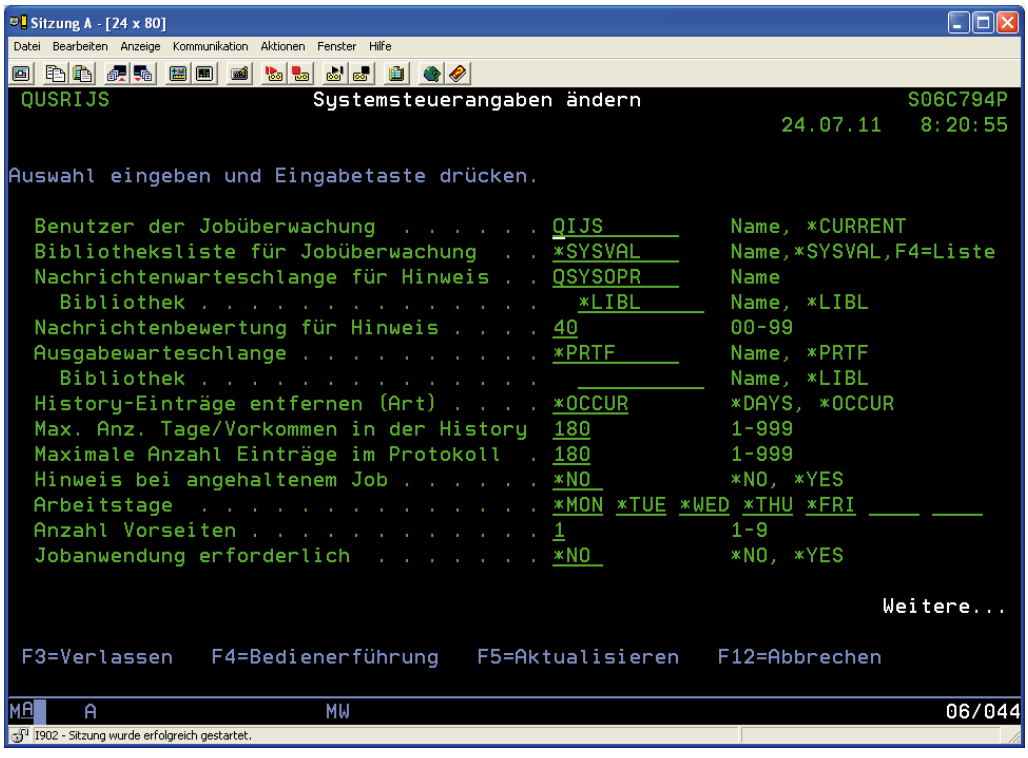
Sollten Sie nach der Eingabe von GO JS nicht in die abgebildete Anzeige gelangen, dann fehlt entweder die notwendige Berechtigung – oder AJS ist nicht installiert.

Grundeinstellungen
Für AJS lassen sich allgemeine Einstellungen im Bereich „Systemsteuerangaben ändern“ hinterlegen. Hier können z. B. die folgenden Angaben definiert werden:

– Benutzerprofil für die Ausführung der AJS-Jobüberwachung
  Standardwert ist das Benutzerprofil QIJS

– Nachrichtenwarteschlange für Hinweise
  Als Standardnachrichtenwarteschlange ist MSGQ QSYSOPR hinterlegt. Diese Einstellung kann bei Bedarf in jede beliebige MSGQ geändert werden.

Über den Angabenbereich „Arbeitstage“ lassen sich die Wochentage als „normale“ Arbeitstage definieren. Hier sind als Standard die Tage „Montag bis Freitag“ hinterlegt.

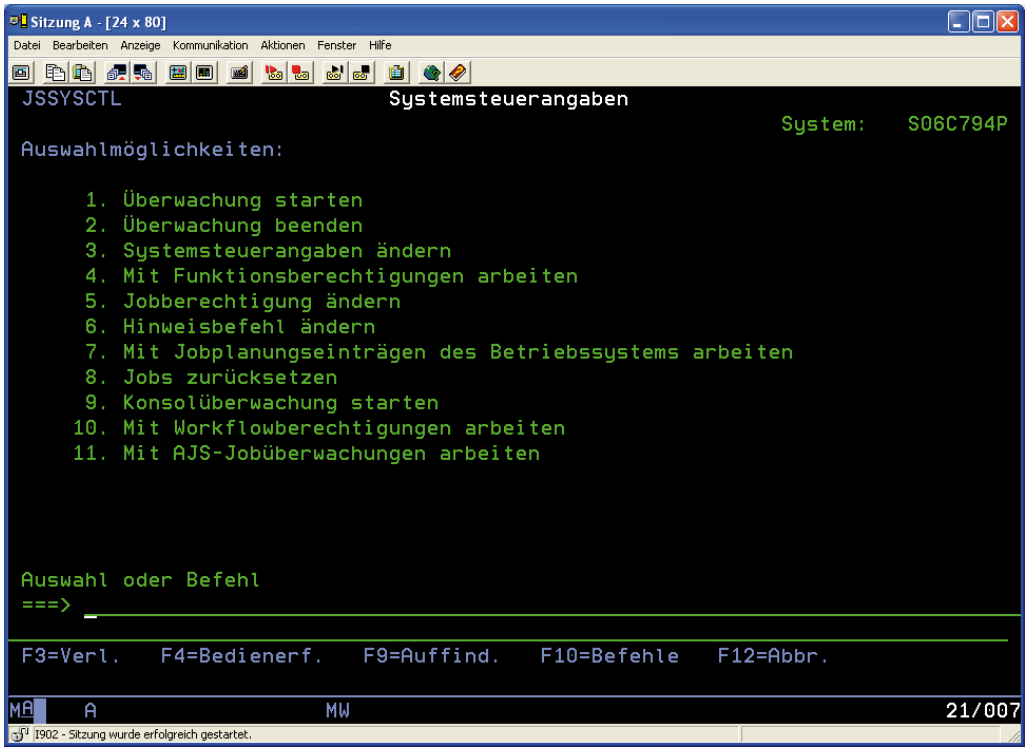


Systemsteuerangaben

2.6.7.2
Seite 6

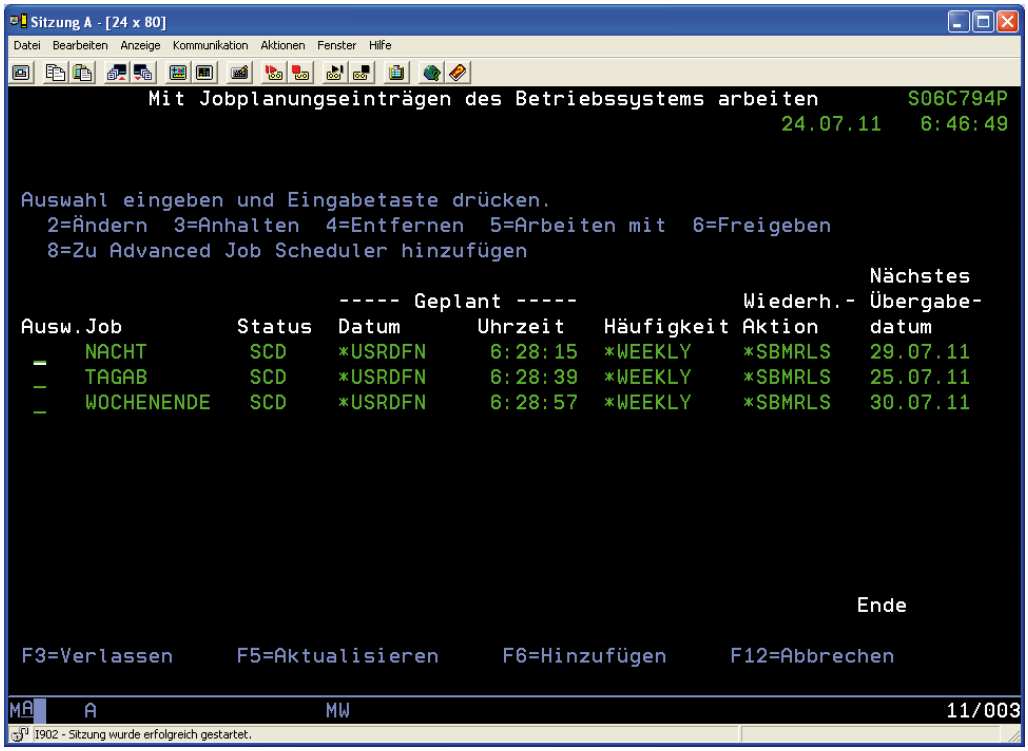
Importieren von bestehenden JOBSCDE-Einträgen

Die meisten System i-Kunden verwenden den im Betriebssystem enthaltenen Job Scheduler – und nicht selten sind in diesem viele verschiedene Jobeinträge enthalten. Sie lassen sich mit Option 7 des Menüs GO JSSYSCTL in den AJS importieren.



GO JSSYSCTL

Im Folgenden werden die wichtigsten Funktionen dieses Menüs gezeigt: Zuerst wird mit dem Menüpunkt 7 begonnen. Da befinden sich die Jobplanungseinträge, die auf Betriebssystemebene mit WRKJOBSCDE angelegt wurden.

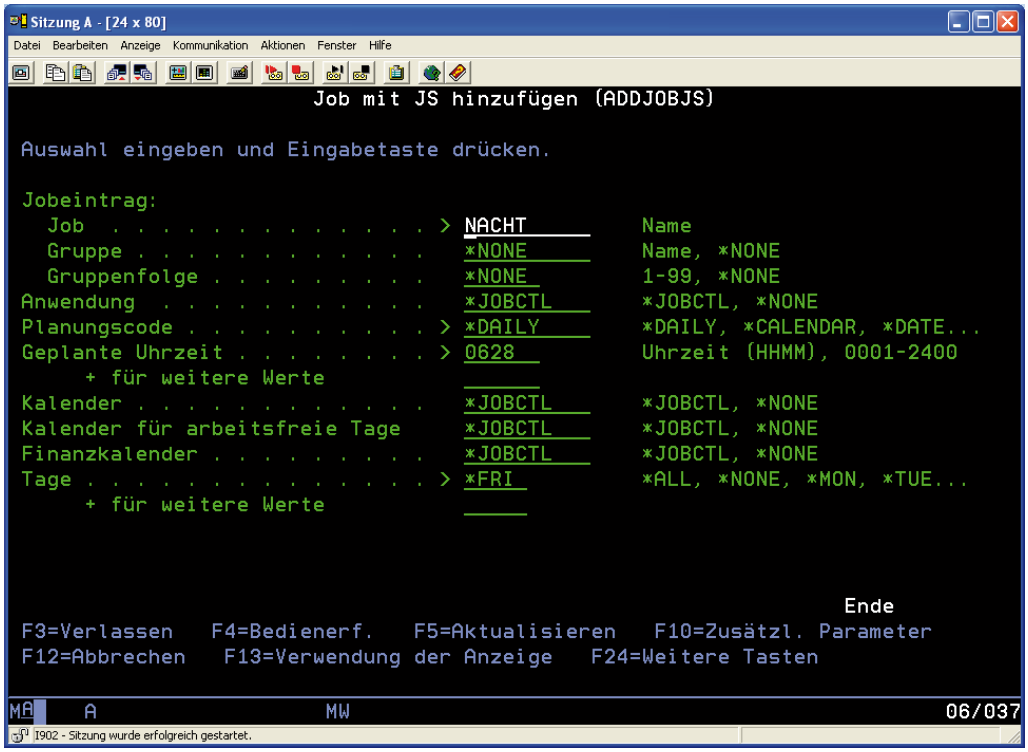


Planungseinträge auf Betriebssystemebene

2.6.7.2
Seite 8

Meist findet sich auf den Systemen eine Fülle von Planungseinträgen. Diese können bei Bedarf in den AJS importiert werden.

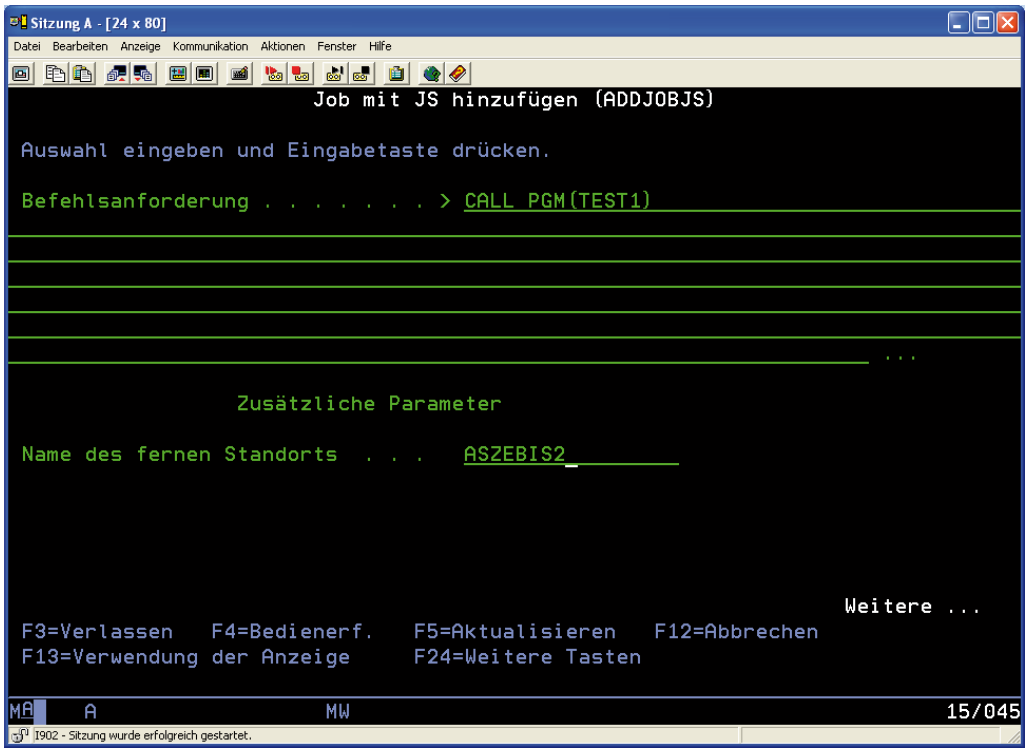
In der Anzeige erkennt man JOBSCDE-Einträge. Die gewünschten Einträge lassen sich mit Option 8 final in den AJS importieren.



ADDJOBJS

Einige der Parameter des ursprünglichen Planungseintrags – wie „Name“, „Ausführungshäufigkeit“, „Auszuführender Befehl“ etc. – werden voreingestellt.

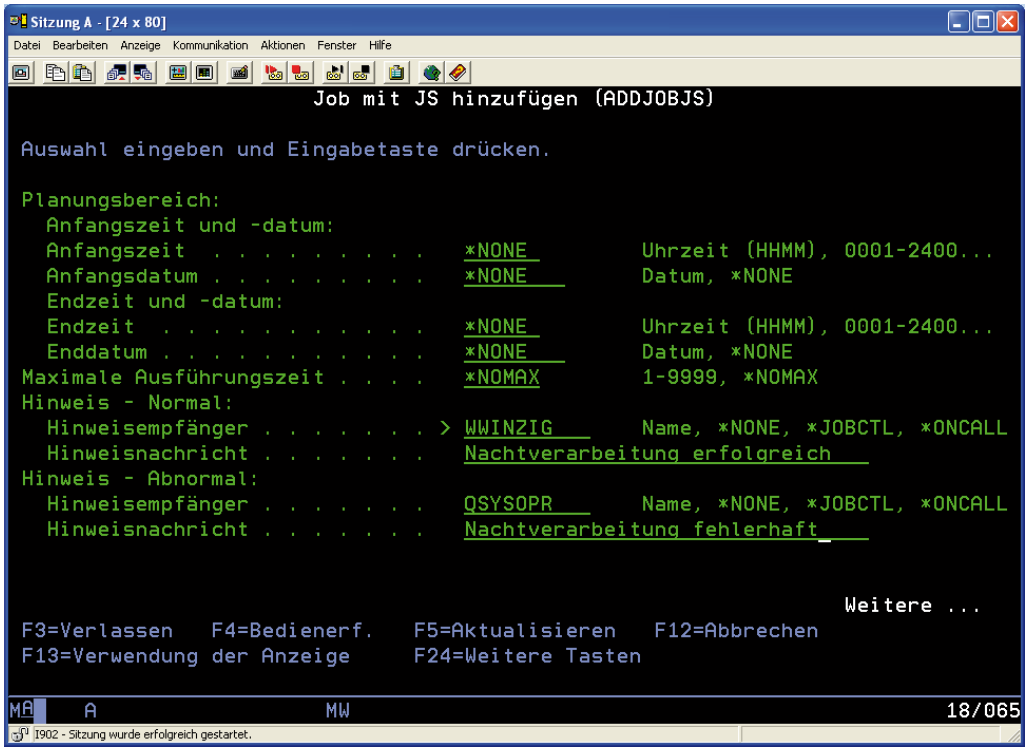
Als Besonderheit kann ein Eintrag des AJS durchaus auch auf einem fernen IBM i-System ausgeführt werden. Wenn das gewünscht wird, dann muss in der Anzeige der zusätzlichen Parameter (F10 Taste) der Name des fernen Standorts angegeben werden. Ein Beispiel dafür finden Sie in der folgenden Abbildung:



Ferner Standort

2.6.7.2
Seite 10

Ein weiterer Vorteil des AJS liegt in der Tatsache begründet, dass geplante Jobs in Bezug auf das Ausführungsergebnis – „erfolgreich“ / „nicht erfolgreich“ – in der Lage sind, Nachrichten zu versenden. Diese Nachrichten können mittels SMTP an vordefinierte Empfänger übertragen werden. Dazu steht der Befehl SNDDSTJS zur Verfügung, den wir uns später noch genauer ansehen werden.



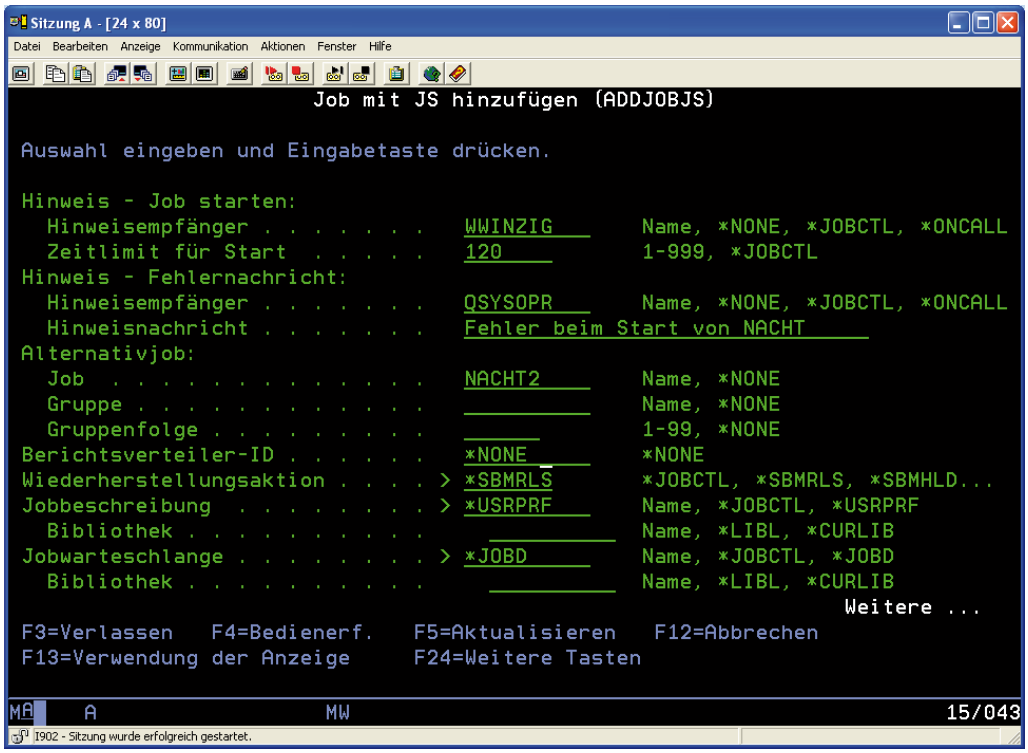
Empfängerinformationen

Um zu verhindern, dass eine zu lange Ausführungszeit des Jobs weitere Verarbeitungen behindert, kann in dieser Anzeige im Bereich „Maximale Ausführungszeit“ eine Zeitangabe in Minuten erfolgen, innerhalb welcher der geplante Job verarbeitet werden muss. Sollte keine Zeitangabe erfolgen, dann wird der Job nach Erreichen der Zeitgrenze beendet.

Anmerkung:

Mit dieser Angabe sollte sorgfältig umgegangen werden, denn unter Umständen kann das abnormale Beenden eines Jobs entsprechende negative Konsequenzen haben!

Auch die Startaktionen eines geplanten Jobs lassen sich mit dem AJS überwachen. Bei Bedarf können auch hier Nachrichten hinzugestellt werden. Sollte ein Job nicht gestartet werden können, dann kann ein Alternativjob angegeben werden.



Startinformationen zu ADDJOBJS

Bei einem solchen Import ist zu beachten, dass die neuen, in AJS importierten Jobs per Standard dazu angehalten sind, den Ausführungsstatus für neu importierte Einträge anzupassen und sie für eine automatische Ausführung zu aktivieren.

Probleme beim Importieren von klassischen Job Scheduler-Einträgen können in den folgenden Fällen auftreten:

– Der Eintrag beinhaltet ein zu übergehendes Datum.

– Der Eintrag beinhaltet als Benutzerprofil ein Systemprofil.

In beiden Fällen muss der Eintrag manuell angepasst werden, damit er erfolgreich in den AJS übertragen werden kann.

Für zu übergehende Datumsangaben kann wahlweise der Sonderwert *NONE angegeben werden. Dann allerdings ist diese „Übergehungsfunktion“ ausgeschaltet. Oder man verwendet einen Ferienkalender, in dem das Datum entsprechend markiert worden ist.

Beim Import eines Eintrags mit Systemprofil muss ein Nicht-Systembenutzer angegeben werden. Dieser sollte allerdings auch über die ausreichenden Berechtigungen für die Ausführung des zu planenden Jobs verfügen.

Mit der Installation des Advanced Job Schedulers werden zwei neue Bibliotheken auf dem System abgelegt:

– QIJS
  Produkt Bibliothek

– QUSRIJS
  Benutzer Bibliothek AJS

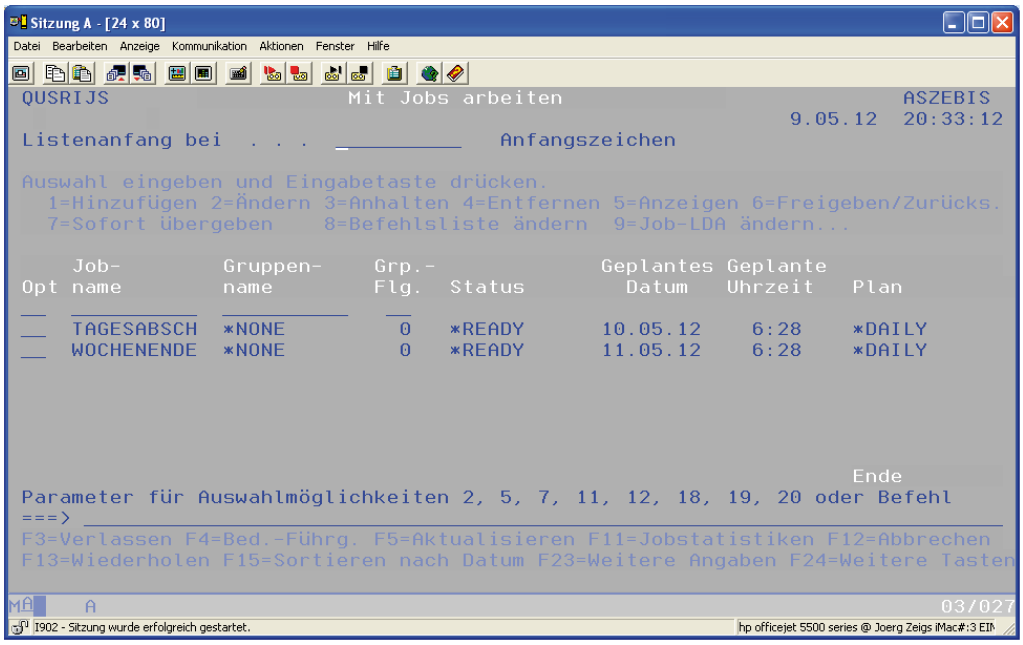
Anmerkung:

In einigen Fällen kann es vorkommen, dass sich DTAQ-Objekte des AJS nicht sichern lassen. Entsprechende Fehlermeldungen können z. B. bei einer Gesamtsicherung ausgegeben werden. Es ist kein Problem, wenn es sich bei den nicht gesicherten Objekten nur um DTAQ-Objekte des AJS handelt, denn die werden – sollten sie nicht schon vorhanden sein – automatisch neu erstellt.

Für AJS bietet die IBM eigene Gruppen-PTFs an, die in jedem Fall installiert werden müssen, um eine korrekte Verwendung des AJS sicherzustellen.

Die Einstellungen des AJS werden in der Bibliothek QUSRIJS gespeichert. Die kann beispielsweise auch dazu genutzt werden, um die AJS-Einstellungen von einem System auf ein anderes zu übertragen.

Ein wichtiger Befehl im Zusammenhang mit der Verwaltung der Advanced Job-Planungseinträge ist – neben dem Befehl ADDJOBJS – der Befehl WRKJOBJS. Er listet alle für AJS definierten Planungsjobs auf und erlaubt ihnen – je nach Berechtigung – sich zu verwalten.



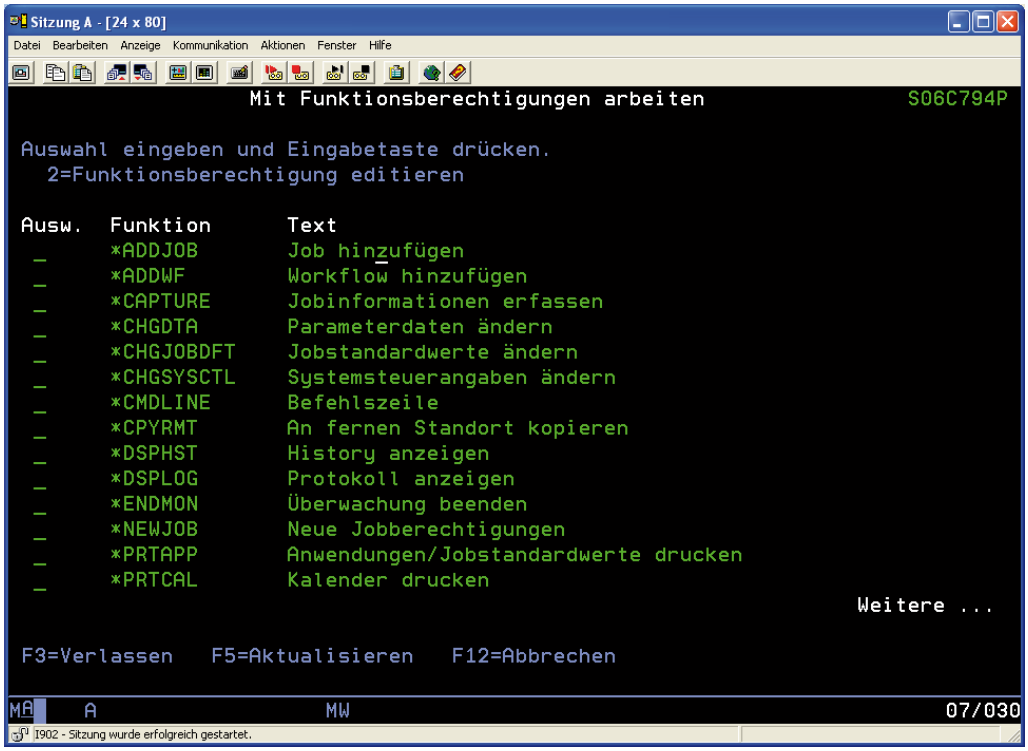
WRKJOBJS

Wir werden uns dazu später noch eingehender mit diesem Befehl und den damit verbundenen Möglichkeiten beschäftigen.

2.6.7.2
Seite 14

Funktionsberechtigungen

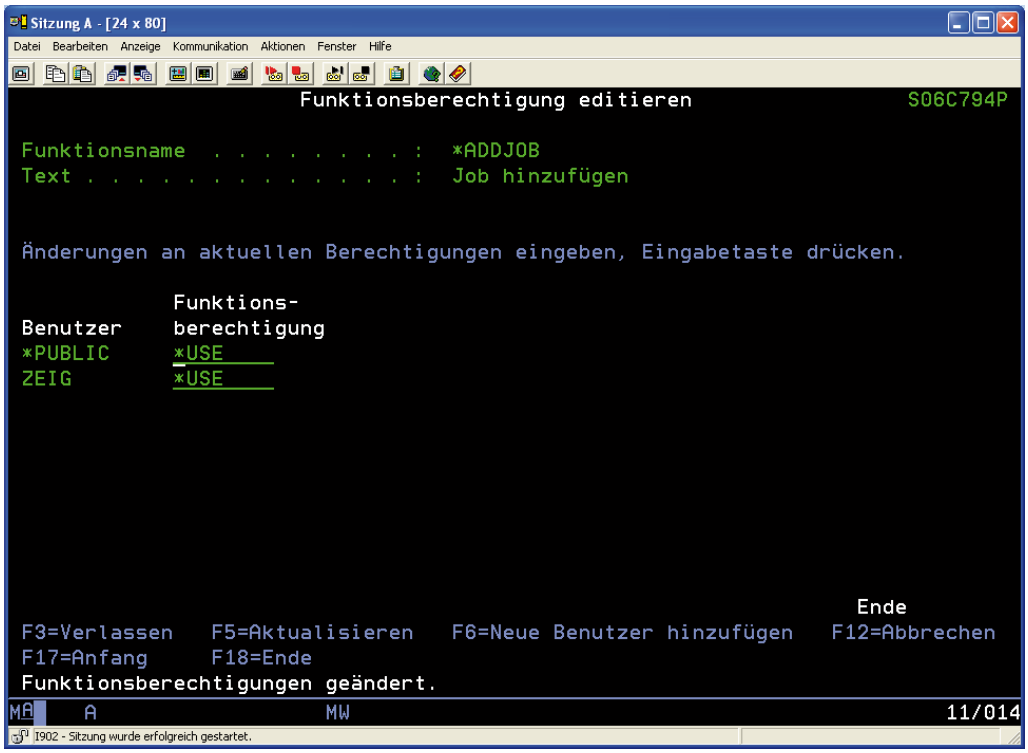
Die Funktionsberechtigungen steuern den Zugriff und die Ausführungsrechte für unterschiedliche Bereiche.



Funktionsberechtigungen

Mit der Auswahl „2“ lassen sich die Funktionsberechtigungen verwalten. Für diese Verwaltung sind ausreichende Berechtigungen erforderlich. Achten Sie deshalb darauf, dass *ALLOBJ und *SECADM als Berechtigungen zur Verfügung stehen.

Die Verwaltung der Funktionsberechtigungen ist an die Berechtigungsverwaltung des Betriebssystems angelehnt – wie es die folgende Abbildung zeigt:



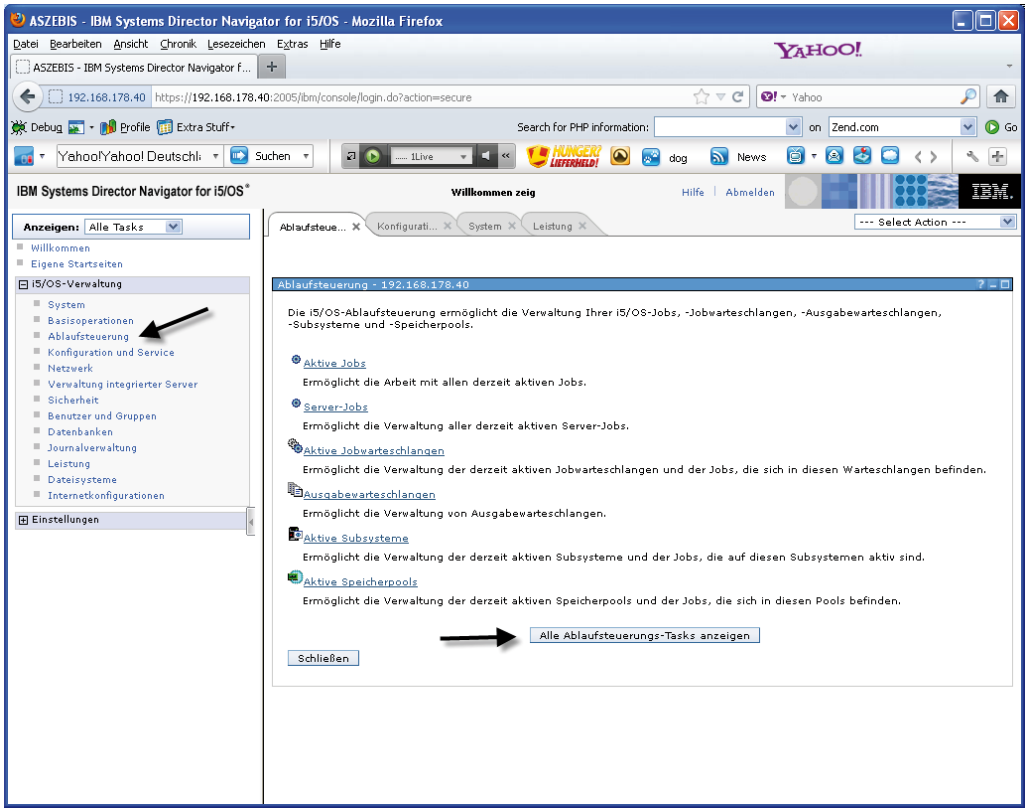
Funktionsberechtigung editieren

Web-Verwaltung

Seit Version 6.1 des IBM i sind wesentliche Erweiterungen in der Web-Version des AJS-Verwaltungsbereichs vorgenommen worden. Die Voraussetzungen für den Web-Verwaltungsbereich des AJS sind in der Basisinstallation jedes Systems IBM i ab Version 6.1 enthalten. Um den Verwaltungsbereich aufzurufen, geben Sie wahlweise die Web-Admin URL http://ServerName:2001 oder den direkten Link für den Verwaltungsbereich des AJS an: http://ServerName:2001/webnav/WnServlet?task=schjob. In beiden Fällen muss der Platzhalter „ServerName“ durch den Hostnamen des Systems IBM i ersetzt werden.

Auch wenn die Grundeinstellungen des AJS noch ausschließlich über 5250-Befehle zur Verfügung stehen, bietet die Web-Verwaltung insbesondere für die alltäglichen Arbeiten eine gute Alternative zu den 5250-Befehlen.

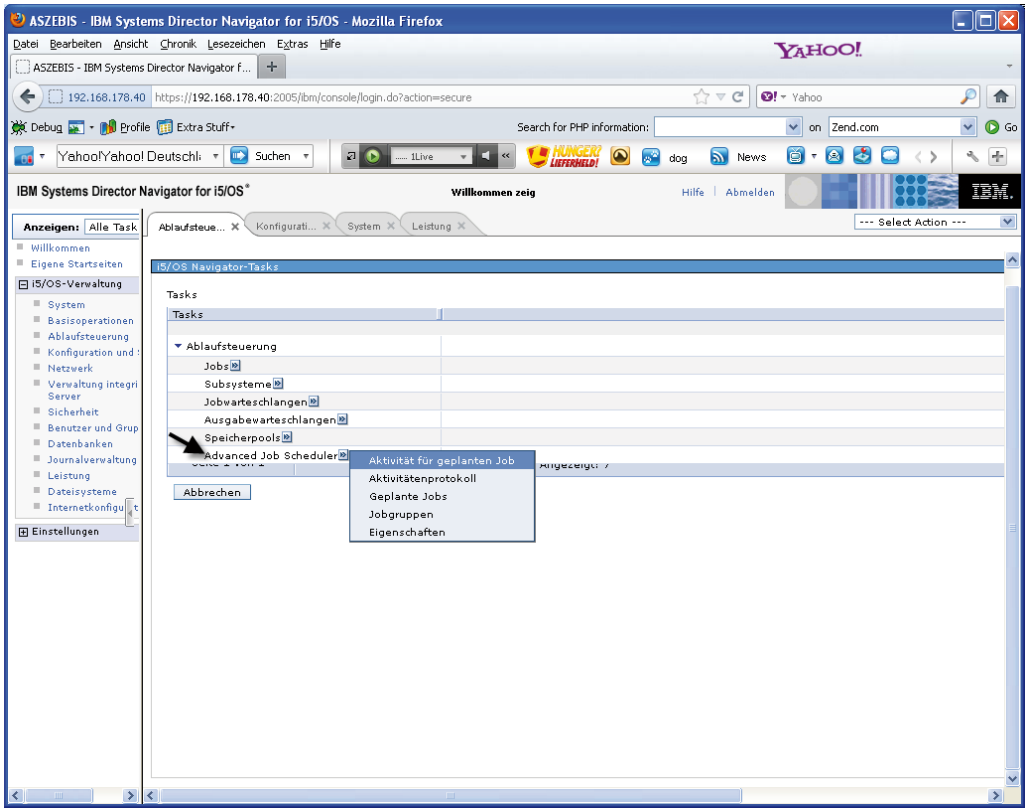
Der IBM Systems Director for IBM i bietet die Grundlage für eine Vielzahl von administrativen Tätigkeiten. So auch Funktionen, die mit dem Advanced Job Scheduler zum Einsatz kommen. Diese Funktionen sind im Bereich „Ablaufsteuerung“ zu finden, der auch die Funktionen für das Arbeiten mit Jobs (WRKACTJOB) sowie weitere Steuertasks enthält.



Ablaufsteuerung im System Director

Für den Einstieg in den Advanced Job Scheduler im linken Navigationsbereich wird zunächst der Eintrag „Ablaufsteuerung“ gewählt. Anschließend sollte im Hauptanzeigenbereich die Schaltfläche „Alle Ablaufsteuerungstasks anzeigen“ selektiert werden.

Damit öffnet sich eine weitere Anzeige im Browser, in der verschiedene Tasks zur Auswahl angeboten werden. Unter anderem befindet sich in der Auflistung auch der Eintrag „Advanced Job Scheduler“. Auf diesen Eintrag ist zu klicken, um die Anzeige zu erweitern. Erzeugt wird dadurch eine Darstellung, wie sie exemplarisch in der folgenden Abbildung zu sehen ist:

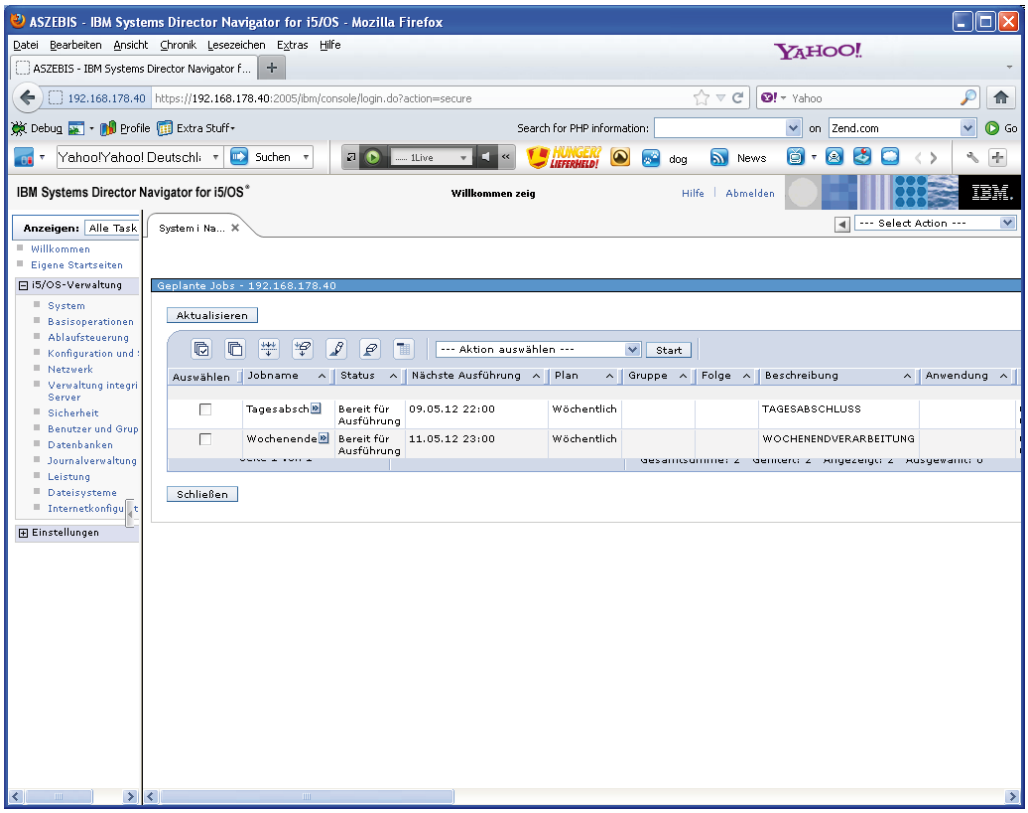


Advanced Job Scheduler

Die verschiedenen Funktions- und Konfigurationsbereiche des Advanced Job Schedulers sind hier in einer tabellarischen Form aufgelistet. Um z. B. mit den Jobplanungseinträgen zu arbeiten, wird auf den Eintrag „Geplante Jobs“ geklickt.

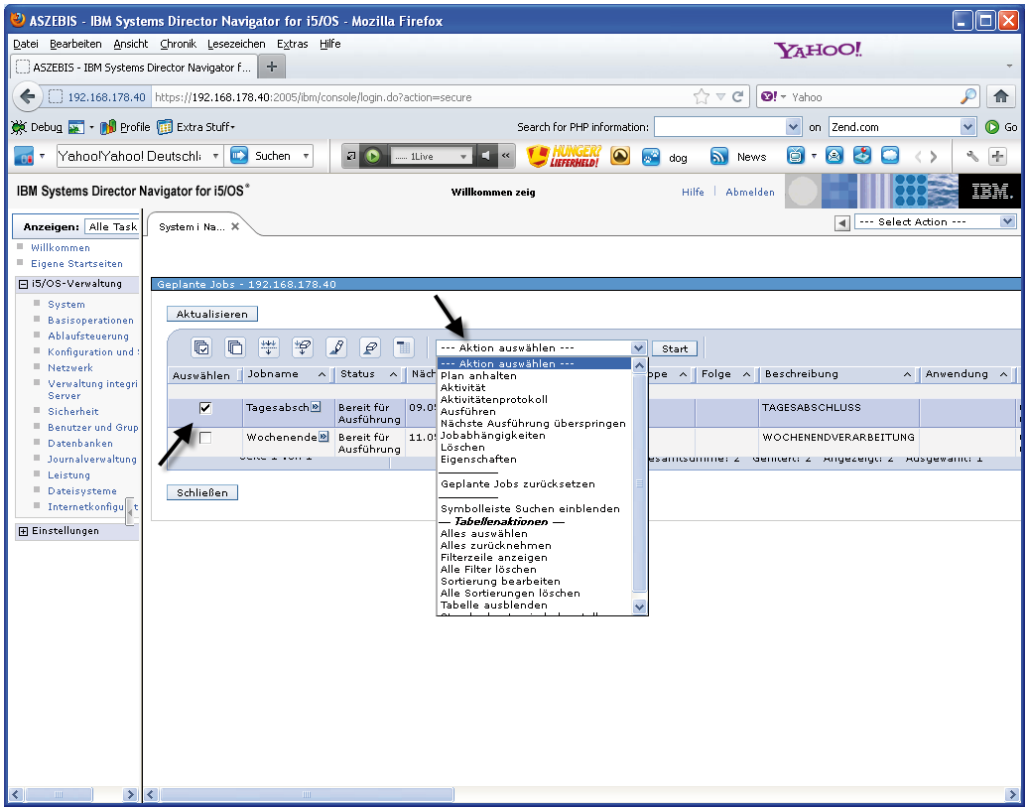
2.6.7.2
Seite 18

Damit gelangt man in eine Übersichtsanzeige der Jobplanungseinträge des AJS.



Planungsjobs anzeigen

Die Details eines Jobplanungseintrags können mit einem Klick auf den gewünschten Eintrag eingesehen und auch verwaltet werden. Allerdings muss vor der Verwaltung noch entschieden werden, was mit dem selektierten Eintrag gemacht werden soll. Mit einem Klick auf die Schaltfläche „Aktion auswählen“ öffnet sich eine Auswahlliste, wie sie in der folgenden Abbildung zu sehen ist:

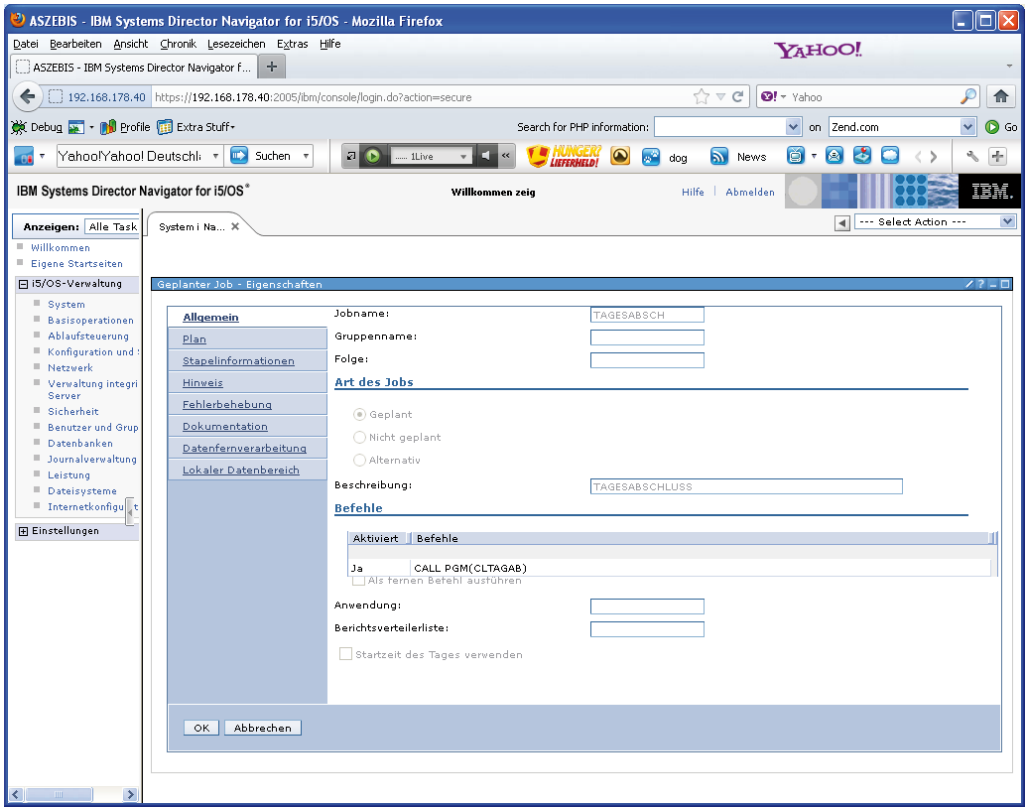


Planungsjob verwalten

Hier kann man beispielsweise den betreffenden Eintrag anhalten, ihn löschen, Konfigurationseinstellungen ändern oder auch Jobabhängigkeiten definieren. Letztere sind eines der wesentlichen Herausstellungsmerkmale für den Einsatz des AJS.

2.6.7.2
Seite 20

In der folgenden Abbildung kann man exemplarisch die Anzeige für die Verwaltung des Jobplanungseintrags sehen, und zwar nach Auswahl der Option „Eigenschaften“. Dahin gelangt man nach Selektion der gewünschten Aktion und dem Bestätigen der Auswahl durch einen Klick auf die Schaltfläche „Start“, die sich direkt neben dem Auswahlfenster befindet.



Eigenschaften des AJS

Eine neue Arbeitsumgebung erstellen

Wir haben in den vorangegangenen Kapiteln viel über den Weg eines Jobs erfahren und gelernt, wie wir den Verlauf eines Jobs verfolgen können. Jetzt nutzen wir dieses Wissen und erstellen uns eine eigene Arbeitsumgebung, ohne auf bestehende Objekte des Systems iSeries zurückzugreifen. Eigene Subsysteme können für die Ausführung bestimmter Anwendungen erforderlich sein. Externe Faxanwendungen laufen z.B. häufig in eigens dafür definierten Subsystemen ab. Ich werde ein neues Subsystem aufbauen, um Querys auszuführen. Das hat einige Vorteile: meine anderen Batch-Jobs werden nicht mehr durch Queryabfragen im Subsystem Qbatch blockiert. Ich kann die Priorität der Abfragen senken und habe immer einen Überblick darüber, wer welche Abfragen zurzeit ausführt. Damit meine Objekte nicht zu sehr mit den Systemobjekten vermengt werden, habe ich mir zunächst eine neue Bibliothek erstellt. Diese Bibliothek ist als Current-Library (=aktuelle Bibliothek) hinterlegt, so dass ich in den meisten CRT-Befehlen die Default-Einstellung *CURLIB stehen lassen kann.

4. Ergänzung 09/2005

2.7
Seite 2

Jobwarteschlange erstellen

Als Erstes benötigen wir eine Jobwarteschlange. Unsere Jobwarteschlange werde ich STDQRY nennen. Die IBM nutzt für die verschiedenen Objekte einer Arbeitsumgebung stets identische Namen. In meinen Schulungen habe ich allerdings bemerkt, dass es anfangs sinnvoll sein kann, die Objekte unterschiedlich zu benennen, um die einzelnen Objekttypen klar voneinander zu unterscheiden. Ich überlasse es Ihnen, eine geeignete Namensgebung zu finden.

```
                    Jobwarteschlange erstellen (CRTJOBQ)

Auswahl eingeben und Eingabetaste drücken.

Jobwarteschlange . . . . . JOBQ          stdqry
  Bibliothek . . . . . .                   *CURLIB
Text 'Beschreibung'  . . . . . TEXT      Jobwarteschlange für RunQry
_____

                Zusätzliche Parameter

Vom Bediener gesteuert . . . . . OPRCTL      *YES
Berechtigung prüfen  . . . . . AUTCHK        *OWNER
Berechtigung . . . . . AUT           *USE

                                                                  Ende
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Jobwarteschlange erstellen (CRTJOBQ)

Der Parameter „Jobwarteschlange“ bestimmt den Namen des Objektes. Der beschreibende Text ist nicht zwingend erforderlich, aber er ist einfach hilfreich, wenn man sich nach einiger Zeit die Frage stellt, „Wofür brauche ich dieses Objekt?“.

Der Parameter „Vom Bediener gesteuert (OPRCTL)“ bestimmt, dass ein Benutzer, der über die Sonderberechtigung JOBCTL verfügt, die Jobwarteschlange steuern darf. In unserem Beispiel darf jeder der über diese Berechtigung verfügt die Jobwarteschlange anhalten, freigeben etc. Die Öffentlichkeit, d.h. die Gruppe der *PUBLICs darf die Jobwarteschlangen nur nutzen. Dies wird durch den Parameter Berechtigung (AUT) festgelegt. Nach der Freigabe des Befehls erhalten wir ein Objekt der Art *JOBQ mit dem Namen „STDQRY“. Die Jobwarteschlange allein ist nicht sehr nützlich. Wir könnten sie natürlich mit einem bestehenden Subsystem verbinden oder einfach nur als Sammelort für Jobs verwenden, die dann in der Nacht abgearbeitet werden, aber unser Ziel ist ja eine völlig eigenständige Arbeitsumgebung. Also brauchen wir jetzt ein Subsystem.

2.7.1 Seite 2

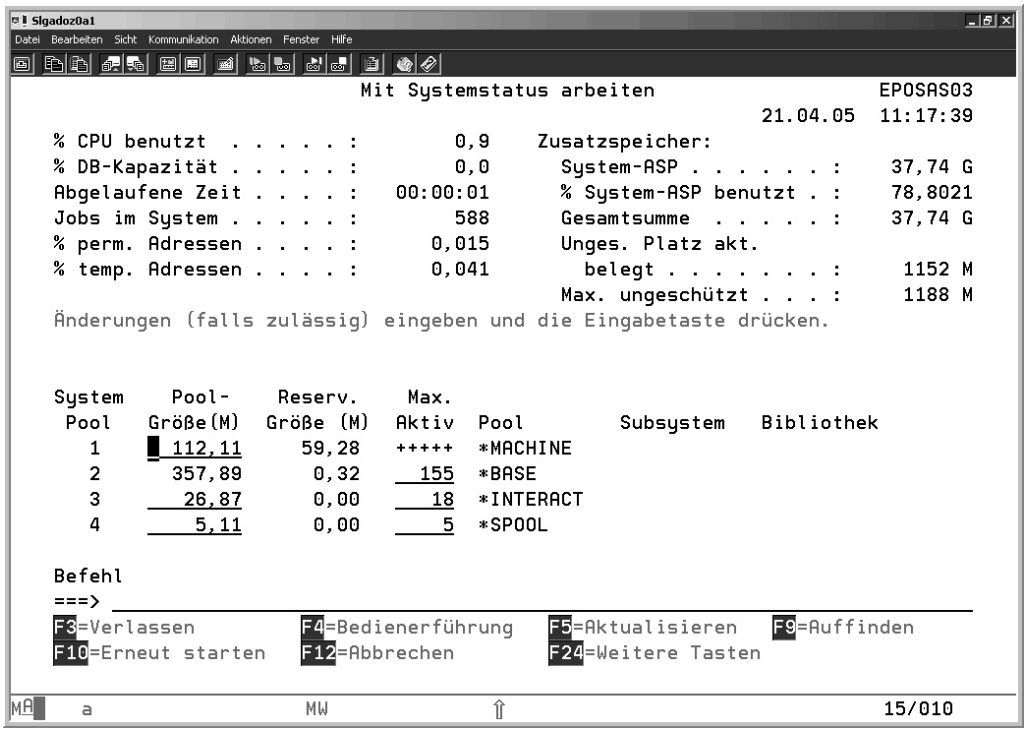
Subsystemdefinition

4. Ergänzung 09/2005

2.7.2
Seite 2

Pooldefinitionen

Wir wissen bereits, dass der Arbeitsspeicher in mehrere Pools unterteilt ist. Innerhalb der Pools laufen die verschiedenen Subsysteme. Die Zuordnung der Subsysteme auf die Pools erfolgt, wenn wir die Subsystembeschreibung erstellen. Jetzt wird es etwas kompliziert. Das Subsystem muss den Speicherpools zugeordnet werden. Für die zu erstellende Subsystembeschreibung können maximal 10 Speicherpooldefinitionen angegeben werden. Doch obwohl in jeder Subsystembeschreibung 10 Definitionen angegeben werden können, ist durch das Betriebssystem festgelegt, dass nur max. 64 Speicherpools einschließlich Base- und Maschinenpool gleichzeitig aktiv sein können. Wird ein Subsystem gestartet, bei dem kein Speicherpool zugeordnet werden kann, ohne die maximal zulässige Anzahl von 64 Pools für das System zu überschreiten, werden bis zu 16 Pools zugeordnet, die überzähligen Pools jedoch nicht. Jobs, die dann den nicht gestarteten Pools zugeordnet werden, laufen im Basepool. Bevor Sie also eine Subsystembeschreibung erstellen ist es erforderlich, dass Sie sich einen Überblick über Ihr System verschaffen. Wie viele Subsysteme gibt es? Wie viele Pools sind den Subsystemen zugeordnet? Handelt es sich hierbei um private Pools oder gemeinsame Pools? Diese Informationen können Sie den jeweiligen Subsystembeschreibungen entnehmen. Sie können sich diese Informationen aber auch mit dem Befehl WRKSYSSTS (work with System Status) beschaffen.

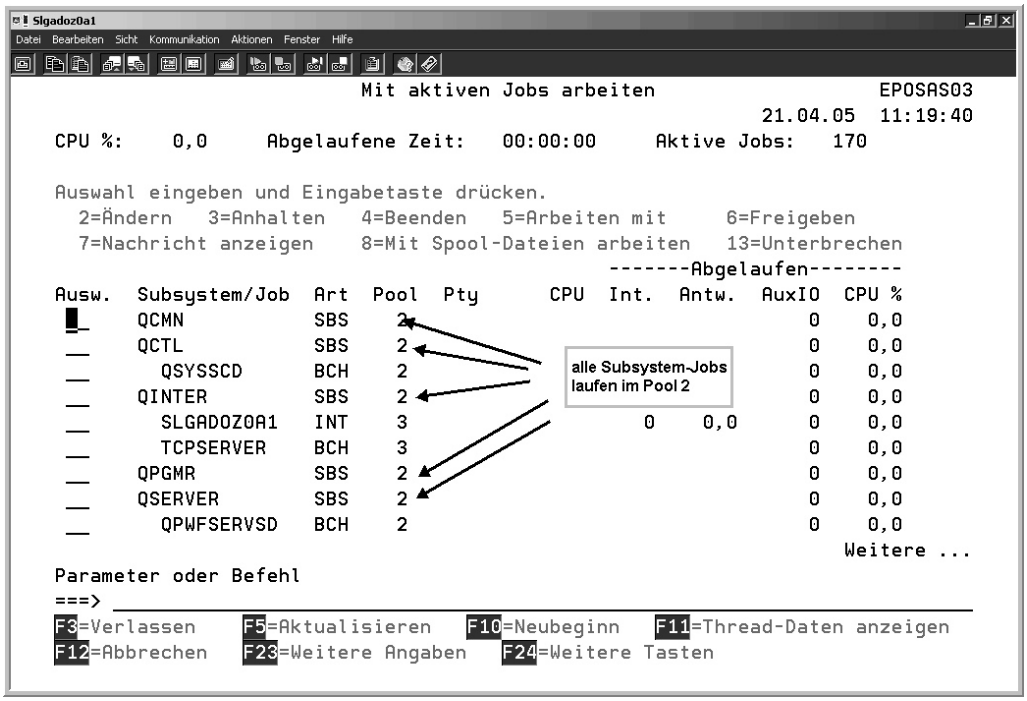


Der Befehl WRKSYSSTS

Sie sehen, dass auf der Maschine EPOSAS03 nur die 4 Standard-Pools definiert sind. Wir werden also kein Problem mit neuen Pooldefinitionen bekom-

2.7.2.1
Seite 2

men. Dem Befehl WRKSYSSTS können Sie darüber hinaus auch die momentan zugeordneten Poolgrößen entnehmen. Wir wissen allerdings nicht, welche Subsysteme den jeweiligen Pools zugeordnet sind. Wenn Sie diese Informationen benötigen, können Sie den Befehl WRKACTJOB (work with aktiv Jobs) benutzen.



WRKACTJOB Funktionstaste F11

Fällt Ihnen etwas auf? Jedes Subsystem läuft zunächst im Pool 2. Warum das? Das Subsystem selbst stellt auch einen Job dar, nämlich einen Job der Art SBS und da alle gleichartigen Jobs in einem Pool gruppiert werden sollten, laufen die SBS-Jobs immer im Pool 2, dem Basepool. Wir wissen also jetzt, dass unser Subsystem auch im Pool 2 laufen sollte. Zusätzlich möchte ich einen neuen privaten Pool definieren, in dem dann später meine Query-Abfragen ausgeführt werden sollen.

```
                Subsystembeschreibg. erstellen (CRTSBSD)

Auswahl eingeben und Eingabetaste drücken.

Subsystembeschreibung  . . . . . > STDQRY      Name
  Bibliothek . . . . . .     *CURLIB   Name, *CURLIB
Speicherpools:                  _
  Pool-ID  . . . . . > 1        1-10
  Speichergröße  . . . . . > *BASE      Zahl, *BASE, *NOSTG...
  Auslastungsgrad  . . . . . >          Zahl
                                _
  Pool-ID  . . . . . > 2        1-10
  Speichergröße  . . . . . > 2048       Zahl, *BASE, *NOSTG...
  Auslastungsgrad  . . . . . > 3        Zahl
    + für weitere Werte         _
Maximale Anzahl Jobs . . . . . > 1      0-1000, *NOMAX
Text 'Beschreibung'  . . . . . > 'Query Subsystem '
_____

                                                                  Ende
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F10=Zusätzl. Parameter
F12=Abbrechen   F13=Verwendung der Anzeige   F24=Weitere Tasten
```

Ein Subsystem beschreiben (CRTSBSD)

Mit dem Befehl CRTSBSD (create Subsystem description) erstelle ich die Subsystembeschreibung. Ich entscheide mich für die Namenskonvention der IBM und nenne mein Subsystem auch „STDQRY“. Der Parameter „Pool-ID“ ist eine fortlaufende Nummer, die die maximal 10 Speicherpooldefinitionen nummeriert. Sie sehen, dass ich für die Definition meines Subsystems zwei verschiedene Pools zuordne. Zuerst (Pool-ID 1) wähle ich als Speichergröße den Basepool. Das ist der Basissystempool, in dem auch das Subsystem QBATCH läuft und natürlich alle SBS-Jobs. Als Zweites (Pool-ID 2) wähle ich einen privaten Pool, indem ich die Anfangsgröße des Pools in Kilobyte bestimme. Außerdem muss ich bei einem privaten Pool noch bekannt geben, wie viele Threads in dem Pool gleichzeitig aktiv werden dürfen. Der private Pool wird nur mein Subsystem unterstützen. Wenn Sie im System nicht die Default-Einstellung verändert haben, wird die iSeries die Poolgrößen später dynamisch verwalten und je nach Anforderung selbstständig verkleinern oder vergrößern.

Der Parameter „Speicherpools“ besteht somit aus drei Elementen:

2.7.2.1
Seite 4

Die Pool-ID

Es können bis zu 10 Wertegruppen angegeben werden, um bis zu 10 Speicherpools für das Subsystem zu definieren.

Die Speichergröße

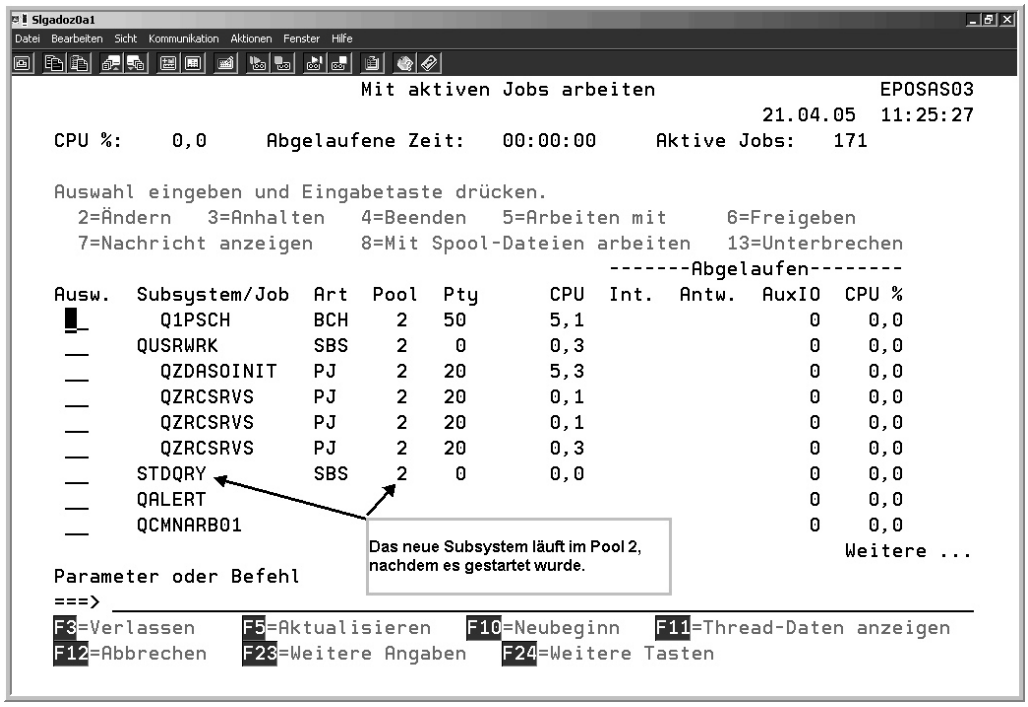
*BASE	Es wird der Basissystempool definiert, auf den andere Subsysteme gemeinsamen Zugriff haben. Mindestgröße und Auslastungsgrad des Basispools werden in den Systemwerten QBASPOOL und QBASACTLVL angegeben. Das Subsystem QBATCH ist dem Basis- pool zugeordnet.
*INTERACT	Der angegebene Pool ist so definiert, dass er als gemeinsamer Pool für interaktive Arbeiten dient. Die Größe und der Auslastungsgrad des gemeinsamen Pools werden mit dem Befehl CHGSHRPOOL (gemeinsamen Speicher-Pool ändern) angegeben. Das Subsystem QINTER ist diesem Pool zugeordnet.
*SPOOL	Der angegebene Pool ist so definiert, dass er als gemeinsamer Pool für Druckaufgaben dient. Die Größe und der Auslastungsgrad des gemeinsamen Pools werden mit dem Befehl CHGSHRPOOL verändert. Das Subsystem QSPOOL ist dem Pool zugeordnet.
*SHRPOOLxx	Der Sonderwert *SHRPOOLxx definiert einen neuen allgemeinen Pool. Es ist möglich insgesamt sechzig allgemeine Pools, die durch die Sonderwerte *SHRPOOL1 bis *SHRPOOL60 identifiziert werden, zu definieren. Die Größe und der Auslastungsgrad eines gemeinsamen Pools werden mit dem Befehl CHGSHRPOOL angegeben.
Ganze Zahl	Durch die Angabe der Speichergröße des Speicherpools, wird ein privater Pool definiert. Die Größenangabe erfolgt in Kilobyte, wobei der Pool mindestens eine Anfangsgröße von 256 Kb benötigt.

Der Auslastungsgrad

Das dritte Element, der Auslastungsgrad, bestimmt die maximale Anzahl Threads, die gleichzeitig im Pool ablaufen können.

Zum Schluss bestimme ich über den Parameter „Maximale Anzahl Jobs“, wie viele Jobs gleichzeitig in dem Subsystem aktiv sein dürfen. Da ich vorhabe, meine Querys in dem Subsystem laufen zu lassen, wähle ich den Wert 1, denn Querys bauen häufig aufeinander auf. D.h. das nachfolgende Query benötigt die Ausgabe einer vorangegangenen Abfrage und damit dies sichergestellt

wird, dürfen in meinem Subsystem die Querys nur nacheinander und nicht parallel abgearbeitet werden. Das Subsystem ist jetzt soweit fertig gestellt und kann anschließend mit dem Befehl STRSBS STDQRY gestartet werden. Wenn Sie Ihr Subsystem starten, können Sie anschließend mit dem Befehl WRKACT-JOB sehen, dass das Subsystem aktiv ist.



Überprüfung des gestarteten Subsystems

2.7.2.1
Seite 6

Jobwarteschlangen-Einträge bearbeiten

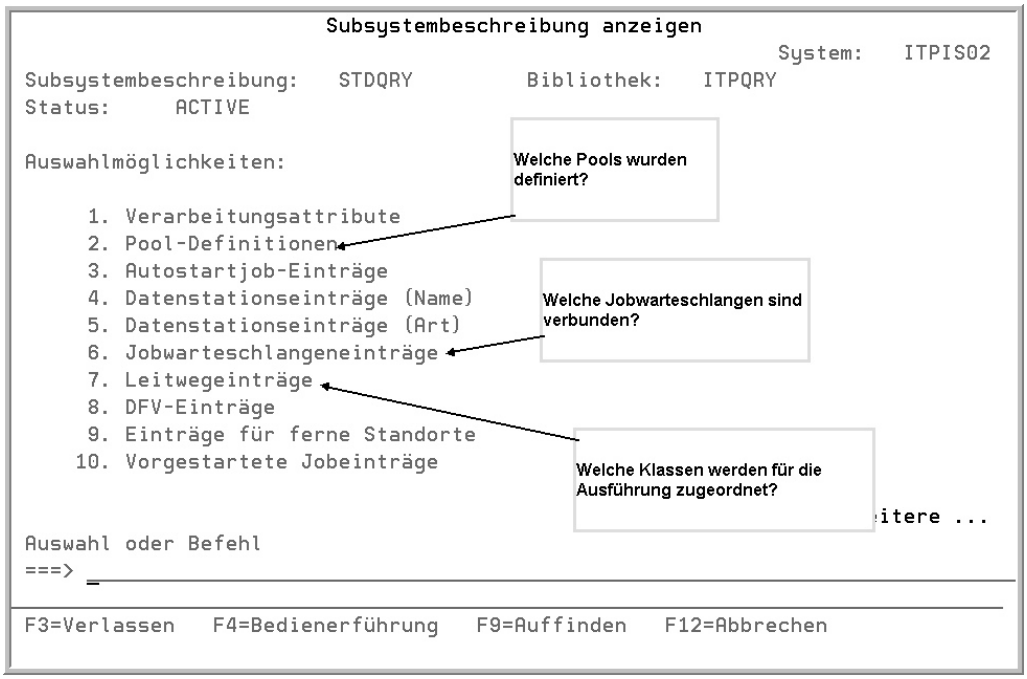
Wir haben jetzt eine Jobwarteschlange und ein Subsystem. Allerdings müssen wir die beiden Objekte noch miteinander bekannt machen, d.h. wir müssen die Jobwarteschlange STDQRY mit dem Subsystem STDQRY verbinden. Jobwarteschlangen-Einträge fügen Sie mit dem Befehl ADDJOBQE (add jobqueue entry) hinzu. Wenn Sie einen Eintrag entfernen wollen, müssen Sie den Befehl RMVJOBQE (remove jobqueue entry) verwenden. Schauen wir uns also den Befehl ADDJOBQE etwas genauer an:

```
                Jobwarteschl.eintr. hinzufügen (ADDJOBQE)

Auswahl eingeben und Eingabetaste drücken.

Subsystembeschreibung  . . . . . SBSD          stdqry
  Bibliothek . . . . . .                         *LIBL
Jobwarteschlange . . . . . . . . JOBQ          stdqry
  Bibliothek . . . . . .                         *LIBL
Maximale Anzahl aktiver Jobs . . MAXACT        1
Folgenummer  . . . . . . . . . . SEQNBR        10
Max. aktiv Priorität 1 . . . . . MAXPTY1       *NOMAX
Max. aktiv Priorität 2 . . . . . MAXPTY2       *NOMAX
Max. aktiv Priorität 3 . . . . . MAXPTY3       *NOMAX
Max. aktiv Priorität 4 . . . . . MAXPTY4       *NOMAX
Max. aktiv Priorität 5 . . . . . MAXPTY5       *NOMAX
Max. aktiv Priorität 6 . . . . . MAXPTY6       *NOMAX
Max. aktiv Priorität 7 . . . . . MAXPTY7       *NOMAX
Max. aktiv Priorität 8 . . . . . MAXPTY8       *NOMAX
Max. aktiv Priorität 9 . . . . . MAXPTY9       *NOMAX

                                                                    Ende
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Jobwarteschlange einem Subsystem zuordnen (ADDJOBQE)

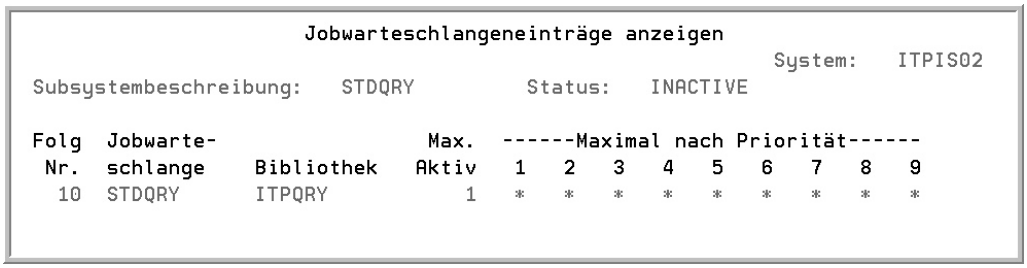
Die Parameter „Subsystembeschreibung“ und „Jobwarteschlange“ sind selbsterklärend. Der Parameter „Maximale Anzahl aktiver Jobs“ bestimmt, wie viele Jobs diese Warteschlange gleichzeitig an das Subsystem übergeben darf. In unserem Fall darf das Subsystem nur einen Job zurzeit ausführen, also darf die Jobwarteschlange auch nur einen Job zurzeit übergeben. Die Folgenummer ist eine fortlaufende Nummerierung der Jobwarteschlangen-Einträge. Da ein Subsystem mit mehr als einer Warteschlange verbunden sein kann, müssen die Einträge eindeutig nummeriert sein. In unserem Beispiel haben wir nur einen Warteschlangen-Eintrag. Um aber den Konventionen zu genügen, wählen wir hier einen beliebigen Wert. Die Verbindung der Subsysteme mit den zugehörigen Warteschlangen können wir in den Subsystembeschreibungen prüfen.

2.7.2.2
Seite 2



Subsystemdefinitionen anzeigen (DSPSBSD)

Der Subsystembeschreibung können Sie die Pooldefinitionen (Pkt. 2) entnehmen und natürlich sehen Sie, welche Jobwarteschlangen (Pkt. 6) zugeordnet wurden.



Menüpunkt 6 „Jobwarteschlangen-Einträge“

Leitwegdateneinträge bearbeiten

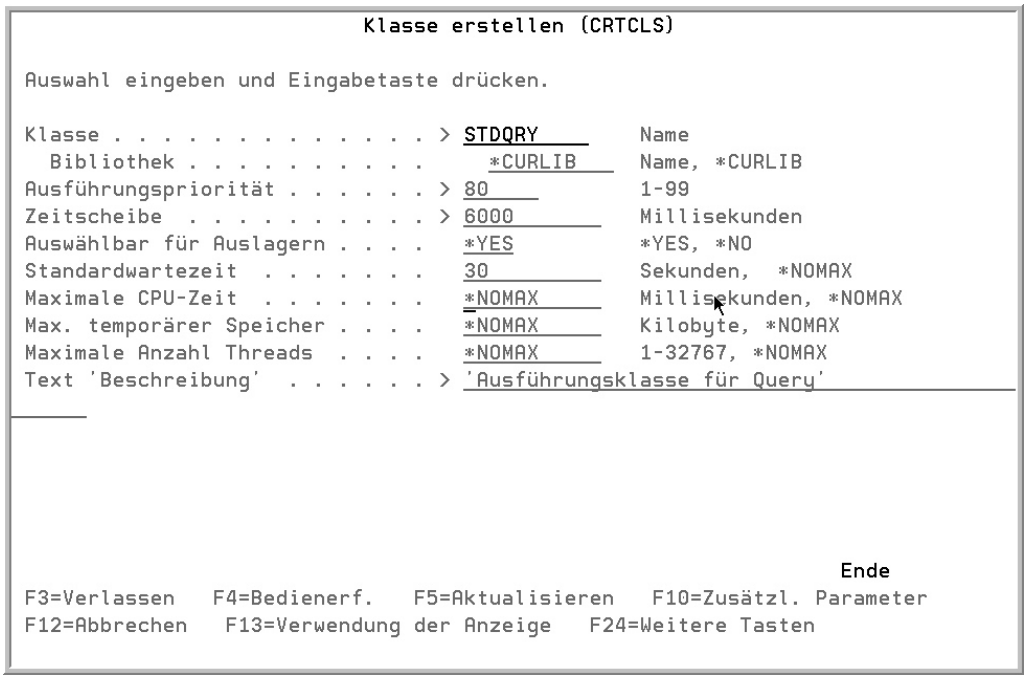
Fertig! Unser Subsystem ist mit der Warteschlange verbunden. Jeder Job, der in die Warteschlange STDQRY gestellt wird, läuft jetzt automatisch im Subsystem STDQRY ab. Oder doch nicht? Wenn Sie einen Job in Ihre Warteschlange übergeben, werden Sie folgendes Fehlerprotokoll (JOBLOG) erhalten:

```
                          Spool-Datei anzeigen
Datei . . . . . :   QPJOBLOG                        Seite/Zeile 1/1
Steuerung . . . .   F-1                             Spalten     38 - 115
Suchen  . . . . .   _____________________________
..4....+....5....+....6....+....7....+....8....+....9....+....0....+....1....+
            Jobprotokoll                        ITPIS02  23.03.05 13:31:50
FTJOBD        Benutzer  . . . . :   LGADOZ02     Nummer . . . . . .
FTJOBD        Bibliothek  . . . :   QGPL
W  DATUM     ZEIT             VON PGM      BIBLIOTHEK  INST     AN PGM      BI
   23.03.05  13:31:50,499696  QWTMMERH     QSYS        0110     *EXT
Von Benutzer  . . . . . :   QSYS
Nachricht . . . :   CPC1117 Leitwegdaten entsprechen nicht Leitwegeintrag.
Fehlerbeseitigung:  Entweder die Leitwegdaten ändern oder mit dem Befehl
  ADDRTGE einen Leitwegeintrag hinzufügen, der den Leitwegdaten in der
  Subsystembeschreibung entspricht.
   23.03.05  13:31:50,502656  QWTPIIPP     QSYS        061C     *EXT
Nachricht . . . :   Job 089393/LGADOZ02/QDFTJOBD im Subsystem STDQRY in ITPQRY
  am 23.03.05 um 13:31:50 gestartet. Job im System am 23.03.05 um 13:30:31.
  angekommen.
   23.03.05  13:31:50,503160  QWTPCRJA     QSYS        0108     *EXT
Nachricht . . . :   Job 089393/LGADOZ02/QDFTJOBD übergeben.
                                                                  Weitere ...
F3=Verlassen  F12=Abbrechen   F19=Links   F20=Rechts   F24=Weitere Tasten
```

Fehlerprotokoll

Das Betriebssystem teilt Ihnen mit, dass der Job nicht verarbeitet wurde aufgrund fehlender Leitwegdaten. Die Fehlermeldung ist nicht sehr hilfreich oder wissen Sie, was Leitwegdaten sind? Und wofür Sie gebraucht werden? Tatsächlich kann die iSeries Ihren Arbeitsauftrag nicht verarbeiten, weil die Ablaufattribute nicht zugeordnet werden können. D.h. dem System fehlen z.B. Informationen darüber mit welcher Priorität und wie lange der Job verarbeitet werden darf. Diese Informationen enthält die Jobklasse, die in unserem Fall noch nicht zugeordnet werden kann.

2.7.2.3
Seite 2



Eine Jobklasse erstellen (CRTCLS)

Ich erstelle also zunächst einmal eine eigene Jobklasse für unsere Querys. Querys belasten das System z. T. erheblich. Ich möchte daher sicherstellen, dass meine Abfragen nicht mit den interaktiven Jobs und auch nicht mit anderen Batch-Jobs des Systems konkurrieren. Aus diesem Grund wähle ich als Ausführungspriorität den Wert 80. Interaktive Jobs haben eine Ausführungspriorität von 20, Batch-Jobs laufen normalerweise mit einer Ausführungspriorität von 50. Meine Abfragen werden also nachrangig bearbeitet werden. Eine Erläuterung der weiteren Parameter finden Sie bereits im Kapitel 2.6.3, so dass ich an dieser Stelle darauf verzichte. Jetzt muss ich die Jobklasse meinem Subsystem bekannt geben. Dafür verwenden wir den Befehl ADDRTGE (add routing entry). Falls Sie einen Leitwegeintrag wieder entfernen möchten, geschieht dies mit dem Befehl RMVRTGE (remove routing entry). Zur Veränderung des Leitwegeintrages steht der Befehl CHGRTGE(change routing entry) zur Verfügung.

```
                Leitwegeintrag hinzufügen (ADDRTGE)

Auswahl eingeben und Eingabetaste drücken.

Subsystembeschreibung  . . . . . SBSD       >   STRQRY
  Bibliothek . . . . . .                        *LIBL
Folgenummer f. Leitwegeintrag  . SEQNBR     >   10
Vergleichsdaten:                 CMPVAL
  Vergleichswert . . . . . .             >   'QUERY'

  Startposition  . . . . . .                    1
Programm aufrufen  . . . . . . . PGM        >   QCMD
  Bibliothek . . . . . .                        *LIBL
Klasse . . . . . . . . . . . . . CLS        >   STRQRY
  Bibliothek . . . . . .                        *LIBL
Max. aktive Leitwegschritte  . . MAXACT         *NOMAX
Speicherpool-ID  . . . . . . . . POOLID     >   2
```

Einen Leitwegeintrag hinzufügen (ADDRTGE)

Der Befehl ADDRTGE besteht aus einigen Parametern, die wir jetzt im Einzelnen besprechen werden.

Der Parameter „Subsystembeschreibung“ (SBSD)

Er bestimmt den Namen und die Bibliothek des Subsystems zu der der Leitwegeintrag hinzugefügt wird.

Der Parameter „Folgenummer f. Leitwegeintrag“ (SEQNBR)

Der ist wieder eine fortlaufende Nummer. Da ein Subsystem mehr als einen Leitwegeintrag erhalten kann, werden die Vergleichswerte (CMPVAL) in aufsteigender Reihenfolge der Folgenummern verglichen. Die Suche endet, wenn eine Übereinstimmung auftritt oder der letzte Leitwegeintrag geprüft wurde. Daher wird, wenn mehr als eine Übereinstimmungsmöglichkeit vorliegt, nur die erste Übereinstimmung verarbeitet.

Der Parameter „Vergleichsdaten“ (CMPVAL)

Er besteht aus zwei Elementen: Das erste Element enthält eine beliebige Zeichenfolge aus max. 80 Zeichen. Beachten Sie bitte hierbei die Groß- und Kleinschreibung. Wenn Sie Kleinbuchstaben in der Zeichenfolge verwenden wollen, müssen Sie die Zeichenfolge in Hochkommata einschließen, weil andernfalls das System die Kleinbuchstaben in Großbuchstaben umsetzt. Das zweite Element kennzeichnet, welche Position der Leitwegdatenzeichenfolge als Startposition beim Vergleich dient. Wenn später die Leitwegdaten des Jobs mit dieser Zeichenfolge übereinstimmen, wird die Verarbeitung anhand dieses Leitwegeintrages ausgelöst. Man spricht in diesem Fall von dem Start des ersten Leitwegschrittes.

2.7.2.3
Seite 4

Der Parameter „Programm aufrufen“ (PGM)

Dieser gibt den Namen und die Bibliothek des Programms an, das im ersten Leitwegschritt ablaufen soll. Dem angegebenen Programm können keine Parameter übergeben werden. Ich nutze hierfür das Befehlsinterpreter-Programm QCMD der IBM.

Der Parameter „Klasse“ (CLS)

Der gibt den Namen der Klasse an, die verwendet wird. Die Klasse definiert die Attribute der Ablaufumgebung.

Der Parameter „Max. aktive Leitwegschritte“ (MAXACT)

Dieser gibt die maximale Anzahl Jobs an, die über diesen Leitwegeintrag gleichzeitig aktiv sein können. Ist ein Subsystem aktiv und die hier definierte maximale Anzahl Leitwegschritte ist erreicht, schlägt jeder nachfolgende Versuch fehl, einen Leitwegschritt über den Leitwegeintrag zu starten. Der Job, der versucht, den Leitwegschritt zu starten, wird beendet und das Subsystem sendet eine entsprechende Nachricht an das Jobprotokoll. Meistens liegt kein Grund zum Steuern der Leitwegschritte vor. In diesem Fall verwendet man den Sonderwert *NOMAX. Andernfalls kann die Anzahl der Leitwegschritte durch die Werte 0 – 1000 begrenzt werden.

Der Parameter „Speicherpool-ID“ (POOL-ID)

Der bestimmt den Pool, in dem der Job abläuft. Die hier angegebene Pool-ID verweist auf die Speicherpools der Subsystembeschreibung. Da ich will, dass die Abfragen im privaten Pool laufen, muss ich die Pool-ID 2 verwenden analog meiner Subsystembeschreibung. Die eingetragenen Leitwegdaten können wir anschließend mit dem Befehl DSPSBSD Menüpunkt 7 prüfen. Unter der Folgenummer 10 finden wir unseren Vergleichswert „QUERY“.

```
                          Leitwegeinträge anzeigen
                                                              System:   ITPIS02
Subsystembeschreibung:   STDQRY         Status:   ACTIVE

Auswahl eingeben und Eingabetaste drücken.
  5=Details anzeigen

Aus-   Folge-                                                          Start-
wahl   nummer   Programm     Bibliothek   Vergleichswert              position
 _        10    QCMD         QSYS         'QUERY'                        1
 _        99    QCMD         QSYS         *ANY
```

Der Befehl DSPSBSD Menüpunkt 7

Wenn wir uns die Details dieses Leitwegeintrages anschauen, können wir auch erkennen, welche Klasse sich hinter dem Eintrag verbirgt.

```
                    Details des Leitwegeintrags anzeigen
                                                          System:   ITPIS02
Subsystembeschreibung:   STDQRY         Status:   ACTIVE

Folgenummer des Leitwegeintrags . . . . . :   10
Programm  . . . . . . . . . . . . . . . . :   QCMD
  Bibliothek  . . . . . . . . . . . . . . :     QSYS
Klasse  . . . . . . . . . . . . . . . . . :   STDQRY
  Bibliothek  . . . . . . . . . . . . . . :     ITPQRY
Maximale Anzahl aktiver Leitwegschritte . . :   *NOMAX
Pool-ID . . . .  . . . . . . . . . . . . . :   2
Vergleichswert  . . . . . . . . . . . . . . :   'QUERY'

Startposition für Vergleich . . . . . . . . :   1
Threads mit Ressourcenaffinität :
  Gruppe  . . . . . . . . . . . . . . . . . :   *SYSVAL
  Ebene . . . . . . . . . . . . . . . . . . :
Gruppe mit Ressourcenaffinität  . . . . . . :   *NO


Eingabetaste --> Weiter

F3=Verlassen   F12=Abbrechen   F14=Vorherigen Eintrag anzeigen
```

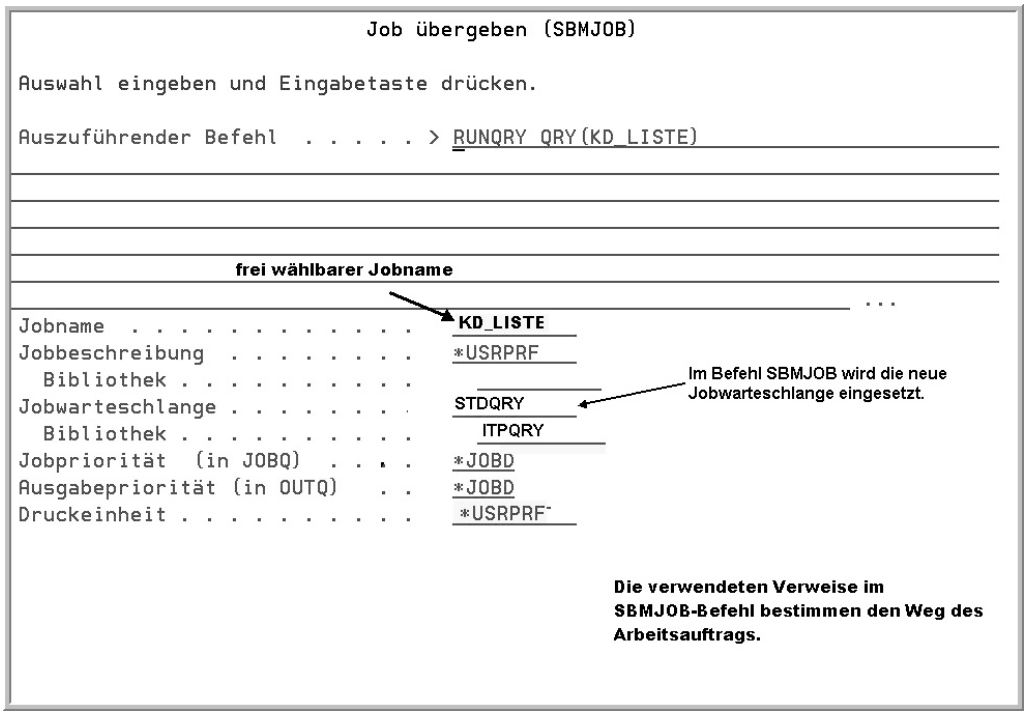
Details anzeigen

Alle Jobs, die als Leitwegdaten die Zeichenfolge „QUERY“ verwenden, werden jetzt in unserem Subsystem mit den entsprechenden Ablaufattributen im Pool 2 ausgeführt. Sobald eine Abfrage andere Leitwegdaten enthält, würden wir wieder ein JOBLOG erhalten. Damit das nicht passiert, ist es möglich einen Leitwegeintrag mit dem Vergleichswert *ANY einzutragen. Durch den Eintrag *ANY werden alle Leitwegdaten akzeptiert. Daher muss dieser Leitwegeintrag den höchsten Folgenummernwert aller Leitwegeinträge in der Subsystembeschreibung besitzen.

2.7.2.3
Seite 6

Die neue Arbeitsumgebung nutzen

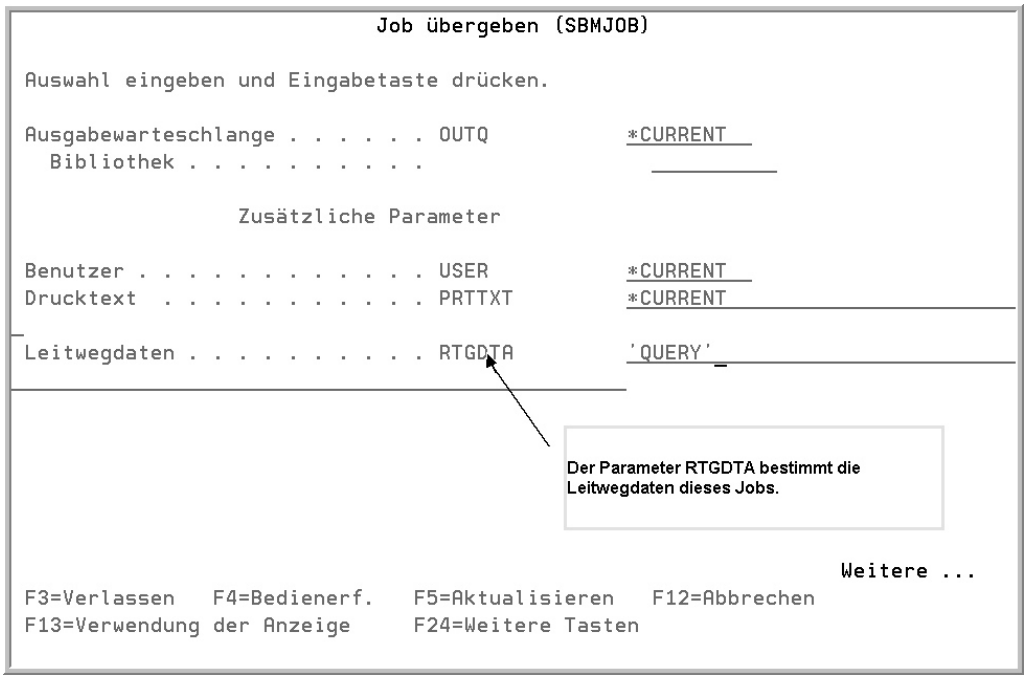
Jetzt sind wir fertig und können unsere Abfrage in der neuen Umgebung ausführen lassen, indem wir im SBMJOB-Befehl unsere neue Jobwarteschlange nutzen.



Die Abfrage starten

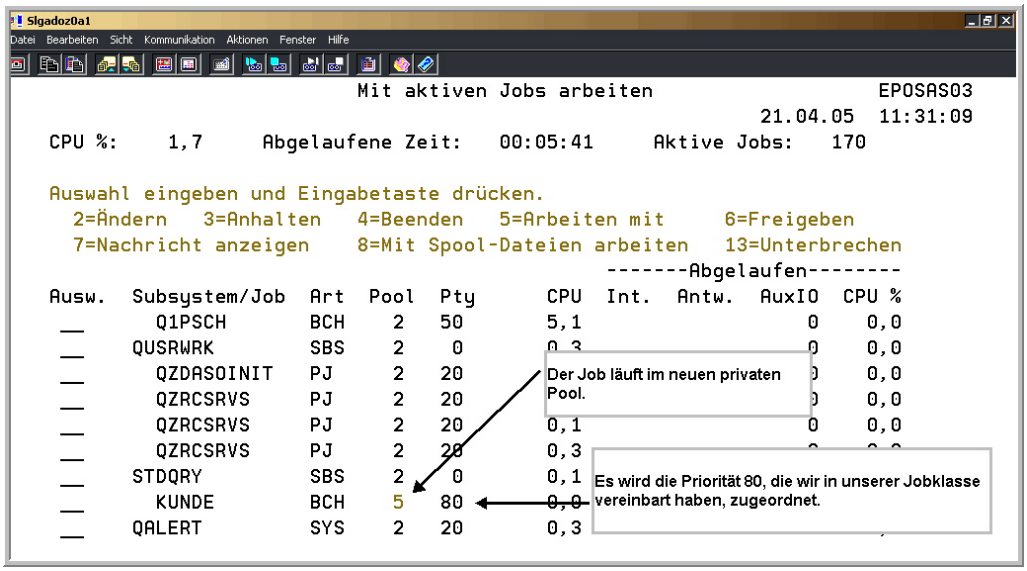
Wir müssen natürlich auch noch unsere Leitwegdaten bekannt geben. Dies geschieht auf der 2. Seite des SBMJOB-Befehls.

2.7.2.4
Seite 2



Die Leitwegdaten bestimmen

Nach der Freigabe werden Sie sehen, dass die Abfrage im neuen Subsystem STDQRY läuft.



Überprüfung des Jobs

Zwei Probleme könnte es ziemlich schnell geben: Wie verhindern Sie, dass Ihre Anwender, die Abfragen interaktiv im Subsystem QINTER ausführen und wie können Sie den Aufruf des Querys vereinfachen?

Die interaktive Ausführung des Befehls RUNQRY ist sehr einfach zu verhindern. Erstellen Sie zunächst eine Kopie des Befehls RUNQRY mit dem Befehl

CRTDUPOBJ. Die Kopie legen wir in unserer eigenen Systembibliothek ab, um unsere Modifikationen vom Original zu trennen.

```
                  Doppeltes Objekt erstellen (CRTDUPOBJ)

Auswahl eingeben und Eingabetaste drücken.

Von Objekt . . . . . . . . . . . OBJ            runqry
Von Bibliothek . . . . . . . . . FROMLIB        qsys
Objektart  . . . . . . . . . . . OBJTYPE        *cmd
               + für weitere Werte
Nach Bibliothek  . . . . . . . . TOLIB          EPSYS
Neues Objekt . . . . . . . . . . NEWOBJ         *OBJ
Von ASP-Einheit  . . . . . . . . ASPDEV         *
In ASP-Einheit . . . . . . . . . TOASPDEV       *ASPDEV
```

Kopie des Befehls RUNQRY

Anschließend modifizieren wir das Objekt mit dem Befehl CHGCMD. Mit dem Befehl CHGCMD können Sie die Attribute der Befehle verändern. So ist es auch möglich durch Abändern des Parameters „Ausführung wo zulässig“ (ALLOW) die interaktive Nutzung des Commands RUNQRY zu verhindern, wenn Sie für diesen Parameter die Werte *BMOD, *BREXX, *BPGM und *BATCH nutzen.

```
                        Befehl ändern (CHGCMD)

Auswahl eingeben und Eingabetaste drücken.

Befehl . . . . . . . . . . . . . > RUNQRY       Name
  Bibliothek . . . . . . . . . . >   ITPQRY     Name, *LIBL, *CURLIB
Befehlsverarbeitungsprogramm . .   QQUDA       Name, *SAME, *REXX
  Bibliothek . . . . . . . . . .     QSYS      Name, *LIBL, *CURLIB
Gültigkeitsprüfprogramm  . . . .   VCP_RUNQRY  Name, *SAME, *NONE
  Bibliothek . . . . . . . . . .     ITPQRY    Name, *LIBL, *CURLIB
Gültig bei Modus . . . . . . . .   *PROD       *SAME, *ALL, *PROD, *DEBUG...
                                   *DEBUG
                                   *SERVICE
Ausführung wo zulässig . . . . .   *BMOD       <-- Der Parameter 'Ausführung wo zulässig',
                                   *BREXX          bestimmt in welchen Umgebungen der
                                   *BPGM           Befehl ausgeführt werden darf.
     + für weitere Werte           *BATCH
Benutzer begrenzt zulassen . . .   *NO         *SAME, *NO, *YES
Bücherfach . . . . . . . . . . .   *LIST       Name, *SAME, *LIST, *NONE

                                                                    Weitere ...
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

RUNQRY in einer Batch-Umgebung nutzen

Nachdem Sie die Attribute verändert haben, erhält jeder Anwender, der die geänderte Version des Betriebssystembefehls RUNQRY nutzt, folgende Information beim Aufruf in einer interaktiven Umgebung:

2.7.2.4
Seite 4

```
                                                                  Ende
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
Befehl RUNQRY in dieser Umgebung unzulässig.
```

Fehlermeldung bei interaktiver Ausführung des Befehls RUNQRY

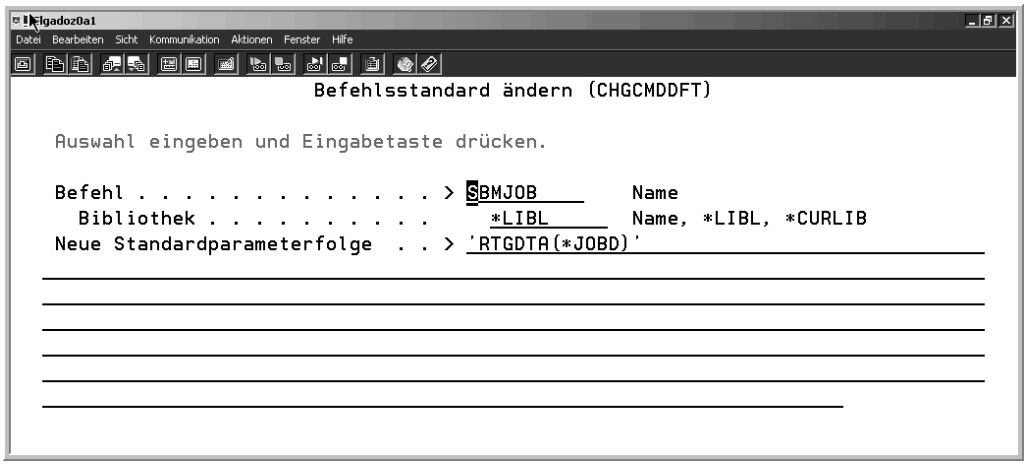
Schwieriger ist die Automatisierung des Aufrufs. Sie können natürlich eine eigene Jobbeschreibung erstellen:

```
SIgadoz0a1
Datei  Bearbeiten  Sicht  Kommunikation  Aktionen  Fenster  Hilfe
                        Jobbeschreibung erstellen (CRTJOBD)

  Auswahl eingeben und Eingabetaste drücken.

  Jobbeschreibung  . . . . . > STDQRY      Name
    Bibliothek . . . . . .     *CURLIB     Name, *CURLIB
  Jobwarteschlange . . . . . > STDQRY      Name
    Bibliothek . . . . . .     *LIBL       Name, *LIBL, *CURLIB
  Jobpriorität  (in JOBQ)  . . .  5        1-9
  Ausgabepriorität (in OUTQ) . . .  5      1-9
  Druckeinheit . . . . . .   *USRPRF       Name, *USRPRF, *SYSVAL...
  Ausgabewarteschlange . . . . .  *USRPRF  Name, *USRPRF, *DEV, *WRKSTN
    Bibliothek . . . . . .                 Name, *LIBL, *CURLIB
  Text 'Beschreibung'  . . . . .  *BLANK
_____

                          Zusätzliche Parameter

  Benutzer . . . . . .   *RQD       Name, *RQD
  Drucktext  . . . . . .   *SYSVAL
```

Bedienerführung des Befehls CRTJOBD Seite 1

Sie sehen, dass wir in der Jobbeschreibung unsere Jobwarteschlange benennen können und es ist ebenfalls möglich auf der 2. Seite, die Leitwegdaten zu hinterlegen. An dieser Stelle bekommen wir aber ein Problem, denn der Befehl SBMJOB hat als Default-Einstellung die Leitwegdaten QCMDB und verweist nicht auf die Leitwegdaten der Jobbeschreibung. Wir haben jetzt verschiedene Möglichkeiten zu reagieren:

1. Wir verzichten auf eigene Leitwegdaten. Was möglich ist, da unser Subsystem durch den Leitwegeintrag *ANY alle Leitwegdaten akzeptiert. Dann können wir den Querys in dem Subsystem STDQRY allerdings nur unterschiedliche Jobklassen zuordnen, wenn wir die Leitwegdaten direkt im Befehl SBMJOB eingeben.
2. Wir kopieren den Befehl SBMJOB und verändern die Default-Einstellungen dahingehend, dass der Befehl die Leitwegdaten über die Jobbeschreibung sucht. Falls Sie sich für die zweite Lösung entscheiden, müssen Sie sicherstellen, dass andere Jobs in Ihrem System dadurch nicht manipuliert werden.



Veränderung der Defaulteinstellung des Befehls SBMJOB

Die neu erstellte Jobbeschreibung können Sie anschließend im Befehl SBMJOB nutzen oder direkt im Benutzerprofil hinterlegen. Bedenken Sie aber, dass alle Batch-Jobs, die auf die Standardjobbeschreibung zugreifen, die neue Jobbeschreibung im Benutzerprofil verwenden. Es wird vermutlich am einfachsten sein, sich einen ganz neuen RUNQRY-Befehl zu schreiben, um all die kleinen aber manchmal doch sehr hartnäckigen Probleme zu lösen.

2.7.2.4
Seite 6

Remote Journal

Datensicherung ist bekanntlich ein Verfahren, mit dem ein aktueller Informationsstand eines IT-Systems gesichert wird. Datensicherungen im IBM i-Umfeld werden häufig in Form einer getrennten Datensicherung durchgeführt, d. h. getrennt in Systemdaten und in Benutzerdaten.

Meist findet man getrennte Datensicherung dort, wo täglich die Benutzerdaten gesichert werden. Die Sicherung der Benutzerdaten dient in Verbindung mit einer (meist am Wochenende erstellten) Systemsicherung zur Wiederherstellung der Informationen im Notfall.

So sicher das Verfahren selbst funktioniert, so sicher ist dabei auch, dass Datenverluste unvermeidbar sind. Denn eine klassische Datensicherung erstellt lediglich ein Abbild des Datenbestands zu einem bestimmten Zeitpunkt.

Was aber ist mit den Daten, die sich seit einer Datensicherung geändert haben, und die deshalb nach einem Systemausfall bei diesem Verfahren als verloren angesehen werden müssen?

Betrachten wir dazu ein kleines Beispiel:

In einem Unternehmen werden einmal wöchentlich die Systemdaten gesichert. Dieser Sicherungsprozess wird am Wochenende durchgeführt. Ein CL-Programm sichert das gesamte System.

Täglich werden im Zuge der Nachtverarbeitung die Bewegungsdaten gesichert. Das garantiert, dass alle Benutzerdaten auf Band gezogen werden. Nehmen wir einmal an, die Datensicherung startet um 22.00 Uhr. Damit werden die Datenstände zum Zeitpunkt 22.00 Uhr auf Band gesichert.

Am nächsten Morgen wird ab 06.00 Uhr wieder mit dem System gearbeitet. Aufträge werden erfasst, Warenausgänge gebucht, elektronische Bestellungen gehen ein und per EDI werden Zahlungsinformationen an Kunden und Lieferanten übermittelt. Um 11.00 Uhr fällt das System aus – Datenverlust – eine Rücksicherung muss durchgeführt werden.

Dieses Horrorszenario kann einem Unternehmen enorme Probleme bereiten. Aufgrund der Integration der Systeme miteinander ist eine Wiederherstellung der Zustands, wie er zum Zeitpunkt des Absturzes gegeben war, meist unmöglich.

Natürlich reduzieren Sicherheitstechniken wie RAID oder Spiegelung die Gefahr deutlich – aber bei einem einzelnen System kann es dennoch zu einer Katastrophe kommen.

2.8.1
Seite 2

Wer auf solche Datenverluste nicht entsprechend reagieren kann, benötigt eine Lösung, die über den Einsatz einer einzelnen Maschine hinausgeht.

Stellen wir uns das zuvor demonstrierte Beispiel nochmals vor. Dieses Mal verfügen wir jedoch über eine Zweitmaschine, auf die in Echtzeit die Daten des Produktionssystems übertragen werden. Die Transaktionen werden vom Produktionssystem aus mit Hilfe von fernen Journalen auf das Backupsystem übertragen. Damit stehen die Daten quasi doppelt zur Verfügung. Fällt nun das Produktionssystem aus, dann kann man mit gewissen Anpassungen die Arbeit auf dem Backupsystem wieder aufnehmen.

Natürlich ist dies nicht ganz so einfach, wie man es hier in wenigen Worten beschreibt, aber vom Grundsatz her ist die Verwendung von fernen Journalen eben die Möglichkeit, Daten eines Systems auf ein anderes System in Echtzeit zu übertragen.

IBM hat dem System i für Fälle, die bei einem einzelnen System zu Datenverlusten führen würden, natürlich auch Funktionen mitgegeben, die mehr oder weniger elegant und komplex dafür Sorge tragen, dass es nicht zum Verlust von Daten kommt, die sich zwischen Datensicherung und Ausfall verändert haben.

Hochverfügbarkeit ist eine Eigenschaft, die wir unserem System i sicher zusprechen können. Aber was ist, wenn der unwahrscheinliche Fall doch einmal eintritt und das System unvorhergesehen ausfällt?

Hochverfügbarkeit ist ein weit reichender Begriff, der allerdings eine Menge Spielraum bei der Interpretation lässt. Muss ein System 24 Stunden am Tag und das 7 Tage die Woche an 365 Tagen zur Verfügung stehen, dann ist das sicher ein Musterbeispiel für eine Hochverfügbarkeitsumgebung. Ist in einer Umgebung hingegen ein kurzer Ausfall (1-4 Stunden) verkraftbar, dann ist das auch eine gewisse Form der Hochverfügbarkeit – aber alleine der Zeitfaktor „Stunde“ unterscheidet sich doch erheblich von der eng umgrenzten Definition der Hochverfügbarkeit, die nicht einmal wenige Sekunden als Toleranz zulässt.

Wie erreicht man eine solche Hochverfügbarkeit? Natürlich mit Hilfe einer zweiten Maschine, die idealer Weise an einem fernen Standort installiert ist. Viele Wege führen nach Rom, und mindestens genauso viele Wege führen dazu, Informationen von einem System auf ein anderes zu übertragen.

Ich möchte an dieser Stelle nicht in die weite und komplexe Welt der Hochverfügbarkeitslösungen einsteigen, die von IBM und auch anderen Anbietern für das System i angeboten werden. Das käme einem Produktvortrag gleich. Vielmehr ist mir an dieser Stelle daran gelegen, Ihnen einen Einblick in eine Standardfunktion des System i zu geben, mit der wir in der Lage sind, Daten und Informationen von einem System auf ein anderes zu übertragen. Diese

Standardfunktion wird fernes Journal oder auch „Remote Journal“ genannt.

Vielleicht kennen und nutzen Sie es bereits: das Journal auf dem IBM i. Wobei – diese Formulierung war ungenau: Denn ein Journal auf dem IBM i kann zu ganz unterschiedlichen Zwecken genutzt werden, wie die folgenden Beispiele auszugsweise demonstrieren:

– Journal als Basis für die Aufzeichnung von Dateiänderungen
– Journal als Basis für die Commit-Steuerung in Anwendungen zum Zwecke der Umsetzung von Transaktionssicherheit
– Journal als Basis für die Aufzeichnung sicherheitsrelevanter Informationen (QAUDJRN)

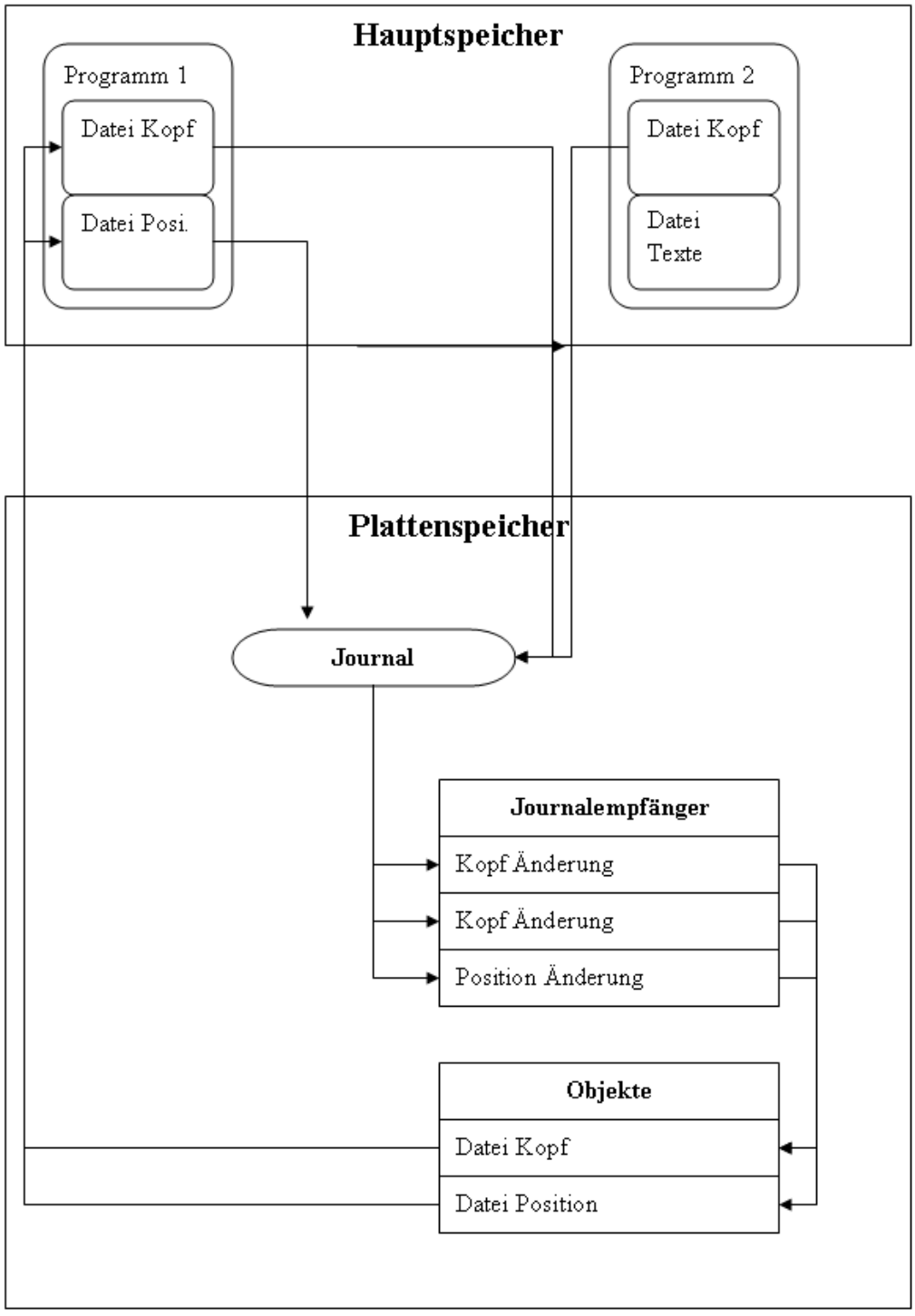
Betrachtet man das Einsatzspektrum von Journalen, dann stellt man sehr schnell fest, dass es enorm weitreichend ist. Sicherheitsaufzeichnungen, Transaktionskontrollen und Änderungsaufzeichnungen sind dabei nur einige Hauptpunkte. Darüber hinaus finden wir auch auf Betriebssystemebene unterschiedliche Journale, die jeweils individuelle Aufgaben übernehmen.

Basierend auf dem lokalen Journal hat IBM bereits mit Version V4R2 die fernen Journale eingeführt. Mit dieser Erweiterung sind wir heute in der Lage, komfortabel und – relativ gesehen – auch sehr einfach Journalinhalte von einem System auf ein anderes System zu übertragen. Der Lösungsansatz basiert dabei auf der Verwendung typischer Konstellationen lokaler Journale und lokaler Journalempfänger, denen dann als Erweiterung ferne Journale bzw. ferne Journalempfänger zugeordnet werden. Damit basieren ferne Journale also auf dem Einsatz der lokalen Journale.

Schauen wir uns den Prozess des Journalings einmal grob an. Bei einem Journal handelt es sich um eine im Betriebssystem verankerte Funktion, mit der beispielsweise Änderungen an einem Objekt aufgezeichnet werden. Jede Änderung an einem in einem Journal aufgezeichneten Objekt führt dazu, dass ein korrespondierender Journaleintrag erstellt wird. Dieser Journaleintrag beinhaltet die geänderten Informationen. Der Detailgrad der Informationen und der Aktualitätsstand können je nach Vorgaben variieren. So finden wir in einer bestimmten Konstellation z. B. nur solche Journaleinträge vor, die den Status eines Datensatzes in einer Datei nach einer Änderung darstellen. In anderen Konstellationen hingegen finden wir zu einer Änderung gleich zwei Journaleinträge: einen mit Stand vor der Änderung und einen mit Stand nach der Änderung.

2.8.1
Seite 4

Die folgende Abbildung zeigt schematisch den Ablauf in einer Journalumgebung:



Schematisches Remote-Journal

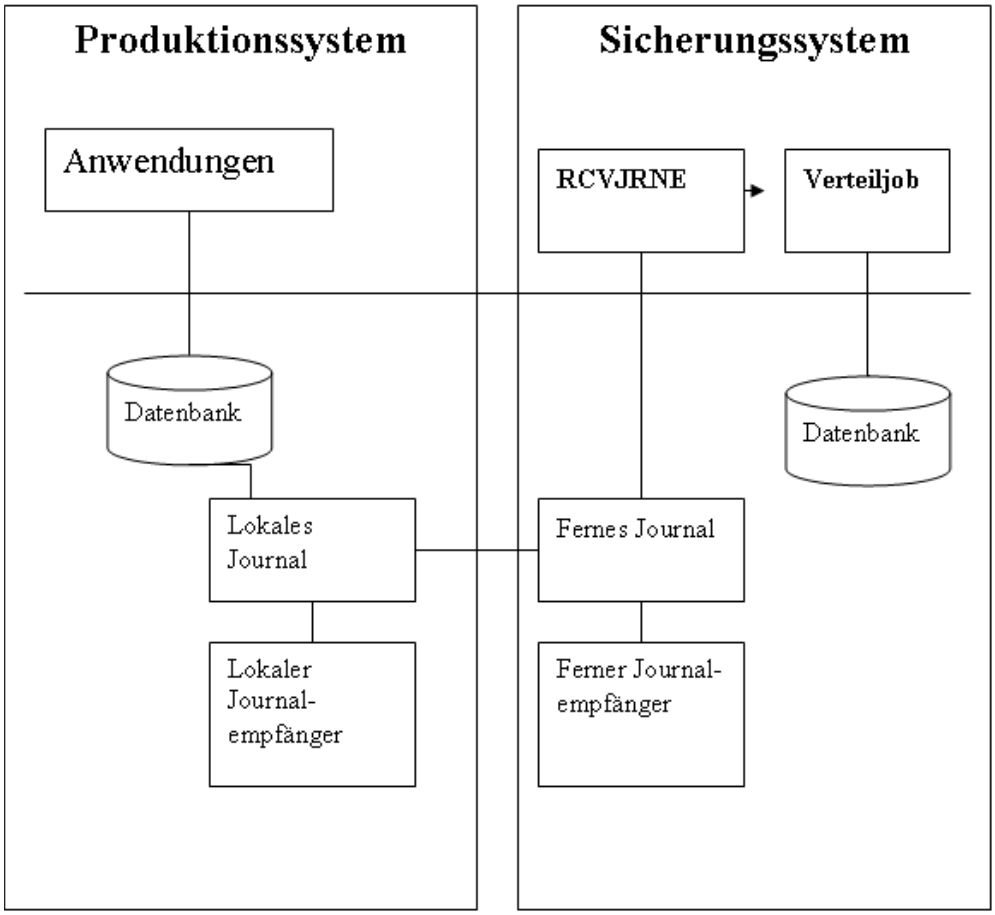
Wie man erkennen kann, werden Änderungen an den Dateien „Kopf“ und „Position“ sowohl in die Dateien selbst, als auch in das zugehörige Journal geschrieben.

Wenn zusätzlich zum lokalen Journal ein fernes Journal eingerichtet wird, dann werden die Änderungen an einem Objekt (in unserem Beispiel: an einer Datei) zusätzlich noch in ein zweites Journal geschrieben, das sich auf einem fernen System befindet. Die Einrichtung eines fernen Journals ist auch auf verschiedenen Maschinen möglich. So kann für ein Quellsystem eine Lösung etabliert werden, bei der z. B. zwei Backupsysteme als Basis für die fernen Journale eingesetzt werden.

Das Ergebnis: quasi eine Echtzeitdatensicherung der in dem Journal aufgezeichneten Informationen. Damit können viele Unternehmen eine bestehende Sicherheitslücke schließen, die anderenfalls in einem Notfall dazu führen würde, dass Daten verloren gehen.

In anderen Bereichen der IT – nehmen wir Domino und Notes als Beispiel – spricht man bei solchen Datenaktionen auch von Replikationen. Im Groben ist dies auch für die fernen Journale zutreffend.

Die folgende Abbildung illustriert dieses Verfahren, bei dem zusätzlich zu einem lokalen Journal noch ein fernes Journal eingesetzt wird.



Ferne Journale

2.8.1
Seite 6

AS/400-Kenner (hier verwende ich bewusst den klassischen Namen) werden sich jetzt an DDM-Zeiten erinnern. DDM ist aber nicht mit fernen Journalen zu verwechseln!

Wie man anhand der vorhergehenden Abbildung erkennen kann, gibt es eine Verbindung zwischen lokalen und fernen Journalen. Bei der Einrichtung des fernen Journals wird einmalig jeder Eintrag des lokalen Journals in das ferne Journal übertragen. Ist diese einleitende Übertragung abgeschlossen, dann werden anschließend alle neuen Einträge aus dem lokalen Journal automatisch in das ferne Journal übertragen. In Abhängigkeit von der Geschwindigkeit der Verbindung zwischen den beiden Systemen können derart schnelle Übertragungen erreicht werden, dass wir quasi von einer Echtzeitübertragung sprechen können.

Soviel zu den Grundlagen von lokalen Journalen und einem ersten kleinen Einblick in ferne Journale. Kommen wir nochmals zu den Grundsätzen des fernen Journals und dem Aufzeigen der Möglichkeiten zurück, die das ferne Journal mit sich bringt.

Stellt man Überlegungen in Bezug auf den Einsatz von fernen Journalen an, dann sind natürlich viele Faktoren wichtig. Zu nennen sind hier sicher Kosten, Performance, administrativer Aufwand. Aber das sind nur einige Punkte, die man bedenken muss bzw. die eine Rolle bei der Entscheidungsfindung spielen. Wichtig ist dabei, dass im Wildwuchs der Lizenzprogramme, die IBM gerade in den letzten Jahren in Sachen Hochverfügbarkeitslösungen ausgegeben hat, die eigentlichen Basisfunktionen rund um ferne Journale Bestandteile des Betriebssystems sind. Damit fallen keine zusätzlichen Lizenzgebühren an. Die Integration dieser fernen Journal-Funktionen ist sehr maschinennah. Das bedeutet, dass viele der Funktionen auf Ebene des Microcodes ausgeführt werden. Das macht sich letztlich insbesondere bei der Ausführungsgeschwindigkeit bemerkbar – und diese ist gerade bei Hochverfügbarkeitslösungen immens wichtig! Dass die Vorgänge des fernen Journals unterhalb der Ebene der Maschinenschnittstelle (MI – Machine Interface) stattfinden, bringt auch eine geringere Belastung der CPU mit sich – zumindest in Bezug auf deren Aufteilung in „normale“ System- und Anwenderjobs gesehen.

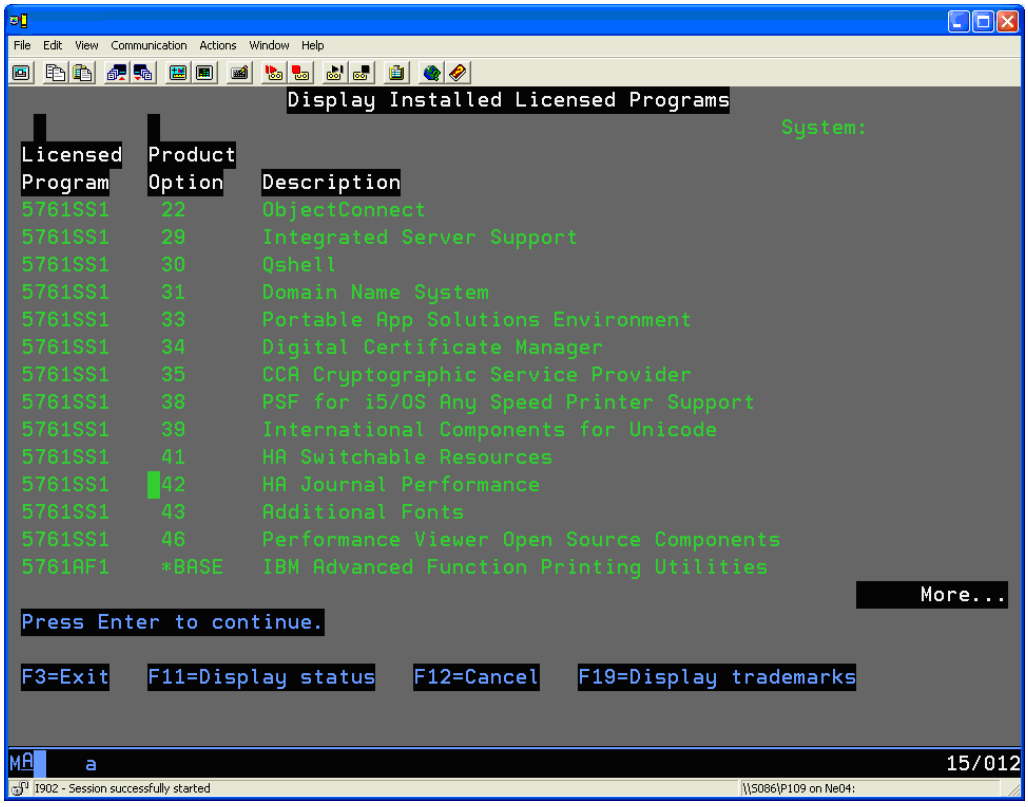
Das Schöne an den Remote Journals ist zudem, dass sie auf einem fernen System i oder auch auf einem iASP angelegt werden können. Mit dieser Eigenschaft sind die Möglichkeiten rund um zusätzliche Sicherheit oder Maschinenverfügbarkeit noch flexibler.

Aber all die Vorzüge und Eigenschaften sollen über eines nicht hinwegtäuschen: Ferne Journale sind keine fertige Lösung zur Implementierung einer Hochverfügbarkeit, wie sie beispielsweise von Zusatzprodukten geboten wird. Ferne Journale stellen lediglich die Basis für eine komplexe Hochverfügbar-

keitslösung dar. Letztere zeichnet sich z. B. durch Verwaltungskomponenten aus, welche die Journalaufzeichnung der relevanten Informationen ebenso überwacht wie nicht korrekte Funktionen, und die Filter zur Verwaltung bietet etc.

Ferne Journale sind heute häufig die Grundlage für Hochverfügbarkeitslösungen und können, Zeit und Geschick vorausgesetzt, auch mit selbsterstellten Programmen und Funktionen so verfeinert werden, dass das Ergebnis dann quasi eine Hochverfügbarkeitslösung darstellt. Nur ist der damit verbundene Aufwand hoch und stellt ein gewisses Risiko dar – denn letztlich geht es um existenzielle Prozesse und Daten.

Auf Basis der im Betriebssystem verfügbaren Funktionen bietet IBM zudem „Erweiterungen“ für den Einsatz der fernen Journale an. Ein Beispiel für solche Erweiterungen ist die Option 42 des 57xx-SS1 (HA Journal Performance).



Anzeige der installierten Lizenzprogramme

Mit dieser Funktion können zum einen Journaleinträge geblockt werden, was zu einer Reduzierung der zu übertragenden Daten führt. Zum anderen bietet diese Lizenzprogrammoption seit Version 7.1 Filterfunktionen, mit denen die zu übertragenden Daten eingeschränkt und für eine Wiederherstellung weniger wichtige Daten ausgeschlossen werden können.

2.8.1
Seite 8

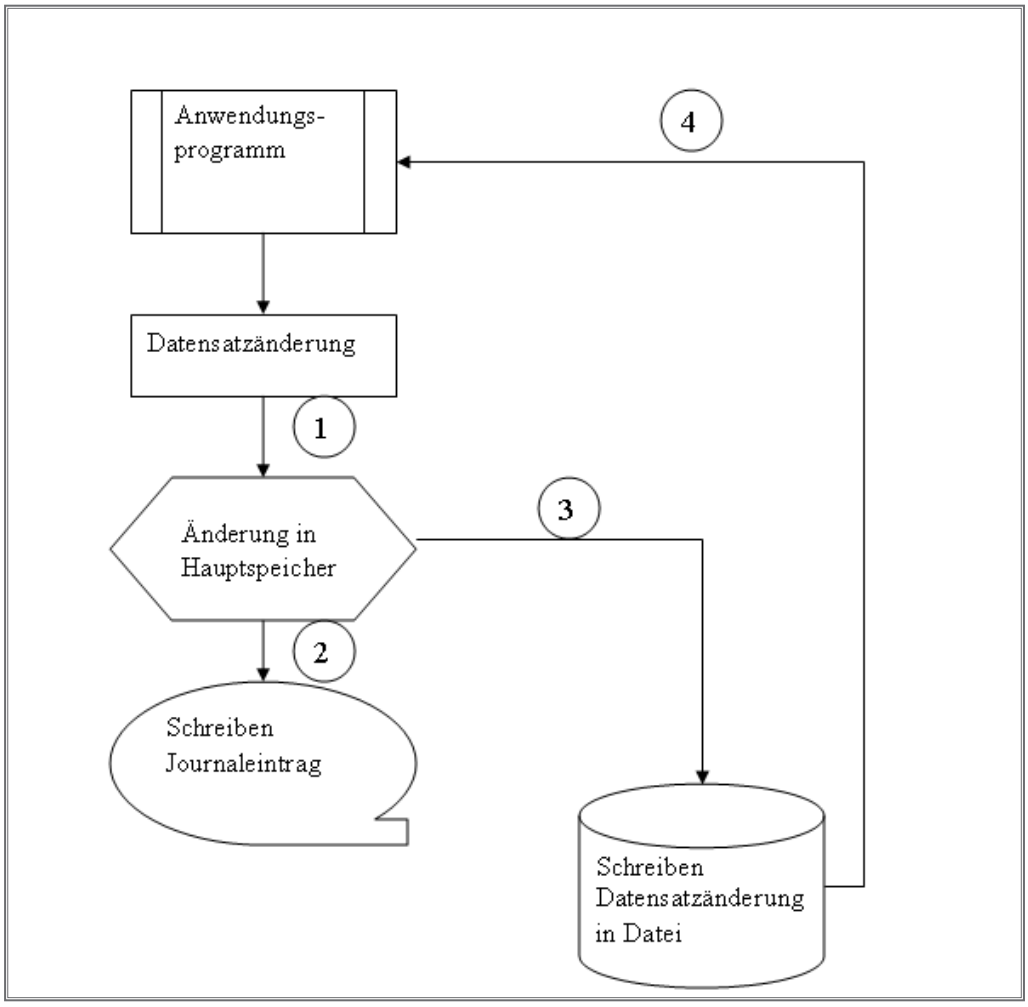
Übertragungsmodus

Ferne Journale können mit unterschiedlichen Übertragungsmodi eingerichtet und betrieben werden. Hier kommen die beiden Verfahren „synchrone“ und „asynchrone“ Übertragung zum Einsatz. Bevor wir uns diese Verfahren anschauen und die Unterschiede aufzeigen, sollten wir einen kurzen Blick auf den Prozess werfen, mit dem Änderungen an Dateiinhalten durchgeführt werden. Dabei ist es unerheblich, ob die Änderung ein Schreiben, Löschen oder Verändern eines Datensatzes darstellt.

Bei einem „normalen“ Prozess der Änderung des Inhalts einer Datei wird das folgende Verfahren angewendet: Der Datensatz wird in den Hauptspeicher geladen, dort geändert und dann erst in die Tabelle oder Datei geschrieben. Der letzte Teilschritt – das Schreiben der Änderungen auf die Platte – ist der langsamste des gesamten Prozesses.

Ähnlich verhält es sich beim zusätzlichen Schreiben von Journaleinträgen: Veränderungen an einer Datei bzw. an einem Datensatz werden zunächst am Image des Datensatzes im Hauptspeicher vorgenommen. Anschließend wird der Journaleintrag in den Journalempfänger geschrieben. Hier erfolgt das Schreiben direkt auf die Platte. Erst danach wird der eigentliche Datensatz verändert in die Ursprungsdatei geschrieben und die Steuerung wieder an das Programm übergeben.

2.8.1.1
Seite 2



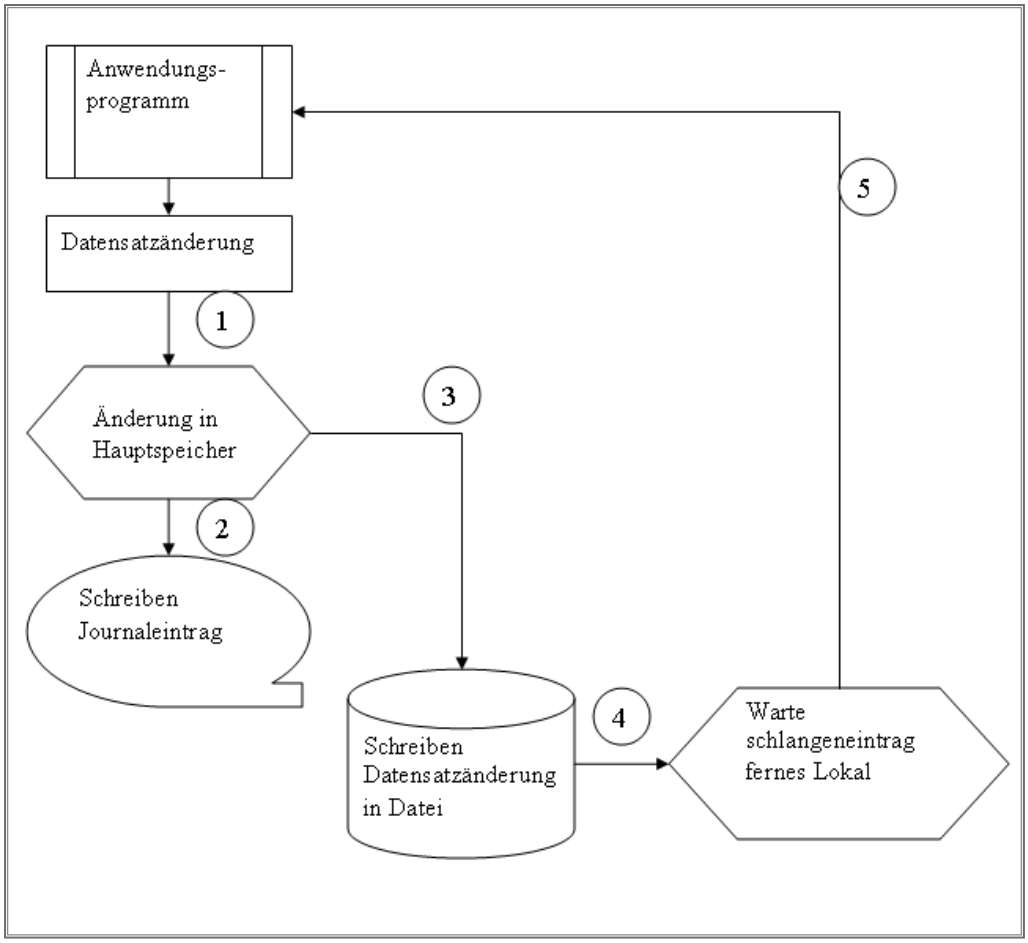
Übertragungsmodus

Es sind gleich zwei Teilprozesse, die sich hier in Bezug auf die Ausführungszeit negativ bemerkbar machen: Das Schreiben des Datensatzes in die Datei und das zusätzliche Schreiben des Journaleintrags. Natürlich ist ein einzelnes Update kaum oder gar nicht spürbar, werden hingegen Massendaten verarbeitet, kann dies durchaus zum Nadelöhr werden! Deshalb gilt auch hier natürlich, die entsprechende Sorgfalt bei der Einrichtung des lokalen Journals walten zu lassen, was insbesondere die Zuordnung zu ASP und damit zu Platten und Plattencontrollern betrifft. Die Empfehlung ist auch hier klar: Trennung der Schreibprozesse bei journalisierten Daten und Journalen auf unterschiedliche Benutzer-ASPs bzw. verschiedene Plattencontroller.

Beim Einsatz des fernen Journals wird der Schreibprozess nochmals komplizierter. Allerdings variieren hier die Komplexität und die damit verbundenen Auswirkungen mit der Wahl der Einsatzvariante des fernen Journals.

Asynchrone Übertragung

Ferne Journale werden meist im asynchronen Verarbeitungsmodus betrieben. Dabei erfolgt der Update-Prozess wie zuvor beschrieben auf dem Quellsystem, ergänzt um Eintragung der anstehenden Änderung in eine Warteschlange zwecks Übertragung an das ferne Journal. Erst nach dem Schreiben des Journaleintrags in die Warteschlange wird die Steuerung wieder an das ausführende Programm zurückgegeben. Diese Ergänzung ist der folgenden Skizze als Punkt 4 hinzugefügt.



Asynchrone Übertragung

Diese vereinfachte Darstellung des Ablaufs berücksichtigt zudem nicht das Aufzeichnen der Datensatzabbilder mit dem Verfahren *both. Damit würde die Anzahl der Schreibprozesse nochmals erhöht und würden auch die Skizzen zu unübersichtlich.

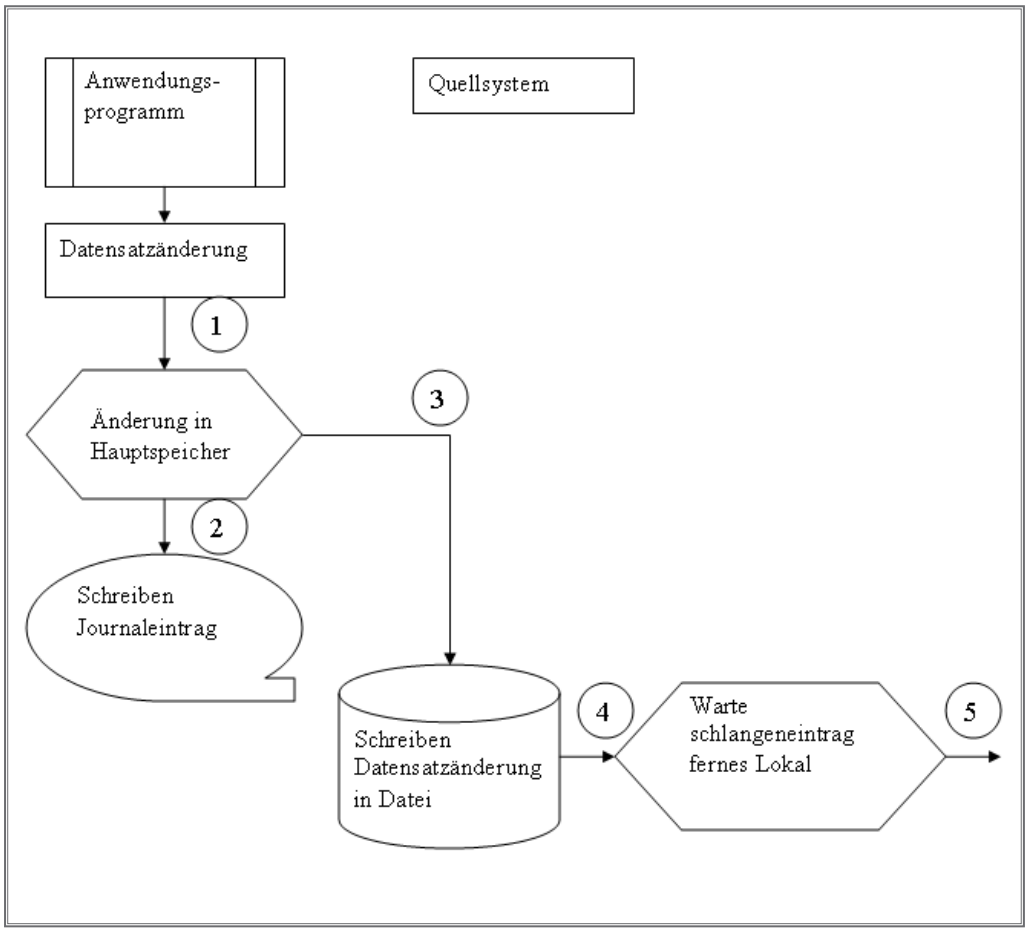
2.8.1.1.1

Seite 2

Synchrone Übertragung

Bei der sogenannten synchronen Übertragung erfolgen die ersten Verarbeitungsschritte gemäß den Verfahren, die bei der asynchronen Verarbeitung beschrieben wurden. Sie werden jedoch noch erweitert. Nach dem Schreiben des Journaleintrags in die Übertragungswarteschlange für das ferne Journal wird der Eintrag an das ferne System übertragen. Dort werden die Journaleinträge zunächst in den Hauptspeicher übertragen, dann auf die Platte des fernen Journalempfängers geschrieben und mittels einer Bestätigung an das Quellsystem quittiert.

Erst nachdem das Backupsystem die Bestätigungsmeldung des Quellsystems erhalten hat, wird der im Hauptspeicher befindliche Journaleintrag in den Journalempfänger des Zielsystems geschrieben und die Steuerung wieder zurück an den initiierenden Prozess (das Programm) übertragen. Die folgende Skizze veranschaulicht dieses Verfahren.



Synchrone Übertragung

2.8.1.1.2
Seite 2

Wie man unschwer erkennen kann, sind die Prozesse bei der asynchronen Verarbeitung komplexer und können damit natürlich auch mehr Zeit in Anspruch nehmen. Besonders dieser Umstand ist wohl häufig der Grund, sich für eine synchrone Übertragung zu entscheiden.

Sind die Anforderungen an das ferne Journal derart, dass die Einträge auf beiden Systemen synchron laufen müssen, dann ist die synchrone Verwendung des fernen Journals einzusetzen. Zwar können damit vereinzelt Verzögerungen bei der Verarbeitung der Daten auftreten, aber dieser Nachteil kann in Hochverfügbarkeitsumgebungen zu Gunsten der Sicherheit in Kauf genommen werden.

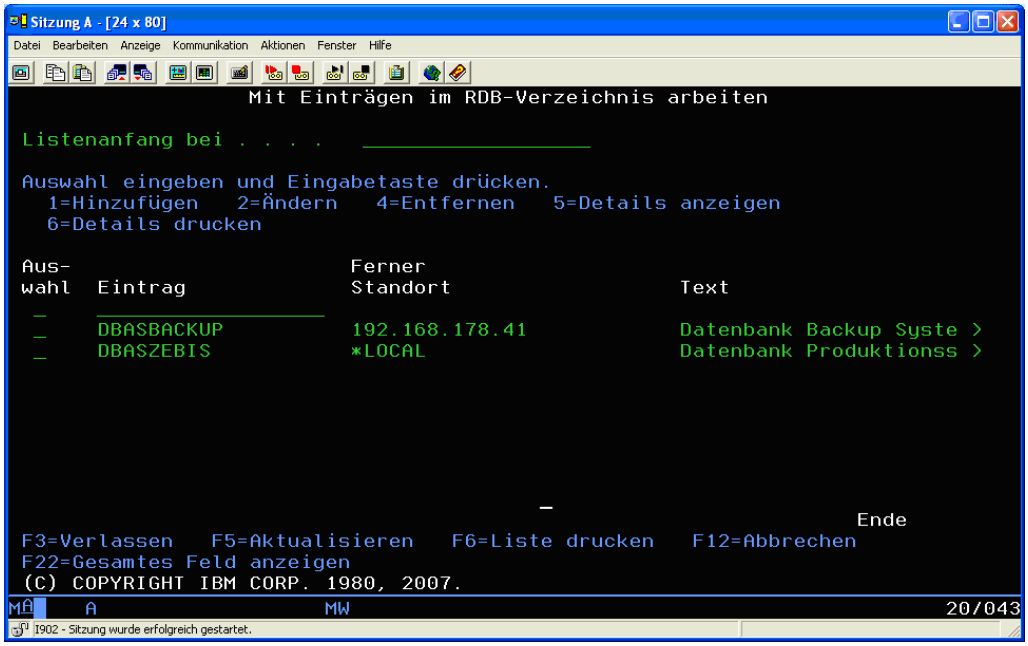
IBM hat seit Version 6.1 des Betriebssystems IBM i eine durchaus nützliche Erweiterung für die fernen Journale implementiert, mit der sichergestellt werden kann, dass die vom Quellsystem gesendeten Informationen auch vollständig auf dem Zielsystem angekommen sind. Das auf einem speziellen Algorhythmus basierende Checksum-Verfahren bietet insbesondere bei der Übertragung der Journalinformationen in Blockform Vorteile und stellt sicher, dass alle notwendigen Bytes die Leitung erfolgreich passiert haben. Es versteht sich dabei leider auch von selbst, dass dieses zusätzlich implementierte Verfahren Zeit kostet. Insbesondere Rechenzeit der CPU – und das auf beiden Systemen! Diese Tatsache sollte man bedenken, wenn es um den Einsatz der Checksum-Funktion des fernen Journals auf kleineren bzw. bereits stark ausgelasteten Maschinen geht.

Verbindungen für ferne Journale

Der Datenaustausch zwischen zwei Systemen kann in der System i-Welt auf unterschiedliche Arten und Weisen erfolgen. Meist ist TCP/IP die Grundlage für die Übertragung der Inhalte eines lokalen Journals in ein fernes Journal. Aber auch SNA, OptiConnect und ab Version IBM i 6.1 DataPort Services sind als Kommunikationsgrundlage möglich.

Schauen wir uns die Einrichtung einmal am Beispiel einer TCP/IP-Verbindung an. Natürlich ist in Sachen Verbindung die Geschwindigkeit wichtig, mit der die Informationen von einem auf ein anderes System übertragen werden. Ideal wäre eine für den Journalverkehr dediziert zur Verfügung stehende TCP/IP-Verbindung. Letztlich lohnt sich eine Auslastungs- bzw. Belastungsanalyse der bestehenden Verbindungen.

Die Verbindung zwischen zwei Systemen basiert auf dem Austausch von Informationen auf Datenbankebene. Dazu müssen in beiden Systemen die Datenbanknamen der jeweiligen Gegenseite eingetragen werden. Diese Einträge werden einmalig mit dem Befehl WRKRDBDIRE vorgenommen:

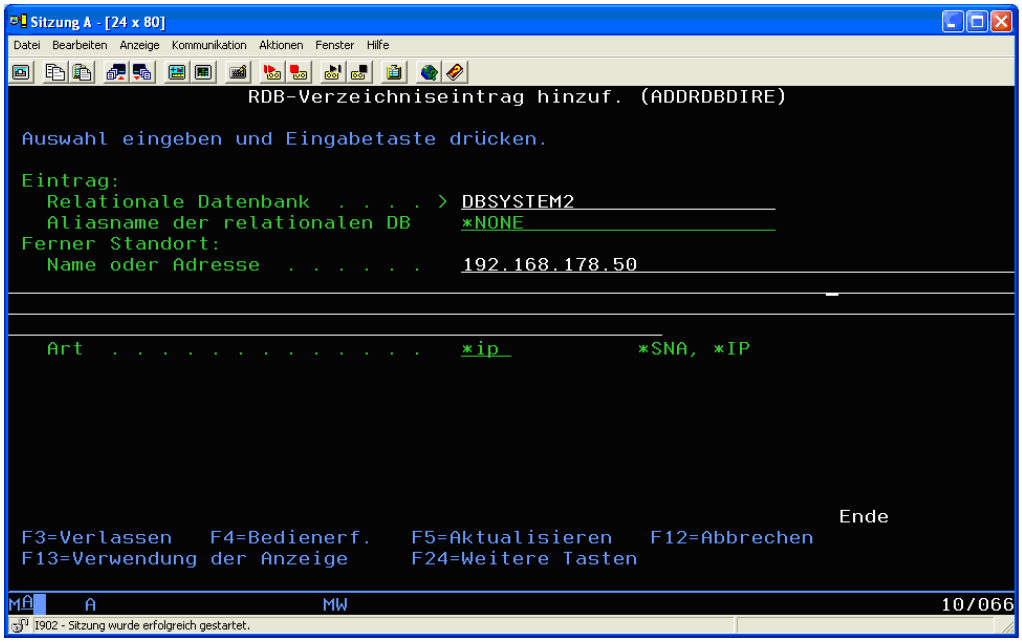


WRKRDBDIRE

Wie Sie sehen, besteht ein solcher Datenbankeintrag aus dem Namen der Datenbank und der Angabe, ob es sich um eine lokale oder um eine ferne Datenbank handelt. Auf jeden Fall muss auf jedem der Systeme auch der lokale Datenbankname enthalten sein. Dieser wird durch die Angabe *LOCALE identifiziert. Fehlt ein solcher Eintrag, dann fügen Sie ihn mit der Auswahl 1 oder dem Befehl ADDRDBDIRE hinzu.

2.8.1.2
Seite 2

Für das ferne System kann wahlweise dessen IP-Adresse oder dessen DNS-Name als Angabe im Parameter „ferne Lokation“ verwendet werden. Außerdem muss der Name der fernen Datenbank angegeben werden, was auch für die Kommunikation wichtig ist – bei auf TCP/IP basierenden Verbindungen muss dort *IP eingetragen werden.



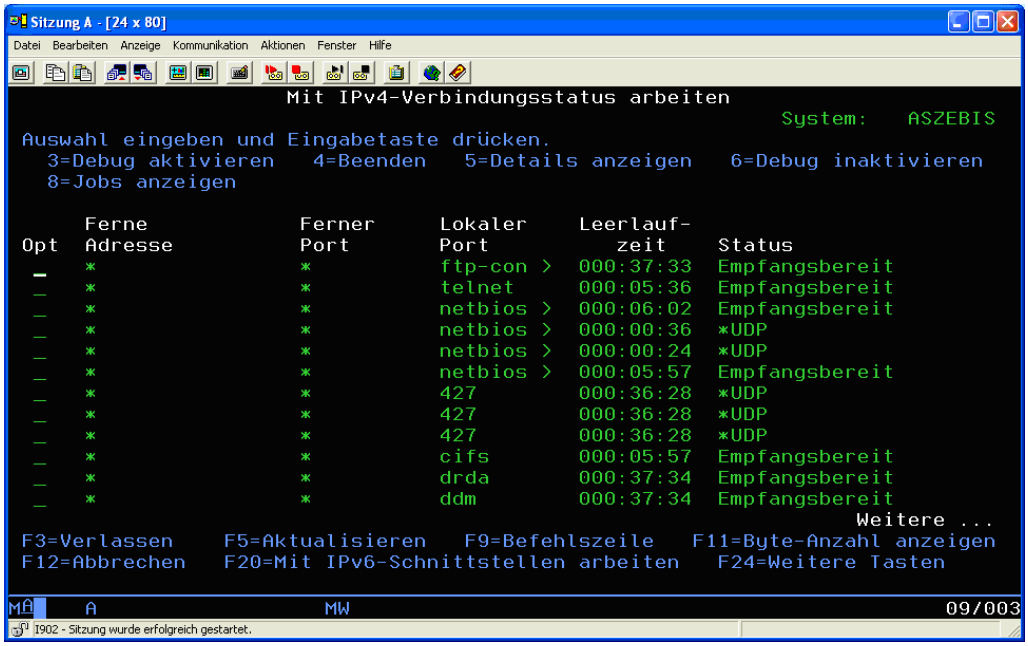
ADDRDBDIRE

Anmerkung

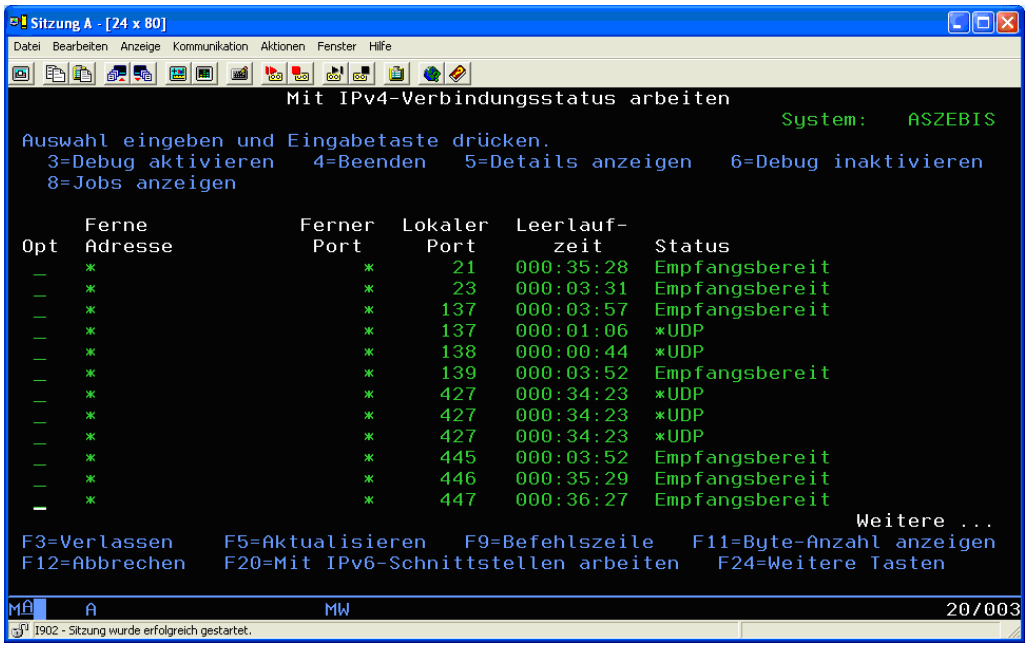
Es kann durchaus vorkommen, dass die Datenbanknamen der beiden Maschinen identisch sind. Grundsätzlich ist das für den Einsatz des fernen Journals zwar kein Problem, stellt aber unter Umständen eine Fehlerquelle für Anwendungen dar, die maßgeblich mit dem Datenbanknamen arbeiten und diesen als eindeutig in einer Umgebung erwarten. Andererseits erleichtert die Namensgleichheit aber auch den Betriebswechsel zwischen Quell- und Backupsystem, da Anwendungen nicht auf den neuen Datenbanknamen des anderen Systems angepasst werden müssen.

Der Datenaustausch zwischen lokalen und fernen Journalen basiert auf den mit dem DDM-Serverdienst bereitgestellten Funktionen. Deshalb sollte auch vor der eigentlichen Einrichtung sichergestellt sein, dass der DDM-Serverdienst gestartet ist. Das können wir mit Hilfe des Befehls NETSTAT OPTION(*CNN) überprüfen. Suchen Sie dort nach dem Port 447. Steht er auf „Empfangsbereit“, so ist die DDM-Funktion aktiviert.

Sollten die Ports nicht angezeigt werden, dann drücken Sie die F14-Taste.



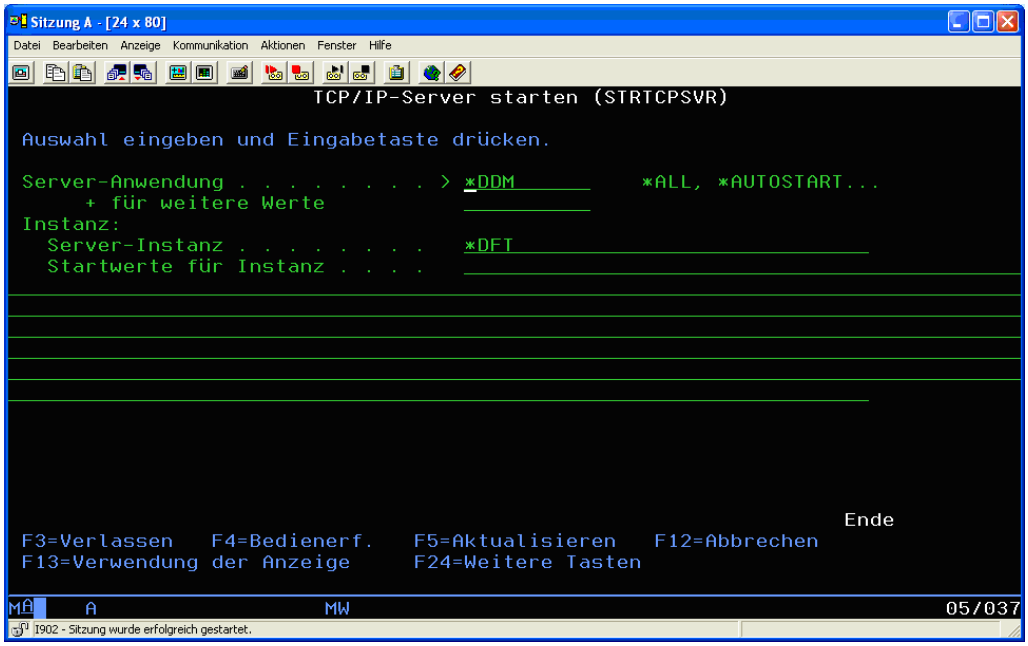
Netstat mit Serverdiensten



*NETSTAT(*CNN) mit Portanzeige*

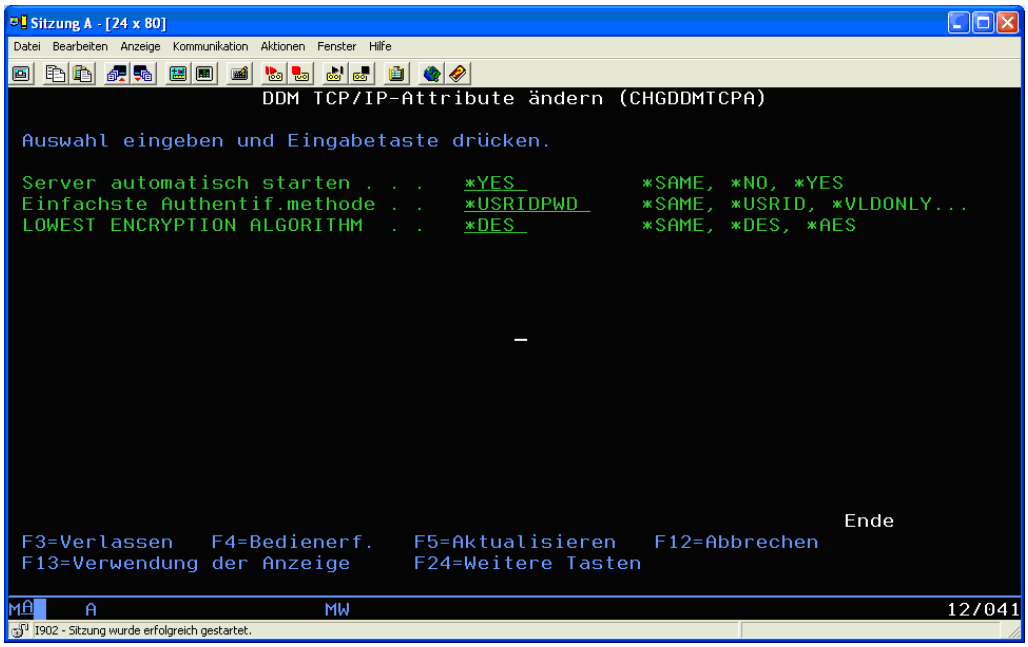
Sollte der Dienst nicht gestartet worden sein, so starten Sie ihn mit dem Befehl STRTCPSVR SERVER(*DDM). Achten Sie darauf, dass das Benutzerprofil mit STRTCPSVR ausgeführt wird und über ausreichende Berechtigungen verfügt. Ideal ist die Verwendung von SECOFR-Berechtigungen. Mindestens sollten Sie jedoch über die Sonderberechtigung *IOSYSCFG verfügen.

2.8.1.2
Seite 4



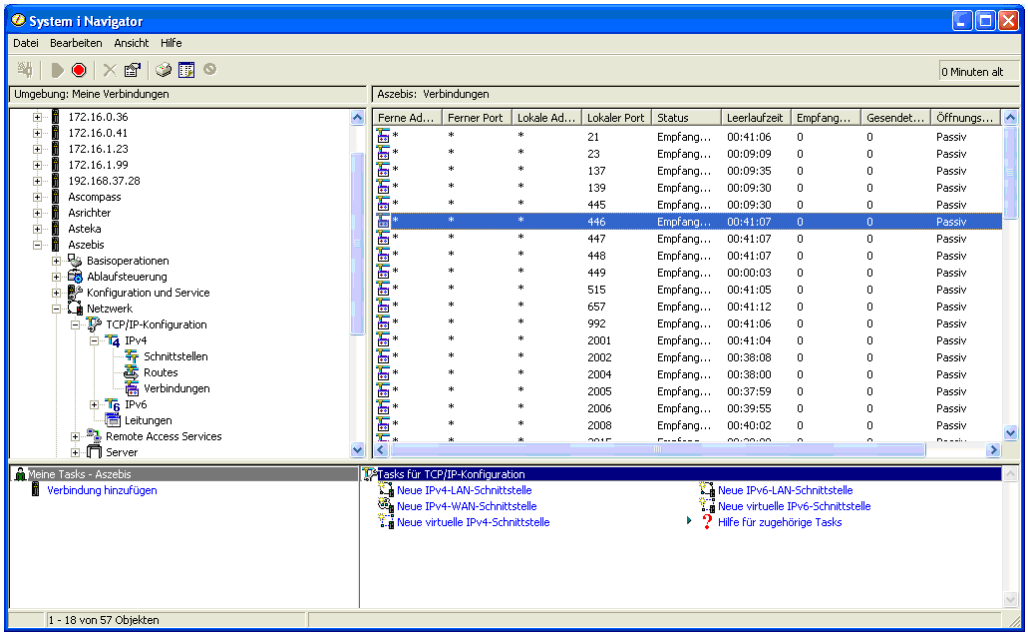
STRTCPSVR

Die DDM-Kommunikation kann mit verschiedenen Einstellungen definiert werden. Beispielsweise kann angegeben sein, dass für einen Verbindungsaufbau sowohl das Benutzerprofil als auch das Kennwort benötigt werden. Alternativ dazu kann man aber auch angeben, dass der Verbindungsaufbau lediglich auf Angabe des Benutzerprofils hin durchgeführt werden kann. Diese Einstellungen können wir mit dem Befehl „DDM TCP/IP-Attribute ändern“ (CHGDDMTCPA) einsehen und auch verwalten.



CHGDDMTCPA

Gerade bei einer Datenübertragung zwischen verschiedenen Lokationen ist eine geschützte Verbindung ein Muss. Firewalls verhindern auch in Umgebungen mit fernen Journalen den unberechtigten Zugriff Dritter. Sollten Sie die beiden Maschinen durch Firewalls geschützt haben, dann muss sichergestellt sein, dass die Firewalls so konfiguriert sind, dass der Datenverkehr zwischen Quell- und Zielsystem über den Standardport für das ferne Journal (oder genauer: für DDM) erreichbar ist. Das ist der Port 446.



Portanzeige System i Navigator

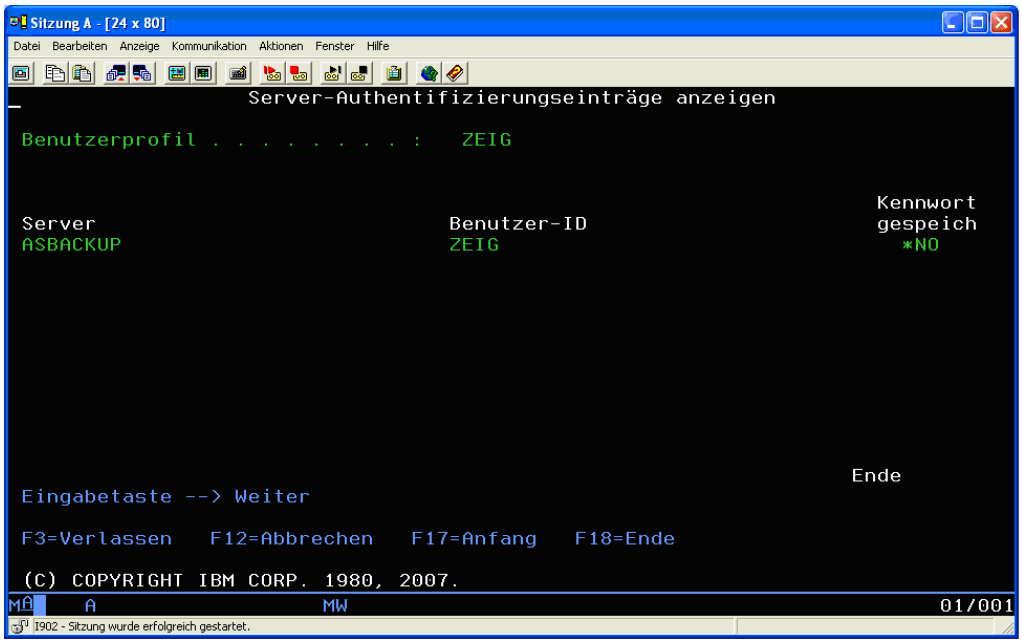
Der Port 446 wird allerdings nur für den Handshake benötigt. Ist der erfolgreich, wird die Übertragung der Informationen mittels einer geschützten Verbindung über den Port 3777 durchgeführt. Natürlich muss auch dieser Port in der Umgebung verfügbar und für die Systeme zugänglich sein.

Bei DDM handelt es sich um einen TCP/IP-Serverdienst. Wie für andere Dienste, können wir auch für DDM angeben, dass der Dienst im Zuge eines IPL mitgestartet werden soll. Damit ist dann in jedem Fall sichergestellt, dass die für das ferne Journal notwendige Kommunikationsbasis aktiv ist. Sollten Sie sicherheitsrelevante Bedenken oder andere Gründe haben, die gegen das automatische Starten von DDM sprechen, dann können Sie diesen Bereich natürlich auch manuell oder über ein CL starten.

Damit die beiden Maschinen und die Journale miteinander verbunden werden können, bedarf es einer Authentifizierung der beiden Systeme. Diese kann wahlweise über ein Benutzerprofil oder das Senden eines Kennworts durchgeführt werden. Damit die Verbindungsprüfung zwischen den Systemen durchgeführt werden kann, legen wir zunächst die Zugangsprüfungen fest. Dabei werden die Datenbanken und die zugehörigen Benutzerprofile

bzw. Kennworte einmalig definiert. Die Serverauthentifizierung bedient sich der Einstellungen, die für einen Benutzer existieren bzw. vorgenommen werden.

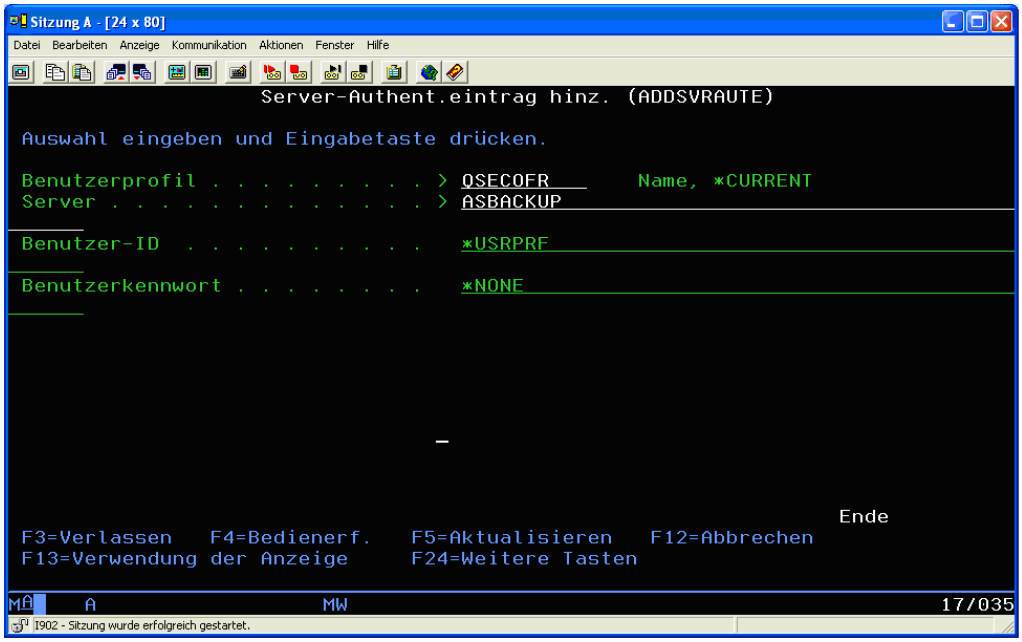
Vor der Definition empfiehlt es sich also zu prüfen, ob ein solcher Eintrag bereits vorhanden ist. Dazu verwenden wir den Befehl DSPSVRAUTE. Wird in der Ergebnisanzeige für diesen Befehl ein Eintrag angezeigt, dann prüfen Sie, ob er für die Einrichtung und den Betrieb des fernen Journals verwendet werden kann.



DSPSVRAUTE

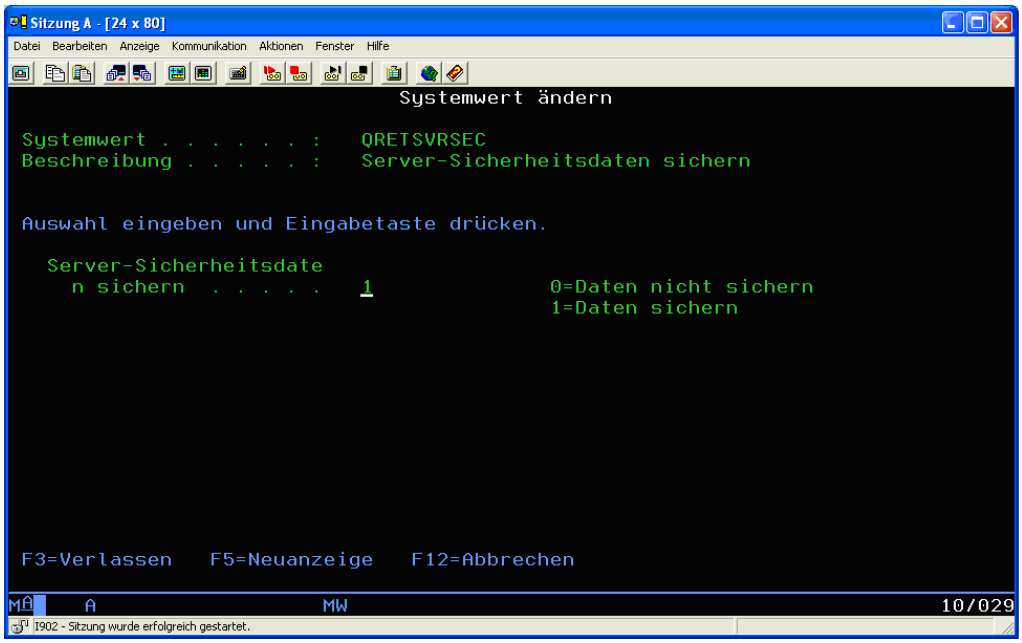
Ein neuer Eintrag kann mit Hilfe der Auswahl 1 oder des Befehls ADDSVRAUTE angelegt werden. Dabei muss als Benutzerprofil das Profil des Quellsystems angegeben werden.

Achten Sie bei der Angabe des fernen Systems darauf, dass Sie den Namen in Großbuchstaben angeben. Anderenfalls kann es beim Starten und beim Betrieb des fernen Journals zu Problemen kommen.



ADDSVRAUTE

Damit das Kennwort für die Verwendung gespeichert wird, muss der Systemwert QRETSVRSEC auf „1“ gestellt sein. Den Systemwert können Sie mittels des Befehls WRKSYSVAL QRETSVRSEC prüfen und bei Bedarf ändern.



WRKUSRPRF QRETSVRSEC

2.8.1.2
Seite 8

Damit die beiden Systeme mit dem Benutzerprofil und dem Kennwort problemlos arbeiten können, müssen sie auf beiden Systemen synchron gehalten werden. Das bedeutet, dass Änderungen am Benutzerprofil und/oder am Kennwort unbedingt auf beiden Systemen durchgeführt werden müssen. Diese Änderungen beziehen sich dann auch auf den Bereich der Server-Authentifizierung und sind mit dem Befehl CHGSVRAUTE zu pflegen!

Typen des fernen Journals

Ferne Journale können auf dem IBM i Server in zwei verschiedenen Versionen oder Typen existieren: *TYPE1 und *TYPE2. Der Unterschied der beiden Typen liegt im Wesentlichen in den Sicherungs- und Wiederherstellungsverfahren, den Weiterleitungsfunktionen und den Zuordnungsmöglichkeiten.

Schauen wir uns diese Unterschiede der beiden Journaltypen einmal genauer an.

Type1

Während *TYPE1 umfangreiche Wiederherstellungsfunktionen beinhaltet, sind die Varianten der Namensvergaben eingeschränkt.

Mit *TYPE1 sind die Informationen im Journalempfänger detaillierter. Sie enthalten z. B. sowohl den Namen des Quell- als auch des Zieljournals. Das gilt auch für die Namen der betreffenden Journalempfänger. Der Vorteil: Bei einem Ausfall können wahlweise die Journalinformationen vom Quell- oder vom Zielsystem für einen Wiederaufbau verwendet werden.

Der Name für das ferne Journal muss dem Namen des lokalen Journals entsprechen. Die Bibliothek, in der sich das ferne Journal befindet, kann von der Bibliothek des lokalen Journals abweichen. Dasselbe gilt auch für die Bibliothek des fernen Journalempfängers.

Type2

Bei der Verwendung von *TYPE2 werden nur die Informationen des Quelljournals und des zugehörigen Quelljournalempfängers gespeichert. Damit verbunden ist aber auch die Einschränkung, dass sie vom Zielsystem zurück auf das Quellsystem gespeichert werden können – nicht umgekehrt. Aber: Mit dieser Variante ist eine geänderte Zuordnung zu den aufgezeichneten Bibliotheken möglich.

Die Angabe von abweichenden Bibliotheken ist hier ebenso möglich – und zwar in detaillierter Form.

Somit gilt es bei der Entscheidung für den Typ des fernen Journals abzuwägen, welche der beiden Varianten für die gewünschte Journalfunktion am besten geeignet ist.

2.8.1.3
Seite 2

Einrichtung eines fernen Journals

Wie in vielen anderen Bereichen auch, so muss vor der eigentlichen Einrichtung eines fernen Journals zunächst eine möglichst genaue Planung durchgeführt werden, mit der sowohl die zu journalisierenden Bereiche analysiert und definiert werden müssen als auch die Gesamtkonfiguration.

Die Grobschritte und Hauptaufgaben der Einrichtung eines fernen Journals sind die nachfolgenden Punkte:

- Auswahl des geeigneten Konzepts für den Einsatz der fernen Journale
- Planung der lokalen Journale
  - Gegebenenfalls Verwendung bereits bestehender Journale
  - Einrichtung neuer Journale
  - Sicherstellung der Aufzeichnung neuer Objekte in Journalen
- Planung des fernen Journals
- Einrichtung des fernen Journals

Details zur Einrichtung eines lokalen Journals finden Sie an anderer Stelle. Dort sind ausführlich Planung und auch Einrichtung der lokalen Journale beschrieben. An dieser Stelle möchte ich mich bewusst auf die Anleitung zur Einrichtung von fernen Journalen beschränken.

Schauen wir uns die wichtigsten Planungsschritte für die Einrichtung eines fernen Journals an.

Natürlich gilt es festzulegen, welche Informationen und Daten für die Verwendung mit fernen Journalen in Frage kommen bzw. benötigt werden. Sicher kann es auch Daten geben, die nicht unbedingt auf zwei Maschinen vorgehalten werden müssen. Einige Informationen sind gegebenenfalls unwichtig und müssen deshalb nicht auf zwei Maschinen existieren, da sie die Lauffähigkeit der Anwendungen nicht gefährden.

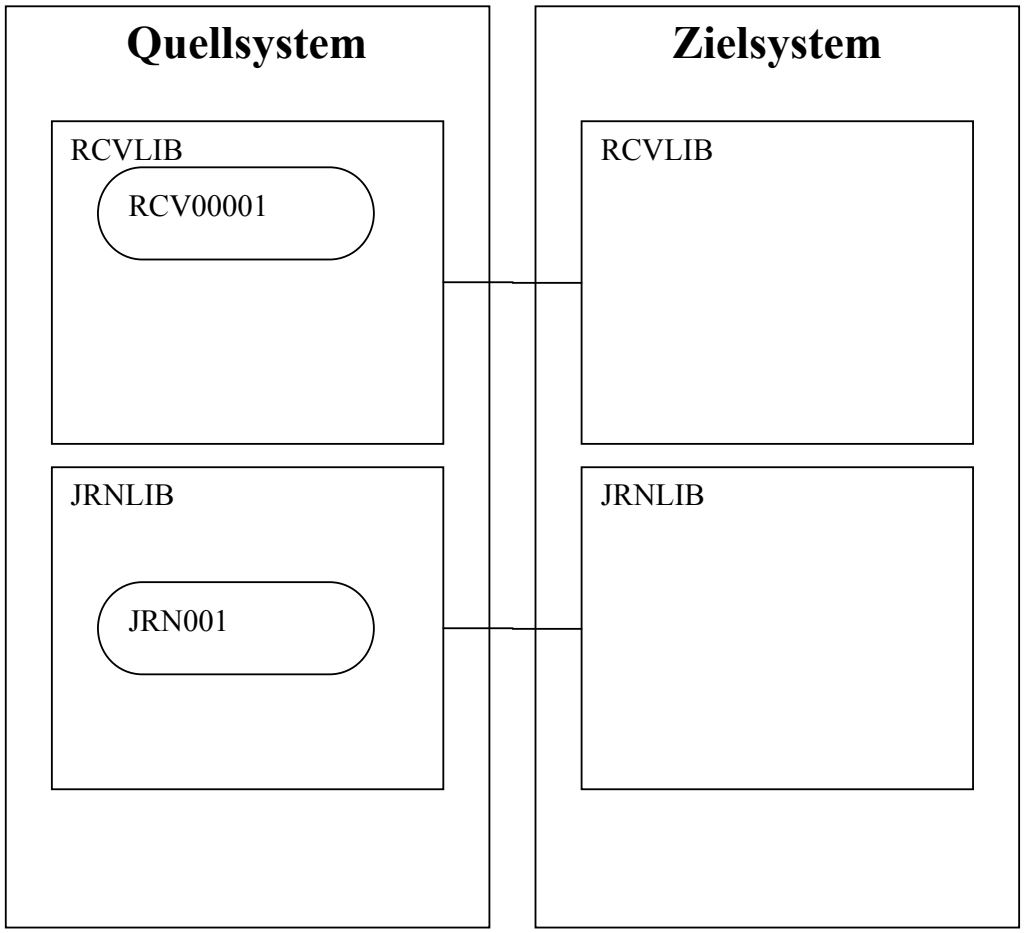
Die Einrichtung von fernen Journalen bringt eine Zusatzbelastung mit sich. Diese betrifft die Schreibprozesse auf den Platten (denn letztlich werden Änderungen quasi doppelt geschrieben – einmal in der Basisdatei bzw. im Objekt und einmal im Journalempfänger). Zum anderen ist aber auch zusätzlicher Hauptspeicher erforderlich, um die Tätigkeiten rund um das ferne Journal, den Datenverkehr und andere Bereiche ausreichend abdecken zu können.

Der Bedarf an zusätzlichem Speicher besteht auf allen Maschinen, die in die Umgebung eines fernen Journals eingebunden sind. Beispielsweise muss auch für Sondersituationen ausreichend Speicher vorhanden sein, etwa wenn die Zielmaschine die Informationen nicht annehmen kann und die Journalempfänger des Quellsystems größer werden, so dass sie die Daten zwischenpuffern müssen.

2.8.1.4
Seite 2

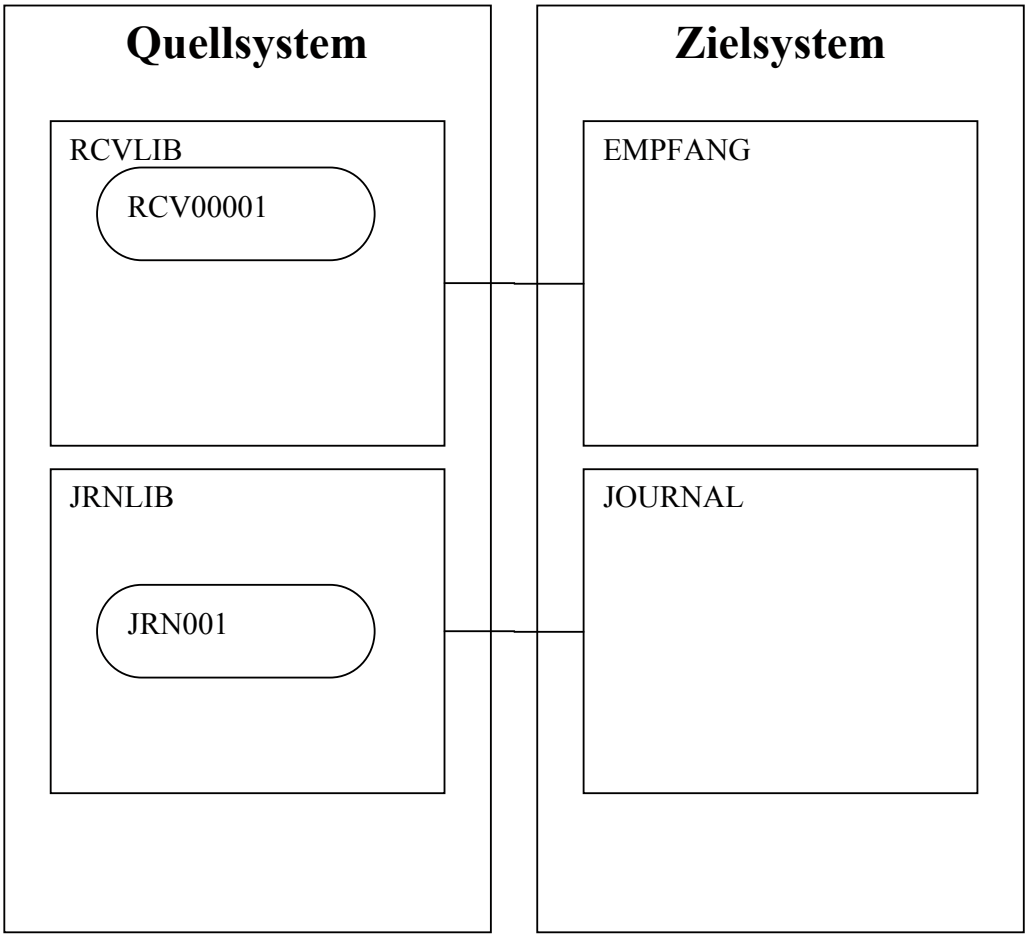
Wichtig ist an dieser Stelle noch der Hinweis, dass man beim Einsatz ferner Journale die Journale und Journalempfänger einer Bibliothek des Quellsystems auf dem Zielsystem in eine abweichende Bibliothek übertragen kann. Es muss lediglich sichergestellt sein, dass die benötigten Bibliotheken und Objekte auf dem Zielsystem existieren.

Die folgende Abbildung zeigt schematisch den Standardfall einer Datenübertragung im Zusammenhang mit einem fernen Journal.



Schematische Darstellung der Datenübertragung bei fernen Journalen (1)

Alternativ zur Standardzuordnung kann in einer Umgebung auch definiert werden, dass die Inhalte einer Bibliothek des Quellsystems auf dem Zielsystem in eine abweichende Bibliothek eingestellt werden sollen. Auch dieser Fall ist nachfolgend schematisch dargestellt:



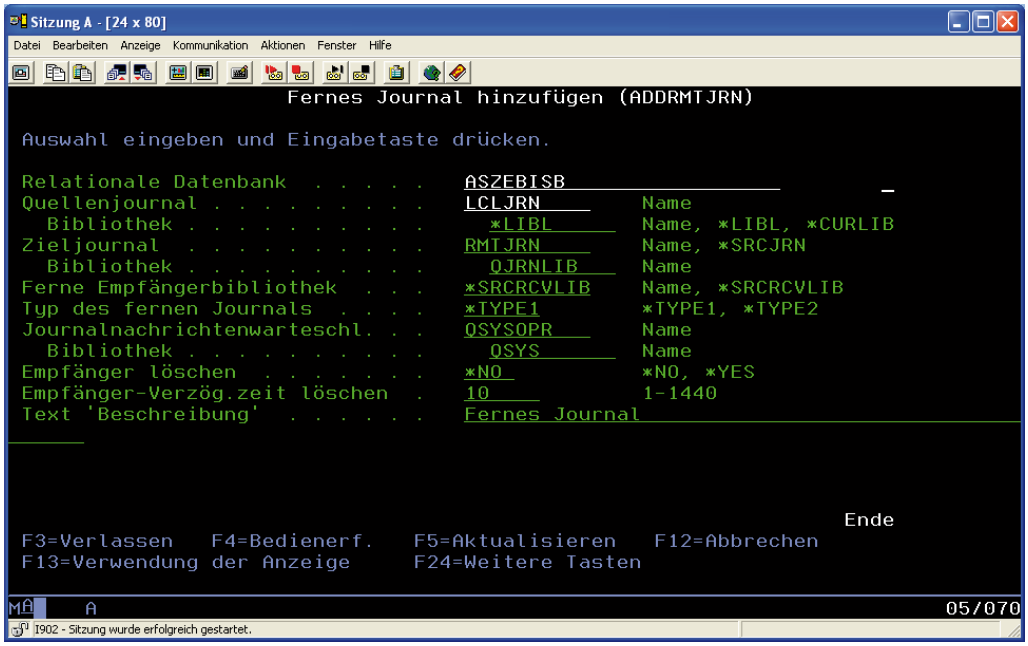
Schematische Darstellung der Datenübertragung bei fernen Journalen (2)

Wird ein fernes Journal bei der Übertragung in eine andere Bibliothek eingestellt, bezeichnet man das als Bibliotheksumleitung oder „library redirection". Dieses Verfahren wirkt sich auf weitere Prozesse aus. Deshalb sollte im ersten Planungsstadium für ein fernes Journal genau abgewogen werden, welche Informationen welcher Bereiche zu übertragen sind, und ob das Ziel dieselbe Bibliothek sein muss oder von ihr abweichen kann.

Nachdem die Datenbanknamen für die beiden Systeme eingetragen, die Kommunikation zwischen beiden Systemen eingerichtet und das lokale Journal auf dem Quellsystem definiert worden sind, kann auf dem Quellsystem mit der Einrichtung des fernen Journals begonnen werden.

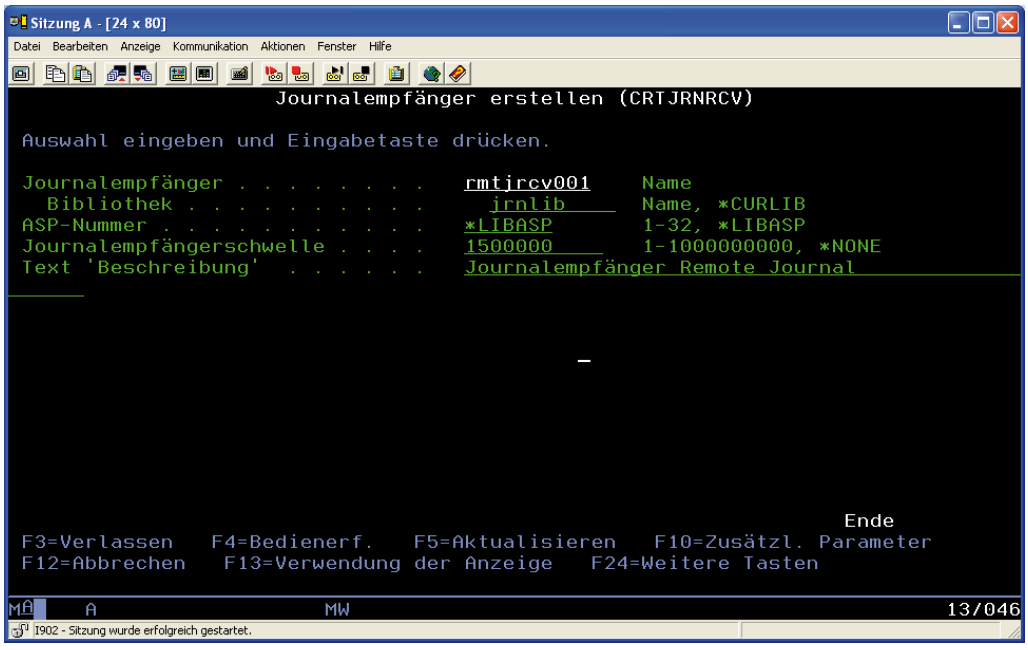
Der Befehl ADDRMTJRN dient der Definition eines neuen fernen Journals. ADDRMTJRN benötigt die folgenden Parameter:
- Name der Datenbank des fernen Systems
- Name des lokalen Journals
- Alle Journal- und Weiterleitungsinformationen
- Typ des fernen Journals (*TYPE1 oder *TYPE2)

2.8.1.4
Seite 4

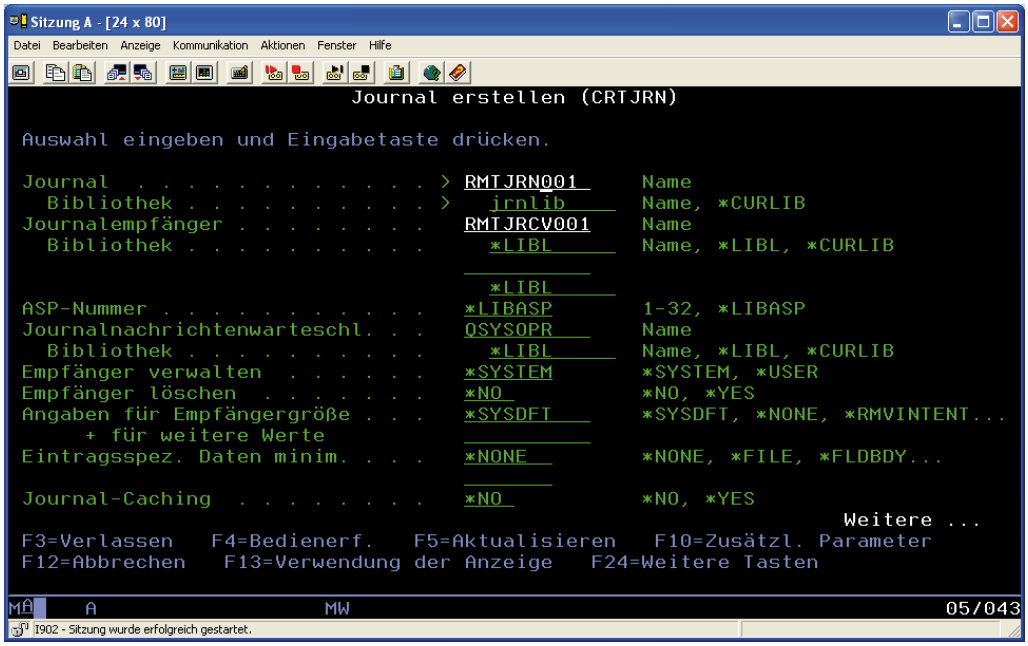


ADDRMTJRN

Sollten die notwendigen Journalkomponenten noch nicht existieren, dann müssen sie vor der Ausführung des Befehls ADDRMTJRN angelegt werden. Dazu bedienen wir uns der bereits bekannten Befehle CRTJRNRCV und CRTJRN.



CRTJRNRCV

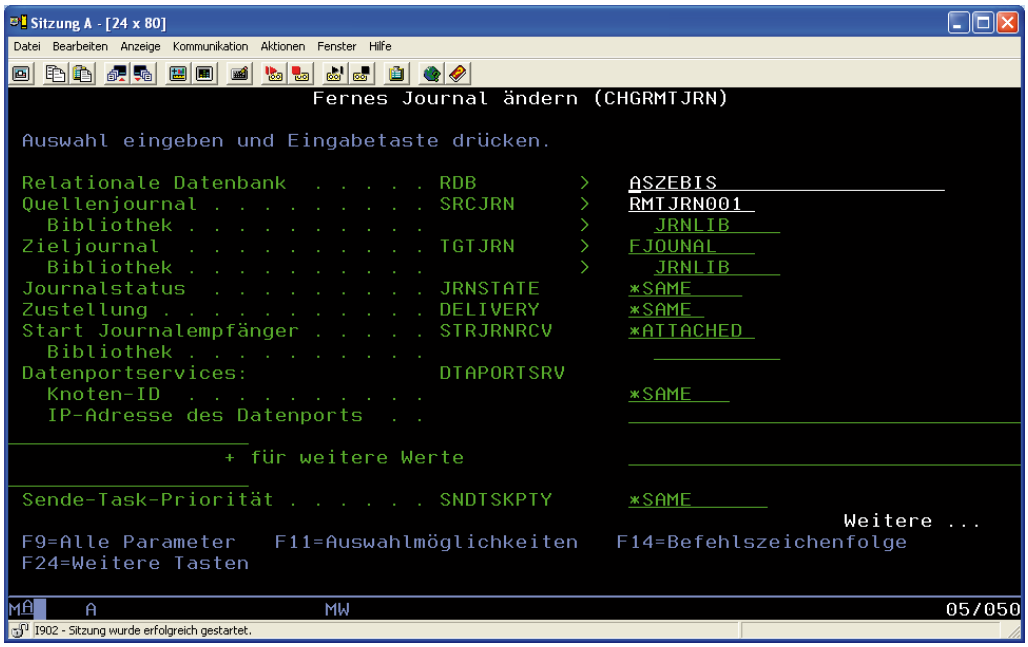

Sitzung A - [24 x 80]
Datei Bearbeiten Anzeige Kommunikation Aktionen Fenster Hilfe
Journal erstellen (CRTJRN)
Auswahl eingeben und Eingabetaste drücken.
Journal . . . . . > RMTJRN001 Name
Bibliothek . . . . . > jrnlib Name, *CURLIB
Journalempfänger . . . . . RMTJRCV001 Name
Bibliothek . . . . . *LIBL Name, *LIBL, *CURLIB
*LIBL
ASP-Nummer . . . . . *LIBASP 1-32, *LIBASP
Journalnachrichtenwarteschl. . . QSYSOPR Name
Bibliothek . . . . . *LIBL Name, *LIBL, *CURLIB
Empfänger verwalten . . . . . *SYSTEM *SYSTEM, *USER
Empfänger löschen . . . . . *NO *NO, *YES
Angaben für Empfängergröße . . . *SYSDFT *SYSDFT, *NONE, *RMVINTENT...
+ für weitere Werte
Eintragsspez. Daten minim. . . . *NONE *NONE, *FILE, *FLDBDY...
Journal-Caching . . . . . *NO *NO, *YES
Weitere ...
F3=Verlassen F4=Bedienerf. F5=Aktualisieren F10=Zusätzl. Parameter
F12=Abbrechen F13=Verwendung der Anzeige F24=Weitere Tasten
MA A MW 05/043
I902 - Sitzung wurde erfolgreich gestartet.

CRTJRN

2.8.1.4
Seite 6

Starten des fernen Journals

Nach der erfolgreichen Einrichtung des fernen Journals sollten die beiden Systeme miteinander synchronisiert werden. Dazu müssen die Einträge des Journals auf dem Quellsystem auf das zugeordnete Journal des Zielsystems übertragen werden. Der Befehl „Fernes Journal ändern“ CHGRMTJRN bietet die notwendigen Funktionen für das Starten der Übertragung zwischen den beiden verbundenen Maschinen.



CHGRMTJRN

Der Befehl CHGRMTJRN verwendet als Standardeinstellungen die asynchrone Übertragung. Ist davon abweichend eine synchrone Übertragung gewünscht, so muss dies bei der Definition angegeben werden.

ACHTUNG!

Wenn Sie den Befehl CHGRMTJRN interaktiv aufrufen, dann kann für den Fall, dass eine Vielzahl an Journaleinträgen zu übertragen ist, die Ausführungszeit extrem lange dauern. Das bedeutet, dass die Sitzung gesperrt ist, solange die Übertragung andauert. Deshalb sollte man genau abwägen, ob CHGRMTJRN interaktiv oder in einem CL – eingebunden als Batch-Aufruf – ausgeführt wird.

Für das Starten des fernen Journals stehen uns folgende Parameter zur Verfügung:

- Datenbankname (RDB)
  - Hier ist der Name der fernen Datenbank anzugeben. Dieser muss in der Auflistung der Datenbanken (ADDRDBDIRE) zuvor eingepflegt worden sein.

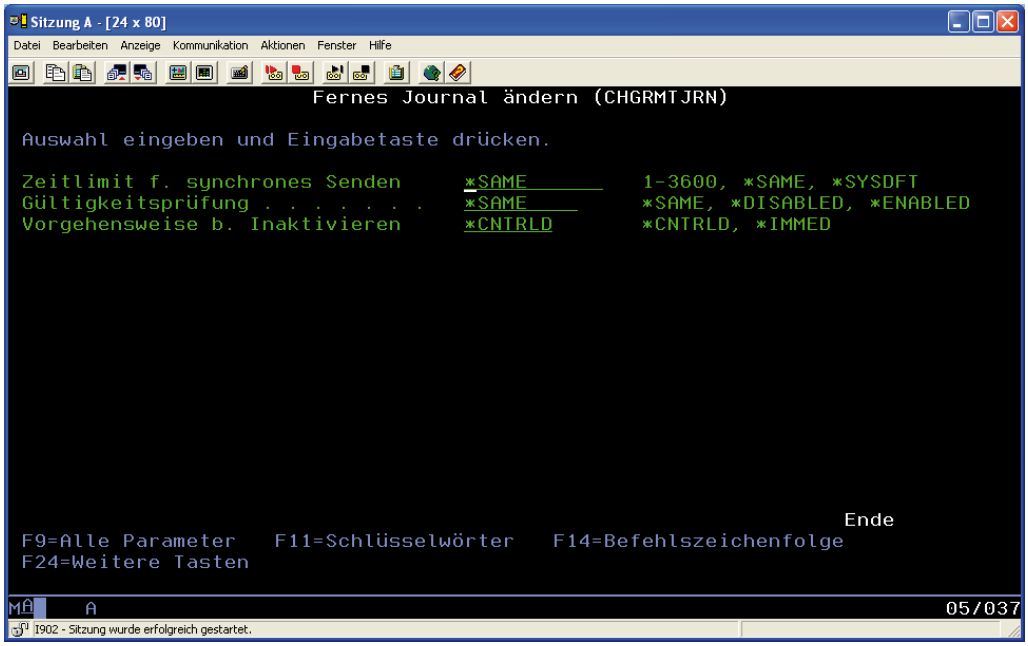
2.8.1.5
Seite 2

- Quelljournal (SRCJRN)
  - Geben Sie hier den qualifizierten Namen des Quelljournals an. Dieses Journal muss zuvor angelegt worden sein.
- Zieljournal (TGTJRN)
  - Geben Sie hier den qualifizierten Namen des Zieljournals an. Dieses Journal muss auf dem Zielsystem existieren.
- Journalstatus (JRNSTATE)
  - Gibt den Status an, der mit CHGRMTJRN erreicht werden soll. Für das Aktivieren des fernen Journals geben wir hier den Wert *ACTIVE ein.
- Zustellung (Delivery)
  - Mit diesem Parameter legen Sie fest, ob das ferne Journal synchron oder asynchron eingesetzt werden soll. Für eine synchrone Übertragung verwenden wir den Wert *SYNC, für eine asynchrone Übertragung *ASYNC.
- Journalempfänger starten (STRJRNRCV)
  - Gibt den Startpunkt für das ferne Journal an.

Wenn keine anderen Vorgaben gemacht werden, dann beginnt die Replikation mit dem aktuell angehängten Journalempfänger auf dem Quellsystem. Beim Erststart ist dies jedoch wenig hilfreich. In einem solchen Fall muss genau geplant werden, wie das erste Aufsetzen der beiden Systeme abläuft. In der Regel geschieht das mit einer Datensicherung, die auf das Zielsystem gespielt wird. Auf Basis dieser Datensicherung kommen anschließend die erstellten Journalempfänger des Quellsystems für die Übertragung zum Einsatz.

Deshalb kann man mit dem Befehl CHGRMTJRN auch den Start-Journalempfänger angeben, mit dem die Informationen des Quellsystems auf das Zielsystem übertragen werden sollen.

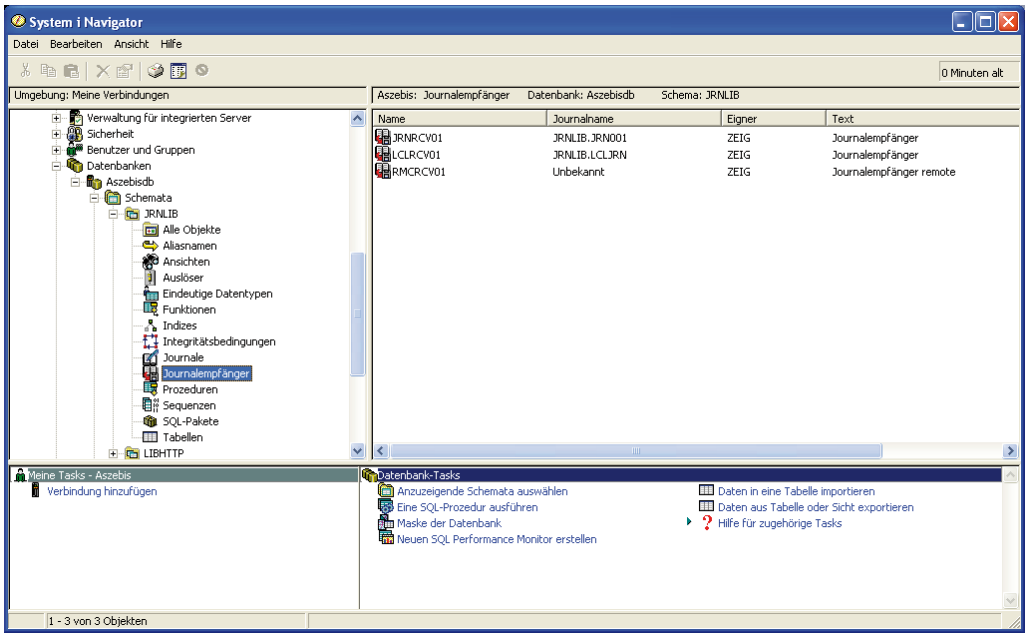
Ein weiterer Parameter des Befehls CHGRMTJRN steht seit Version 6.1 des IBM i Servers zur Verfügung. Er betrifft das Einschalten des neuen Checksum-Verfahrens, mit dem die vollständige Übertragung der Informationen zwischen Quell- und Zielsystem gewährleistet werden soll.



CHGRMTJRN-2

Alternativ kann auch der IBM Systems Director Navigator für die Einrichtung eines Journals genutzt werden.

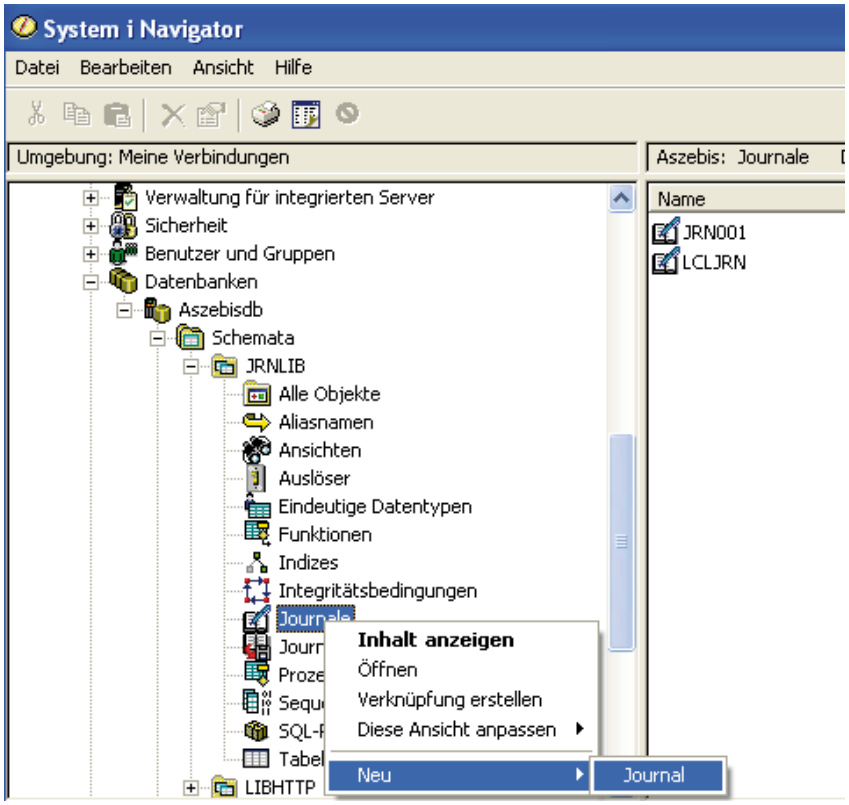
Die allgemeinen Einstellungen für Journale werden im Bereich „Datenbank“ des System i Navigators verwaltet. Hier finden sich auf Bibliotheksebene die beiden Bereiche „Journale“ und „Journalempfänger“.



Journalempfänger im System i Navigator

2.8.1.5
Seite 4

In dieser Anzeige lassen sich beispielsweise auch neue Journale anlegen. Dazu wählt man im linken Navigationsbereich den Eintrag „Journale“ mit der rechten Maustaste und legt über die Option „Neu“ ein neues Journal an.

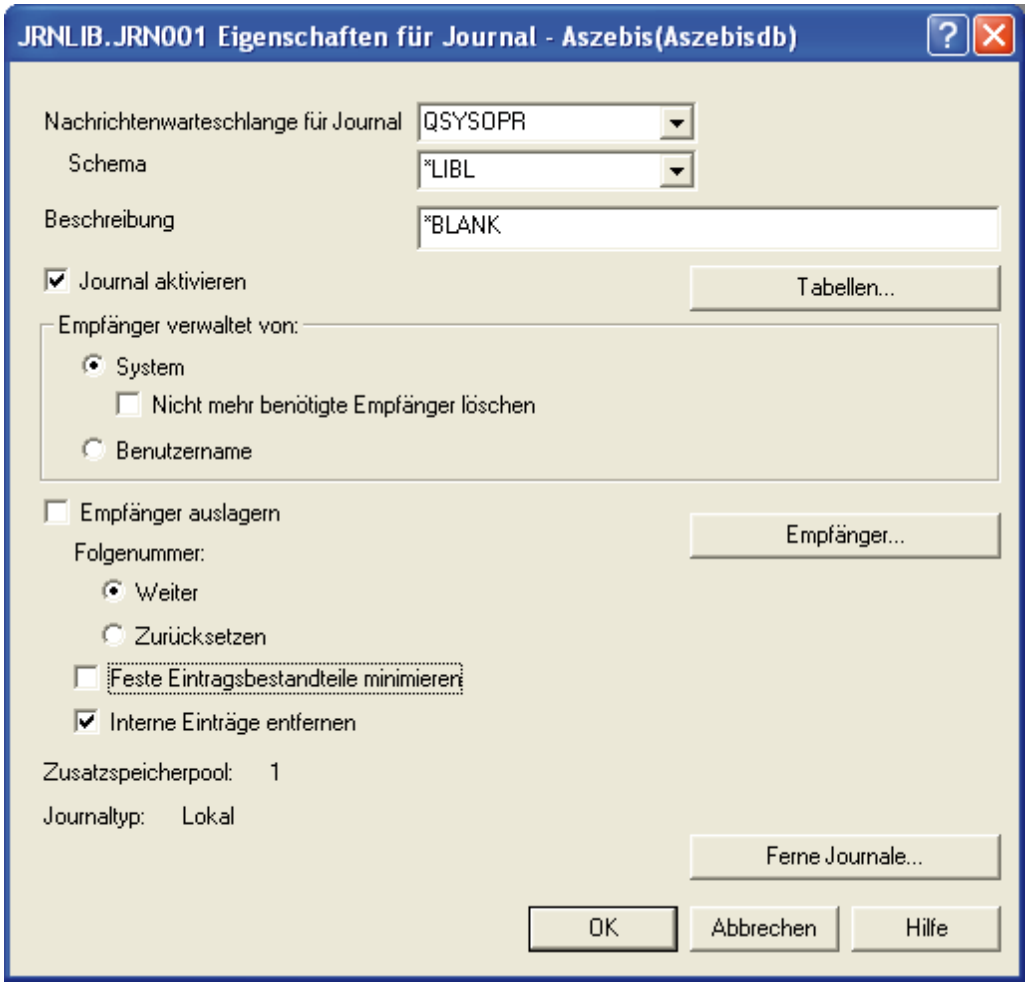


Neues Journal anlegen

Wir sollten an dieser Stelle davon ausgehen, dass bereits ein Journal existiert. Basierend auf einem existenten Journal kann jetzt ein fernes Journal definiert werden. Dazu geht man wie folgt vor:

- Erweitern des Eintrags „Journalverwaltung“
- Auswahl der „Datenbank / Bibliothek“
- Auswahl des „Journals“ mit einem Doppelklick

Mit diesem Vorgehen gelangen wir in den allgemeinen Definitionsbereich des Journals. Ein Beispiel dafür finden Sie in der folgenden Abbildung:

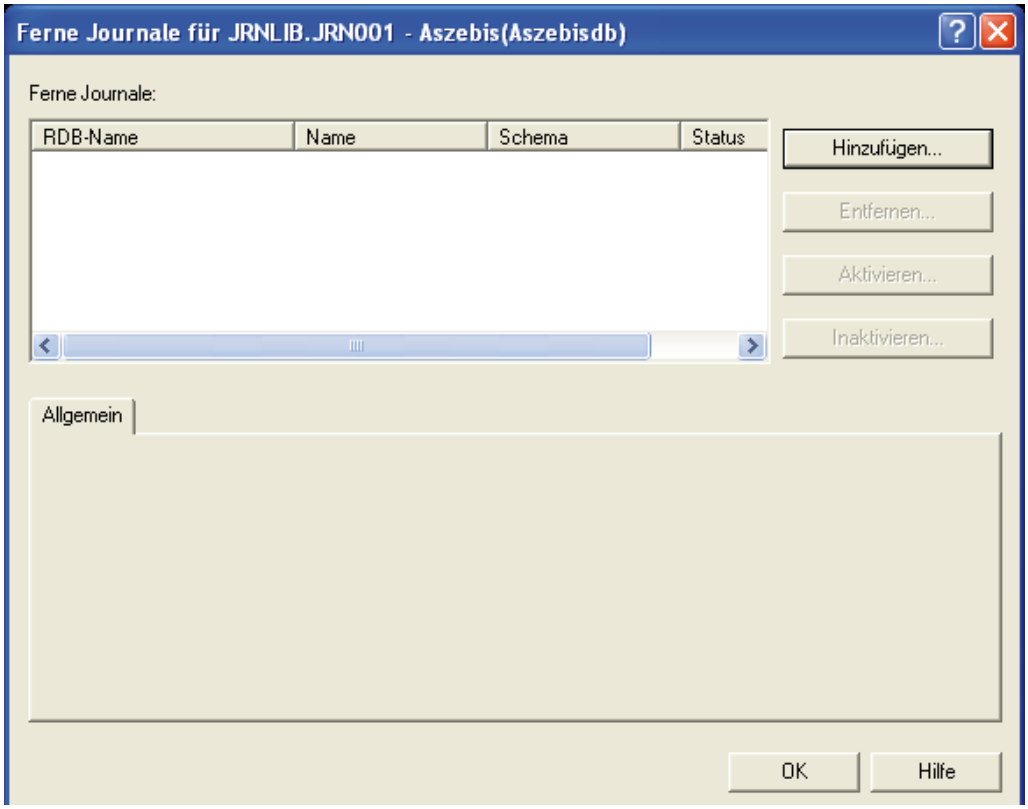


Journaleigenschaften

Entgegen der Verwaltung im Green Screen bietet die grafische Variante im Verwaltungsbereich der Journale weitere Funktionen – wie zum Beispiel das direkte Verwalten der Journalempfänger zu diesem Journal oder auch die Verwaltung der fernen Journale.

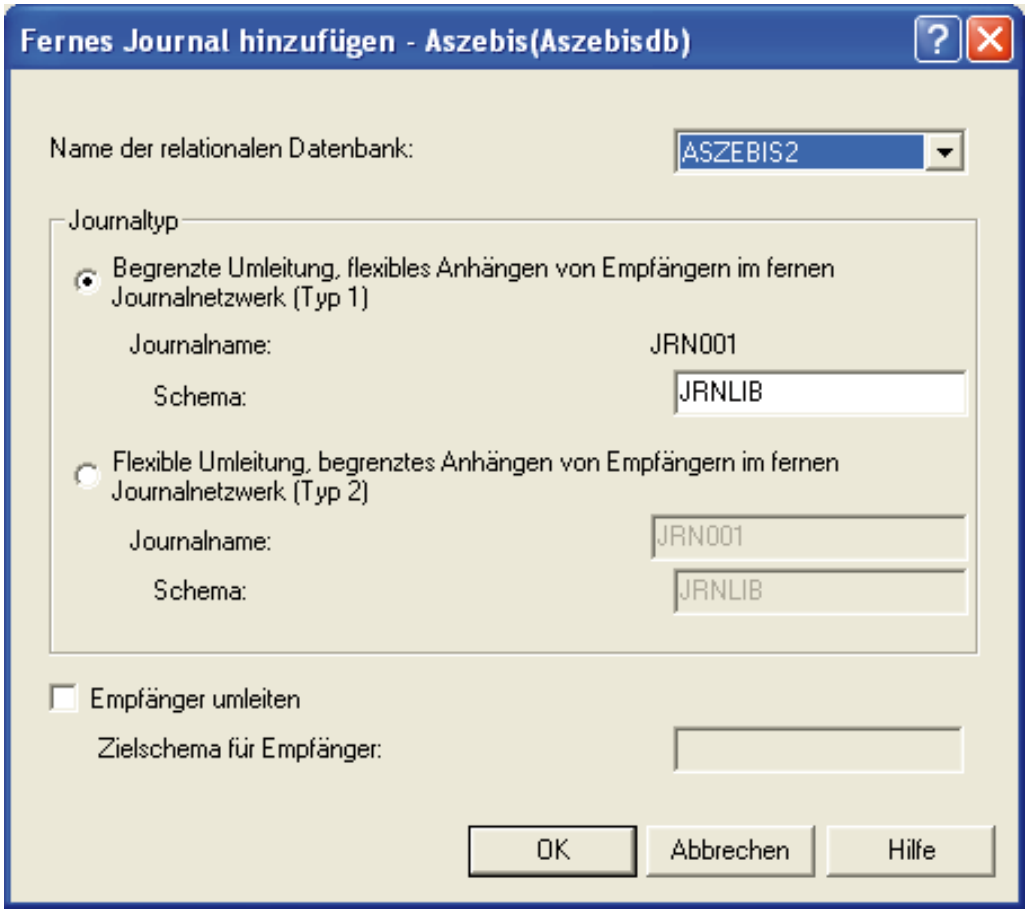
Da wir hier ein fernes Journal erstellen oder verwalten wollen, klicken wir auf den Eintrag „Ferne Journale“. Damit gelangen wir in den Verwaltungsbereich der fernen Journale.

2.8.1.5
Seite 6



Neues fernes Journal hinzufügen

Mit einem Klick auf „Hinzufügen...“ kann ein neues fernes Journal definiert werden. Voraussetzung ist hier, dass die Verbindung zur fernen Maschine bzw. zu deren Datenbank vorab definiert wurde.



Details zum neuen fernen Journal

Das Starten des fernen Journals kann natürlich auch über die grafische Administrationsoberfläche in Form des IBM Systems Director Navigator durchgeführt werden.

Der Startvorgang in dieser Umgebung wird wie folgt durchgeführt:
- Erweitern des Eintrag „Journalverwaltung“
- Auswahl der Datenbank/Bibliothek
- Auswahl des Eintrags „Journal“
- Selektion „Ferne Journalaktionen“
- Auswahl des gewünschten fernen Journals und Klick auf „Aktivieren“.

ACHTUNG!

Das initiale Starten eines fernen Journals kann durchaus längere Zeit in Anspruch nehmen. Das hängt natürlich von der Anzahl der Journaleinträge ab. Bedenken Sie das auf jeden Fall vor dem Start des fernen Journals. Insbesondere dann, wenn es interaktiv gestartet werden soll!

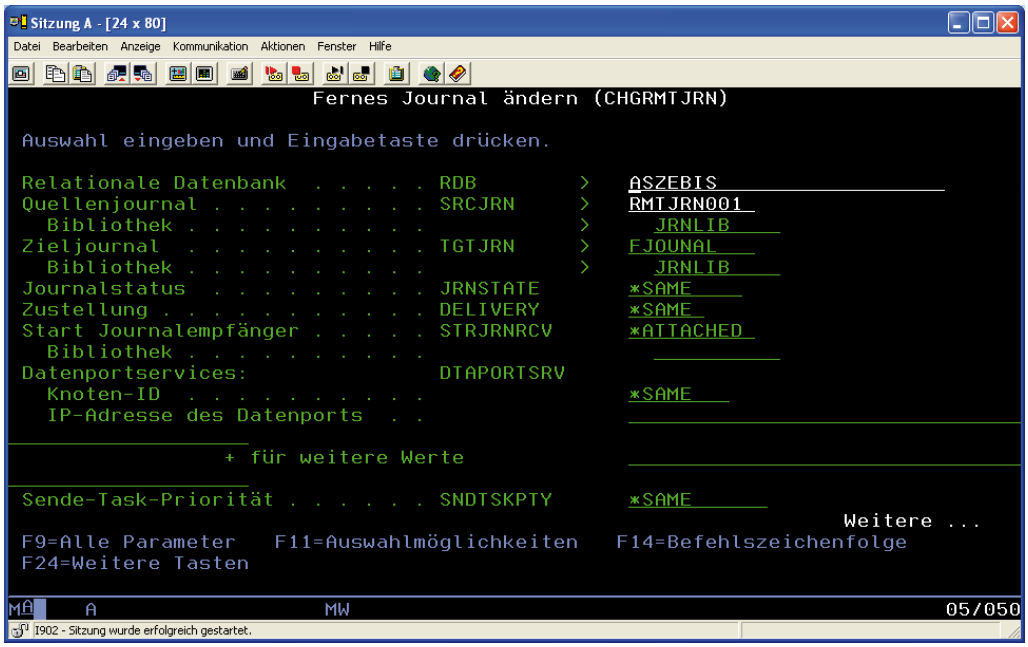
2.8.1.5
Seite 8

Beenden des fernen Journals

Im Regelfall sollte das Beenden eines fernen Journals ein beabsichtigter Prozess sein, der die Übertragung der Daten vom Quellsystem auf das Zielsystem unterbricht.

In der Praxis kann es aber immer mal wieder vorkommen, dass ein fernes Journal plötzlich beendet wird.

Bleiben wir aber zunächst beim normalen Beenden des fernen Journals. Ähnlich wie der Startprozess ist der Ausgangsbefehl für das Beenden des fernen Journals erneut CHGRMTJRN.

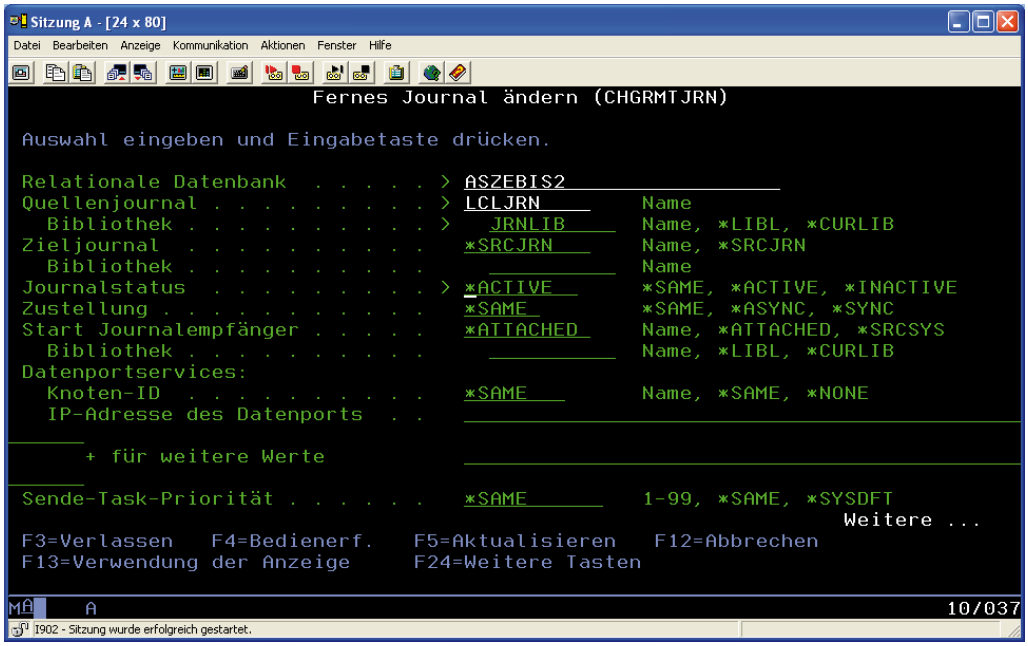


CHGRMTJRN

Das ferne Journal ist dann inaktiv, wenn der Status *INACTIVE anzeigt. Damit können auf dem Zielsystem keine neuen Journaleinträge eingehen und verarbeitet werden. Auf dem Quellsystem werden die anstehenden Journaländerungen fortgeschrieben und warten quasi auf die Übertragung auf das ferne System.

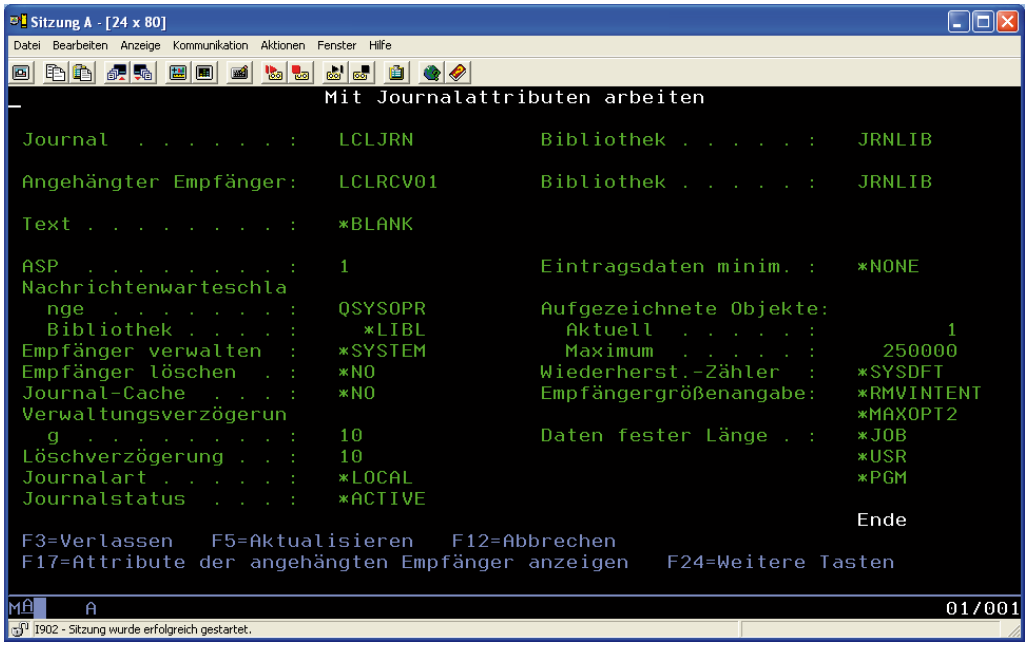
Auch ein Abschalten des Quellsystems führt natürlich zum Beenden des fernen Journals. Nicht automatisch wird hingegen der Start des fernen Journals nach einem IPL durchgeführt. Nach einem Neustart der Maschine muss das ferne Journal manuell oder mittels eines CL gestartet werden. Wichtig ist dabei, dass der Status in CHGRMTJRN auf *ACTIVE steht:

2.8.1.6
Seite 2



*Status *ACTIVE*

Jedes Journal sendet beim Beenden eine Nachricht in die bei der Erstellung des Journals angegebene Warteschlange. Diese Warteschlange kann auch geändert und mit dem Befehl WRKJRNA eingesehen werden:



WRKJRNA

Seit Version 6.1 des IBM i Servers werden die Nachrichten über das Beenden eines Journals nicht nur in die Nachrichtenwarteschlange, sondern auch in das System History-Protokoll geschrieben.

Details in der Nachricht weisen auf den Grund des Beendens hin. Wenn das ferne Journal beendet wird, dann sollte man sich zunächst die Einträge in der Nachrichtenwarteschlange des lokalen Systems anschauen und die Ursache ermitteln.

Je nachdem, welche der beiden Seiten (Quellsystem oder Zielsystem) Ursache für das Beenden war, finden wir unterschiedliche Informationen in den Nachrichtenwarteschlangen der beiden Systeme. Lag die Ursache auf dem Quellsystem, dann ist die Wahrscheinlichkeit groß, dass der Eintrag für das Beenden auf dem Zielsystem zu finden ist.

Hier eine Auflistung der am häufigsten mit fernen Journalen auftretenden Fehler- bzw. Ursachencodes für das Beenden des Datentransfers.

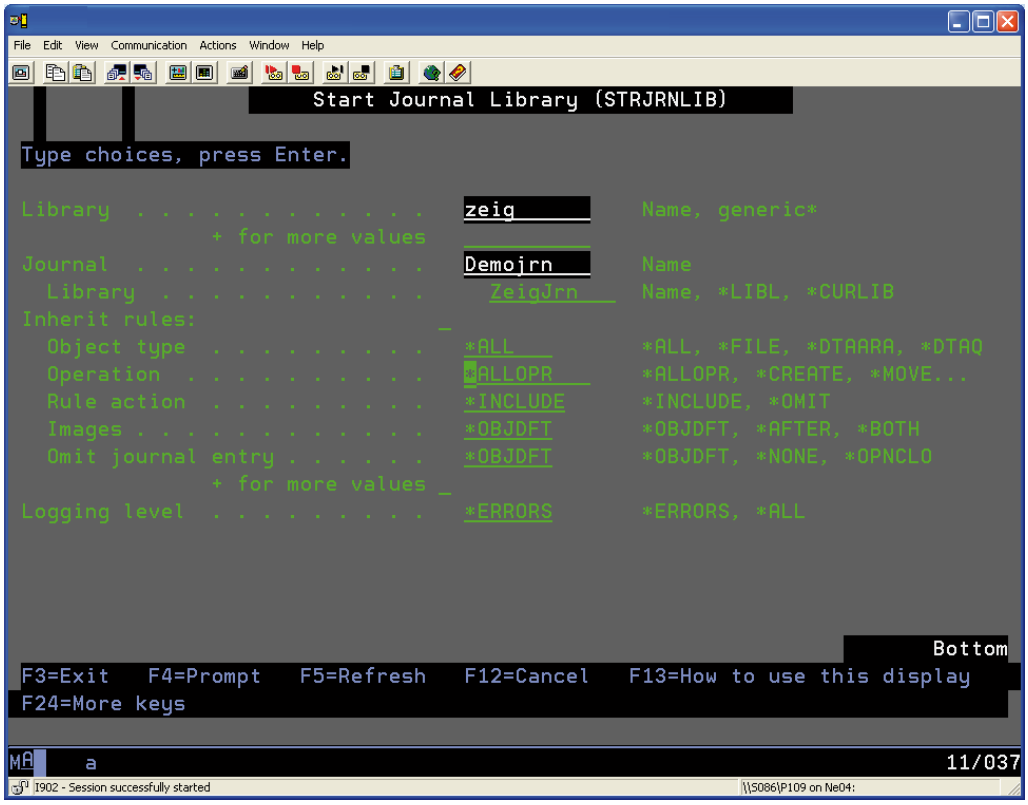
Diese Fehlercodes beziehen sich auf dem Quellsystem liegende Ursachen:
- CPF70D3
- CPF70D5
- CPC6984
- CPC6983
- CPF70C5
- CPI7012
- CPI7016

Diese Fehlercodes beziehen sich auf dem Zielsystem liegende Ursachen:
- CPF70D4
- CPF70D5
- CPF70DB
- CPF70DC
- CPF70D7
- CPC6983
- CPF70C4
- CPF70C5
- CPI7012
- CPI7016

Dass neue Release-Versionen wirklich sinnvolle Neuerungen beinhalten, zeigt auch in diesem Fall das Erscheinen der Version 7.1. Hier sind Funktionen implementiert, mit denen die fernen Journale nach einem Ausfall automatisch wieder gestartet werden können. Die Basis dafür bilden die Nachrichten CPF70C5 und CPF70D5.

Einträge für ferne Journale können seit 6.1 (optimiert in 7.1) gefiltert werden. Dazu steht mit dem Befehl STRJRNLIB der Parameter INHRULES zur Verfügung. Ein Beispiel zeigt die folgende Abbildung:

2.8.1.6
Seite 4



STRJRNLIB

Hier können Filter für Ein- oder Ausschluss von Informationen bei Verwendung mit dem fernen Journal definiert werden.

Verwalten des fernen Journals

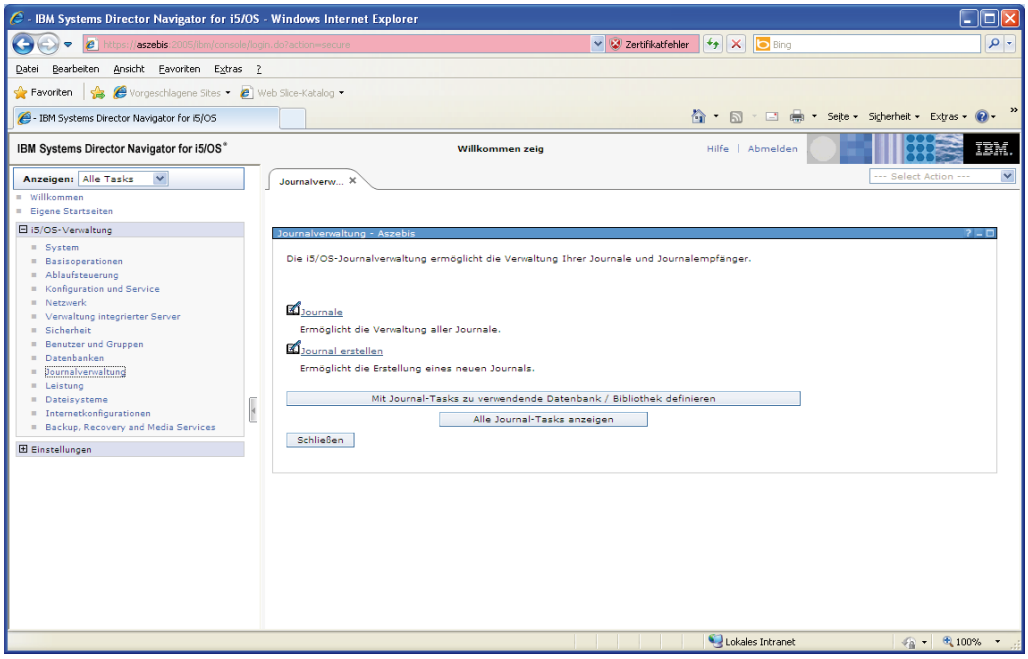
Wir haben bereits eine grafische Verwaltungsfunktion für Journale grob skizziert kennengelernt – den System i Navigator. Dieses Verwaltungstool wird seit Jahren von IBM für die Administration des IBM i präferiert. Seit kurzem gilt nunmehr die browserbasierte Version des IBM Systems Director als zukünftiges strategisches Verwaltungswerkzeug für die Maschine bzw. deren Betriebssysteme. Die Funktionen zur Verwaltung des fernen Journals sind in der grafischen Oberfläche des System Director Navigator for i elegant gelöst.

Dort finden wir die entsprechenden Funktionen im Abschnitt „Journalverwaltung".

Journalverwaltung im IBM Systems Director

Um ein fernes Journal zu verwalten, erweitern wir den Eintrag „Journalverwaltung". Wählen Sie den Eintrag „Datenbank setzen / mit Journaltasks zu verwendende Bibliothek". Wählen Sie abschließend den Eintrag „Journale", selektieren Sie das gewünschte Journal und klicken Sie auf „Remote Journalaktionen".

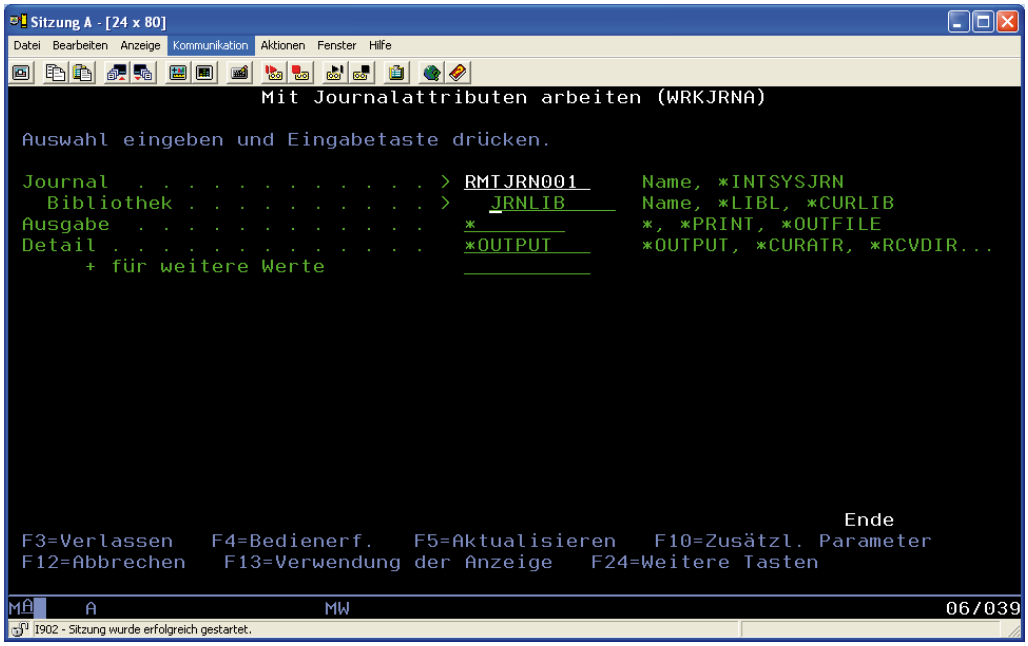
2.8.1.7
Seite 2



Verwaltungsbereich Journale

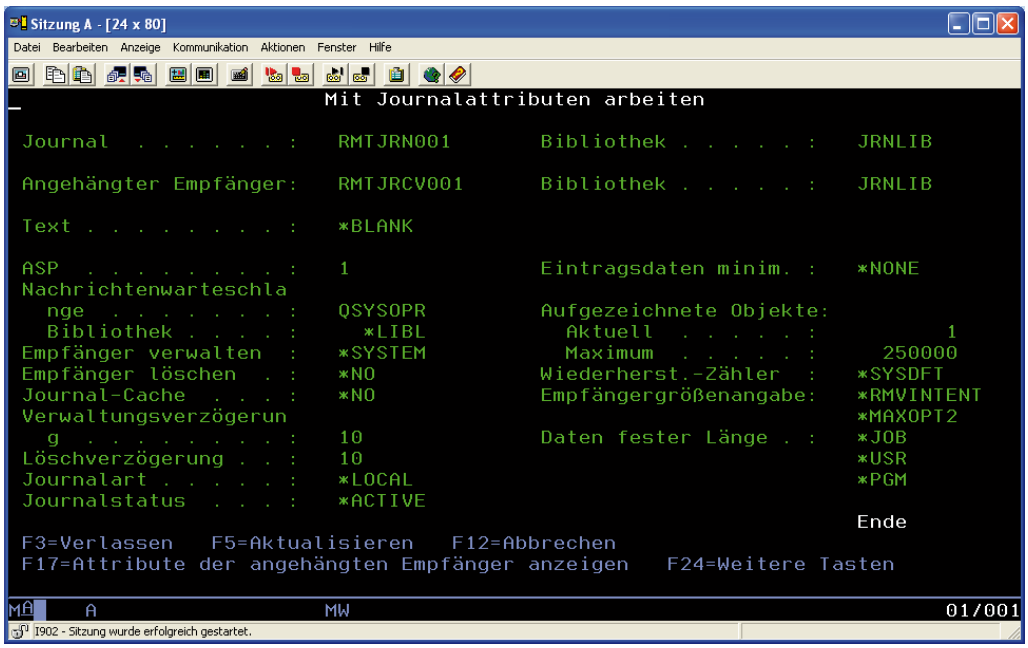
Im Eigenschaftsbereich dieser Anzeige finden wir eine Fülle an Informationen zum fernen Journal.

Als Alternative zur grafischen Oberfläche können wir zur Verwaltung eines fernen Journals auch den Befehl WKRJRNA verwenden. Dieser liefert jeweils die Informationen zu dem auf dem System befindlichen Journalen. Dabei wird zunächst einmal nicht zwischen lokalen oder fernen Journalen unterschieden.



WRKJRNA

Ergebnis von WRKJRNA:



Detailanzeige WRKJRNA

2.8.1.7

Seite 4

Filter für das ferne Journal

Die Menge der mit fernen Journalen zu übertragenden Daten kann imposante Ausmaße annehmen und dementsprechend auch die Kommunikation zwischen dem Quell- und dem bzw. den Zielsystemen beeinflussen. Nun ist die Einrichtung von fernen Journalen meist mit der Notwendigkeit verbunden, kritische Daten und Informationen redundant auf mehreren Systemen zur Verfügung zu haben. Dem steht häufig eine Beschränkung der zu übertragenden Daten entgegen. Was aber, wenn genau die nicht übertragenen Daten in einem Notfall benötigt werden?

Und dennoch: in bestimmten Fällen kann es durchaus Sinn machen, dass die in einem Journal aufgezeichneten Daten nicht auf ein Zielsystem übertragen werden. Natürlich sollte man solche Definitionen nur vornehmen, wenn man absolut sicher sagen kann, dass die für das ferne Journal ausgeschlossenen Informationen nicht für einen Weiterbetrieb auf dem Zielsystem benötigt werden.

Ein Beispiel für den Einsatz von solchen Filtern ist das Ausschließen der „Before"-Einträge in einem Journal. Journale sind von Haus aus für unterschiedliche Einsätze geeignet. So z. B. auch für das Sicherstellen von Transaktionen innerhalb von Anwendungen. Commit-Steuerungen benötigen in der Regel die Aufzeichnung des Standes eines geänderten Datensatzes vor oder nach der Änderung. Deshalb werden diese Informationen auch in einem Journal gespeichert. Wenn ein solches Journal nicht nur als Basis für die Commit-Steuerung in Anwendungen, sondern als Sekundäreffekt und somit als Basis für das ferne Journal dient, dann sind die „Before"-Einträge nur unnötiger Ballast, der für das ferne Journal weggelassen werden kann.

Mit der (kostenpflichtigen!) Option 42 des Betriebssystems IBM i bietet IBM u. a. die Möglichkeit zum Einsatz von Filtern für ferne Journale.

Diese Filter basieren auf drei Hauptkriterien:
- Auszuschließende Objekte
- „Before"-Einträge
- Verarbeitungsprogramm für den Eintrag

2.8.1.8

Seite 2

Automatisieren des fernen Journals

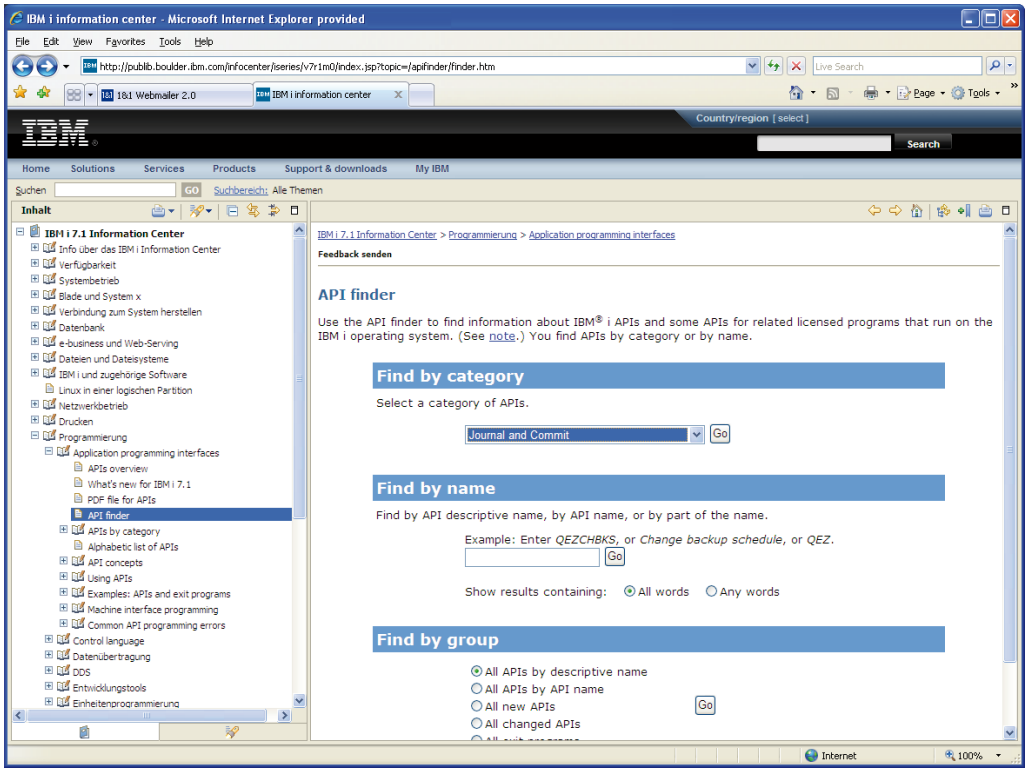
Die von IBM bereitgestellten Befehle sind natürlich für das Einrichten und auch für den Betrieb eines fernen Journals vollkommen ausreichend. Allerdings wünscht man sich hier und da doch ein wenig mehr Automatisierung. Aber hier überlässt IBM das Feld wohl bewusst den Herstellern solcher Produkte, die auf den fernen Journal-Funktionen aufbauen und erweiterte Funktionen oder Lösungen entwickelt haben.

Neben den verschiedenen CL-Befehlen, die wir im Verlauf der Behandlung der fernen Journale kennengelernt haben, stehen – wie in vielen anderen Bereichen auch – APIs für den Einsatz mit den fernen Journalen zur Verfügung.

Eine vollständige Auflistung der verfügbaren APIs, die für die Steuerung der fernen Journale genutzt werden können, finden Sie auf der folgenden Webseite:

http://publib.boulder.ibm.com/infocenter/iseries/v7r1m0/index.jsp?topic=/apifinder/finder.htm

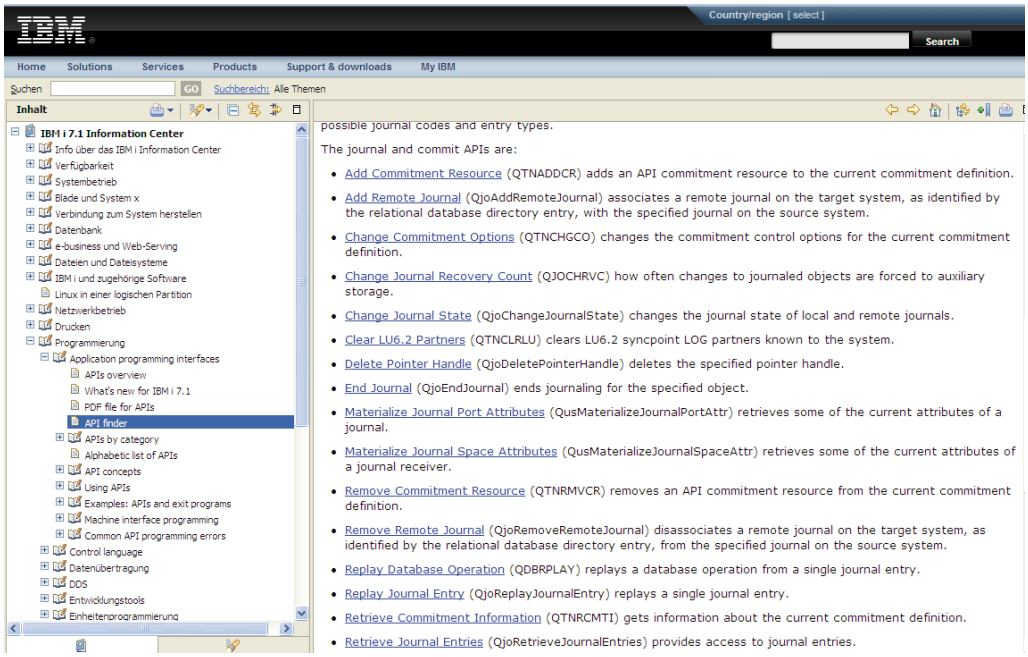
Ideal ist hier das Suchen nach Kategorien. Also wählen wir „Journal and Commit“ aus, wie es die folgende Abbildung zeigt:



API-Auflistung

Als Ergebnis erhalten wir eine Übersicht der mit Journalen einsetzbaren APIs. So z. B. das API „QjoAddRemoteJournal“.

2.8.1.9
Seite 2



Übersicht Journal-APIs

Systempflege

Ein Computer ist grundsätzlich nur eine Maschine, die hin und wieder gewartet werden muss, damit sie ihren Dienst ordnungsgemäß verrichten kann. Um das Risiko eines Systemausfalls zu minimieren, sollten Sie aber in gewissen Zeitabständen bestimmte Wartungsarbeiten an Ihrem System ausführen. Hierzu gehören beispielsweise regelmäßige IPLs. Sicher müssen Sie Ihr System heute nicht mehr so häufig starten wie in der Vergangenheit, aber ein bis zweimal pro Jahr ist eine sicher vertretbare und realisierbare Anzahl von Systemneustarts. Während eines IPLs führt das System einige Bereinigungsaufgaben durch. Bei einem IPL werden zum Beispiel die Zugriffspfade logischer Dateien neu erstellt und nicht mehr erforderlicher Speicherplatz auf den Platten freigegeben. Wer jetzt allerdings glaubt, alles richtig zu machen, da er bereits seit Jahren wöchentlich seinen Server neu startet, den muss ich an dieser Stelle enttäuschen. IBM baut mittlerweile im Arbeitsspeicher temporäre Objekte auf, die sich positiv auf die Performance Ihrer Datenbankabfragen auswirken. So enthält der Arbeitsspeicher seit Release V5R2 beispielsweise einen Cachebereich, in dem generierte SQL Runtime-Objekte gespeichert werden, die die neue Query Engine SQE erstellt. Mit jedem IPL wird der 512MB große Cache-Bereich geleert; mit der Folge, dass nach dem IPL alle SQL Abfragen neu vom SQL Optimizer optimiert und unter Umständen auch temporäre Indizes neu aufgebaut werden müssen. Besser ist es also nur viertel- oder halbjährlich den Systemneustart zu planen und einige andere Aufgaben manuell zu realisieren. Aber auch das regelmäßige Einspielen von Ptfs bzw. Fixes gehört zu den regelmäßigen Wartungsaufgaben eines Operators. Auch wenn Ihr System i Server eines der zuverlässigsten Systeme am Markt ist. Drei von vier gemeldeten Problemen, die Fehler betreffen, sind Neuauflagen bereits gemeldeter Probleme. Viele Benutzer hätten das Problem oder den Ausfall möglicherweise vermieden, wenn der verfügbare Fix – auch PTF (Program Temporary Fix) genannt – in ihrem System angelegt worden wäre.

21. Ergänzung 12/2009

2.9
Seite 2

Vorläufige Programmkorrekturen (Fixes)

Haben Sie sich schon einmal ein neues Auto gekauft? Sie werden sicherlich ohne zu zögern alle vorgeschriebenen Wartungen durchführen lassen: Ölwechsel alle 10.000 Kilometer, große Inspektion nach 30.000 Kilometern und natürlich regelmäßige Reifenwechsel. Diese Wartungsstrategie wird von den Automobilherstellern für neue Fahrzeuge empfohlen, um kostspielige Probleme zu vermeiden. Auch wenn Ihre iSeries bzw. i5 eines der zuverlässigsten Systeme am Markt ist, gibt es bei IBM ähnliche Richtlinien für regelmäßige „Wartungen". Drei von vier gemeldeten Problemen, die Fehler betreffen, sind Neuauflagen bereits gemeldeter Probleme. Viele Benutzer hätten das Problem oder den Ausfall möglicherweise vermieden, wenn der verfügbare Fix – auch PTF (Program Temporary Fix) genannt – in ihrem System angelegt worden wäre. Ungeplante Ausfallzeiten stellen eine starke Beeinträchtigung der Mitarbeiterproduktivität, des Geschäftsbetriebs und nicht zuletzt des Umsatzes dar. Hinzu kommt, dass die Zeiten lange vorbei sind, als die PTFs lediglich der Fehlerbehebung dienten. Heute implementiert die IBM über die Fixes auch neue Funktionen. IBM erstellt zwar die PTFs, doch es ist Aufgabe des Benutzers diese zu beschaffen und zu installieren.

Leider gibt es keine einfache Empfehlung für die Fixverwaltung. Jedes System und jede Umgebung muss individuell betrachtet werden. Bevor Sie also eine eigene Fixstrategie entwickeln, müssen wir uns zunächst einmal mit einigen Begriffen auseinandersetzen. Hierzu gehört die Frage: Welche Fixtypen gibt es und welchen Fixstatus hat Ihre iSeries?

In Bezug auf vorläufige Programmkorrekturen (PTFs) oder Fixes gibt es viele Begriffe. Ich verwende im folgenden die Begriffe „vorläufige Programmkorrektur (PTF)" und „Fix" gleichbedeutend, denn die PTF-Terminologie gibt es schon lange vor dem iSeries-Server und ist heute hauptsächlich mit der zeichenorientierten Schnittstelle, dem „Green Screen" verknüpft. Dies wird in CL-Befehlen wie DSPPTF (PTF anzeigen) oder SNDPTFORD (PTF-Bestellung senden) deutlich. Die Fixterminologie wurde mit der grafischen Benutzerschnittstelle von iSeries Navigator eingeführt. Beide Begriffe haben allerdings dieselbe Bedeutung.

2.9.1 Seite 2

Einzelne Fixes

Einzelne PTFs (Fixes) werden von der IBM erstellt und anschließend veröffentlicht, um gemeldete Fehler zu beheben. Es gibt unterschiedliche Einzel-PTFs:

HIPER-PTFs
HIPER-PTFs beheben ernsthafte Probleme Ihres Systems. Sie sind grundsätzlich für zwei unterschiedliche Arten von Fehlern zuständig: Fehler, die eine starke Auswirkung haben oder die jederzeit und überall auftreten können. IBM verwaltet eine HIPER-PTF-Gruppe, die eine Liste aller aktuellen HIPER-Fixes für jedes unterstützte i5/OS-Release enthält, um die Verwaltung von HIPER-Fixes zu erleichtern.

Vorausgesetzter oder zugehöriger Fix
In Ihrem Fix-Begleitschreiben oder in den PSP (Preventive Service Planning)-Informationen können andere Fixes angegeben sein, die vor oder gleichzeitig mit dem Fix, den Sie installieren wollen, installiert werden müssen. Aber keine Angst, im Gegensatz zu früher, können Sie heute keine Fixes installieren, deren Voraussetzungen nicht erfüllt sind. Das System überprüft, ob die zugehörigen Fixes vorhanden sind.

Für die Verteilung erforderlicher Fix
Heute ist es möglich mit Hilfe des Management Central Fixes auf andere iSeries/i5-Server zu verteilen. Für die Verteilung sind bestimmte PTFs erforderlich. Diese PTFs werden nur für Verteilungszwecke benötigt. Das System macht es nicht erforderlich, dass Sie die für die Verteilung erforderlichen Fixes anlegen. Es wird jedoch empfohlen, sie mit dem zugehörigen PTF anzulegen. Für die Verteilung erforderliche Fixes werden nicht über „Management Central“ gesendet.

Verzögerte und sofortige Fixes

Einige PTFs (Fixes) können angelegt werden, ohne dass ein IPL erforderlich ist; andere Fixes können nur während eines IPL angelegt werden. Einige Ihrer Systemanwendungen sind immer dann aktiv, wenn Ihre iSeries gestartet ist. Ein typisches Beispiel hierfür ist der lizenzierte interne Code (LIC) und große Teile des Betriebssystems. Der LIC ist ein geschützter Bereich, der sehr viele Funktionen ausführt: Speicherverwaltung, Zeigeradressierung, Ausnahme- und Ereignisverwaltung, Datenfunktionen und Sicherheit. Das System enthält zwei Kopien vom LIC. Eine Kopie wird als permanente Kopie verstanden und mit dem Buchstaben A bezeichnet. Die andere Kopie ist ein Duplikat von A und enthält alle temporär angelegten PTFs. Dieser Bereich wird mit B gekennzeichnet. Falls Sie nicht eingreifen, wird diese Kopie (B-Bereich) bei jedem IPL verwendet.

2.9.1.1
Seite 2

- **Verzögerte Fixes**
  Verzögerte Fixes können nur während eines IPL angelegt werden. Eine Ausnahme bilden verzögerte Fixes, die den lizenzierten internen Code betreffen. Sie können sofort angelegt werden, wenn der Speicherbereich A verwendet wird.
- **Sofortige Fixes**
  Sofortige Fixes können ohne IPL angelegt werden, wenn die betroffenen Objekte nicht im Gebrauch sind.

Temporäre und permanente PTFs
PTFs können entweder permanent oder temporär auf ein IBM-Programm angewendet werden. Werden die PTFs temporär angewendet, wird zunächst eine Kopie der ursprünglichen Objektversion gesichert. Bei permanenten PTFs wird die alte Version zerstört. Permanente Fixes können demzufolge nicht mehr gelöscht werden. Aus diesem Grund werden die PTFs in der Regel zunächst temporär angelegt. Erst wenn Sie erneut PTFs einspielen und sichergestellt ist, dass Sie keine Probleme nach dem Einspielen der PTFs haben, werden die alten PTFs in den permanenten Zustand übernommen.

Kumulative PTF-Pakete

Kumulative PTF-Pakete werden von der IBM in regelmäßigen Abständen veröffentlicht. Sie enthalten Einzelfixes für ein bestimmtes Release des Betriebssystems i5/OS. Die in einem kumulativen PTF-Paket enthaltenen Einzelfixes sind für den lizenzierten internen Code und die meisten Lizenzprogramme bestimmt, die unter diesem Release des Betriebssystems installiert werden können.

Wie der Name sagt, ist jedes Paket kumulativ; das heißt, es enthält alle Fixes des vorherigen Cum-PTFs sowie zusätzliche Fixes, die seit dem vorherigen Paket herausgegeben wurden. Viele, aber nicht alle neuen Fixes werden zu kumulativen Paketen zusammengefasst. Nicht enthalten sind normalerweise Fixes, die spezifische Benutzersituationen oder -anwendungen betreffen. Ausschlaggebend für die Übernahme ist die Anzahl der Verwendung, d.h., wenn ein Einzel-PTF weltweit mehr als 35 mal angefordert wurde, wird die IBM dieses PTF in das Cum-PTF integrieren. Für spezifische Fehler auf Ihrer Maschine könnte es sich daher lohnen, gezielt nach veröffentlichten Einzel-PTFs zu suchen und nicht allein auf das CUM-PTF zu vertrauen.

Zusätzlich zum kumulativen PTF-Paket erhalten Sie stets die aktuelle Datenbank PTF-Gruppe und die HIPER PTF-Gruppe. Wie oft ein kumulatives Paket zur Verfügung steht, ist abhängig von der Stufe des OS/400-Release-Lebenszyklus. In den ersten Monaten der allgemeinen Verfügbarkeit eines neuen Release kann ein kumulatives Paket alle vier bis sechs Wochen veröffentlicht werden. Danach werden kumulative Pakete vierteljährlich geplant. Wenn das Release längere Zeit zur Verfügung steht, kann die Häufigkeit der Cum-PTFs noch weiter abnehmen.

Man kann nicht sagen wie oft Sie die kumulativen PTF-Pakete installieren sollten. Die IBM empfiehlt eine vierteljährliche Installation in dynamischen Umgebungen und in stabilen Umgebungen seltener. Auf alle Fälle sollten Sie an die Installation von kumulativen Fixpaketen vor größeren Hardware- oder Softwareänderungen in Ihrer Umgebung denken.

2.9.1.2
Seite 2

Fixgruppen

Ein Gruppen-PTF ist ein PTF-Paket, das aus einer logischen Gruppe von Einzel-PTFs besteht, die eine bestimmte Funktion (z. B. Datenbank, JAVA, HIPER, etc.) betreffen. Wenn neue PTFs für die Funktion verfügbar werden, werden Gruppen-PTFs dynamisch von der IBM aktualisiert. Ein Charakteristikum der PTF-Gruppen ist darin zu sehen, dass die PTF-Nummer stets beibehalten wird. Vielmehr verfügen die Gruppen-PTFs über eine Level-Stufe. Sie gibt Auskunft über die Aktualität des Gruppen-PTFs. Mit dem Release V5R2 wurde zusätzlich der Befehl WRKPTFGRP von der IBM eingeführt, damit Sie auch auf der „Green Screen"-Seite komfortabel mit den Gruppen-PTFs arbeiten können. Mittlerweile gibt es Gruppen-PTFs für die folgenden Funktionsbereiche:

- HIPER PTF-Gruppe
- DB2 UDB for iSeries
- WebSphere Application Server
- Electronic Service Agent
- TCP/IP PTF-Gruppe
- JAVA PTF-Gruppe
- Cumulative PTF-Gruppe
- PTF-Gruppe für IBM HTTP Server for iSeries

Bitte achten Sie darauf, dass Sie die Java-Gruppe regelmäßig anlegen. Auch wenn Sie selbst kein Java einsetzen, nutzt die IBM Java und viele Tools sind in Java geschrieben. Also denken Sie beim nächsten Mal einfach auch an die JAVA-PTF-Gruppe.

2.9.1.3

Seite 2

Servicepakete

Ein Servicepaket ist eine Reihe von Programmkorrekturen (keine PTFs) für iSeries Access für Windows-Produkte.

Spezifische Servicepaketinformationen finden Sie in:

http://www-03.ibm.com/servers/eserver/iseries/access/

Auf dieser Website erfahren Sie, welche Servicepakete verfügbar sind, wie Sie eine Nachricht erhalten, wenn ein neues Servicepaket zur Verfügung steht und wie ein Servicepaket bestellt und herunter geladen wird.

http://publib.boulder.ibm.com/infocenter/iseries/v5r4/index.jsp?topic=/rzaij/rzaijsp.htm

Dieses Thema enthält Informationen zum Überprüfen des Serviceziels, zum Abrufen des neuesten Servicepakets und zur Installation des Pakets.

2.9.1.4

Seite 2

PTF-Informationen anzeigen

Wir haben jetzt viel über Fixtypen gesprochen. Wahrscheinlich werden Sie sich jetzt folgende Fragen stellen:

- Welche PTFs befinden sich zurzeit auf unserem System?
- Welchen Status haben diese PTFs – permanent oder temporär?
- Welche PTFs stehen zurzeit zur Verfügung?

Es ist also an der Zeit herauszufinden, welche Möglichkeiten Sie haben, sich über den Fixstatus Ihres Systems zu informieren. Generell können Sie die zeichenorientierte Schnittstelle oder die grafische Schnittstelle, den iSeries Navigator nutzen.

8. Ergänzung 09/2006

2.9.2
Seite 2

PTF-Informationen mit der zeichenorientierten Schnittstelle anzeigen

Ihnen stehen verschiedene Möglichkeiten zur Verfügung, um die Fixstatusbeschreibungen anzuzeigen. Sie können Ihre PTFs mit der Option 5 (Vorläufige Programmkorrektur anzeigen) im Menü PTF verwalten. Die Option 5 zeigt Ihnen den Status Ihrer PTFs. Neben dem Status eines PTF finden Sie auch allgemeine Informationen zu einem PTF. Auch den Befehl DSPPTF können Sie nutzen.

Wir wollen jetzt mit dem Befehl DSPPTF prüfen, welche PTFs sich auf dem System befinden.

```
                              PTF anzeigen (DSPPTF)

Auswahl eingeben und Eingabetaste drücken.

Produkt . . . . . . . . . . . .   *ALL      F4=Liste
PTF-Nummern auswählen . . . . .   *ALL      Zeichenwert, *ALL...
Release . . . . . . . . . . . .   *ALL      *ALL, VxRxMx
Nur PTF-Begleitschreiben . . . .  *NO       *NO, *YES
Ausgabe . . . . . . . . . . . .   *         *, *PRINT, *OUTFILE

                                                                Ende
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Bedienerführung des Befehls DSPPTF

Geben Sie die Lizenzprogrammnummer und die Fix-ID für den Parameter LICPGM und für den Parameter SELECT an, wenn Sie die PTFs für ein spezielles Produkt anzeigen wollen.

Die Produkt-IDs können Sie selektieren, wenn Sie die Funktionstaste F4 verwenden.

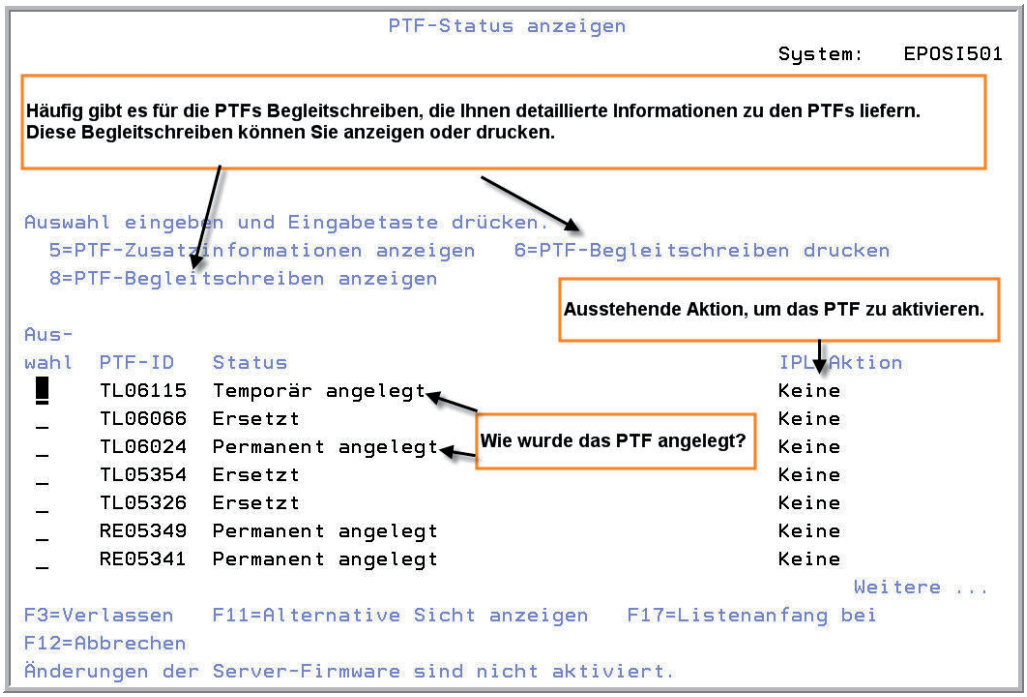
2.9.2.1
Seite 2

```
        Wert für Parameter angeben            LICPGM

Auswahl eingeben und Eingabetaste drücken.

Art  . . . . . . . . . . . . . :   ZEICHEN
Produkt  . . . . . . . . . . .    *ALL

*ALL                                5722IP1 V5R3M0 IBM Infoprint Serve
5722999 V5R4M0 Lizenzierter intern  5722JC1 V5R4M0 IBM Toolbox für Jav
5722SS1 V5R4M0 i5/OS                5722JV1 V5R4M0 IBM Developer Kit f
INFOAS4 V5R2M0 Information APARs     5722LSV V5R4M0 IBM i5/OS Integrati
INFOAS4 V5R3M0 Information APARs     5722NAE V5R4M0 Network Authenticat
INFOAS4 V5R3M5 Information APARs     5722PT1 V5R4M0 Performance Tools f
INFOAS4 V5R4M0 Information APARs     5722QU1 V5R4M0 Query for iSeries
INFODSL V0R0M1 Information APARs     5722ST1 V5R4M0 DB2 Query Manager a
1TMELCF V4R1M0 Tivoli Management A   5722TC1 V5R4M0 TCP/IP-Konnektivitä
5722BZ1 V5R1M0 IBM Business Soluti   5722WDS V5R4M0 WDS für iSeries
5722CM1 V5R4M0 Communications Util   5722XE1 V5R4M0 iSeries Access für
5722DE1 V5R4M0 DB2 UDB Extender      5722XH2 V5R4M0 iSeries Access für
5722DG1 V5R4M0 IBM HTTP-Server für   5722XP1 V5R3M0 iSeries Access for      +

F3=Verlassen   F5=Aktualisieren   F12=Abbrechen   F13=Verwendung der Anzeige
F24=Weitere Tasten
```

Produkt-IDs anzeigen

Soll der Status für alle Produkte auf dem System angezeigt werden, geben Sie für beide Parameter des Befehls DSPPTF *ALL an und drücken die Eingabetaste.

Anschließend werden die permanent und temporär angelegten PTFs angezeigt. Außerdem wird die Aktion angezeigt, die erforderlich ist, um das PTF zu aktivieren.



Die Ausgabe des Befehls DSPPTF

Beachten Sie, dass der Status „Anstehend“ und der Status „Aktion“ anzeigt, dass weitere Aktionen Ihrerseits erforderlich sind, um das PTF zu aktivieren oder zu deaktivieren. PND bedeutet in diesem Zusammenhang, dass der PTF-Status aktualisiert wird, nachdem die Aktionen durchgeführt worden sind. ACN bedeutet, dass der PTF-Status erst beim nächsten IPL aktualisiert wird. Falls Sie den Status „Vorübergehend angelegt – ACN“ finden, kann das System zurzeit die Wirksamkeit des PTFs nicht prüfen.

2.9.2.1
Seite 4

Die folgende Tabelle enthält eine Übersicht der möglichen Statusbedingungen:

Status	Beschreibung
Bestellt	Bestellt, aber nicht im System empfangen.
Nur Begleitschreiben	Zeigt an, dass nur das Begleitschreiben des PTF empfangen wurde.
Nur Sicherungsdatei	PTF ist in einer Sicherungsdatei in der Bibliothek QGPL vorhanden. Das PTF wurde nie geladen oder wurde permanent entfernt.
Nicht angelegt	PTF wurde geladen, aber nicht angelegt oder wurde vorübergehend entfernt.
Nicht angelegt – IPL	PTF wurde geladen, aber nicht angelegt. Er wird beim nächsten IPL angelegt. Ein Server-IPL ist erforderlich, um eine aktive Ausführung des PTF auf dem System zu ermöglichen.
Vorübergehend angelegt	Ein temporärer Fix, der aus dem System entfernt oder permanent angelegt werden kann. Das PTF ist kein permanenter Bestandteil des Systems.
Vorübergehend angelegt – IPL	Ein PTF, der ein Server-IPL erfordert, um eine aktive Ausführung auf dem System zu ermöglichen. Das PTF ist kein permanenter Bestandteil des Systems.
Vorübergehend angelegt – PND	Das PTF wird vorübergehend angelegt, zum Aktivieren des PTF ist jedoch eine Aktion erforderlich. Dieses Feld wird aktualisiert, wenn die PTF-Details angezeigt werden.
Vorübergehend angelegt – ACN	Das PTF wird vorübergehend angelegt, zum Aktivieren des PTF ist jedoch eine Aktion erforderlich. Dieses Feld wird erst beim nächsten IPL aktualisiert. Wenn Sie die erforderlichen Aktionen zur Aktivierung des PTF ausgeführt haben, müssen Sie jetzt kein IPL ausführen. Der Inhalt dieses Felds wird beim nächsten IPL gelöscht.
Permanent angelegt	Das PTF ist ein permanenter Bestandteil des Systems.
Permanent angelegt – IPL	Das PTF ist ein permanenter Bestandteil des Systems. Ein Server-IPL ist erforderlich, damit das PTF aktiv auf dem System ausgeführt wird.
Permanent angelegt – PND	Das PTF wurde permanent angelegt, zum Aktivieren des PTF ist jedoch eine Aktion erforderlich. Dieses Feld wird aktualisiert, wenn die PTF-Details angezeigt werden.

Permanent angelegt – ACN	Das PTF wurde permanent angelegt, zum Aktivieren des PTF ist jedoch eine Aktion erforderlich. Dieses Feld wird erst beim nächsten IPL aktualisiert. Wenn Sie die erforderlichen Aktionen zur Aktivierung des PTF ausgeführt haben, müssen Sie jetzt kein IPL ausführen. Der Inhalt dieses Felds wird beim nächsten IPL gelöscht.
Vorübergehend entfernt	Vorübergehend entfernt.
Vorübergehend entfernt – PND	Das PTF wurde entfernt, zum Aktivieren des alten Codes ist jedoch eine Aktion erforderlich. Dieses Feld wird aktualisiert, wenn die PTF-Details angezeigt werden.
Vorübergehend entfernt – ACN	Das PTF wurde entfernt, zum Aktivieren des alten Codes ist jedoch eine Aktion erforderlich. Dieses Feld wird erst beim nächsten IPL aktualisiert. Wenn Sie die erforderlichen Aktionen zur Aktivierung des PTF ausgeführt haben, müssen Sie jetzt kein IPL ausführen. Der Inhalt dieses Felds wird beim nächsten IPL gelöscht.
Permanent entfernt – IPL	Das PTF wurde permanent aus dem System entfernt, der Code ist jedoch noch auf dem System aktiv. Ein Server-IPL ist erforderlich, damit das PTF nicht mehr aktiv auf dem System ausgeführt wird.
Permanent entfernt – PND	Das PTF wurde permanent entfernt, zum Aktivieren des alten Codes ist jedoch eine Aktion erforderlich. Dieses Feld wird aktualisiert, wenn die PTF-Details angezeigt werden.
Permanent entfernt – ACN	Das PTF wurde permanent entfernt, zum Aktivieren des alten Codes ist jedoch eine Aktion erforderlich. Dieses Feld wird erst beim nächsten IPL aktualisiert. Wenn Sie die erforderlichen Aktionen zur Aktivierung des PTF ausgeführt haben, müssen Sie jetzt kein IPL ausführen. Der Inhalt dieses Feldes wird beim nächsten IPL gelöscht.
Überlagert	Ein neuerer PTF ist auf dem System vorhanden, aber möglicherweise nicht angelegt. Wenn dieser PTF überlagert wurde, rufen Sie die PTF-Details auf, um die PTF-Nummer zu bestimmen, durch die dieser PTF überlagert wurde. Zeigen Sie dann diesen PTF an, um seinen Status zu ermitteln.
Beschädigt	Es wird kein PTF-Objekt gefunden (evtl. versehentlich entfernt). Das PTF muss erneut geladen werden, damit es angelegt oder entfernt werden kann.

2.9.2.1
Seite 6

Manchmal sind diese Informationen nicht ausreichend. Dann können Sie sich zusätzlich Informationen anzeigen lassen. Die folgende Tabelle enthält die zusätzlich verfügbaren Informationen.

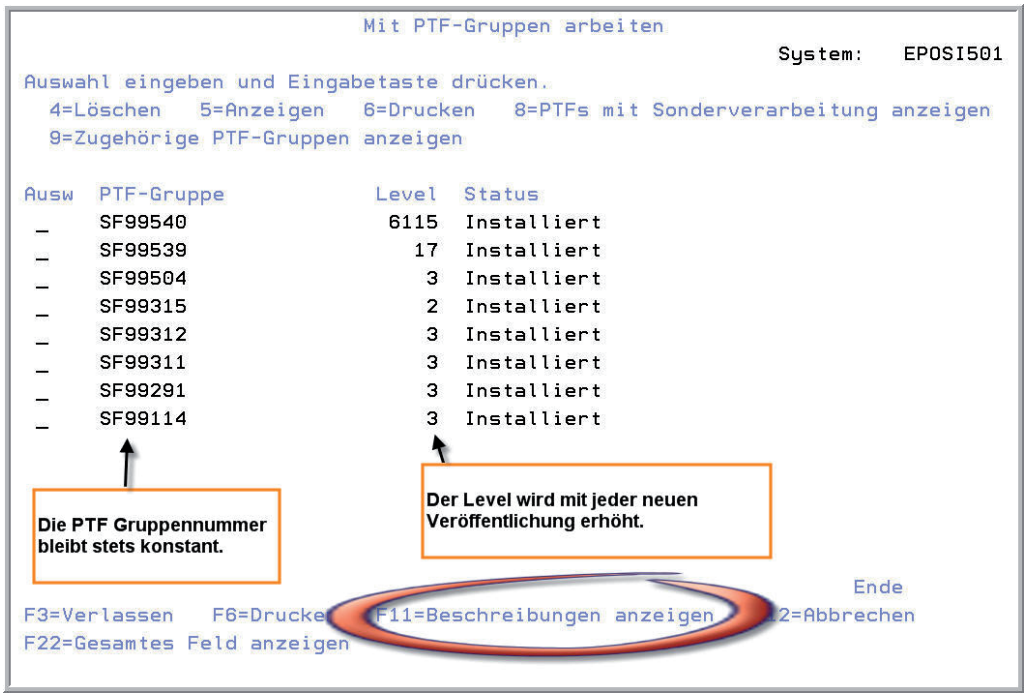
Feld	Beschreibung
Bestellt	Gibt an, ob der Fix bestellt ist. „Ja" bedeutet, dass der Fix bestellt wurde, aber möglicherweise noch nicht auf dem System ist.
PTF-Sicherungsdatei	Gibt an, ob für den Fix eine Sicherungsdatei vorhanden ist.
PTF-Status	Gibt den aktuellen Status des Fixes an. Das Fixstatusfeld in der Anzeige „PTF-Zusatzinformationen anzeigen" kann aus mehreren Gründen von dem Statusfeld in der Anzeige „PTF-Status anzeigen" abweichen: • Wenn es seit dem Anzeigen der Anzeige „PTF-Status anzeigen" eine Statusänderung gegeben hat, zeigt die Anzeige „Allgemeine Informationen" den aktualisierten Status. • Das Statusfeld in der Anzeige „PTF-Status anzeigen" ist eine Kombination des Fixstatusfelds und der Felder „Aktion anstehend" und „Aktion erforderlich" in der Anzeige „Allgemeine Informationen". • Wenn bei der Erstellung der Fixinformationsdatei ein Fehler im System aufgetreten ist, wird der Fixstatus aktualisiert, sobald die Anzeige „Allgemeine Informationen" angezeigt wird.
Neuestes ersetzendes PTF	Gibt die aktuelle (ersetzende) PTF-Version des momentan angezeigten PTF an.

Datum/Uhrzeit des Status	Gibt Datum und Uhrzeit der letzten Änderung des PTF-Status an. Das PTF erhielt den aktuellen Status an dem aufgeführten Datum und zu der aufgeführten Zeit.
Art	Gibt an, ob es sich um einen sofortigen oder um einen verzögerten Fix handelt.
Unbeaufsichtigte IPL-Aktion	Gibt die Aktion an, die beim nächsten unbeaufsichtigten IPL ausgeführt wird.
Produktoption	Gibt die Option an, für die dieser Fix gilt.
PTF-Bibliothek	Gibt die Bibliothek an, in der dieser Fix installiert ist.
PTF-Begleitschreiben	Gibt an, ob es ein Begleitschreiben für diesen Fix gibt.
Verbindliche Anweisungen	Gibt an, ob es Anweisungen gibt, die ausgeführt werden müssen, bevor dieser Fix angelegt wird.
Test-PTF	Gibt an, dass Sie einen Testfix nicht permanent auf dem System anlegen können.
Aktion anstehend	Gibt an, ob eine Aktion vorliegt, die ausgeführt werden muss. Ist in diesem Feld „Ja" angegeben, muss eine Aktion ausgeführt werden, um den Fix zu aktivieren oder zu inaktivieren. Ist „Nein" angegeben, bedeutet dies, dass der Fix aktiv ist.
Aktion erforderlich	Gibt an, ob eine Aktion erforderlich ist, um diesen Fix nach dem Anlegen zu aktivieren.
i5/OS-Zielrelease	Gibt das Release des Betriebssystems an, in das dieser Fix kopiert oder geladen werden kann.
Niedrigster/Höchster Stand	Gibt die niedrigste und höchste Ebene des Produkts, auf der dieser Fix installiert werden kann, an. Einige Fixes können mehrere Ebenen eines Produkts abdecken.
Server-IPL erforderlich	Gibt an, ob das PTF ein Server-IPL erfordert, um die Änderungen für das PTF zu aktivieren.
Datum/Uhrzeit der PTF-Erstellung	Gibt Datum und Uhrzeit der Erstellung des PTF an.

2.9.2.1
Seite 8

In regelmäßigen Abständen sollten Sie sich entschließen, dass aktuelle Cum-PTF einzuspielen. Doch zuvor müssen Sie ja erst einmal den aktuellen Cum-PTF-Stand Ihrer Maschine kennen – übrigens auch eine beliebte Frage der IBM Service-Mitarbeiter. Jetzt haben Sie verschiedene Möglichkeiten:

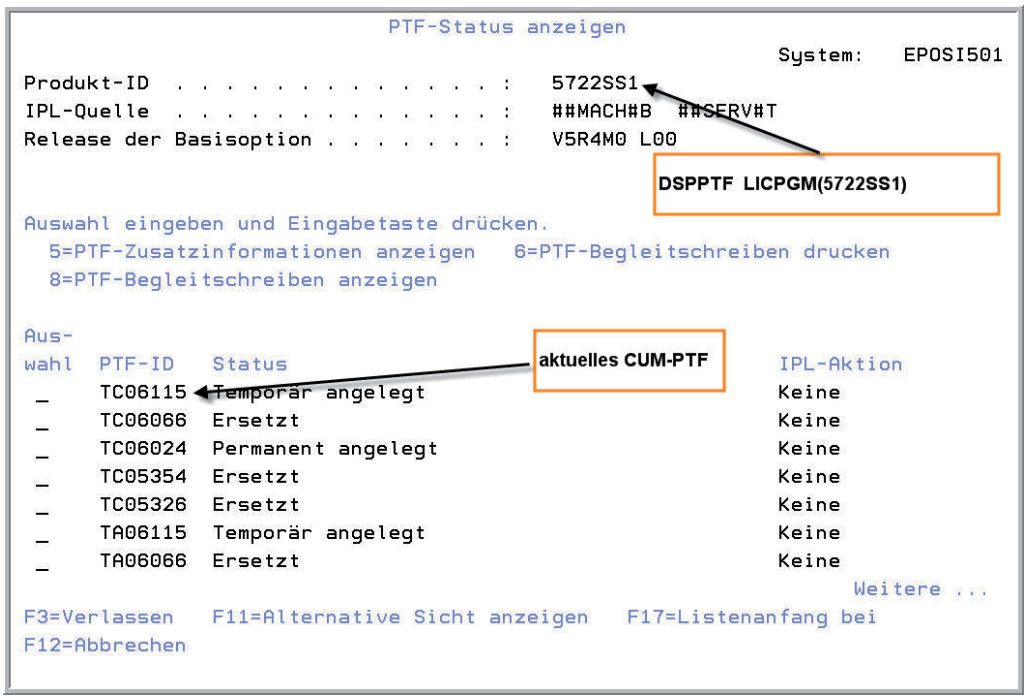
Der Befehl WRKPTFGRP (Mit PTF-Gruppen arbeiten) könnte Ihnen helfen, die Antwort zu finden.



Ergebnis des Befehls WRKPTFGRP

Suchen Sie die PTF-Gruppe SF99540 (Release V5R4) oder die PTF-Gruppe SF99530 (Release V5R3). Der Level der PTF-Gruppe bestimmt das kumulative PTF-Paket. Lautet der Level – wie in unserem Beispiel – 6115, ist die ID Ihres kumulativen PTF-Pakets C6115xxx. XXX wird ersetzt durch die jeweilige Version Ihres Betriebssystems. Wenn Sie das Release V5R4 nutzen, lautet die ID des kumulativen PTF Pakets also C6115540. Der Status der PTF-Gruppe zeigt an, dass alle PTFs des kumulativen PTF-Paketes auf Ihrem System installiert sind.

Alternativ können Sie auch den Befehl DSPPTF LICPGM(5722SS1) eingeben.



Anzeige des Befehls DSPPTF LICPGM(5722SS1)

Es werden alle installierten kumulativen PTF-Pakete aufgeführt. Sie werden feststellen, dass alle IDs für kumulative PTF-Pakete mit dem Buchstaben T beginnen. Das aktuelle Paket steht oben.

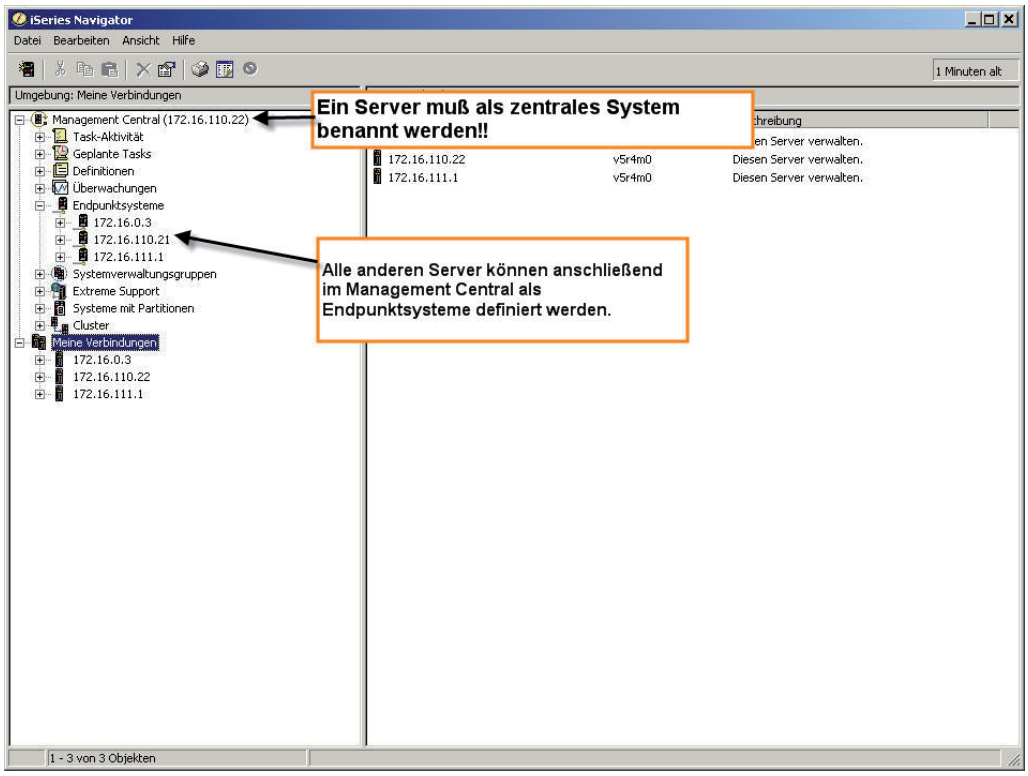
Wenn Sie Version 5 Release 4 Modifikation 0 verwenden und wenn PTF TC06115 auf Ihrem System installiert ist, ist der Stand Ihres kumulativen Pakets auch hier wieder C6115540. Anhand der Fix-ID erhalten Sie aber noch weitere Informationen:

- Fix-IDs, die mit TC beginnen, zeigen an, dass das gesamte kumulative PTF-Paket angelegt wurde.
- Fix-IDs, die mit TA beginnen, zeigen an, dass die HIPER-Fixes und HIPER-Fixes für lizenzierten internen Code angelegt wurden.
- Fix-IDs, die mit TL beginnen, zeigen an, dass nur die HIPER-Fixes für den lizenzierten internen Code angelegt wurden.

2.9.2.1
Seite 10

PTF-Informationen mit iSeries Navigator anzeigen

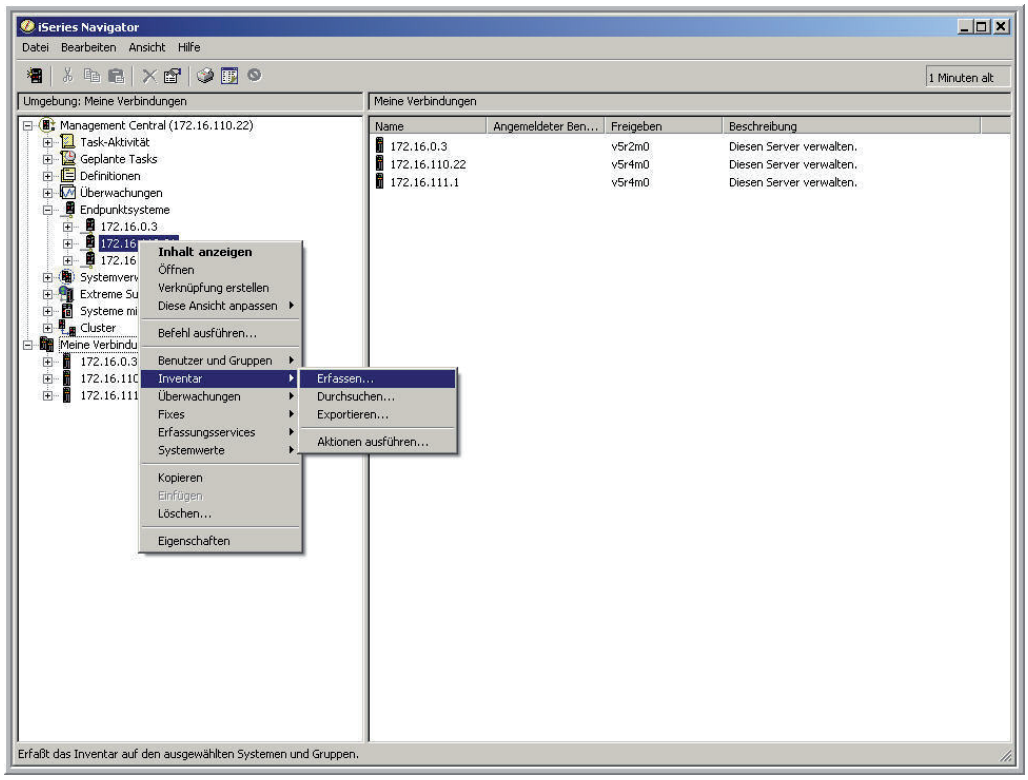
Sie können aber auch Ihr Fixinventar mit dem Grafik-Assistenten „*Management Central*“ des iSeries Navigator verwalten. Mit dem Vergleichs- und Aktualisierungsassistenten können Sie z. B. ein System oder eine Systemgruppe automatisch mit einem Modellsystem vergleichen, fehlende und zusätzliche Fixes ermitteln und die fehlenden Fixes an jedes System senden und dort installieren. Sie können den Vergleichs- und Aktualisierungsassistenten von einem Endpunktsystem, einer Systemverwaltungsgruppe oder von einem System in Ihrer Verbindungsliste starten. Dazu aber in einem späteren Kapitel mehr. Schauen wir uns zunächst einmal ‚*Management Central*’ des iSeries Navigators an.



iSeries Navigator Management Central

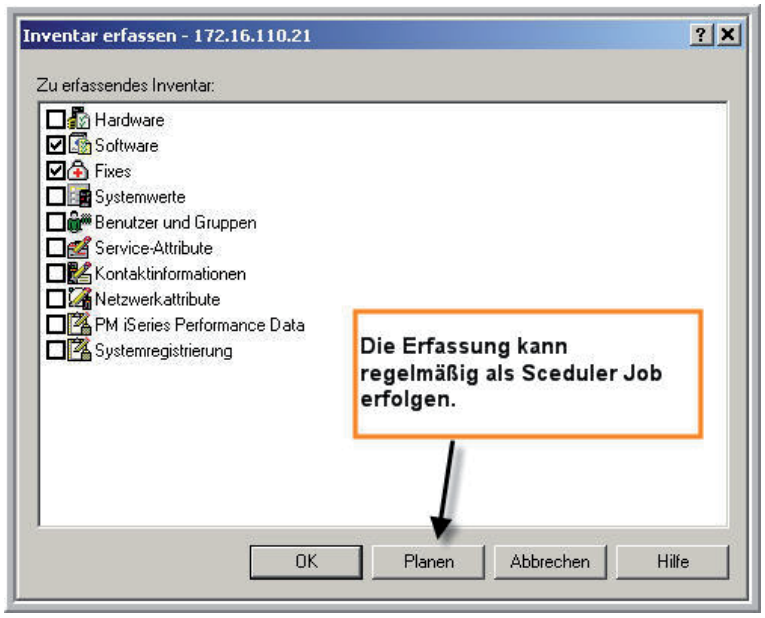
Voraussetzung für die Fixverwaltung mit ‚*Management Central*’ ist die Erfassung eines Fixinventars. Das Inventar wird auf dem zentralen System für die Endpunktsysteme gespeichert. Sie arbeiten also mit einer Kopie der Originaldaten. Wenn Sie jetzt das Inventar erfassen wollen, gehen Sie wie folgt vor:

2.9.2.2
Seite 2

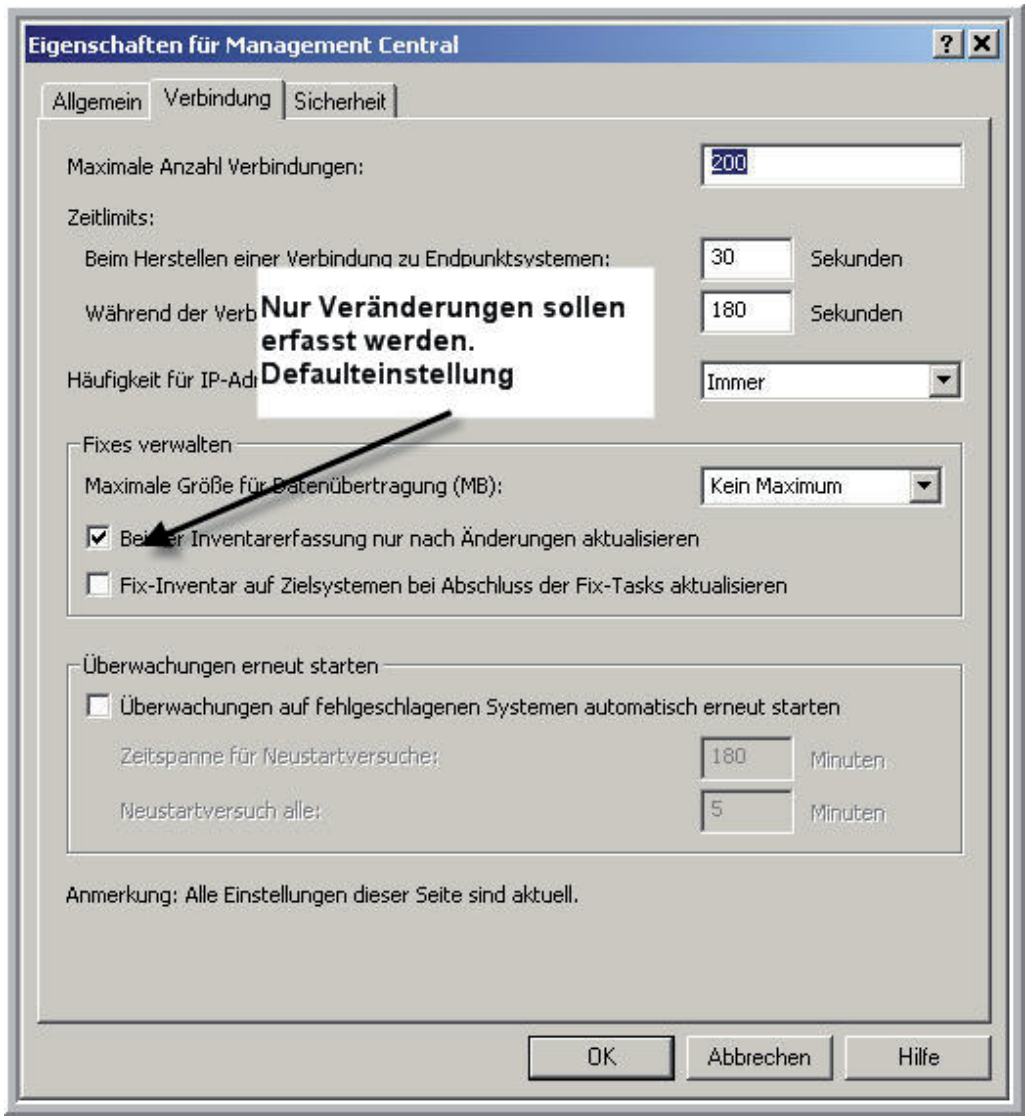


Fixinventar erfassen

Anschließend erhalten Sie ein weiteres Fenster.



Sie sehen, dass Sie das Fixinventar nur gemeinsam mit dem Softwareinventar erfassen können. Außerdem wird das Inventar automatisch für die Fixgruppen erfasst. Sie können die Erfassung dahingehend beschränken, dass die Erfassung nur gestartet wird, wenn sich die Fixdaten geändert haben. Fixgruppen werden hingegen immer erfasst. Diese Einstellungen werden auf der Registerkarte ‚*Verbindung*' im Management Central-Eigenschaftenfenster festgelegt.



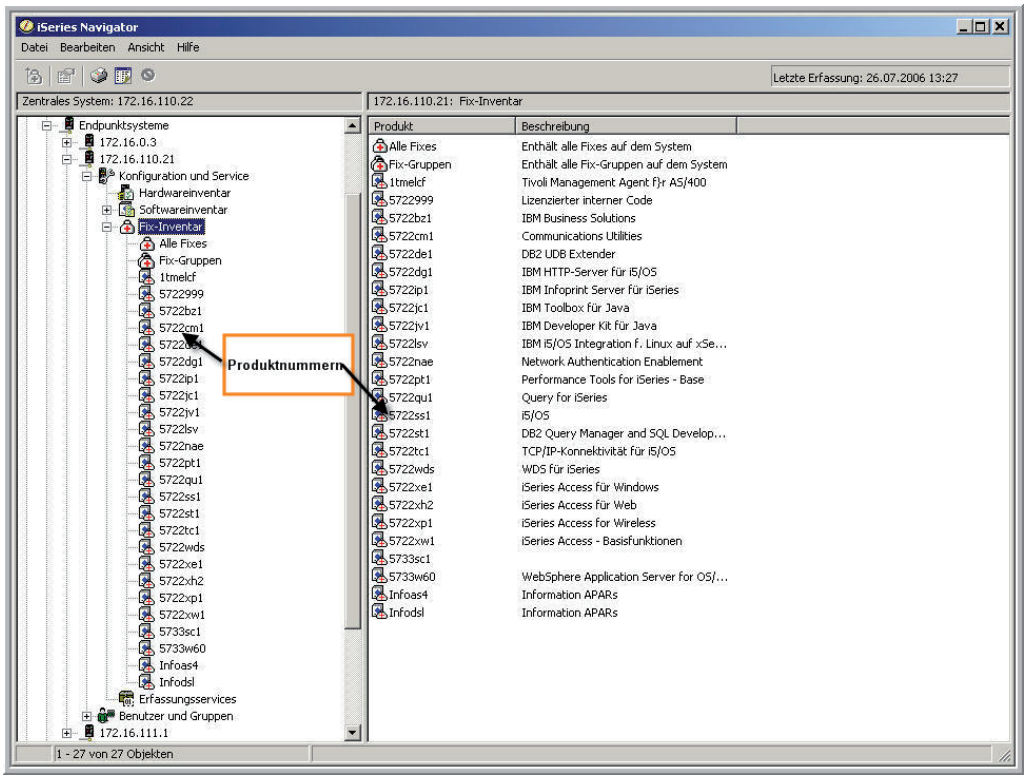
Eigenschaften der Inventarisierung

2.9.2.2
Seite 4

Sobald Ihnen die Inventarliste zur Verfügung steht, können Sie die folgenden Aktionen ausführen:

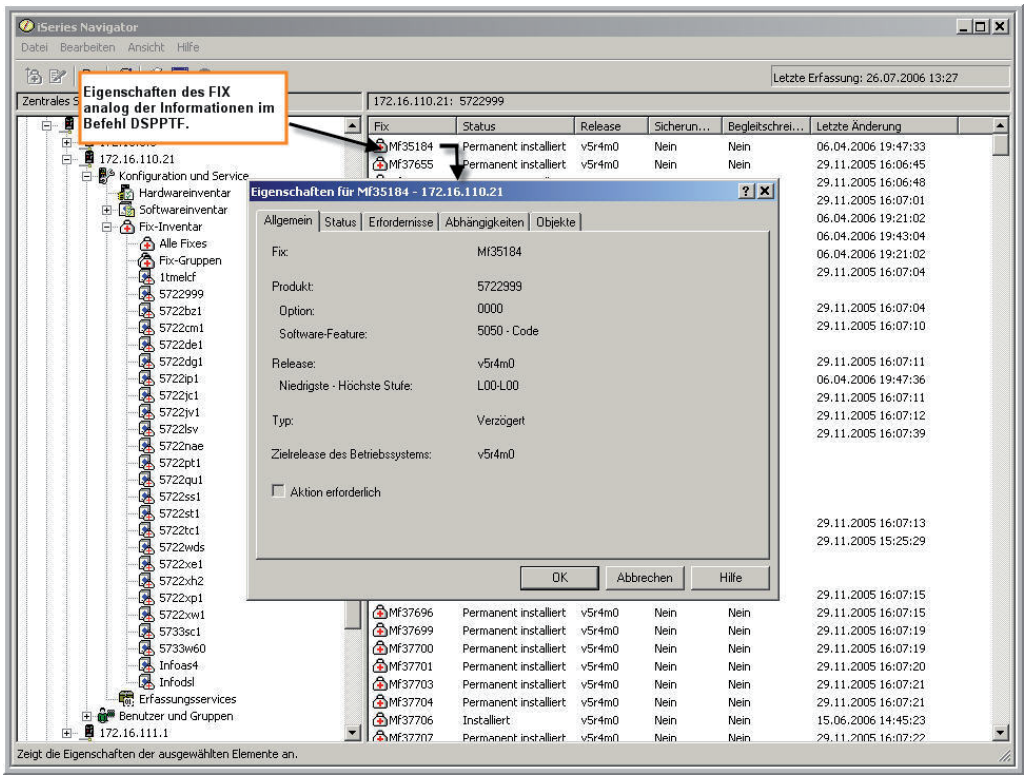
- Assistenten zum Senden und Installieren von Fixes nutzen.
- Fixes permanent installieren.
- Fixes deinstallieren.
- Sicherungsdateien und Begleitschreiben löschen.
- Andere erweiterte Funktionen ausführen, z. B. Fixaktionen abbrechen.
- Zeitplan für die Ausführung dieser Aktionen erstellen.
- Fixes von Datenträgern kopieren.

Die Fixverwaltungsfunktion von iSeries Navigator gestattet Ihnen somit eine sehr einfache Verwaltung Ihrer Fixes. Dazu aber in einem der folgenden Beiträge mehr. Wie wir gesehen haben ist als Voraussetzung ein Inventar notwendig. Also müssen Sie darauf achten, dass das Inventar stets auf dem neuesten Stand ist, damit Sie keine veralteten Informationen erhalten. Nutzen wir Management Central jetzt, um uns einen Überblick über die aktuell installierten Fixes zu verschaffen:



Um die oben dargestellten Informationen zu erhalten, müssen Sie ‚*Management Central*' öffnen. Anschließend öffnen Sie das Endpunktsystem, für das Sie den Status anzeigen wollen. Im Menüpunkt „Konfiguration und Service" finden Sie das Fix-Inventar.

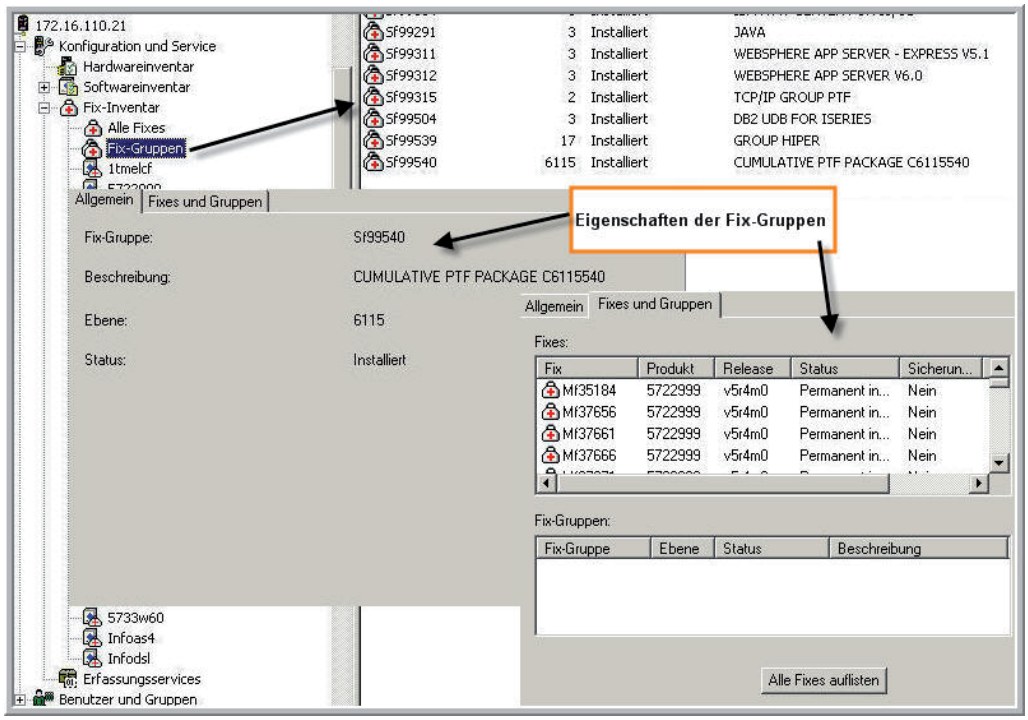
In iSeries Navigator werden natürlich auch zusätzliche Informationen angezeigt. Es gibt mehrere Eigenschaftsfenster, die Ihnen Informationen zu Fixes bieten, die denen sehr ähnlich sind, die nach Eingabe des Befehls DSPPTF angezeigt werden. Sie können die Fix-ID, das zugehörige Produkt, Release und den Status anzeigen und ob ein Begleitschreiben oder eine Sicherungsdatei auf dem System vorhanden ist. Außerdem werden alle vorausgesetzten und zugehörigen Fixes angezeigt. Schließlich sehen Sie die Objekte in dem Fix sowie den Namen der Bibliothek, in der sich die Objekte nach der Installation des Fixes befinden.



FIX Eigenschaften anzeigen

Wenn Sie die Fixgruppen auswählen, erhalten Sie zunächst alle Fixgruppen mit dem aktuellen Level. Über die Kontexttaste (rechte Maustaste) können Sie die Eigenschaften anzeigen.

2.9.2.2
Seite 6



FIX-Gruppen und deren Eigenschaften.

Erläuterung der PTF-Stati

Wir haben gesehen, dass der iSeries Navigator eine grafische Benutzerschnittstelle für die Verwaltung Ihrer Fixes und Fixgruppen bietet. Viele der iSeries Statusbedingungen sind mit denen identisch, die der Befehl DSPPTF (PTF anzeigen) anzeigt. Einige weichen jedoch ab. Die folgende Tabelle enthält eine Beschreibung der Fix- und Fixgruppenstatuswerte, die in „Management Central" verfügbar sind.

PTF-Status		
Management Central-Status	Beschreibung	DSPPTF-Status
Verfügbar	Der Fix ist für die Installation auf dem lokalen System bereit. Er kann als Sicherungsdatei vorhanden sein oder wurde mit dem Befehl LODPTF geladen, aber ist noch nicht oder er wurde vorübergehend entfernt. Damit der Fix an andere Systeme verteilt werden kann, muss die Sicherungsdatei auf dem System vorhanden sein.	Nur Sicherungsdatei oder Nicht angelegt
PTF-Begleitschreiben	Das Begleitschreiben für den Fix ist auf dem System, der Fix selbst jedoch nicht.	Nur Begleitschreiben
Beschädigt	Der Fix ist beschädigt. Wenn die Sicherungsdatei verfügbar ist, können Sie den Fix erneut deinstallieren oder installieren. Ist die Sicherungsdatei nicht verfügbar, müssen Sie sie besorgen und den Fix erneut installieren oder deinstallieren.	Beschädigt
Beim nächsten Neustart installieren	Der Fix wird beim nächsten Neustart des Systems installiert.	Nicht angelegt mit Angabe „Ja" für IPL-Aktion
Beim nächsten Neustart permanent installieren	Der Fix wird beim nächsten Neustart des Systems permanent installiert.	Vorübergehend angelegt mit Angabe „Ja" für IPL-Aktion oder Nicht angelegt mit Angabe „Ja" für IPL-Aktion
Installiert	Der Fix ist installiert. Sie können ihn im System deinstallieren oder permanent installieren. Der Fix ist kein permanenter Bestandteil des Systems.	Vorübergehend angelegt

2.9.2.3
Seite 2

PTF-Status		
Management Central-Status	Beschreibung	DSPPTF-Status
Permanent installiert	Der Fix ist permanent installiert. Sie können ihn nicht deinstallieren. Wenn ein Fix permanent installiert ist, bedeutet das, dass Sie nicht mehr auf die alten Objekte zurückgreifen können. Der Fix ist jetzt ein permanenter Bestandteil des Systems.	Permanent angelegt
Permanent installiert – Aktion anstehend	Der Fix ist permanent installiert, Sie müssen jedoch eine Aktion ausführen, damit der Fix aktiv wird. Lesen Sie die erforderlichen Aktionen im Begleitschreiben nach. Wenn Sie die erforderlichen Aktionen zur Aktivierung des Fixes ausgeführt haben, müssen Sie das System jetzt nicht erneut starten. Der Status „Aktion anstehend" wird beim nächsten Systemstart aktualisiert.	Permanent angelegt – PND oder Permanent angelegt – ACN
Installiert – Aktion anstehend	Der Fix ist installiert, Sie müssen jedoch eine Aktion ausführen, damit der Fix aktiv wird. Lesen Sie die erforderlichen Aktionen im Begleitschreiben nach. Wenn Sie die erforderlichen Aktionen zur Aktivierung des Fixes ausgeführt haben, müssen Sie das System jetzt nicht erneut starten. Der Status „Aktion anstehend" wird beim nächsten Systemstart aktualisiert.	Vorübergehend angelegt – PND oder Vorübergehend angelegt – ACN
Bestellt	Der Fix ist bestellt, aber noch nicht auf dem System eingetroffen.	Angefordert
Überlagert	Ein neuerer Fix ist auf dem System vorhanden, aber möglicherweise nicht angelegt.	Überlagert

PTF-Status		
Management Central-Status	Beschreibung	DSPPTF-Status
Nur unterstützt	Der Fix ist als Sicherungsdatei auf dem System vorhanden, kann aber nicht auf dem System installiert werden. Ein Fix mit dem Status „Nur unterstützt" kann nur an andere Systeme verteilt und dort installiert werden.	Nur Sicherungsdatei
Beim nächsten Neustart deinstallieren	Der Fix wird beim nächsten Neustart des Systems deinstalliert.	Vorübergehend angelegt mit Angabe „Ja" für IPL-Aktion oder Nicht angelegt mit Angabe „Ja" für IPL-Aktion
Beim nächsten Neustart permanent deinstallieren	Der Fix ist installiert oder verfügbar und wird beim nächsten Neustart des Systems permanent deinstalliert.	Vorübergehend angelegt mit Angabe „Ja" für IPL-Aktion oder Nicht angelegt mit Angabe „J" für IPL-Aktion
Permanent deinstalliert – Aktion anstehend	Der Fix ist permanent deinstalliert, Sie müssen jedoch eine Aktion ausführen, damit der Fix nicht mehr aktiv ist. Lesen Sie die erforderlichen Aktionen im Begleitschreiben nach. Wenn Sie die erforderlichen Aktionen ausgeführt haben, müssen Sie das System jetzt nicht erneut starten. Der Status „Aktion anstehend" wird beim nächsten Systemstart aktualisiert.	Permanent entfernt – PND oder Permanent entfernt – ACN
Deinstalliert – Aktion anstehend	Der Fix ist deinstalliert, Sie müssen jedoch eine Aktion ausführen, damit der Fix nicht mehr aktiv ist. Lesen Sie die erforderlichen Aktionen im Begleitschreiben nach. Wenn Sie die erforderlichen Aktionen ausgeführt haben, müssen Sie das System jetzt nicht erneut starten. Der Status „Aktion anstehend" wird beim nächsten Systemstart aktualisiert.	Vorübergehend entfernt – PND oder Vorübergehend entfernt – ACN

Fixgruppenstatus		
Management Central-Status	Beschreibung	WRKPTFGRP-Status
Nicht gültig	Alle Fixes der Fixgruppe und alle zugehörigen Fixgruppen gelten für Produkte, die nicht installiert sind und auf diesem System nicht unterstützt werden. Für die angegebenen Fixes gibt es keine Aktionen.	Nicht gültig
Nur unterstützt	Keiner der Fixes in der Fixgruppe oder in einer der zugehörigen Fixgruppen gilt für Produkte, die auf diesem System installiert sind. Mindestens ein Fix gilt jedoch für ein Produkt, Release, eine Option und Lade-ID, das bzw. die auf diesem System unterstützt wird.	Nur unterstützt
Nicht installiert	Mindestens ein Fix in der Fixgruppe oder in mindestens einer der zugehörigen Fixgruppen gilt für ein Produkt, das auf diesem System installiert ist, und weder der Fix noch ein überlagernder Fix ist installiert oder permanent installiert.	Nicht installiert
Installiert	Alle Fixes in der Fixgruppe oder in einer der zugehörigen Fixgruppen für ein Produkt, das auf diesem System installiert ist, sind installiert, permanent installiert oder überlagert. Wenn ein Fix überlagert wird, wird der überlagernde Fix entweder installiert oder permanent installiert.	Installiert
Fehler	Die Fixgruppeninformationen sind fehlerhaft. Löschen Sie die Fixgruppe mit dem Befehl WRKPTFGRP (Mit PTF-Gruppen arbeiten) oder ersetzen Sie die Fixgruppeninformationen, die momentan auf dem System vorhanden sind.	Fehler

Fixgruppenstatus		
Management Central-Status	Beschreibung	WRKPTFGRP-Status
Nicht gefunden	Zeigt an, dass die zugehörige Fixgruppe, die durch die Fixgruppe angegeben wird, für die sie Eigenschaften anzeigen, auf dem System nicht gefunden wird. Dieser Status wird nur angezeigt, wenn Sie die Eigenschaften einer zugehörigen Fixgruppe anzeigen. Wenn Sie das Fixinventar anzeigen, sehen Sie diesen Status nicht.	Nicht gefunden
Unbekannt	Der Fixgruppenstatus kann nicht aufgelöst werden, weil die Fixgruppe mindestens eine zugehörige Fixgruppe angibt, die auf dem System nicht gefunden wird oder fehlerhaft ist. Für die angegebenen Fixes gibt es keine Aktionen, weil sie nicht für alle Fixes, die die Fixgruppe angibt, erfolgreich ausgeführt werden können.	Unbekannt

2.9.2.3
Seite 6

Informationen über veröffentlichte IBM Fixes finden

Jetzt müssen wir aber noch auf dem Laufenden bleiben: Welche Fixes stehen seitens der IBM zur Verfügung, was können bestimmte Fixes in der Umgebung tun und welche Fixes werden nicht benötigt? Erst wenn wir Zugang zu diesen Informationen haben, können wir entscheiden, ob die entsprechenden Fixes für unser System nützlich sind, ob sie bestellt und anschließend installiert werden sollen.

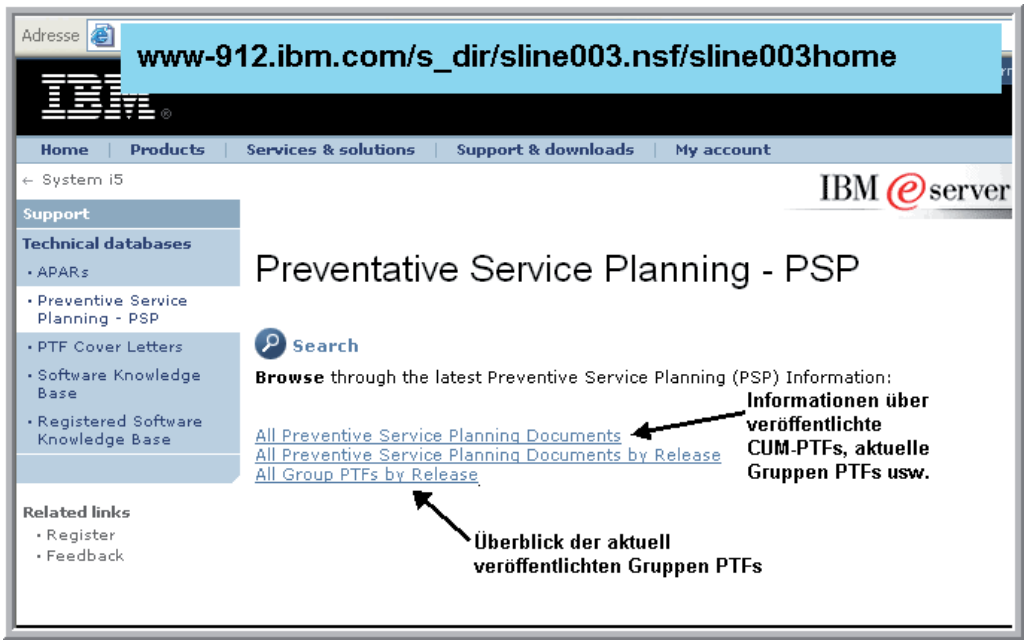
PSP-Informationen (PSP = Preventive Service Planning, Planung für vorbeugenden Service) enthalten die neuesten Informationen, die Ihre Hardware oder Software betreffen. Sie sollten sich daran gewöhnen die PSP-Informationen regelmäßig zu überprüfen. Eine effektive Verwendung der PSP-Informationen ist ein wichtiges Element Ihrer Fixverwaltungsstrategie. Sie sollten diese Informationen aber spätestens lesen, bevor Sie Softwareinstallationen vornehmen, um über Bedingungen informiert zu sein, die Ihren Softwareinstallationsprozess behindern könnten.

8. Ergänzung 09/2006

2.9.3 Seite 2

PSP-Informationen nutzen

Am einfachsten ist es die PSP-Website direkt aufzurufen oder auf die Homepage für IBM eServer iSeries Support zuzugreifen. Wenn Sie diese Website aufrufen, finden Sie die PSP-Informationen in der Kategorie „Technical Databases“.



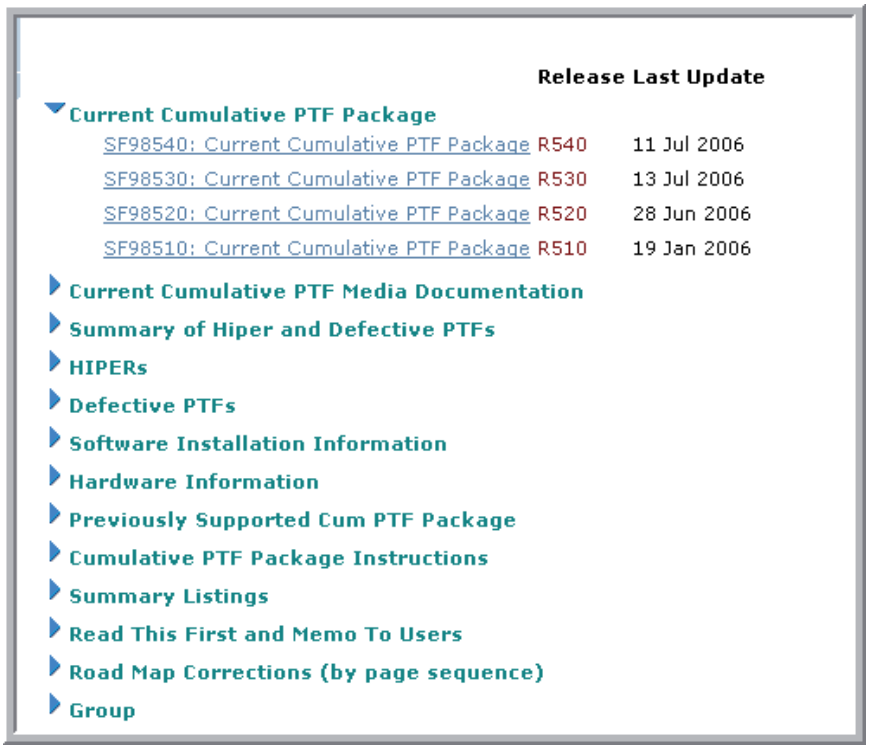
Homepage Preventive Service Planning (PSP)

Wählen Sie den Punkt „All Preventive Service Planing Documents“. Nachdem Sie Ihr Realease gewählt haben, erhalten Sie eine Übersicht aller wichtigen Informationen bzgl. der PTF-Pflege.

2.9.3.1
Seite 2

PSP-Kategorien

Hier finden Sie auch die aktuell veröffentlichten kumulativen PTF-Pakete mit entsprechenden Informationen und können prüfen, ob Ihr System auf einem aktuellen Stand ist.

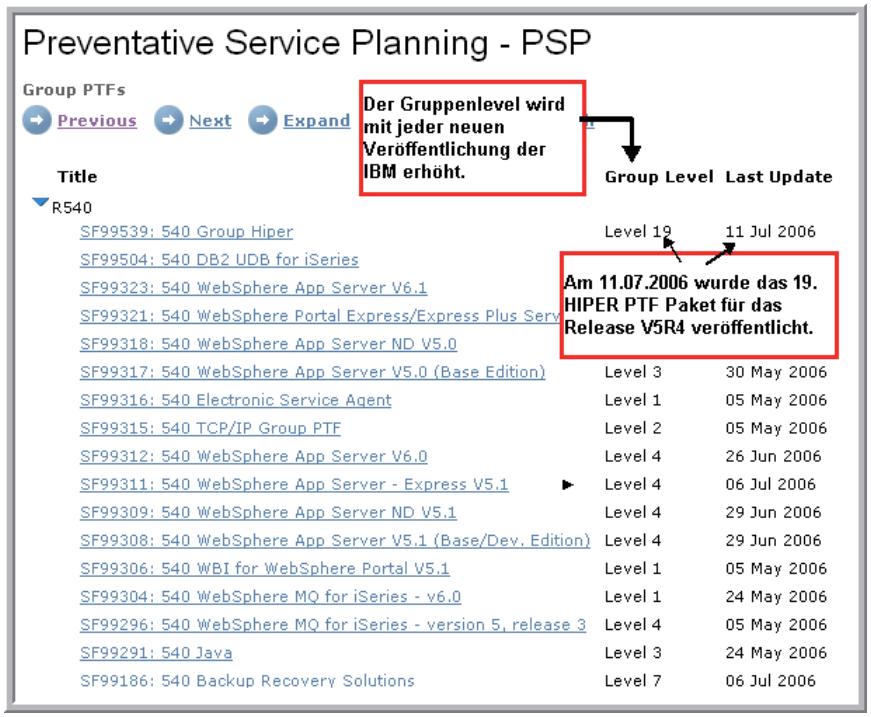


Informationen über aktuelle kumulative PTF-Pakete anzeigen

Darüber hinaus bietet Ihnen diese Internetseite eine Vielzahl von Informationen. So finden Sie ausführliche Informationen über fehlerhafte PTFs, aktuell veröffentlichte Hiper PTFs u.v.m.

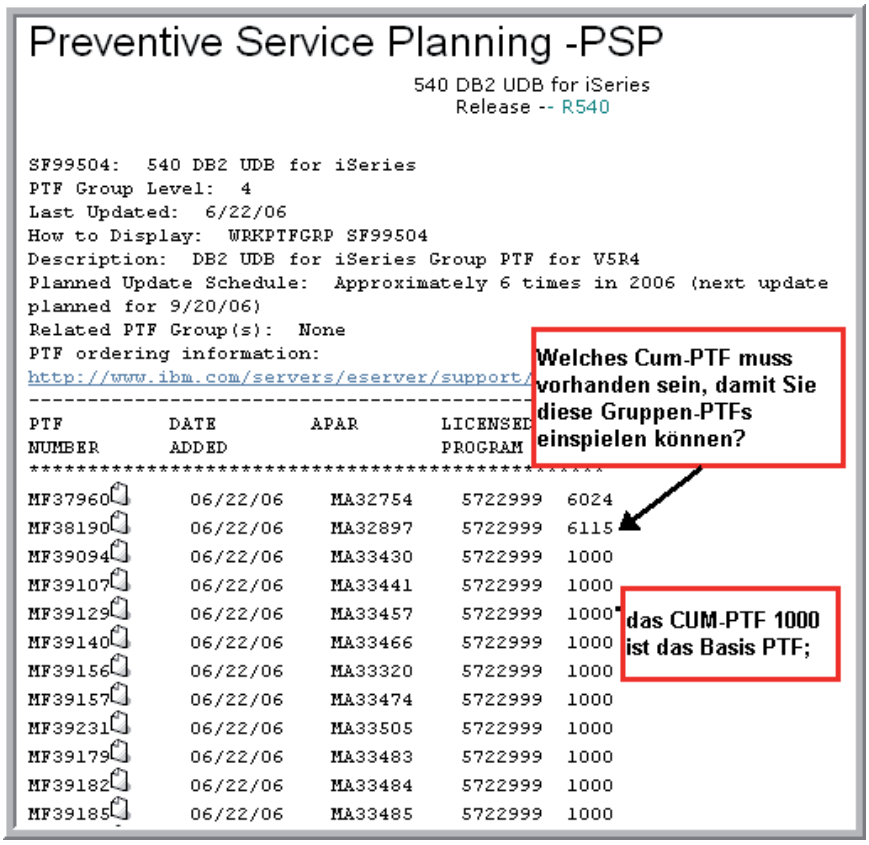
2.9.3.1
Seite 4

Die PSP-Homepage ist aber auch sehr hilfreich, um zu verfolgen, welche Gruppen-PTFs aktuell veröffentlicht sind. Dafür nutzen Sie den Punkt „All Group PTFs by Release“. Hier erhalten Sie detaillierte Informationen über den Gruppenlevel. Sie haben vielleicht bereits bemerkt, dass die PTF-Nummer konstant bleibt, während der Level mit jeder Aktualisierung erhöht wird.



Veröffentlichte Gruppen-PTFs

Natürlich können Sie sich auch anzeigen lassen, welche Einzelfixes Bestandteil der Gruppe sind und welche Voraussetzungen erfüllt sein müssen, um das Gruppen-PTF einzuspielen:



Detaillierte Informationen zum Gruppen-PTF

Jetzt verfügen Sie über alle nötigen Informationen, um zu entscheiden, ob und wann Sie die PTFs bestellen. Wie Eingangs erwähnt beheben die PTFs nicht nur Fehlersituationen, sondern bringen auch neue Funktionen auf Ihr System. Die Zyklen für neue PTFs sind mittlerweile stark verkürzt worden und die alte Weisheit: „Never Change a Running System“ sollte daher vielleicht einmal überprüft werden?

2.9.3.1
Seite 6

PTF-Bestellung

Gehen wir davon aus, dass Sie sich über den aktuellen Fixstatus auf Ihrem iSeries Server informiert haben und neue Fixes bestellen wollen. Sie haben dazu mehrere Möglichkeiten:

- Sie können sich an Ihren zuständigen IBM-Vertriebsmitarbeiter wenden,
- Ihr ECS-Modem verwenden
- oder das Internet nutzen.

Anhand der Informationen im nachfolgenden Kapitel werden Sie feststellen, welche Methode für Ihre individuelle Umgebung am besten geeignet ist.

2.9.4
Seite 2

PTFs mit dem Befehl SNDPTFORD bestellen

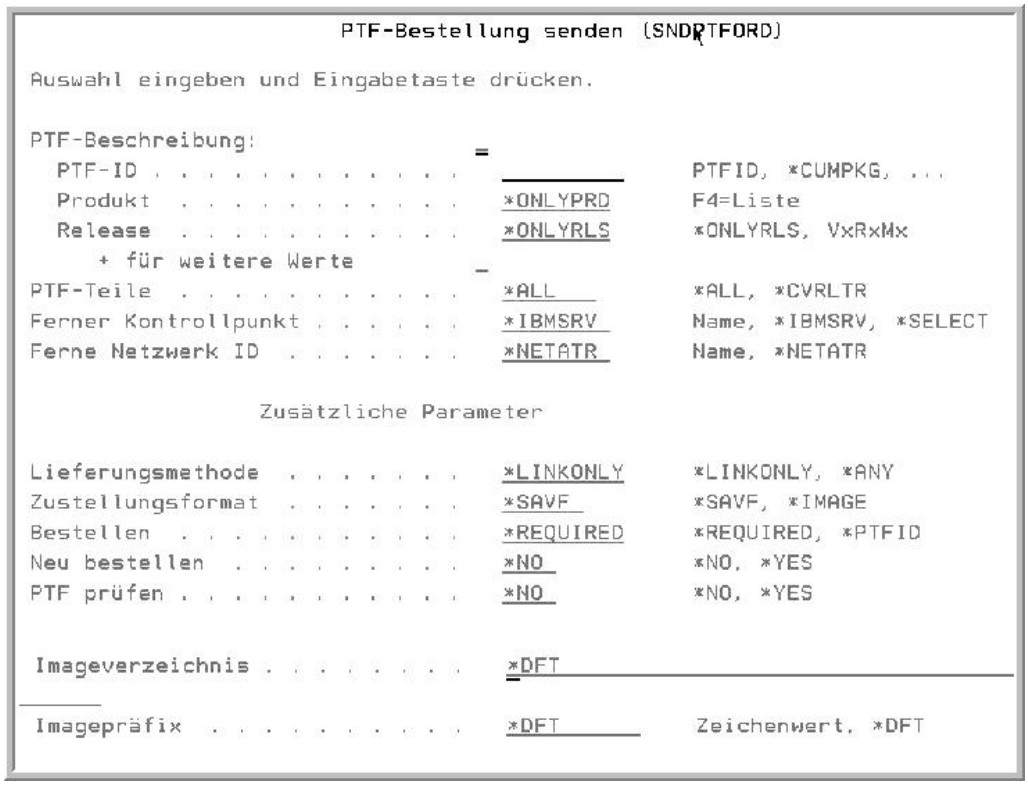
Mit dem Befehl SNDPTFORD (send PTF Order) bestellen Sie PTFs über die Konfiguration des Electronic Customer Support (ECS). Bevor Sie einzelne Fixes bestellen und anlegen, sollten Sie stets das aktuelle CUM-PTF anlegen. Dadurch verringert sich die Anzahl der Fixes, die Sie später einzeln bestellen und ggf. verteilen müssen. Lassen Sie uns daher zunächst einmal ein kumulatives PTF-Paket mit dem Befehl SNDPTFORD über die zeichenorientierte Schnittstelle ordern:

Wenn Sie die zeichenorientierte Schnittstelle bevorzugen, geben Sie den Befehl SNDPTFORD ptf-name ein, um einzelne Fixes, mehrere Fixes, PTF-Gruppen und Übersichten zu erhalten. PTFs werden mit Hilfe von zwei Zeichen dargestellt, auf die fünf Ziffern folgen. Mit diesem Schema können alle PTFs, die Zusammenfassungen und auch die kumulativen PTF-Pakete dargestellt werden:

MF98vrm	Paket für Hardwareserviceplanung
SF98vrm	Paket für die Planung des vorbeugenden Service
SF99vrm	Kumulatives PTF-Paket
SF99xxx	PTF-Gruppe
SF97vrm	PTF-Übersichtspaket

Die Kennung „vrm“ gibt hierbei stets die Version, Release und Modifikationsstufe des Betriebssystems an, für das Sie die PTFs bestellen. Um das kumulative PTF-Paket zu bestellen, müssen Sie also zunächst einmal den Release-Stand Ihres Servers kennen. Auf unserem Server ist das Release V5R4M0 installiert. Wenn wir jetzt das kumulative PTF-Paket bestellen wollen, müssen wir den Befehl SNDPTFORD eingeben:

2.9.4.1
Seite 2



Bedienerführung des Befehls SNDPTFORD

Falls Sie vorhaben einzelne PTFs zu ordern, reicht es, wenn Sie im Befehl SNDPTFORD die entsprechende PTF-Nummer eingeben. Maximal können 20 Einzel-PTFs bestellt werden. Sie können PTFs mit Begleitschreiben oder nur die Begleitschreiben bestellen, diese dann durchsehen und sich danach entscheiden, ob Sie die jeweiligen PTFs benötigen oder nicht. Alle vorausgesetzten PTFs erhalten Sie automatisch. Lassen Sie sich also nicht verwirren, wenn Sie nur ein PTF bestellen und dann drei oder vier PTFs erhalten.

Wenn Sie anschließend die Eingabetaste drücken, erscheint ein Bildschirm, der es Ihnen ermöglicht, Ihre Kontaktinformationen zu überprüfen und natürlich ggf. zu verändern.

Es ist auch möglich, die Bestellung im Stapelbetrieb zu verarbeiten. Dafür geben Sie den Befehl SNDPTFORD im Parameter CMD des Befehls SBMJOB ein. Anschließend wird Ihre Bestellung an die IBM gesendet. Dort wird entschieden, ob die Fixes elektronisch oder auf Datenträgern verteilt werden. IBM wird immer versuchen, Ihnen die PTFs über die ECS-Verbindung zuzusenden. Falls Sie die Fixes über die Serviceleitung erhalten, haben Sie die Auswahl, die Fixes als Sicherungsdateien oder als optische Imagedateien zu empfangen. Der Dateityp wird durch die Angabe im Parameter DLVRYFMT (Zustellungsformat) bestimmt. Für DLVRYFMT(*IMAGE) wird allerdings vorausgesetzt, dass in Ihrer Systemkonfiguration die Verwendung von Universal Connection definiert ist. Sicherungsdateien werden in der Bibliothek QGPL gespeichert. Meistens ist der Name der Sicherungsdatei die Fix-ID mit dem Präfix Q. Nachdem die PTFs geladen sind, sind die Sicherungsdateien eigentlich überflüssig. Mit dem Befehl DLTPTF oder dem Befehl WRKF können Sie die Sicherungsdateien und deren Begleitschreiben löschen.

Optische Imagedateien enthalten PTFs und Begleitschreiben und werden in dem Verzeichnis gespeichert, das Sie im Parameter IMGDIR (Imageverzeichnis) angegeben haben.

Falls Sie Management Central des Operation Navigators einsetzen oder zukünftig einsetzen wollen, um Ihre Fixes zu verwalten, sollten Sie darauf achten, dass Sie den Befehl SNDPTFORD stets auf dem System eingeben, das Sie als Quellensystem beim Vergleichen und Aktualisieren der Fixversionen verwenden.

Was passiert aber, wenn die Bestelldaten für die ECS-Verbindung zu umfangreich sind? Falls Sie *ANY für den Parameter DELIVERY METHOD angegeben haben, erhalten Sie in diesem Fall die Bestellung auf Datenträgern per Post zu gestellt. Wenn Sie dagegen DELIVERY(*LINKONLY) angeben, kann es unter Umständen passieren, dass Sie keine PTFs erhalten, da die Größe Ihrer Fixes die Übertragungsbegrenzung der Serviceverbindung überschreitet.

Und was ist zu tun, wenn Ihre PTFs verloren gehen? Normalerweise werden Sie bereits bestellte PTFs kein zweites Mal erhalten. Wenn Sie aber die PTFs – aus welchen Gründen auch immer – erneut benötigen, setzen Sie bei Ihrer Bestellung den Parameter REORDER auf *YES.

2.9.4.1
Seite 4

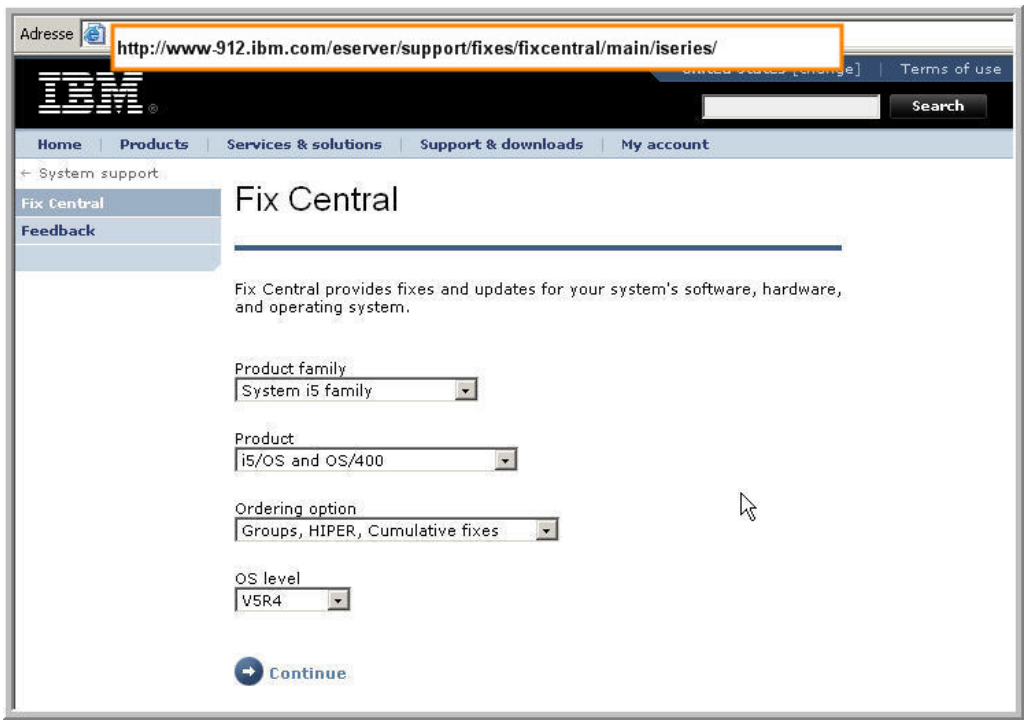
Sie können aber mit dem Befehl SNDPTFORD nicht nur kumulative PTF-Pakete bestellen, sondern eine Reihe von Fixes mit den dazugehörigen Informationen. Die nachfolgende Tabelle zeigt Ihnen, wie mit dem Befehl SNDPTFORD die verschiedenen Arten der Fixes und Fixinformationen bestellt werden.

Fixinformationen	Befehl
Spezifische Fixes und Begleitschreiben	SNDPTFORD nnnnnnn (nnnnnn ist die FIX-ID)
Nur Fix-Begleitschreiben	SNDPTFORD nnnnnnn PTFPART(*CVRLTR)
Kumulative PTF-Pakete	SNDPTFORD SF99vrm (v = Version, r = Release, m = Modifikation)
PTF-Gruppe	SNDPTFORD SF99nnn (nnn = Gruppen-ID)
PTF-Übersichtsliste	SNDPTFORD SF97vrm (v = Version, r = Release, m = Modifikation)
Fix-Querverweisübersichtsliste Betriebssystem VRM510 bis VRM530 Betriebssystem VRM520 bis VRM530 Lizenzierter interner Code V5R3M0 bis lizenzierter interner Code V5R3M54	– SNDPTFORD SF97071 SNDPTFORD SF97072 SNDPTFORD SF97081 SNDPTFORD SF9706
PSP-Informationen für Lizenzprogramme	SNDPTFORD SF98vrm (v = Version, r = Release, m = Modifikation)
PSP-Informationen für lizenzierten internen Code und Hardwareeinheiten	SNDPTFORD MF98vrm (v = Version, r = Release, m = Modifikation)

Fixes über das Internet bestellen

Die schnellste Methode Fixes zu bestellen, haben Sie im Internet. Doch häufig werden PTFs immer noch telefonisch bei der IBM bestellt, dann in Dänemark auf CD-Rom gebrannt und anschließend an Sie verschickt. Gerade bei akuten Problemen können Sie sich hier wertvolle Zeit sparen. Die Voraussetzungen für die Bestellung über das Internet sind sehr gering. Sie benötigen lediglich einen Internetanschluss mit einer Bandbreite größer als 56 k und eine IBM Benutzer-ID und ein IBM-Kennwort. Warten Sie mit der IBM-Registrierung und der Konfiguration dieser Funktion nicht bis zu einem Notfall, bei dem Sie einen Fix schnell anfordern müssen. Seien Sie vorbereitet: Führen Sie Registrierung und Konfiguration zunächst einmal Test halber aus und fordern Sie einen PTF an, um den Prozess im Internet kennen zu lernen.

Zunächst einmal rufen Sie die Startseite von Fix Central auf.



Fix Central Startseite

Hier können Sie für verschiedene Server und deren Produkte Fixes downloaden. Für i5/OS und OS/400 sind die Fixes derzeit zurück bis zum Release V5R1 verfügbar. Wenn Sie den Bestellvorgang jetzt fortsetzen, werden Sie aufgefordert sich anzumelden. Spätestens jetzt benötigen Sie die gültige IBM-ID.

2.9.4.2
Seite 2

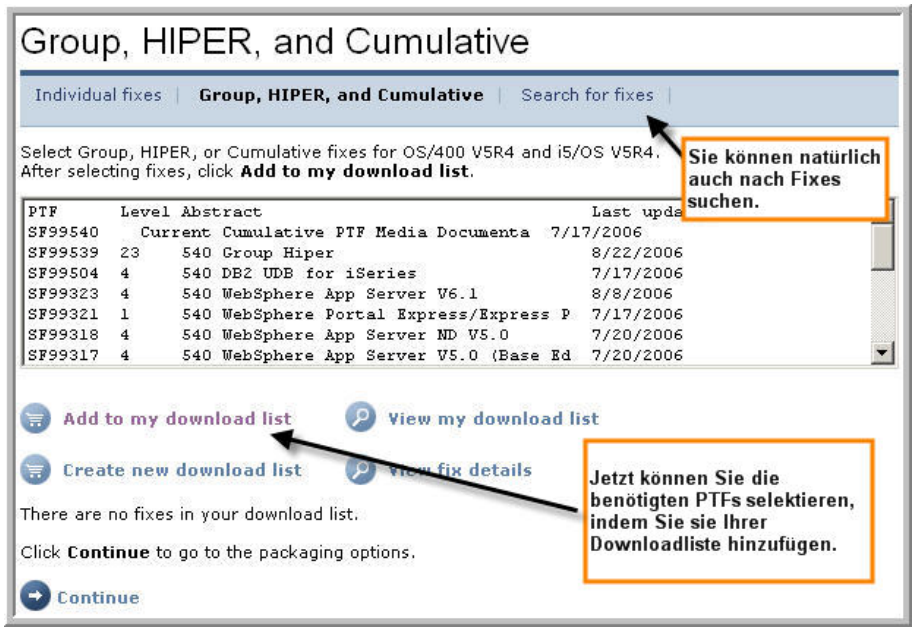
Die Registrierung

Falls Sie sich noch nicht registriert haben, sollten Sie es jetzt tun. Die Registrierung ist ganz einfach:

Sie geben als IBM-ID Ihren zukünftigen Benutzernamen an. Der Benutzername muss dem Aufbau einer gültigen Email-Adresse entsprechen und darf maximal 80 Zeichen lang sein. Der Benutzername kann später nicht mehr geändert werden. Aber keine Sorge, es ist nicht schlimm, wenn Ihre tatsächliche e-Mail Adresse von dem Benutzernamen abweicht, denn Sie müssen ohnehin die e-Mail Adresse separat angeben und die ist dann auch später jederzeit änderbar. Zusätzlich benennen Sie ein Passwort, das aus mindestens 8 Zeichen besteht und geben als Land „Germany“ an. Auf der zweiten Seite müssen Sie jetzt noch Ihre Adressdaten angeben und das war's. Sobald Sie die Registrierung durchgeführt haben, können Sie sich entsprechend anmelden.

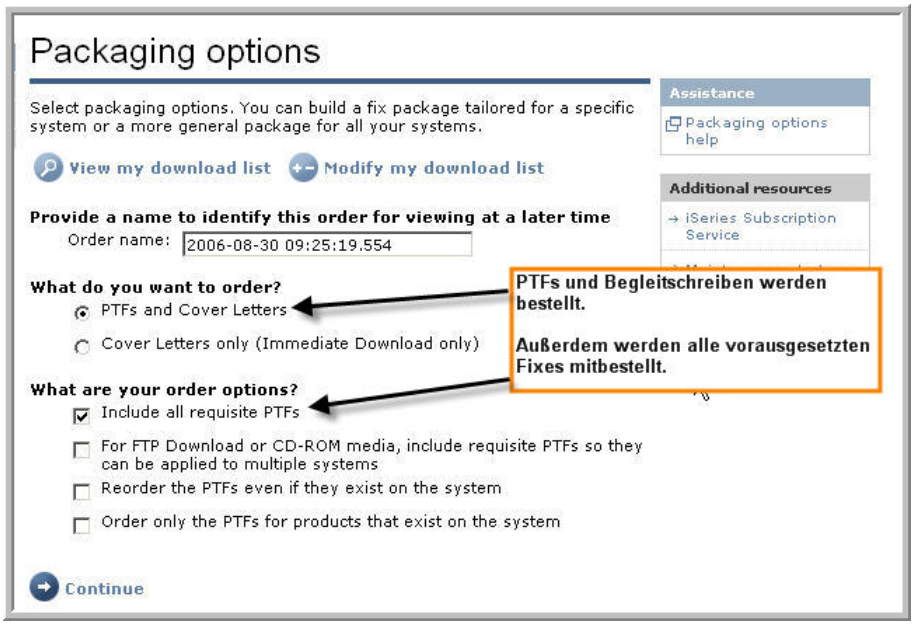
Die Anmeldung

Anschließend müssen Sie die Fixes auswählen, die Sie benötigen. Dabei können Sie nicht nur nach den Fixes suchen, sondern Sie können sich an dieser Stelle auch noch mal die detaillierten Informationen zu den Fixes anschauen.



Fixes selektieren

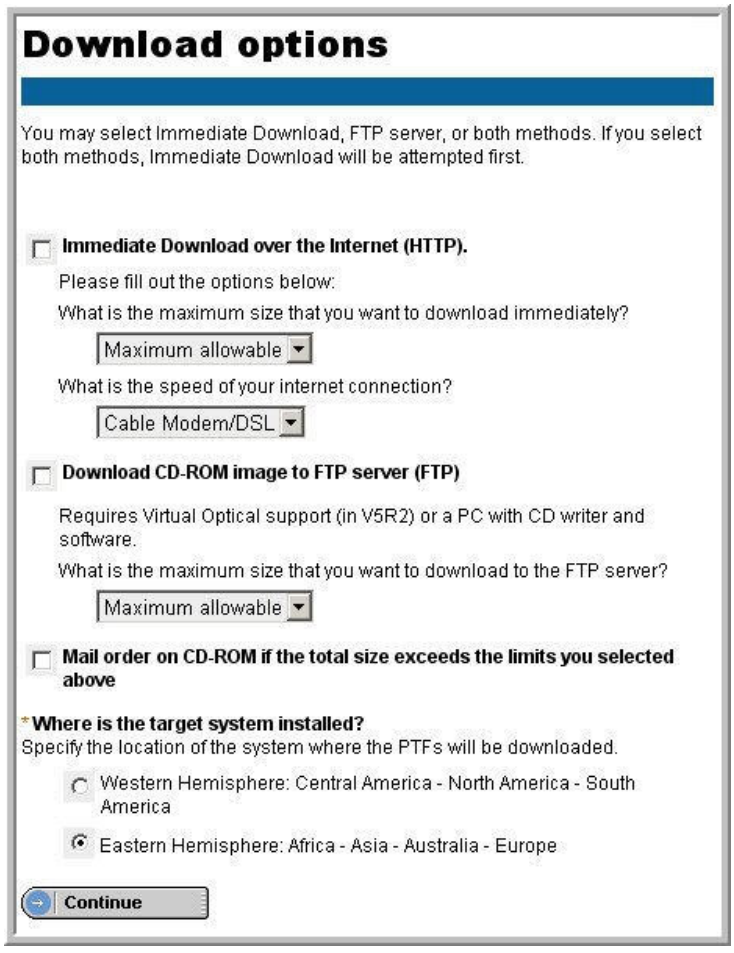
Wenn Sie Ihre Downloadliste zusammengestellt haben, wählen Sie „Continue“. Auf der nächsten Seite müssen Sie die Packaging-Optionen einstellen. Hier finden Sie alle Informationen analog dem Befehl SNDPTFORD.



Die Bestelloptionen

2.9.4.2
Seite 4

Ähnlich dem SNDPTFORD-Befehl können Sie die Fixes mit Begleitschreiben oder auch nur die Begleitschreiben bestellen. Zusätzlich entscheiden Sie, ob alle vorausgesetzten Fixes mitgeliefert werden und falls Sie die PTFs ein zweites Mal bestellen, müssen Sie die Reorder-Option wählen. Auf der nächsten Seite geben Sie dann noch die Download-Optionen an.



Die Optionen für den Download

Falls Sie einen sofortigen Download beabsichtigen, sollten Sie die Datenmenge für den Transfer beschränken. Unterschätzen Sie nicht die zu übertragende Datenmenge. Dadurch, dass meistens alle vorausgesetzten Fixes mitbestellt werden, können schnell mehrere MB an Datenvolumen zusammenkommen. Falls die Maximalgrenze erreicht wird, steht als Alternative immer noch der Transfer mittels FTP oder der Versand per CD-ROM zur Verfügung. Sie dürfen nur nicht vergessen, diese Alternativen zu selektieren.

„Download CD ROM Images to the FTP Server" ist somit eine Alternative zum direkten Download auf Ihren Server. Auf dem FTP-Server werden Ihre PTFs als Image Files abgelegt. Sie benötigen also mindestens einen PC mit CD-Brenner oder Ihr Release muss virtuelle optische Laufwerke unterstützen. Hierzu kommen wir sicherlich noch zu einem späteren Zeitpunkt. Außerdem werden Sie aufgefordert, einen Verzeichnis- und einen Dateinamen einzugeben. Der Verzeichnisname wird immer in Kleinbuchstaben umgesetzt, der Dateiname hingegen ist „case sensitive". Beide Angaben werden auf dem FTP-Server zur Datenablage genutzt und erleichtern Ihnen später die Identifikation Ihrer Image Files. Sie geben z. B. als Verzeichnisnamen „EPS001" an und entscheiden sich für „DBGruppe" als Dateinamen.

Die Verzeichnisstruktur auf dem Server könnte dann wie folgt aussehen:

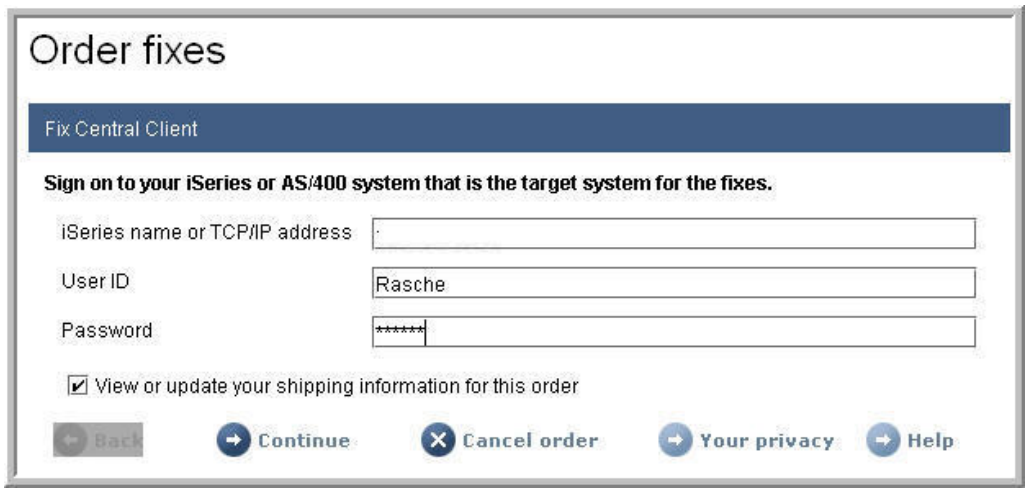
```
/B1234567/ eps001/C/DBGruppe_1.bin
/B1234567/eps001/C/DBGruppe_2.bin
    ...
/B1234567/eps001/C/DBGruppe_n.bin
```

Auch hier können Sie die Datenmenge wieder beschränken und falls dieser Fall eintreten sollte, werden die PTFs auf CD-ROM geliefert.

Zu guter Letzt müssen Sie noch daran denken, den Standort Ihres Quellsystems zu spezifizieren. Hier wählen Sie für gewöhnlich: Eastern Hemisphere: Africa – Asia – Australia –Europe.

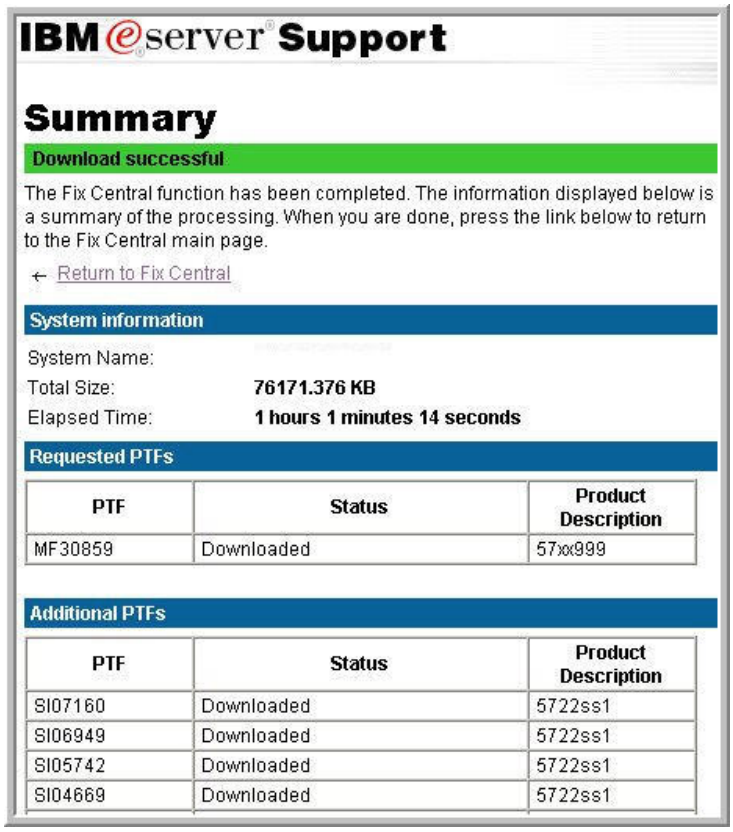
2.9.4.2
Seite 6

Jetzt lösen Sie den Bestellvorgang aus:



Der Bestellvorgang

Im nächsten Schritt wird ein Java-Applet im Browser geladen, das den weiteren Download ausführt. Die Sicherheitshinweise müssen Sie daher auch stets positiv beantworten. Anschließend wird der Downloadfortschritt angezeigt und zum Abschluss erhalten Sie noch eine Zusammenfassung über den Datentransfer.



IBM eserver Support

Summary

Download successful

The Fix Central function has been completed. The information displayed below is a summary of the processing. When you are done, press the link below to return to the Fix Central main page.

← Return to Fix Central

System information

System Name:
Total Size: **76171.376 KB**
Elapsed Time: **1 hours 1 minutes 14 seconds**

Requested PTFs

PTF	Status	Product Description
MF30859	Downloaded	57xx999

Additional PTFs

PTF	Status	Product Description
SI07160	Downloaded	5722ss1
SI06949	Downloaded	5722ss1
SI05742	Downloaded	5722ss1
SI04669	Downloaded	5722ss1

Die Zusammenfassung

Die PTFs stehen jetzt im *SERVICE-Bereich zur Verfügung und können installiert werden.

Falls Sie die Daten nicht direkt auf Ihren Server downloaden, erhalten Sie eine Email mit allen nötigen Informationen für den FTP-Zugriff auf den IBM-Server.

2.9.4.2

Seite 8

Fixes installieren

Bei der Installation von Fixes müssen Sie eine Reihe von Schritten ausführen, damit die Fixes korrekt installiert werden.

Zunächst einmal sollten Sie feststellen, ob Sie vor der Installation besondere Anweisungen zu beachten haben. Überprüfen Sie also unbedingt die Anweisungen, die zusammen mit dem kumulativen PTF-Paket geschickt wurden oder lesen Sie die Begleitschreiben der einzelnen Fixes. Jedes dieser Schreiben könnte spezielle Anweisungen enthalten, die dabei helfen, Probleme zu vermeiden.

Stellen Sie anschließend fest, wie Ihre Fixes installiert werden. Einige Fixes können ohne IPL angelegt werden (sofortige Fixes) andere benötigen ein IPL (verzögerte Fixes). Es gilt also herauszufinden, ob Sie verzögerte Fixes installieren und wann ein IPL Ihres Systems am besten passt. Bedenken Sie auch, dass es erforderlich sein könnte, sekundäre Partitionen abzuschalten, bevor Sie einen Fix auf die primäre Partition eines Systems mit logischen Partitionen anlegen.

Vor der Installation ist eine aktuelle Sicherung der Benutzerdaten, des Betriebssystems und der Lizenzprogramme notwendig.

Sie benötigen außerdem in Ihrem Benutzerprofil die Berechtigung *SECOFR für die weiteren Installationsschritte und Ihr System muss im Speicherbereich B ausgeführt werden. Auf dem System befinden sich zwei Kopien des gesamten lizenzierten internen Codes. Eine Kopie wird als permanente Kopie betrachtet und ist im Speicherbereich A des Systems gespeichert. Die andere Kopie wird als temporäre Kopie betrachtet und ist im Speicherbereich B des Systems gespeichert. Ein B in der Datenanzeige auf der Frontverkleidung der Systemeinheit zeigt an, dass das nächste System-IPL über den Speicherbereich B (temporär) erfolgt. Sie können auch den Befehl DSPPTF 5722999 in eine beliebige Befehlszeile eingeben. In der Anzeige „PTF-Status anzeigen“ ist der Speicherbereich im IPL-Quellenfeld angegeben. ##MACH#A ist der Speicherbereich A und ##MACH#B verweist auf den Speicherbereich B. Falls Sie feststellen, dass aktuell der Bereich B verwendet wird, müssen Sie den Speicherbereich wechseln.

Nachdem die Vorbereitungen beendet sind, können Sie jetzt Ihre Fixes installieren.

2.9.5
Seite 2

Kumulative PTF-Pakete mit einer virtuellen Einheit installieren

Die Bedeutung von PTFs ist unbestritten. Doch kann das Einspielen umfangreicher PTF-Gruppen oder CUM-PTFs bisweilen mühselig werden, da immer wieder neue CDs in das Laufwerk eingelegt werden müssen. Mit OS/400 V5R2 kann man mit Imagekatalogen arbeiten und dadurch das Einspielen der CDs deutlich eleganter lösen. Imagekataloge sind OS/400-Objekte der Art *IMGCLG. Um Imagekataloge zu nutzen, sind folgende Arbeitsschritte erforderlich:

- Erstellen eines virtuellen CD-Rom Laufwerks
- Erstellen eines IFS-Verzeichnisses
- Erstellen des Imagekatalogs
- Imagekatalog und IFS-Verzeichnis verbinden
- Übernahme der Daten in den Imagekatalog
- Imagekatalog und optisches Laufwerk verbinden
- PTF-Installation

Mit der Einführung virtueller optischer Laufwerke ist somit auch die direkte Verwendung der mittels FTP empfangenen Fixes möglich und das Brennen von CDs erübrigt sich.

2.9.5.1

Seite 2

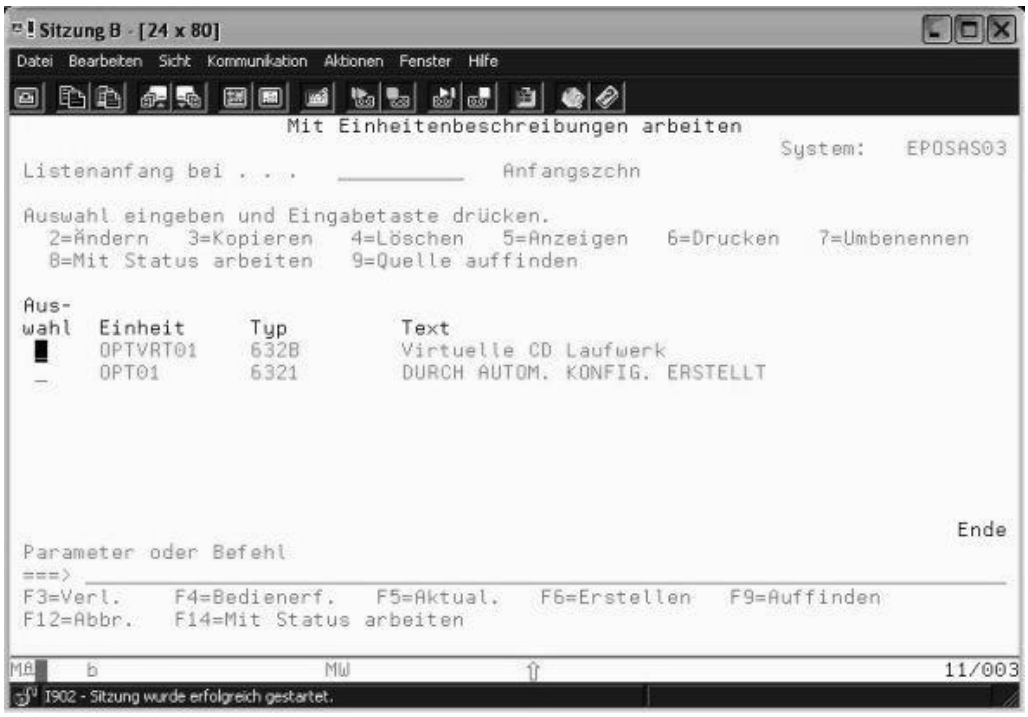
Eine virtuelle optische Einheit erstellen

Um ein virtuelles optisches Laufwerk zu erstellen, geben Sie den CRTDEVOPT ein und drücken anschließend die Funktionstaste F4. Sie erhalten den folgenden Bildschirm:

```
                  Einh.beschr. erstellen (Opt.) (CRTDEVOPT)

Auswahl eingeben und Eingabetaste drücken.

Einheitenbeschreibung . . . . .   OPTVRT01     Name
Ressourcen name . . . . . . . .   *vrt_____     Name, *NONE, *VRT, OPT01
Einheitentyp . . . . . . . . . .  *RSRCNAME    *RSRCNAME, 4531, 632A, 632B...
```

Der Befehl CRTDEVOPT

Es ist ausreichend, wenn Sie einen Einheitennamen (frei wählbar) vergeben und als Ressourcenname *VRT verwenden. Die virtuelle optische Einheit wird durch den Einheitentyp 632B angezeigt.



Die Einheitenbeschreibung für das optische Laufwerk

2.9.5.1.1
Seite 2

Überprüfen Sie anschließend, ob die Einheit aktiv ist. Drücken Sie hierfür die Taste F14 „Mit Status arbeiten“ oder nutzen Sie die Option 8 „Mit Status arbeiten“. Sie sollten dann folgenden Bildschirm erhalten:

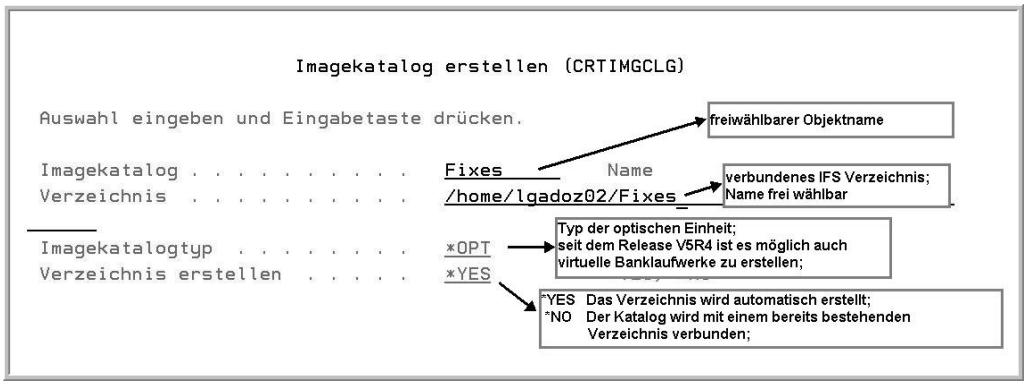
Das aktivierte optische Laufwerk

Falls Ihre Einheit nicht aktiv ist, geben Sie eine 1 ein, um sie anzuhängen.

Der Imagekatalog

2.9.5.1.2
Seite 1

Erstellen Sie jetzt einen Imagekatalog. Der Befehl CRTIMGCLG (Imagekatalog erstellen) ordnet einem Imagekatalog ein Zielverzeichnis zu und erstellt außerdem ein OS/400-Objekt der Art *IMGCLG. In dem Zielverzeichnis müssen später die einzelnen Images geladen werden. Generell kann ein Verzeichnis nur mit einem Imagekatalog zurzeit verbunden sein, aber ein Imagekatalog kann mit mehreren Verzeichnissen verbunden werden. Geben Sie also den Befehl CRTIMGCLG ein und prompten Sie den Befehl mit der Funktionstaste F4. Daraufhin erhalten Sie den folgenden Bildschirm:



Einen Imagekatalog erstellen

Jetzt sind alle Vorbereitungen abgeschlossen. Diese Arbeit müssen Sie also auch nur das erste Mal ausführen. Danach können Sie jederzeit die Imagekatalogeinträge für jeden physischen Datenträger hinzufügen oder Einträge für Fixes hinzufügen, die Sie per FTP erhalten haben, z. B. von Fix Central.

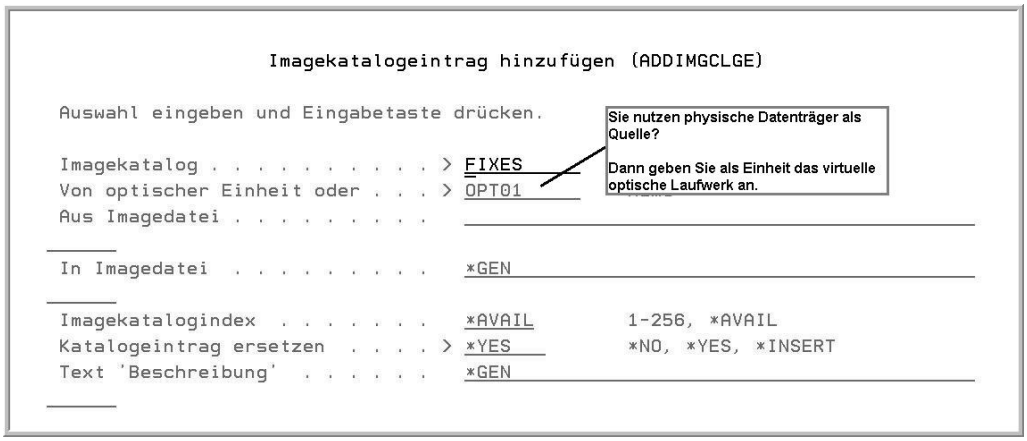
2.9.5.1.2
Seite 2

Images hinzufügen, laden und verwalten

Wenn Sie den Imagekatalog und die optische Einheit verwenden wollen, müssen Sie zunächst einmal einen Imagekatalogeintrag für jeden vorhanden physischen Datenträger hinzufügen. Bitte prüfen Sie vorher, ob für die Images genügend freier Plattenspeicherplatz vorhanden ist.

Anschließend verwenden Sie den folgenden Befehl, um die Einträge hinzuzufügen.

```
ADDIMGCLGE IMGCLG(FIXES) FROMDEV(OPT01) TEXT(Physische Datenträger).
```

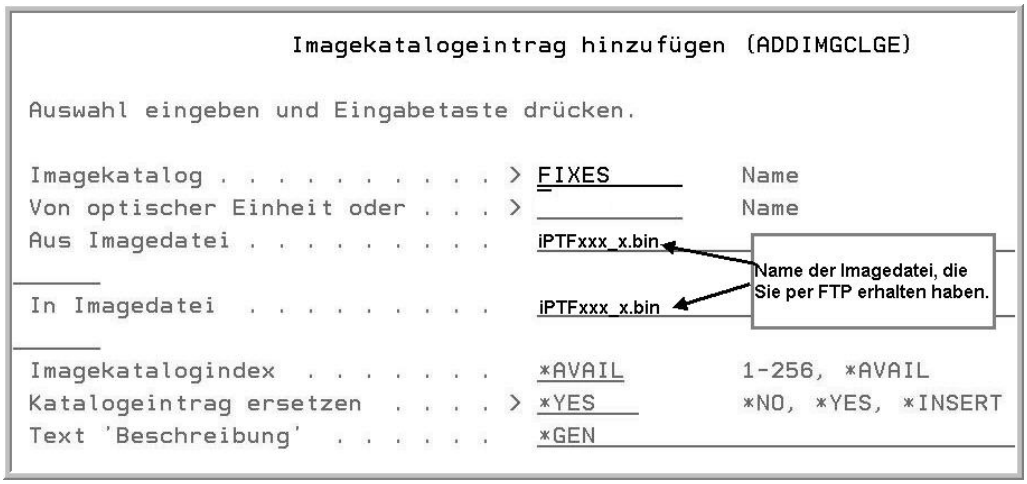


Physische Datenträger laden

Falls Sie Einträge für Ihre per FTP erhaltenen Images hinzufügen wollen, können Sie das IFS-Verzeichnis, das die Daten enthält auf die folgenden Arten hinzufügen:

1. Die Daten befinden sich in demselben Verzeichnis, das dem Imagekatalog zugeordnet ist:

2.9.5.1.3
Seite 2

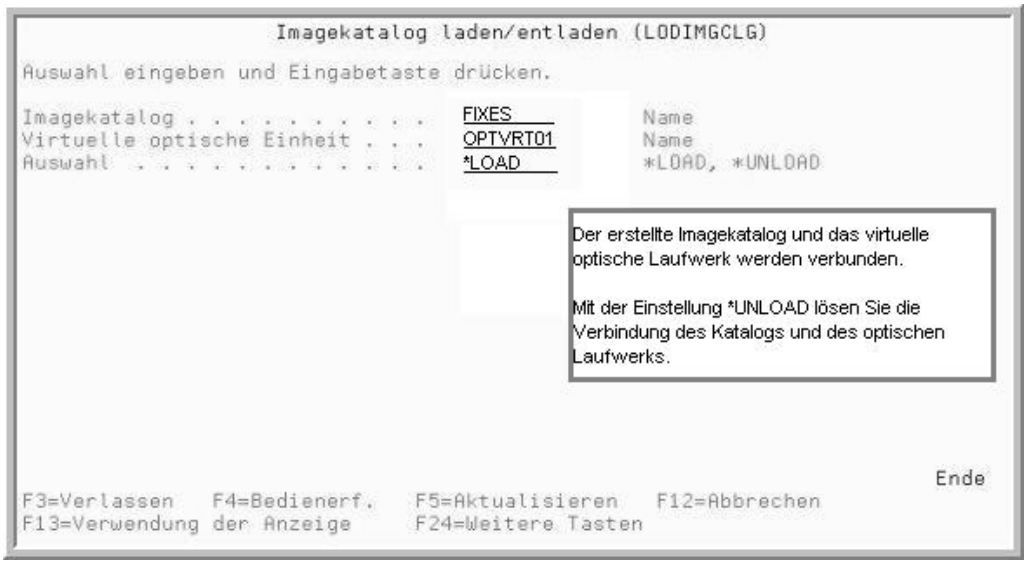


FTP-Images laden

Falls sich Ihre FTP-Dateien in einem anderen Verzeichnis befinden, können Sie die Quellendatei (Parameter FROMFILE) auch qualifizieren. Der entsprechende Befehl könnte dann z. B. heißen:

```
ADDIMGCLGE IMGCLG(FIXES) FROMFILE('HOME/FIXES/iPTFxxx_x.bin')
                         TOFILE(iPTFxxx_x.bin)
```

Als nächstes müssen Sie den Imagekatalog laden. Laden bedeutet in diesem Zusammenhang, dass der Imagekatalog einer optischen Einheit zugeordnet wird. Hierfür geben Sie den Befehl LODIMGCLG ein:

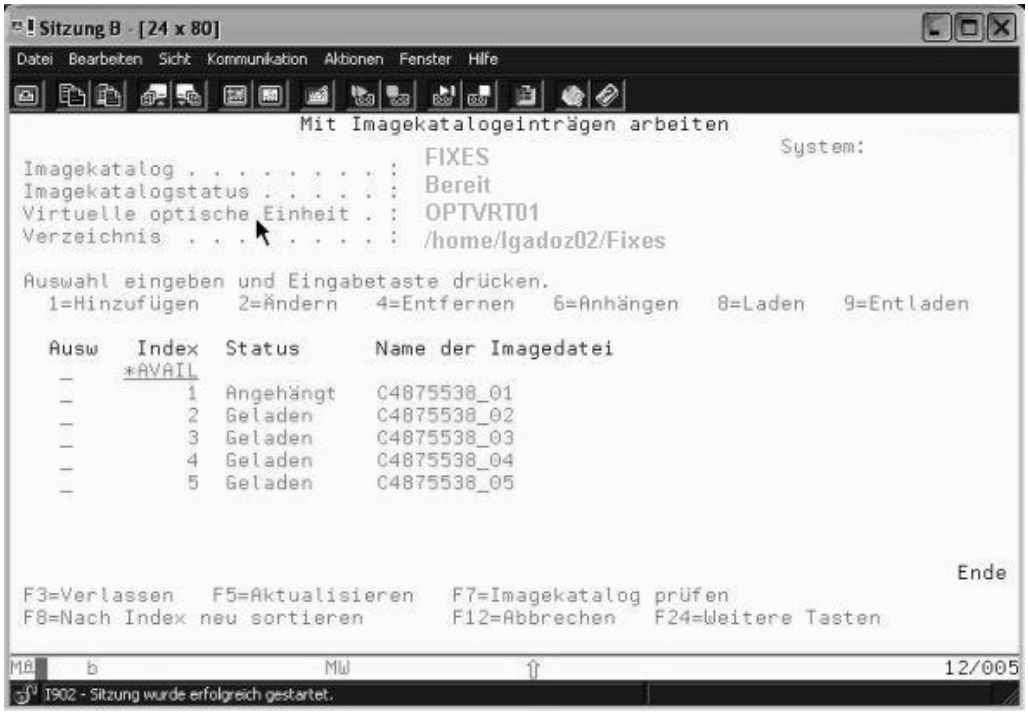


Der Befehl LODIMGCLG

Falls Sie später einmal die Imageeinträge wieder entfernen wollen, müssen Sie aber zuvor die Verbindung zwischen dem optischen Laufwerk und dem Imagekatalog aufheben. Dies geschieht ebenfalls mit dem Befehl LODIMGCLG. Allerdings verwenden Sie dann den Wert *UNLOAD in dem Parameter OPTION (Auswahl). Sobald der Befehl LODIMGCLG korrekt verarbeitet wurde, steht Ihr Katalog jetzt zur Verarbeitung bereit.

Prüfen Sie den Katalog vor der Verarbeitung ruhig noch mal, denn vielleicht haben Sie ja vergessen, einen Imageeintrag hinzuzufügen oder vielleicht haben Sie die Reihenfolge der Images oder CD-ROMs verwechselt. Sie können mit den Katalogeinträgen arbeiten, wenn sie den Befehl WRKIMGCLGE eingeben. Anschließend erhalten Sie folgende Bildschirmausgabe:

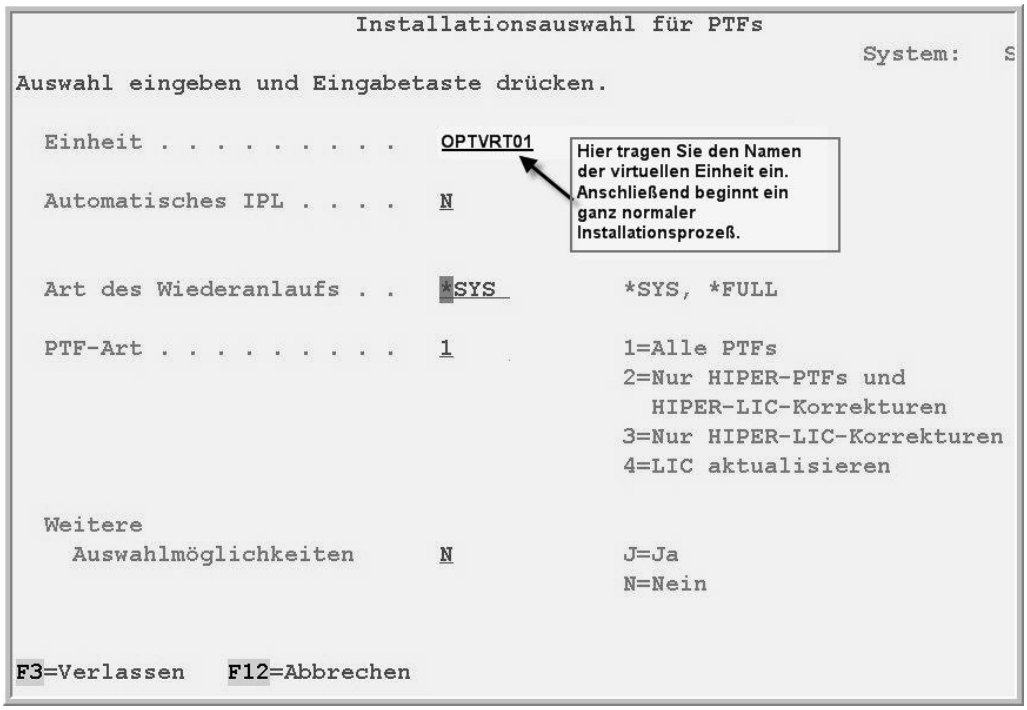
2.9.5.1.3
Seite 4



Ergebnis des Befehls WRKIMGCLGE

Standardmäßig sollte der Datenträger mit dem niedrigsten Index angehängt und alle weiteren Datenträger-Images nur geladen sein. Falls dies nicht so ist, können Sie die Einträge mit dem Befehl VRYIMGCLG IMGCLG(FIXES) TYPE(*PTF) SORT(*YES) vom System überprüfen lassen und dabei auch die Images in die richtige Reihenfolge bringen.

Nun können Sie Ihre Images installieren. Rufen Sie hierzu das Menü PTF auf und wählen Sie die Option 8 (PTF Paket installieren). Anschließend erhalten Sie die Anzeige „Installationsauswahl für PTFs“.



Die Installationsauswahl

Sobald Sie in der Installationsauswahl das virtuelle Laufwerk angegeben haben, beginnt ein ganz normaler Installationsprozess. Mit einer Ausnahme: Alle Images, die Sie in den Katalog geladen haben, werden automatisch und nacheinander installiert. Niemand muss sich um das leidige Wechseln der CDs kümmern.

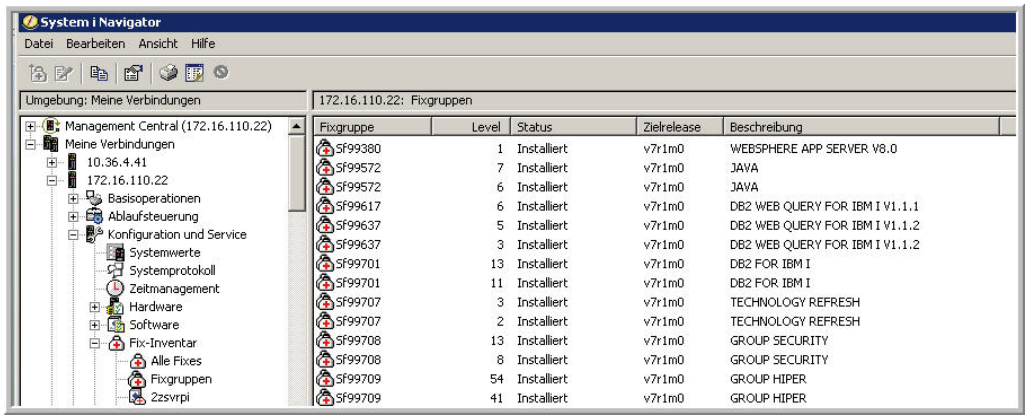
Wenn die Installation Ihrer PTFs abgeschlossen ist, sollten Sie die Images mit dem Befehl RMVCLGIMGE (Imagekatalogeinträge entfernen) aus dem Katalog entfernen. Dies schafft Plattenplatz und verhindert, dass das System bei der nächsten Installationsverarbeitung versucht, die alten Fixes erneut zu installieren.

2.9.5.1.3

Seite 6

Technology Refresh für IBM i7.1

Derzeit ist für den i-Server kein neues Release geplant bzw. angekündigt. Zu schnell folgten in den letzten Jahren die Releases i6.1 und i7.1. Die System i-Kunden brauchen nach diesen schnellen Release-Freigaben einfach etwas Zeit, um auf das aktuelle Release i7.1 oder zumindest auf i6.1 zu wechseln. Um trotzdem neue Technologien zu implementieren, gibt es jetzt die neue PTF-Gruppe **Technology Refresh.** Seit Release i7.1 nutzt IBM diese neue PTF-Gruppe für „Serverupdates".



Fixgruppe	Level	Status	Zielrelease	Beschreibung
Sf99380	1	Installiert	v7r1m0	WEBSPHERE APP SERVER V8.0
Sf99572	7	Installiert	v7r1m0	JAVA
Sf99572	6	Installiert	v7r1m0	JAVA
Sf99617	6	Installiert	v7r1m0	DB2 WEB QUERY FOR IBM I V1.1.1
Sf99637	5	Installiert	v7r1m0	DB2 WEB QUERY FOR IBM I V1.1.2
Sf99637	3	Installiert	v7r1m0	DB2 WEB QUERY FOR IBM I V1.1.2
Sf99701	13	Installiert	v7r1m0	DB2 FOR IBM I
Sf99701	11	Installiert	v7r1m0	DB2 FOR IBM I
Sf99707	3	Installiert	v7r1m0	TECHNOLOGY REFRESH
Sf99707	2	Installiert	v7r1m0	TECHNOLOGY REFRESH
Sf99708	13	Installiert	v7r1m0	GROUP SECURITY
Sf99708	8	Installiert	v7r1m0	GROUP SECURITY
Sf99709	54	Installiert	v7r1m0	GROUP HIPER
Sf99709	41	Installiert	v7r1m0	GROUP HIPER

Gruppen-PTF Technology Refresh

Warum sollten Sie die Technology Refreshes einspielen. Schließlich galt lange Zeit die Aussage: „Never change a running System". IBM führt mehrere Gründe an:

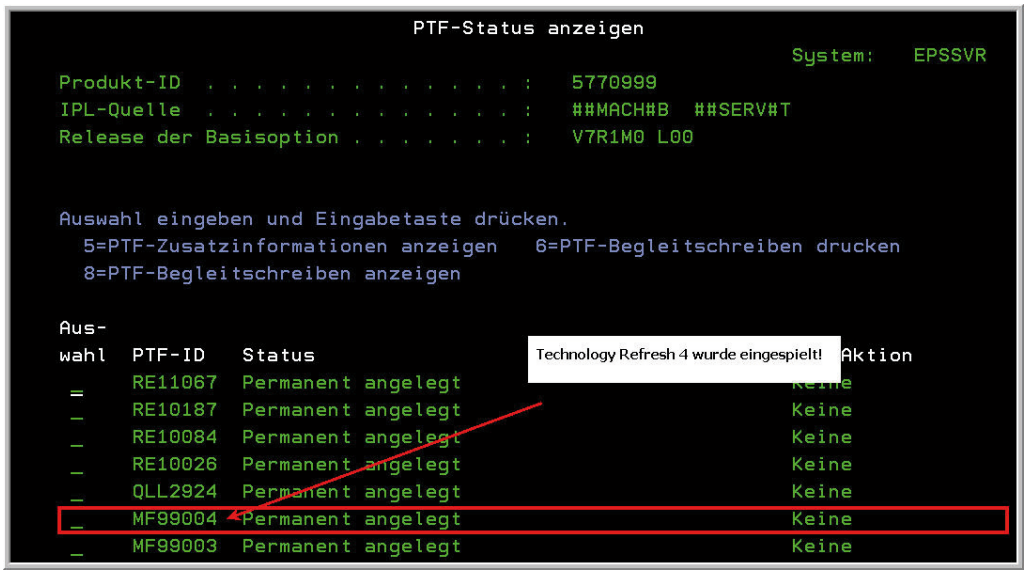
- Mit Hilfe der Technology Refreshes können neue Funktionen, aber auch modifizierte Funktionen zeitiger integriert werden.
- Die Auslieferung in Form von PTFs ist einfacher, geht schneller und ist weniger aufwendig als ein Release-Wechsel.
- Ein Technology Refresh hält Ihre Server auf dem aktuellen Stand und verhindert ggf. unnötige Down-Zeiten des Systems, weil mögliche Fehlersituationen, die ohne Technology Refresh auftreten könnten, bereits behoben sind.
- Und es ist sehr viel einfacher Systeme synchron zu halten, wenn lediglich ein Technology Refresh eingespielt werden muss.

2.9.5.2
Seite 2

Was beinhaltet der Technology Refresh?
Der Technology Refresh ist letztendlich Systemsoftware, die als Paket entwickelt, getestet und für ein Basis-Release freigegeben wird. Ein Technology Refresh ist also kein neues Release, aber ein „Releaseupdate“ für neue Hardware oder Virtualisierungsfunktionen.

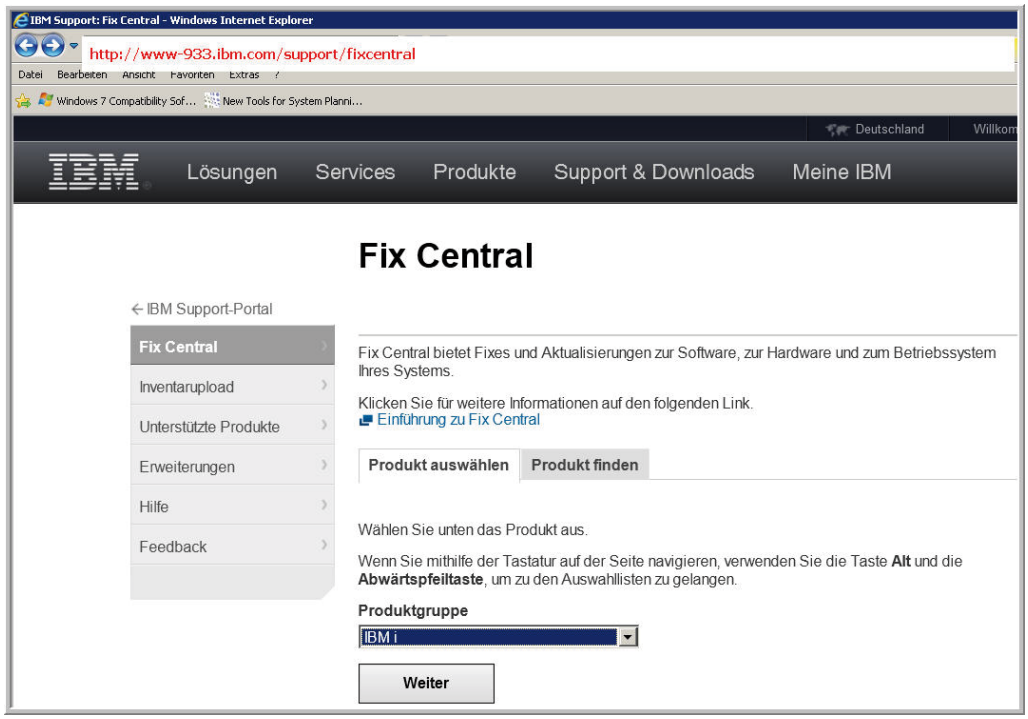
Meistens dienen die Technology Refreshes also dazu, neue Hardware oder neue hardwarenahe Technologien zu implementieren. TR4 beispielsweise brachte als neues Feature u. a. IBM i Live Partition Mobility und wird daher als wichtiger Schritt auf dem Weg in die Cloud betrachtet, denn Live Partition Mobility (LPM) ist ein Feature der PowerVM Enterprise Edition, das ein „Verschieben“ virtueller Maschinen im aktiven, inaktiven oder suspendierten Modus zwischen Power-Systemen ermöglicht. Aber auch Performanceverbesserungen beispielsweise für Record Locks gehören zu TR4.

Wie ermitteln Sie, welchen Technologie-Stand Ihr Server hat?
Der Technology Refresh ändert weder den Release-Stand noch den Release-Level, denn es handelt sich um ein PTF! Wenn auf dem Server also Release i7.1 installiert ist, ändert der Technology Refresh (TR) nicht den Level in beispielsweise i.7.1.1, sondern es bleibt beim Release i7.1. Um den Technology Level zu bestimmen, müssen Sie sich die PTFs für den Licensed Internal Code (5770-999) anschauen. TR-PTFs werden im Format MFxxx gespeichert. Die letzten drei Zeichen bestimmen dabei den Technologie-Stand.



TR-Level prüfen

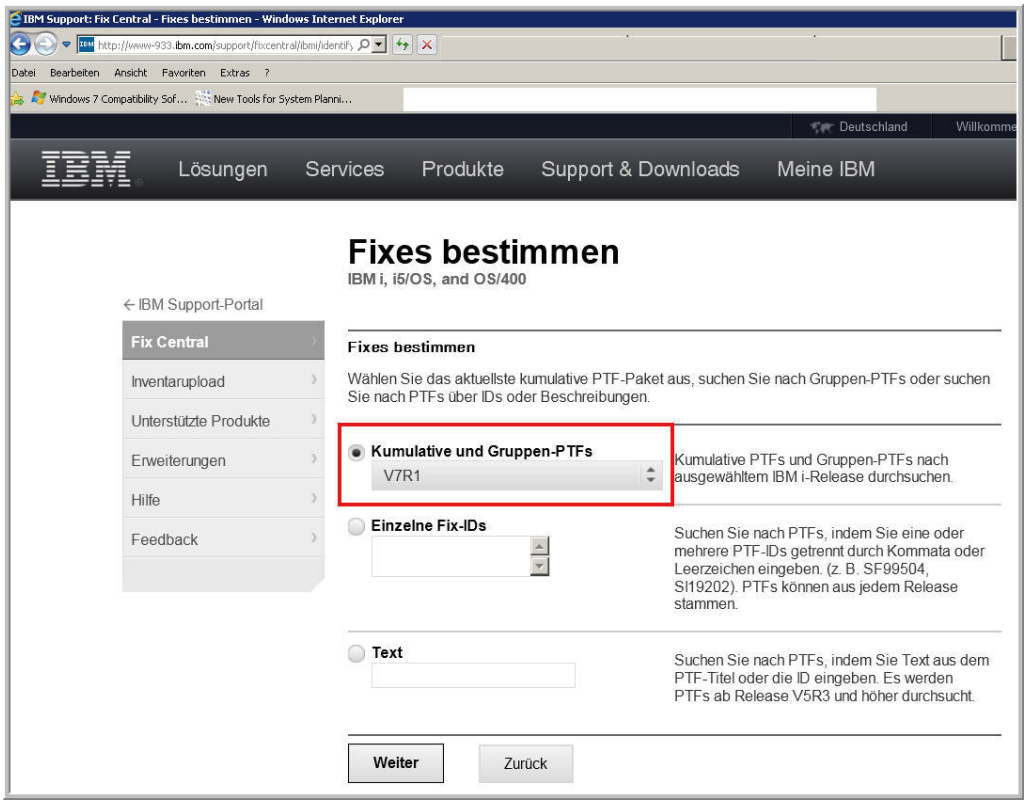
Wie erhalten Sie die Technology Refreshes?
Technology Refreshes können wie alle anderen Gruppen-PTFs im Fix Central bestellt werden.



Fix Central – Optionen

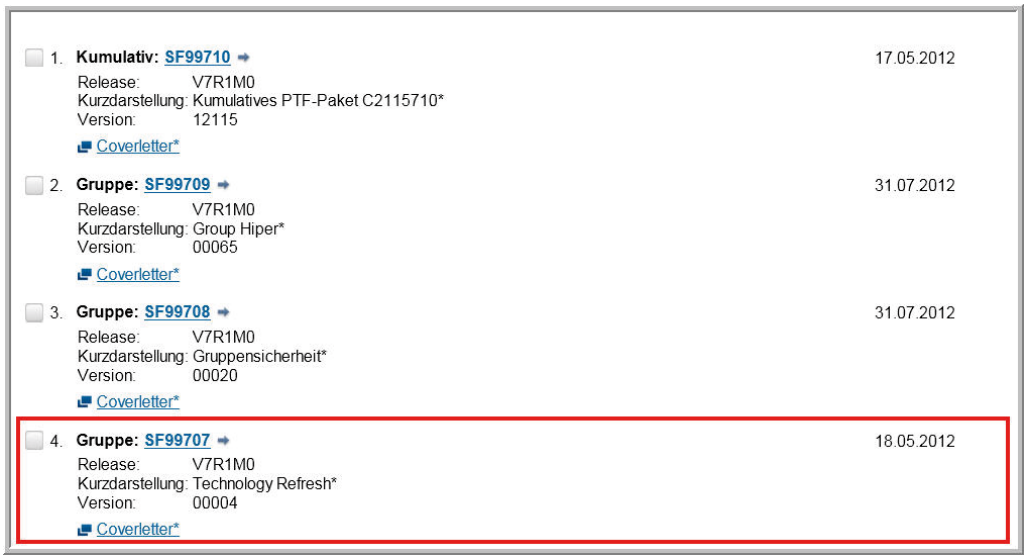
Wenn Sie die Produktgruppe IBM i auswählen und anschließend auf WEITER klicken, gelangen Sie in einen neuen Dialog.

2.9.5.2
Seite 4



Fix Central – PTFs bestimmen

Der TR ist natürlich Bestandteil der kumulativen PTFs sowie der Gruppen-PTFs.



TR ist ein Gruppen-PTF

Derzeit aktuell ist Technology Refresh 4.

Müssen Sie das neue Gruppen-PTF installieren?
Vielleicht sind Sie an dem neuen Hardwaresupport und den neuen Funktionen gar nicht interessiert und entscheiden sich daher, die Gruppe nicht zu installieren. In diesem Fall sollten Sie bedenken, dass spätere PTFs – die Sie vielleicht zwingend benötigen – den TR voraussetzen. Und dann müssten Sie zunächst den TR nachholen, bevor Sie das gewünschte und vielleicht dringend benötigte PTF installieren können. Die Down-Zeit Ihres Servers könnte sich so unerwartet erhöhen.

Falls Sie Angst haben, die neuen Funktionen könnten Ihren Betrieb gefährden, weil Sie noch „alte" Hardware einsetzen, so lassen Sie sich sagen, dass diese Angst unbegründet ist. Alle neuen Funktionen werden zunächst ausführlich auf „alten" Plattformen getestet, bis sich die IBM sicher ist, dass die neuen Funktionen keine negativen Auswirkungen auf bestehende Funktionen haben werden.

2.9.5.2
Seite 6

Virtuelle Laufwerke

Ein virtuelles Laufwerk ist die funktionale Nachbildung eines Laufwerks – inklusive seiner Datenträger. Man könnte auch sagen, dass das Laufwerk und sein Datenträger emuliert werden. Virtuelle Laufwerke existieren daher nicht als physische Hardware, werden aber vom Benutzer genauso verwendet und funktionieren auch genau so wie physische Laufwerke.

Häufig werden optische Laufwerke – also CD- und DVD-Laufwerke – emuliert. Diese Möglichkeit haben wir bereits seit Release V5R2. Aber wir können seit V5R4 auch virtuelle Bandlaufwerke – sogenannte Virtual Tapes – erstellen. Sowohl der Einsatz der virtuellen optischen Laufwerke als auch die Nutzung der Virtual Tapes basieren auf virtuellen Einheitenbeschreibungen sowie auf Imagekatalogen.

Imagekataloge kennen wir seit Release V5R2. Hierbei handelt es sich um Objekte des Typs *IMGCLG. Sie sind Voraussetzung für die Nutzung der virtuellen Einheiten. Doch damit die Imagekataloge auch wirklich etwas nützen, benötigen wir sowohl für die virtuellen optischen Einheiten als auch für die Virtual Tapes zunächst einmal die entsprechenden Einheitenbeschreibungen.

2.9.6
Seite 2

Virtuelle Einheitenbeschreibungen erstellen

Um ein virtuelles optisches Laufwerk zu erstellen, geben Sie den Befehl CRTDEVOPT (Create Device Optical) auf einer Befehlszeile ein und drücken anschließend die Funktionstaste F4. Sie erhalten daraufhin den folgenden Bildschirm:

```
                    Einh.beschr. erstellen (Opt.) (CRTDEVOPT)

Auswahl eingeben und Eingabetaste drücken.

Einheitenbeschreibung . . . . . > VRTOPT      Name
Ressourcen name . . . . . . . . > *VRT        Name, *NONE, *VRT...
Einheitentyp . . . . . *RSRCNAME  *RSRCNAME, 632A, 632B, 632C...
Lokale Internet-Adresse . . . .   *NONE       *NONE, *SRVLAN
```

Befehl CRTDEVOPT

Normalerweise ist es ausreichend, wenn Sie den Einheitennamen vergeben und als Ressourcennamen *VRT verwenden. Die virtuelle optische Einheit erhält dann automatisch den Einheitentyp 632B. Neu in diesem Zusammenhang ist der Parameter „Lokale Internet-Adresse (LCLINTNETA)“. Dieser Parameter ist erforderlich, wenn virtuelle Images innerhalb des Netzwerks verwendet werden sollen und nicht lokal auf dem Server liegen. Dieser Parameter ist daher auch ungültig, wenn Sie den Wert *NONE für den Parameter „Ressourcen name (RSRCNAME)“ verwenden. Oder wenn Sie als Einheitentyp etwas anderes als den Wert 632B oder *RSRCNAME für den Parameter TYPE nutzen.

Überprüfen Sie Ihre Arbeit, indem Sie sich alle optischen Einheiten Ihres System i Servers anzeigen lassen. Hierfür können Sie den Befehl WRKDEVD DEVD (*OPT) verwenden. Daraufhin sollte folgender Bildschirm erscheinen:

```
                    Mit Einheitenbeschreibungen arbeiten
                                                        System:   EPSSVR
Listenanfang bei . . .  __________   Anfangszchn

Auswahl eingeben und Eingabetaste drücken.
  2=Ändern   3=Kopieren   4=Löschen   5=Anzeigen   6=Drucken   7=Umbenennen
  8=Mit Status arbeiten   9=Quelle auffinden

Aus-
wahl  Einheit     Typ         Text
  _   OPTVRT01    632B        Optisches virtuelles Laufwerk
  _   OPT01       6331        DURCH AUTOM. KONFIG. ERSTELLT
```

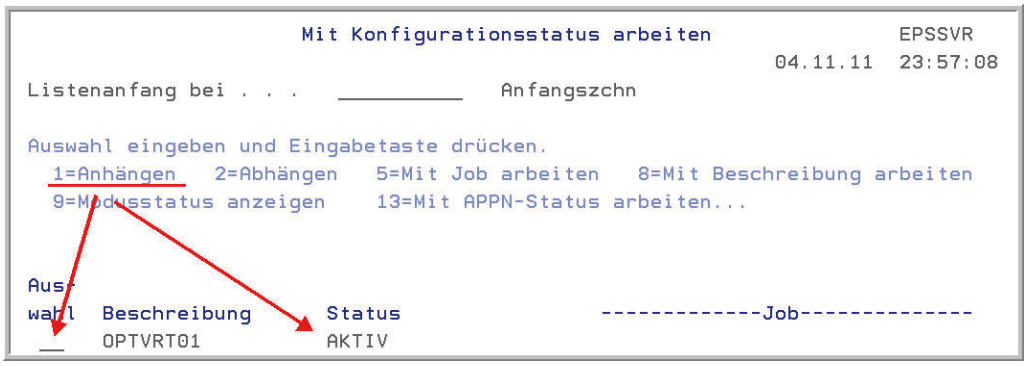
Die optische virtuelle Einheitenbeschreibung

Wenn Sie sich die Einheitenbeschreibung anschauen, werden Sie feststellen, dass die Einheit beim IPL – wie jede physische Einheit – automatisch ange-

2.9.6.1
Seite 2

hängt wird. Fehlermeldungen werden an den QSYSOPR gesendet. Hier wurde kein IPL ausgeführt. Daher müssen wir in einem letzten Schritt das virtuelle Laufwerk noch anhängen; erst dann ist es betriebsbereit. Hierfür kann man entweder den Befehl WRKCFGSTS *DEV oder die Option 8 im oben gezeigten Bildschirm verwenden.

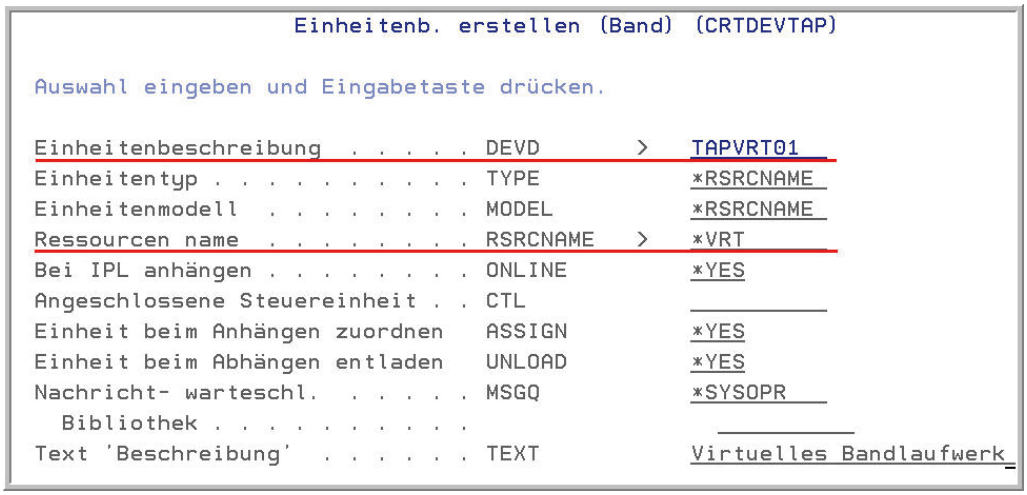
Daraufhin erscheint der folgende Bildschirm:



Konfigurationsstatus prüfen

Falls Ihre Einheit noch nicht aktiv ist, geben Sie eine „1“ ein, um Sie anzuhängen.

Virtuelle Bandeinheiten werden auf ähnliche Art und Weise erstellt. Nur wird dann der Befehl CRTDEVTAP (Create Device Tape) genutzt.



Virtuelle Bandeinheiten erstellen

Wie schon zuvor werden eigentlich nur zwei Parameter benötigt:

- Einheitenbschreibung (DEVD) Name des Virtual Tapes
- Ressourcen name (RSRCNAME) *VRT

Alle weiteren Parameter können Sie einfach übernehmen. Anschließend muss auch diese Einheit angehängt werden. Gehen Sie wie zuvor beschrieben vor.

Der Imagekatalog

Nachdem unsere Einheitenbeschreibungen erstellt und aktiviert wurden, muss der Imagekatalog erstellt werden. Hierfür ist es möglich, sowohl das 5250-Interface oder den System i Navigator zu verwenden.

Der Befehl CRTIMGCLG (Create Imagecatalog) ordnet einem Imagekatalog ein Zielverzeichnis im Integrated File System (IFS) zu. Erstellt wird hierbei ein i5/OS-Objekt der Art *IMGCLG. Anschließend wird der Befehl CRTIMGCLG eingegeben und der Befehl mit der Funktionstaste F4 gepromptet. Daraufhin erhalten wir folgenden Bildschirm:

```
                    Imagekatalog erstellen (CRTIMGCLG)

Auswahl eingeben und Eingabetaste drücken.

Imagekatalog . . . . . .   GrpPtf      Name
Verzeichnis  . . . . . .   /ptf/GruppenPTF
_____
Imagekatalogtyp  . . . . . .   *OPT          *OPT, *TAP
Verzeichnis erstellen  . . . . .   *YES          *YES, *NO
Imagekatalog importieren . . . .   *NO           *NO, *YES
ASP-Schwellenwert für Katalog  .   *CALC         1-99, *CALC, *MAX
Virt. Datenträger hinzufügen . .   *NONE         1-256, *NONE
```

Imagekatalog für optische virtuelle Einheiten

Der Name des Katalogs ist frei wählbar. Mit dem Parameter „Verzeichnis (DIR)“ ordnen Sie dem Imagekatalog einen Bereich im IFS zu. Dieser Pfad ist unbedingt anzugeben, denn in das Zielverzeichnis müssen später die einzelnen Images geladen werden. Generell kann ein Verzeichnis nur mit einem Imagekatalog verbunden sein, aber ein Imagekatalog kann mit mehreren Verzeichnissen verbunden werden. Man könnte daher problemlos ein IFS-Verzeichnis für Gruppen-PTFs erstellen und ein weiteres für die CUM-PTFs, ohne weitere optische virtuelle Einheiten zu erstellen. Der Parameter „Imagekatalogtyp (Type)“ bestimmt, ob das Verzeichnis einem Virtual Tape oder einem virtuellen optischen Laufwerk zugeordnet wird. Außerdem kann das IFS-Verzeichnis automatisch erstellt werden, wenn es bislang noch nicht existiert.

Mit dem Parameter „ASP-Schwellenwert für Katalog (CLGASPTHLD)“ kann die Größe des Imagekatalogs begrenzt werden. Wenn es sich beim Imagekatalog um einen Katalog für eine virtuelle Bandeinheit handelt, berechnet das System den maximalen Speicher, sobald Sie die Einstellung *CALC wählen. Wenn daraufhin mehr als 95 Prozent des verbleibenden Speichers belegt ist oder wenn 5 GB benötigt werden, wird die Bandoperation mit der Fehlermeldung „Ende des Datenträgers“ gestoppt. Beim optischen Laufwerk wird die-

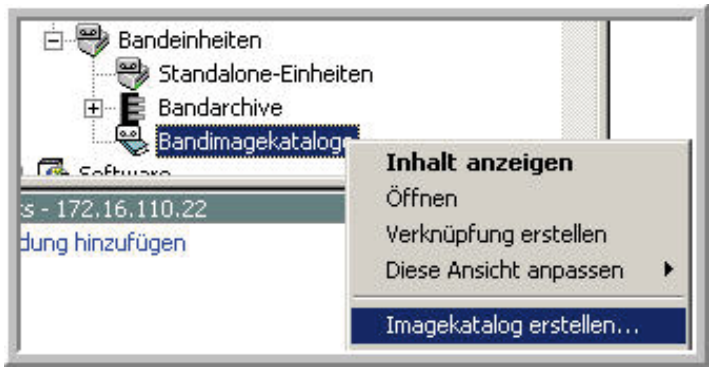
2.9.6.2
Seite 2

ser Grenzwert ebenfalls berücksichtigt, vorausgesetzt Sie nutzen später beim Laden der Images im Parameter „Allocate Storage (ALCSTG) den Wert *IMGSIZ.

Alternativ zum Wert *CALC können Sie auch den maximal verfügbaren Speicher im entsprechenden ASP für die virtuellen Datenträger erlauben. Oder Sie begrenzen den Bereich durch eine entsprechende Prozentangabe. D. h. – wenn Sie beispielsweise den Wert 20 im Parameter „ASP-Schwellenwert für Katalog (CLGASPTHLD)“ eintragen, dann dürfen auch nur 20 Prozent des ASP-Speicherbereichs von den virtuellen Datenträgern verwendet werden. Ein virtueller Datenträger soll zum jetzigen Zeitpunkt noch nicht zugeordnet werden, so dass wir unsere Eingaben bestätigen können, nachdem wir noch einen beschreibenden Text eingegeben haben.

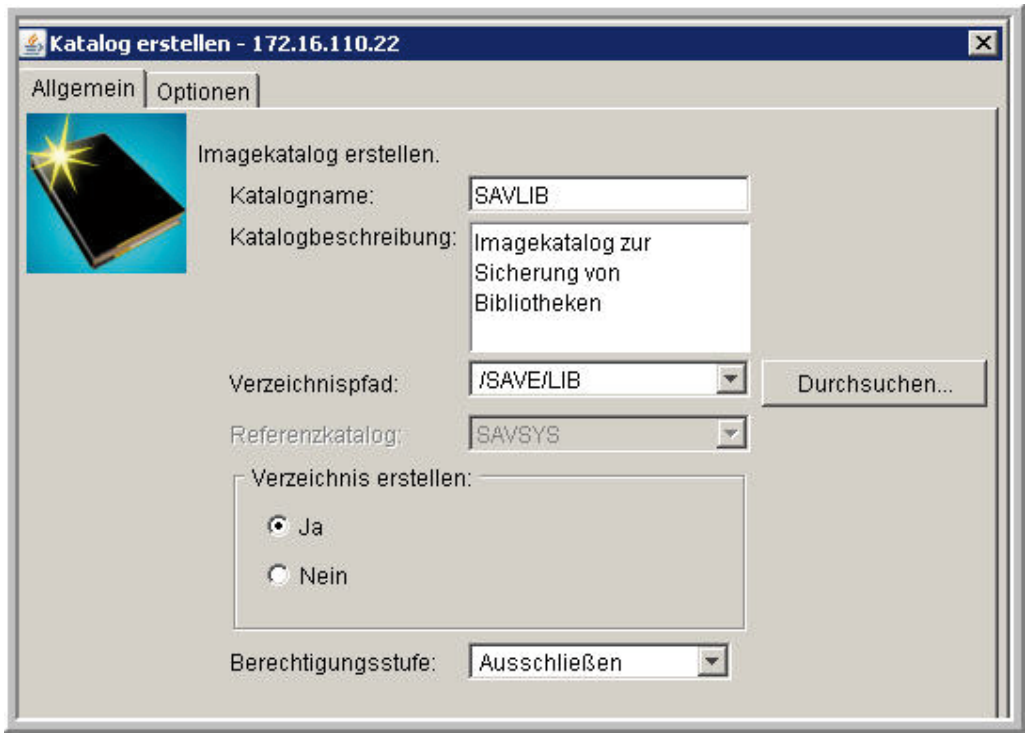
Wir wollen auch einen Imagekatalog für unser Virtual Tape erstellen. Hierfür wird jetzt allerdings der System i Navigator genutzt.

Stellen Sie als Erstes die Verbindung zu Ihrem Server her und öffnen Sie anschließend den Eintrag „Hardware ➜ Bandeinheiten ➜ Bandimagekataloge“:



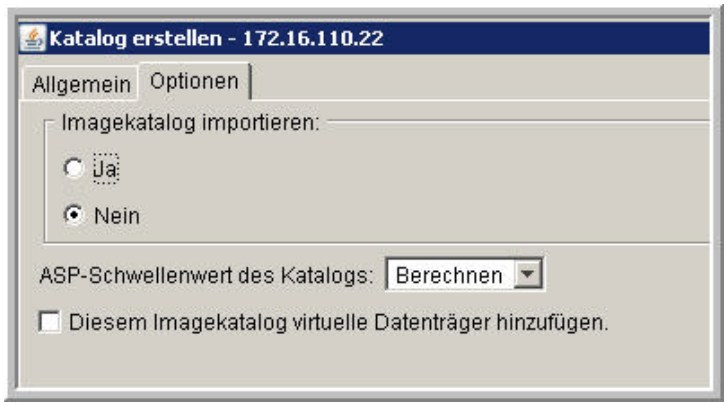
Bandimagekatalog erstellen

Mit Hilfe der Kontexttaste können Sie den Dialog zur Imagekatalogerstellung aufrufen. Daraufhin erscheint der folgende Dialog:



Allgemeine Angaben

Wie zuvor müssen wir dem Katalog einen Namen geben und ein IFS-Verzeichnis zuordnen. Anschließend schauen wir uns noch die weiteren Optionen an.



Allgemeine Optionen

Hier finden wir wieder die bereits bekannten Parameter „ASP-Schwellenwert für Katalog (CLGASPTHLD)“ und „Virtuelle Datenträger hinzufügen“. Wie zuvor belassen wir es bei den Standardeinstellungen. Anschließend bestätigen wir diesen Dialog mit “OK“, so dass der Imagekatalog erstellt wird.

2.9.6.2
Seite 4

Die Vorbereitungen sind jetzt sowohl für unsere optische virtuelle Einheit als auch für das Virtual Tape abgeschlossen. Diese Arbeiten werden nur einmal ausgeführt! Wenn Sie später die Imagekataloge nutzen, müssen Sie nur noch mit den Imagekatalogeinträgen arbeiten – nicht aber mit den Imagekatalogen selbst.

Virtual Tapes zur Sicherung von Bibliotheken nutzen

Handelt es sich um virtuelle Bandlaufwerke, muss dem virtuellen Bandlaufwerk vor der Nutzung noch mindestens ein Eintrag zum Imagekatalog hinzugefügt werden. Solch einen Imagekatalogeintrag fügen wir über den Befehl ADDIMGCLGE (Add Imagecatalogentry) hinzu. Wir können aber auch einfach den Befehl WRKIMGCLG (Work Imagecatalog) verwenden und anschließend die Option 12 „Imagekatalogeinträge hinzufügen“ nutzen. Daraufhin erscheint der folgende Dialog:

```
                    Mit Imagekatalogeinträgen arbeiten
                                                          System:   EPSSVR
Katalog  . . :   SAVLIB                 Status . . . :   Nicht ber.
Typ  . . . . :   Band                   Einheit  . . :
Nächs.Datentr:   *NONE
Verzeichnis  :   /SAVE/LIB

Auswahl eingeben und Eingabetaste drücken.
  1=Hinzufügen   2=Ändern   4=Entfernen   6=Anhängen   8=Laden   9=Entladen
  10=Datenträger initialisieren           11=Band anzeigen       13=Auszug
  14=Duplizieren

                         Datentr.-
Ausw   Index  Status        name     Dichte      Typ  Schreibschutz
 _     *AVAIL
```

Mit Imagekatalogeinträgen arbeiten

Wie erwartet ist derzeit noch kein Eintrag zugeordnet. Wir könnten auch sagen, derzeit ist noch kein Band eingelegt. Dies sollte jetzt passieren:

```
                Imagekatalogeintrag hinzufügen (ADDIMGCLGE)

Auswahl eingeben und Eingabetaste drücken.

Imagekatalog . . . . . > SAVLIB        Name
Von optischer Einheit oder . . > TAPVRT01     Name, *VOL
Aus Imagedatei . . . . . > *new
_____
In Imagedatei  . . . . . *GEN
_____
Imagekatalogindex  . . . . . > *AVAIL        1-256, *AVAIL
Katalogeintrag ersetzen  . . . . *NO         *NO, *YES, *INSERT
Speichermedientyp  . . . . . *RAM        *RAM, *WORM, *ERASE
Imagegröße . . . . . *IMGCLGTYPE   Megabyte, *CD650, *DVD2600..
Text 'Beschreibung'  . . . . . Initialisierender Katalogeintrag
```

Imagekatalogeintrag hinzufügen

Als Erstes müssen wir die im Kapitel 2.9.6.1 erstellte Bandeinheit mit dem Imagekatalog verbinden. Für den Parameter „Aus Imagedatei“ verwenden wir den Wert *NEW, da wir ein neues virtuelles Band, aber kein bereits vorhandenes verwenden wollen. Der Parameter „In Imagedatei (TOFILE)“ gibt den Namen der Zieldatei an, die in das Zielverzeichnis kopiert werden soll.

Wir könnten auch sagen, hier weisen wir dem virtuellen Band einen Namen zu. Wenn wir – wie im Beispiel – den Wert *GEN verwenden, wird der Name aus dem Ursprungsimage erzeugt, der in unserem Beispiel GEN01 lautet. Es kann daher sinnvoll sein, hier eine individuelle Benamung zu wählen. Jetzt sollte noch die Imagegröße festgelegt werden. Wir belassen es bei der Standardeinstellung und überlassen es dem System, die Größe aufgrund des Einheitentyps zu bestimmen.

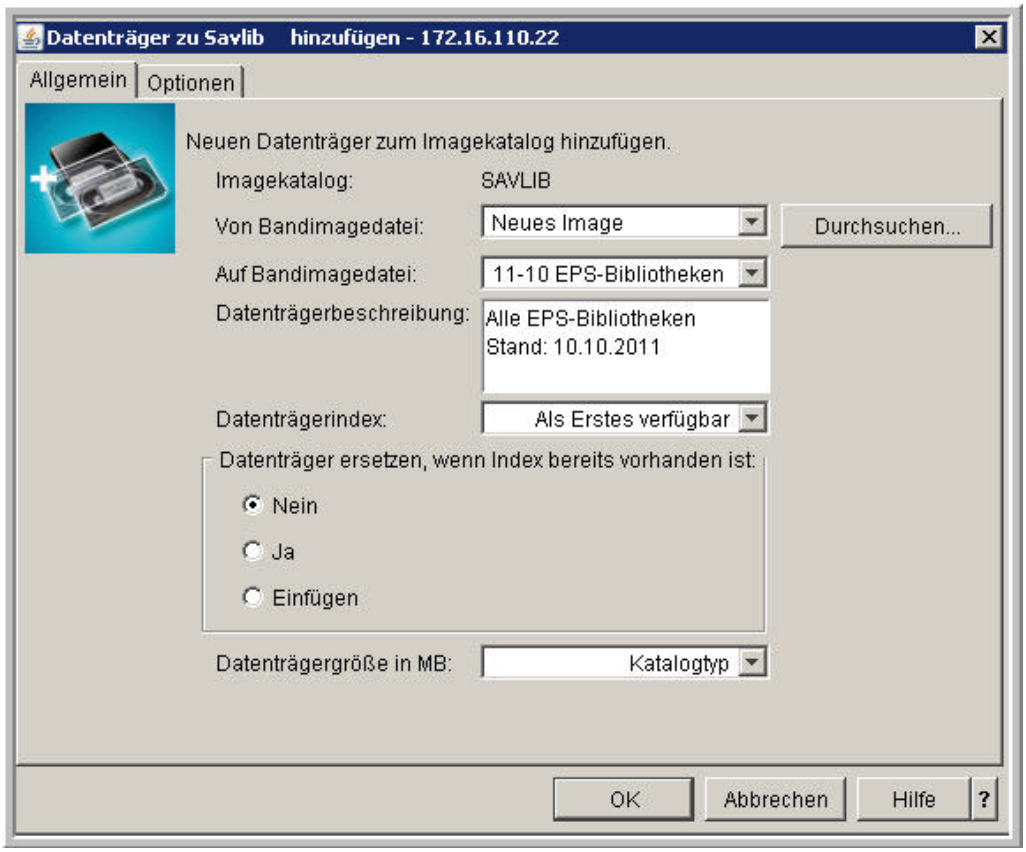
Das Laden eines Imagekatalogeintrags wird auch wieder über den System i Navigator ausgeführt.

Dazu öffnen Sie erneut den Eintrag „Hardware ➜ Bandeinheiten ➜ Bandimagekataloge“ und nutzen die Kontexttaste für das entsprechende Bandlaufwerk. In diesem Dialog wählen Sie – wie nachfolgend gezeigt – den Eintrag „Datenträger hinzufügen“:

Datenträger mit dem System i Navigator hinzufügen

Jetzt ist der Dialog dem 5250-Interface sehr ähnlich. Sie wählen ein neues virtuelles Band, indem Sie den Wert „Neues Image“ eingeben. Natürlich können Sie auch Daten zurückspeichern. Für diesen Fall durchsuchen Sie den IFS-Bereich und laden den Datenträger, der die notwendigen Informationen enthält. Doch dazu später mehr!

Jetzt wollen wir in einem ersten Schritt eine Datensicherung ausführen und alle Bibliotheken sichern, die mit der Zeichenfolge EPS beginnen.

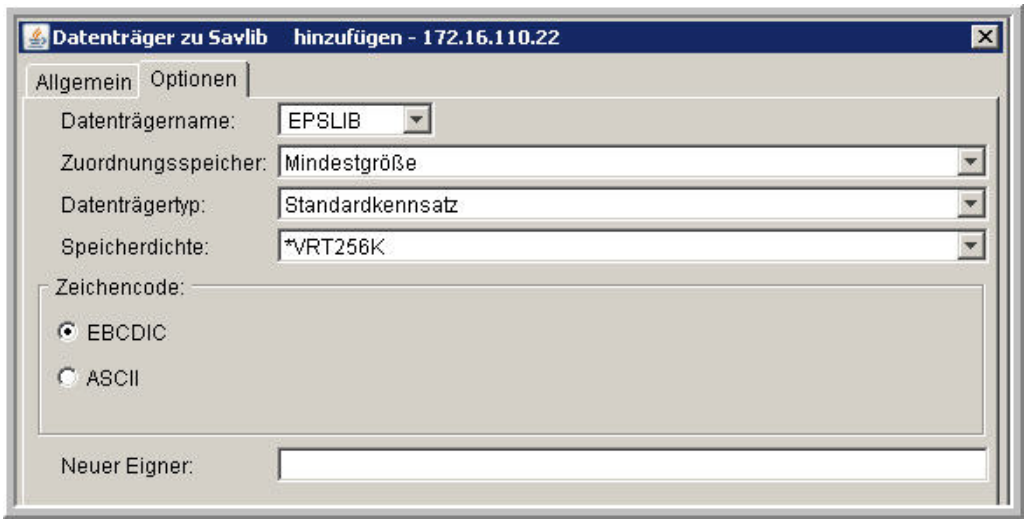


Datenträger hinzufügen – Allgemein

Wie Sie in der Abbildung oben sehen, benötigen wir eine neue Imagedatei. Es folgt der Parameter „Auf Bandimagedatei“. Hier können wir den Namen wieder generieren oder eine Bezeichnung vorgeben. Letztendlich benennen wir mit diesem Parameter das IFS-Verzeichnis, das später die Sicherung enthält. Und da IFS-Verzeichnisnamen auf maximal 255 Zeichen begrenzt sind, stehen uns auch jetzt 255 Zeichen für die Beschriftung zur Verfügung.

2.9.6.3
Seite 4

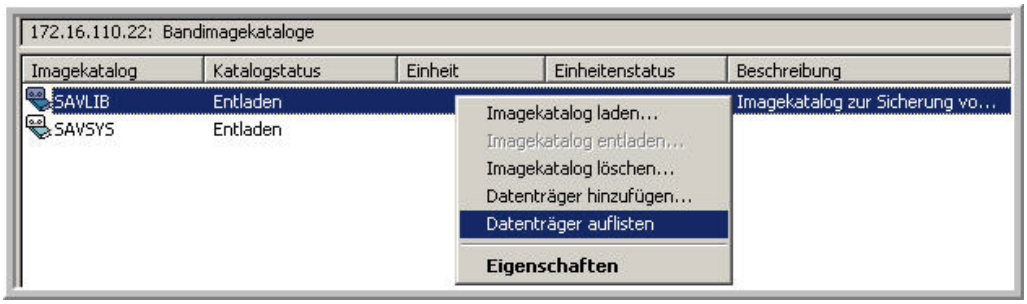
Auf der zweiten Seite dieses Dialogs können wir den Datenträgernamen zuweisen:



Datenträger hinzufügen – Optionen

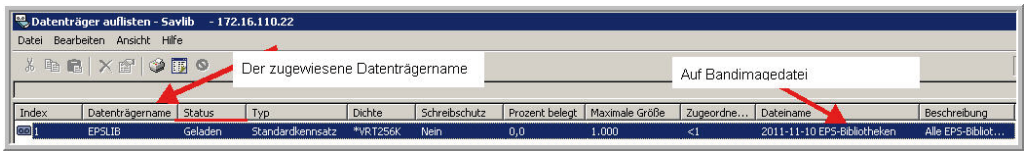
Ähnlich einem INZTAP-Befehl ist es hier möglich, den Datenträgernamen und die Eigner-ID zuzuweisen. Der Datenträgername kann eine Länge von 1 bis 6 Zeichen haben und darf nur alphanumerische Zeichen (A bis Z, $, #, @ sowie 0 bis 9) sowie kein Präfix oder Leerzeichen enthalten. Jeder Datenträgername im Imagekatalog muss eindeutig sein. Schreibdichte und Zeichensatz kennen Sie ebenfalls vom INZTAP-Befehl. Falls es erforderlich ist, weisen sie hier dem virtuellen Band eine Eigner-ID zu. Ich belasse es im Beispiel bei der Vergabe eines Datenträgernamens, alle weiteren Parameter übernehme ich unverändert. Anschließend bestätige ich den Dialog.

Nun wollen wir das Ergebnis überprüfen und listen – wie nachfolgend gezeigt – die Bandlaufwerke dieses Imagekatalogs auf:



Datenträger auflisten

Im nachfolgenden Dialog sehen Sie den zuvor erstellten Katalogeintrag:



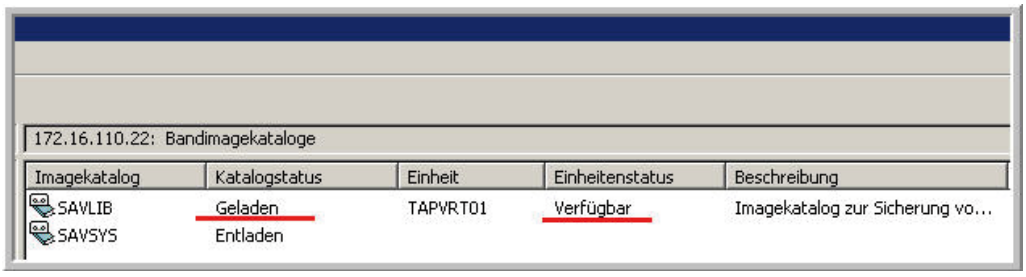
Die geladenen Datenträger

So weit, so gut! Nun muss man dem Imagekatalog die virtuelle Bandeinheit zuordnen. Im 5250-Dialog geschah dies in einem Schritt. Im System i Navigator muss ein weiterer Dialog geöffnet werden, um die Verbindung herzustellen:

Datenträger laden

Oben stehender Dialog ist selbsterklärend und muss nicht weiter erläutert werden.

Wenn alle Dialoge korrekt bearbeitet sind, dann ist sowohl das virtuelle Band als auch das virtuelle Bandlaufwerk für die anschließende Sicherung vorbereitet, wie Sie in der folgenden Übersicht sehen:



Ergebnis der Vorbereitungen

Jetzt kann die Sicherung durchgeführt werden.

2.9.6.3
Seite 6

Bibliotheken auf virtuelle Bandlaufwerke sichern

Virtuelle Tapes können sehr nützlich sein, wenn es darum geht, Daten zu sichern. Diese Objekte können nicht nur für Ad-hoc-Sicherungen eingesetzt werden, sondern auch für Systemsicherungen. Außerdem werden virtuelle Tapes einschließlich der virtuellen Tape Devices auch in einer BRMS-Umgebung unterstützt.

An einem Beispiel soll festgestellt werden, welche Schritte für die Sicherung erforderlich sind. Ziel ist es – wie bereits angekündigt –, einige EPS-Bibliotheken auf unserem Server zu sichern.

Im folgenden SAVLIB-Befehl verwende ich einfach die virtuelle Bandeinheit, die wir im vorangegangenen Kapitel vorbereitet haben:

```
                        Bibliothek sichern (SAVLIB)

Auswahl eingeben und Eingabetaste drücken.

Bibliothek . . . . . > EPS002CR     Name, generisch*, *NONSYS...
     + für weitere Werte          > EPSW07
Einheit . . . . . > TAPVRT01     Name, *SAVF, *MEDDFN
     + für weitere Werte
Datenträger-ID . . . . . *MOUNTED
     + für weitere Werte
```

Der Sicherungsprozess

Weitere Schritte sind nicht erforderlich, da die Einheit und der Imagekatalog im vorangegangenen Kapitel entsprechend vorbereitet wurden.

Und tatsächlich: Nach erfolgreicher Sicherung bekommt man die bereits aus den Sicherungen auf physische Bänder bekannte Erfolgsmeldung CPC3721:

```
+>> SAVLIB LIB(EPS002CR EPSW07) DEV(TAPVRT01)
   Bibliothek QBRM der Bibliotheksliste hinzugefügt.
   Datenträger GEN001 nicht unter Datenträgerverwaltungssteuerung.
   59 Blöcke für Sequenz 1, Datenträger GEN001, auf Einheit TAPVRT01
     verarbeitet.
   Empfänger TT1065 in EPS002CR gesichert, während er zugeordnet war.
   206 Objekt(e) aus Bibliothek EPS002CR gesichert.
   78 Blöcke für Sequenz 2, Datenträger GEN001, auf Einheit TAPVRT01
     verarbeitet.
   Empfänger QSQJRN0001 in EPSW07 gesichert, während er zugeordnet war.
   1 logische Zugriffspfade gesichert oder zurückgespeichert.
   98 Objekt(e) aus Bibliothek EPSW07 gesichert.
   2 Bibliothek(en) gesichert.
```

Das Ergebnis der Datensicherung

Beide Bibliotheken wurden erfolgreich gesichert! Die Daten liegen im IFS im vereinbarten Verzeichnis „/SAVE/LIB/“. Sie müssen aber nicht diesen Weg gehen, um den Bandinhalt zu prüfen. Wie bisher können Sie einfach den

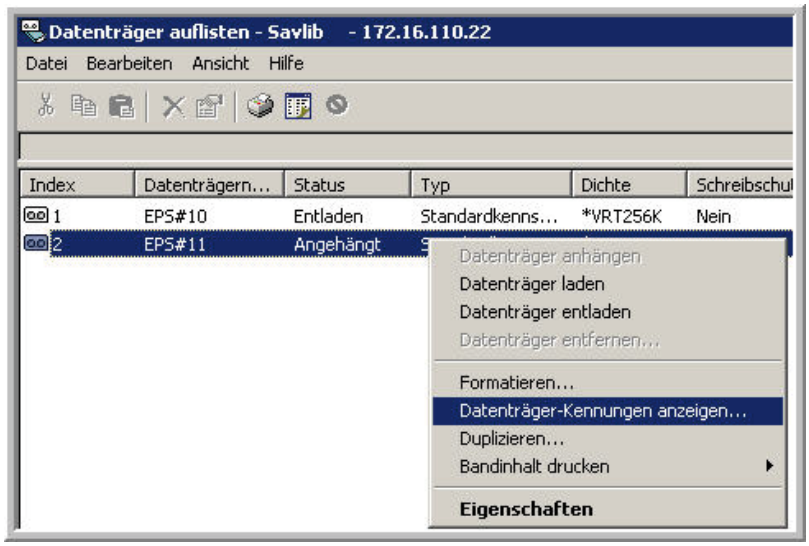
2.9.6.4
Seite 2

5250-Befehl „Bandlaufwerk anzeigen (DSPTAP)“ verwenden, um den Bandinhalt zu prüfen:

```
                  Informationen über Banddatenträger anzeigen

Einheit . . . . . :   TAPVRT01          Datenträger-ID . . :   EPSLIB
Eigner-ID . . . . :                     Schreibdichte  . . :   *VRT256K
Art . . . . . :   *SL                   Code . . . . . :   *EBCDIC


Kennsatz der Datendatei . . . . . :   EPS002CR
Dateifolge . . . . . :   0000000001
```

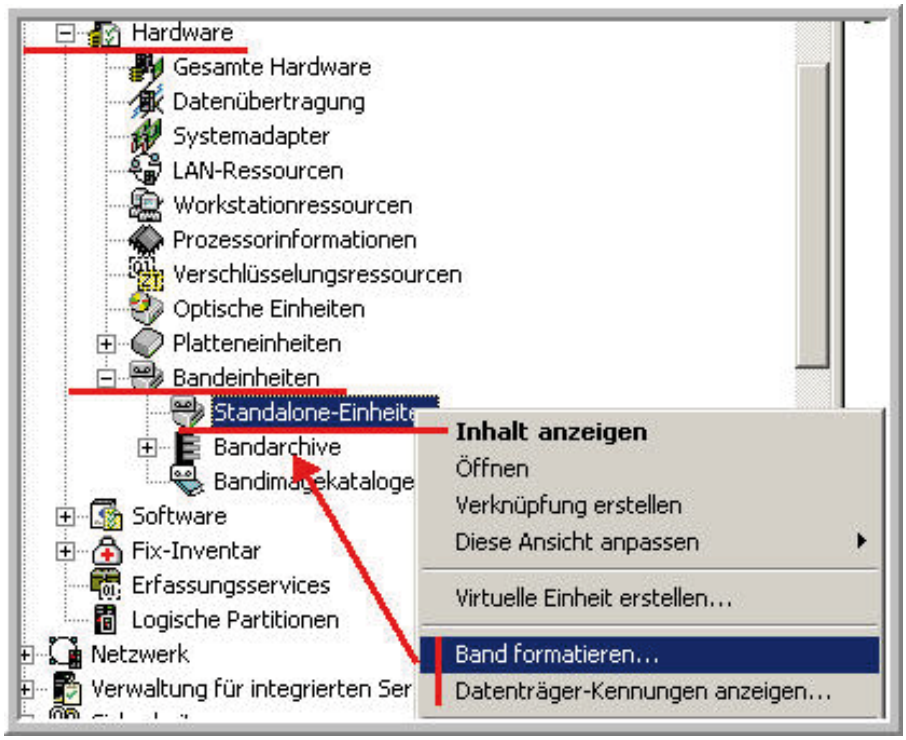
Befehl „Display Tape (DSPTAP)“ – Überprüfung der Datensicherung

Diesen Befehl stellt natürlich auch der System i Navigator zur Bearbeitung virtueller Bänder zur Verfügung:



Virtuelle Bänder mit System i Navigator bearbeiten

Und falls Sie physische Bänder formatieren oder anzeigen möchten, geht das natürlich auch:



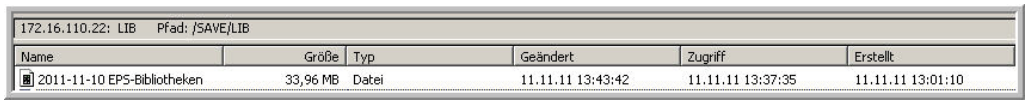
Physische Bänder mit dem System i Navigator verwalten

Anhand dieses einfachen Beispiels erkennen Sie bereits einen entscheidenden Vorteil bei der Verwendung von virtuellen Tapes:

- Bei der Verwendung von virtuellen Tapes kann man mehrere Bibliotheken „gleichzeitig“ sichern, während man bei der Verwendung von Save Files (*SAVF) nur eine Bibliothek zur gleichen Zeit sichern kann.
- Außerdem muss zum Zeitpunkt der Sicherung kein physisches Bandlaufwerk vorhanden sein. Sie können die virtuellen Tapes zu einem beliebigen Zeitpunkt mit dem Befehl DUPTAP (Duplicate Tape) auf physische Bänder kopieren. Das physische Bandlaufwerk kann also zum Zeitpunkt Ihrer Sicherung von einer anderen Partition oder einem anderen System verwendet werden. Vielleicht lässt sich auf diesem Weg die Gesamtanzahl der physischen Bandlaufwerke reduzieren.

Aber es gibt natürlich auch Nachteile:

- Für die virtuellen Tapes benötigen Sie zusätzlichen Plattenplatz, da die Imagedatei im IFS Ihres System i Servers abgelegt wird. Unsere Beispiel-Datensicherung hat aktuell eine Größe von 33,96 MB.


172.16.110.22: LIB Pfad: /SAVE/LIB

Name	Größe	Typ	Geändert	Zugriff	Erstellt
2011-11-10 EPS-Bibliotheken	33,96 MB	Datei	11.11.11 13:43:42	11.11.11 13:37:35	11.11.11 13:01:10

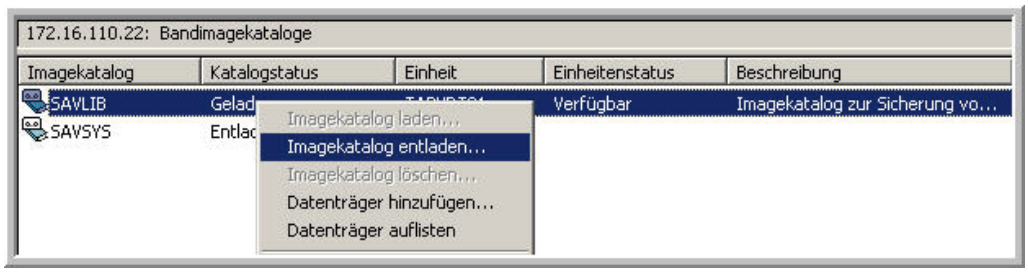
Die Imagedateien benötigen Plattenplatz

2.9.6.4
Seite 4

Für weitere Datensicherungen kann anschließend das geladene Image verwendet oder einen neuer Katalogeintrag erstellt werden. Dieser Vorgang gleicht der Arbeit mit physischen Bändern: entweder Sie benutzen das eingelegte Band für eine weitere Sicherung oder Sie legen einen neuen Datenträger ein.

Virtuelle Bänder wechseln

Wenn das virtuelle Band gewechselt werden soll, ist es am einfachsten, wenn zunächst die Verbindung zwischen dem virtuellen Bandlaufwerk und dem Imagekatalog getrennt wird, indem Sie den Imagekatalog „entladen“:

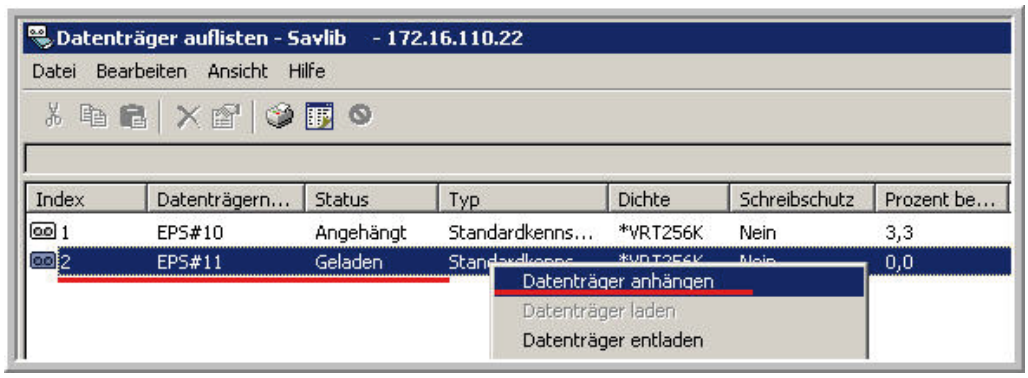


Imagekatalog entladen

Erst danach sollte in einem weiteren Schritt ein neuer Datenträger angelegt werden:

Neuen Datenträger hinzufügen

Anschließend müssen die verfügbaren Datenträger geprüft werden:

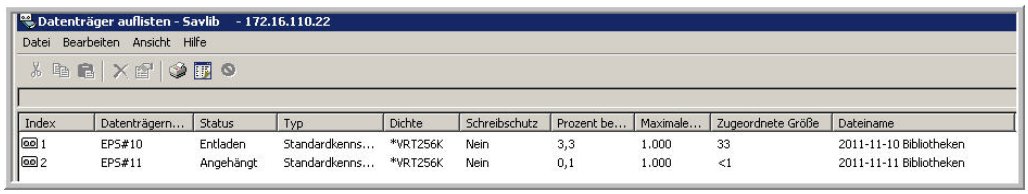


Mehrere Datenträger sind vorhanden

Wie erwartet – sind jetzt zwei virtuelle Datenträger vorhanden. Angehängt ist der alte Datenträger EPS#10. Am einfachsten ist es hier, den neuen Datenträger zu markieren und über das Kontextmenü den Eintrag „Datenträger anhängen“ zu wählen. Sofort wird der alte Datenträger abgehängt. Aber Vorsicht! Der alte Datenträger ist nach wie vor geladen und sollte der angehängte Datenträger für die Sicherung nicht ausreichen, würden die geladenen Datenträger zusätzlich verwendet werden. Um also ganz sicher zu gehen, dass das alte Band nicht verwendet wird, müssen Sie es entladen.

2.9.6.5
Seite 2

In der folgenden Abbildung sehen Sie, dass das neue Band geladen und angehängt und das Band mit der alten Sicherung entladen ist:



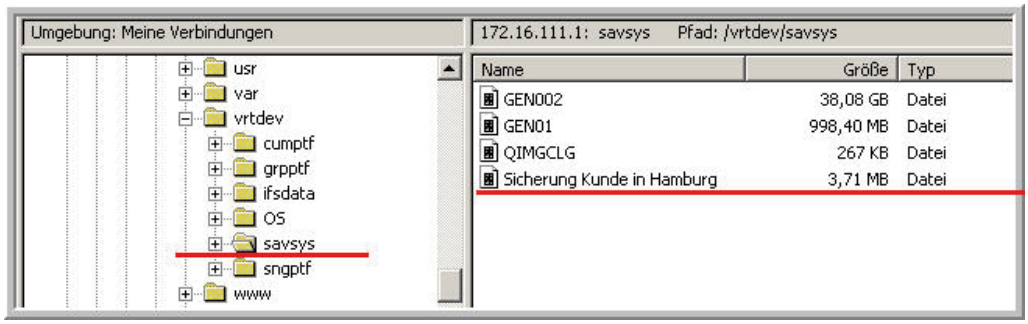
Das Band ist gewechselt

Es bleibt Ihnen überlassen, ob Sie die entladene Imagedatei löschen oder vielleicht in der nächsten Woche wiederverwenden, indem Sie es vor der Sicherung formatieren, so wie Sie es vermutlich mit Ihren physischen Bändern auch tun würden.

Datenrücksicherungen

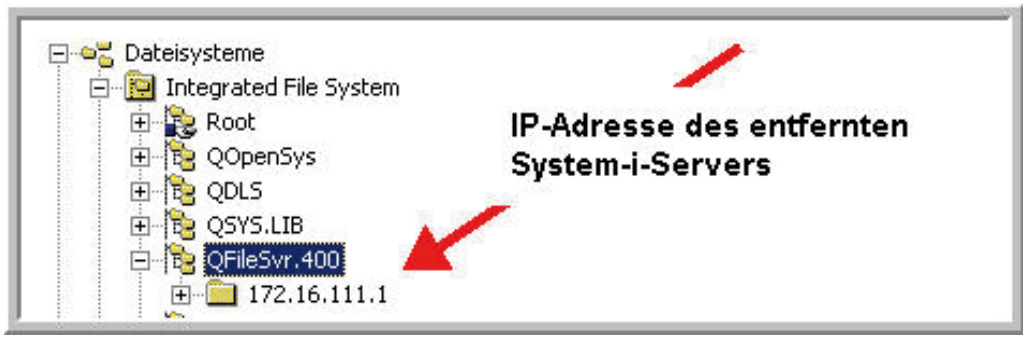
Wie bereits erwähnt, können wir die virtuellen Bänder natürlich auch nutzen, um Daten wiederherzustellen. Diesen Weg wollen wir uns jetzt gemeinsam anschauen.

Allerdings liegt in diesem Beispiel die Datensicherung auf einem anderen Server. Nehmen wir an, dass auf dem Server ITPSRV die Bibliothek eines Hamburger Kunden auf virtuellen Bändern gesichert wurde, die jetzt im IFS-Verzeichnis dieses Servers liegt:



Die Ausgangssituation

Nun soll dieser Imagekatalogeintrag mit unserem virtuellen Bandlaufwerk verbunden werden. Um über Servergrenzen hinweg zu agieren, wird das IFS-Verzeichnis „QFile.svr.400“ auf unserem EPOS-Server genutzt. Das QFilesvr.400-Dateisystem ermöglicht den direkten Zugriff auf die Daten eines anderen System i Servers:



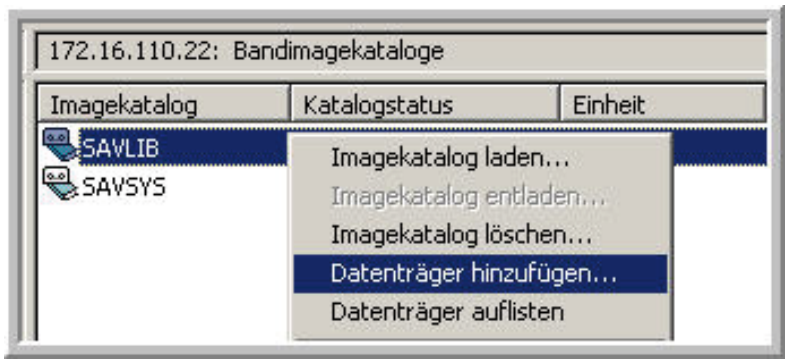
QFilesvr400 nutzen

Der Zugriff auf den entfernten Server ist sehr einfach. Sie erstellen lediglich ein neues Verzeichnis im QFilesvr.400-Dateisystem mit der IP-Adresse oder dem Netzwerknamen des zweiten Servers. Achten Sie hierbei auf Groß- und Kleinschreibung. Das Dateisystem QFilesvr.400 ist case-sensitive. Voraussetzung für die Nutzung ist zudem das Subsystem QSERVER, das gestartet sein muss. Und Ihr aktueller Benutzername muss auf dem entfernten Server vorhanden sein; eine separate Anmeldung erfolgt nämlich nicht. Weitere Konfigurationen oder Informationen benötigt man nicht, um mit QFileSvr.400 zu

2.9.6.6
Seite 2

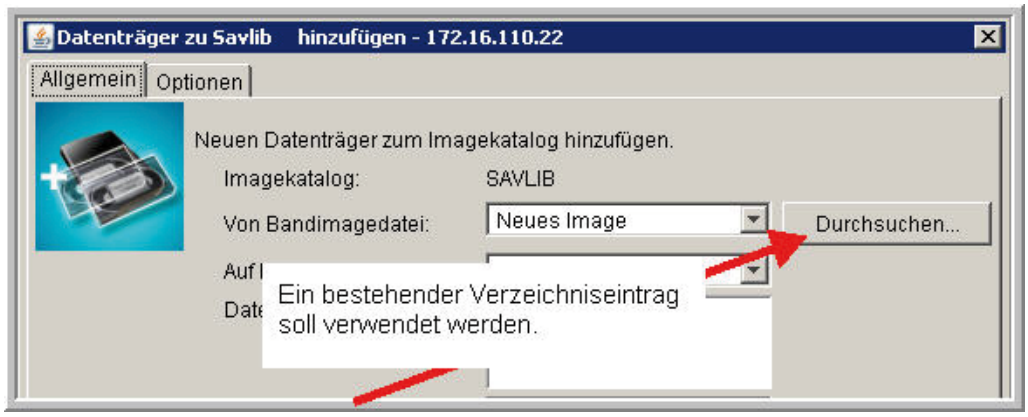
arbeiten. Sie können jetzt einfach den erstellten Verzeichniseintrag expandieren und bewegen sich mühelos auf dem ITPSVR. Dort finden Sie hoffentlich auch die Imagedatei, die die Datensicherung enthält.

Auf eine separate Prüfung wird verzichtet. Dafür wird sofort für den Imagekatalog ein neuer entsprechender Verzeichniseintrag erstellt:



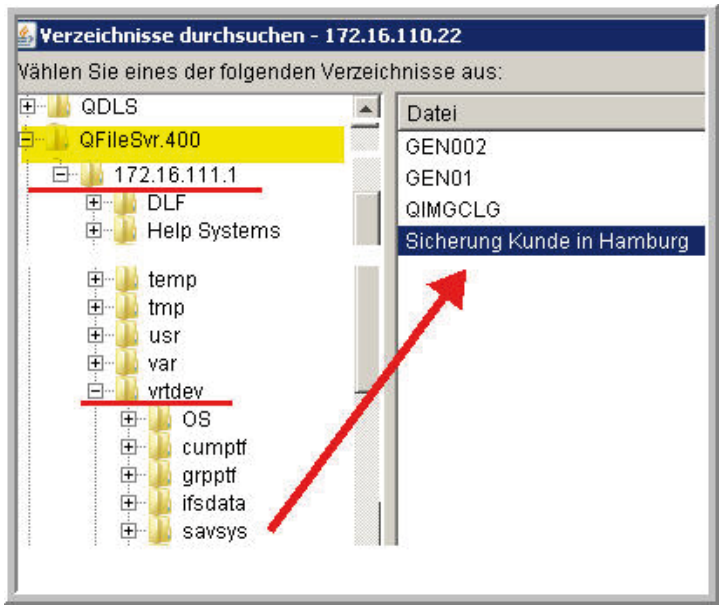
Verzeichniseintrag hinzufügen

Da wir den bereits vorhandenen Imagekatalogeintrag des ITPSRV verwenden wollen, klicken wir auf „Durchsuchen…“:



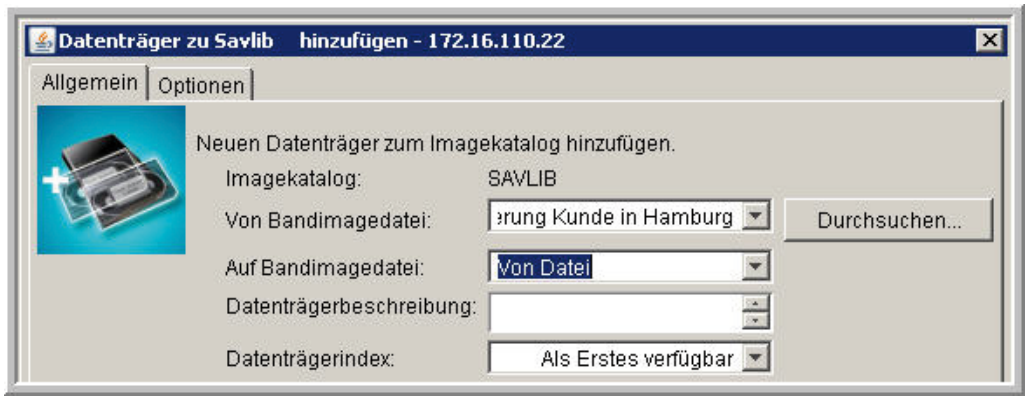
Bestehenden Verzeichniseintrag suchen

Das zuvor erstellte Verzeichnis im QFileSvr.400-Dateisystem wird geöffnet, um auf den Pfad „\QFileSvr.400\172.16.111.1\vrtdev\savsys\Sicherung Kunde in Hamburg“ zuzugreifen.



Verzeichnis des ITP-Servers durchsuchen

Der so gefundene Imagekatalogeintrag wird im Dialog mit „OK“ bestätigt. Daraufhin sollten Sie in den Ausgangsdialog zurückkehren:



Datenträger eines fernen Systems hinzufügen

Sie sehen in der Abbildung oben, dass beim Parameter „Auf Bandimagedatei“ der Wert „Von Datei“ eingestellt ist, so dass die Quell- und Zielimagedatei den gleichen Namen tragen.

Anschließend muss wieder der Imagekatalog geladen werden, damit der entsprechende RESTORE-Befehl ausgeführt werden kann.

2.9.6.6

Seite 4

Überwachung von System- und Plattenstatus

In einer Standard i5 Installation verfügt das System über verschiedene Platten. Diese Platten werden als Einheit gesehen. Achten Sie auf die Auslastung der Platten. Bereits eine AS/400 reagierte „bockig“ und kam zeitweilig zum Stillstand, wenn die Platten zu hoch ausgelastet wurden. Heute – angesichts stark gefallener Hardwarepreise – sollten Sie versuchen, nur noch 60 – 70 % der Platten zu belegen. Der Server wird es Ihnen danken durch verbessertes Laufzeitverhalten, da Teile der Platten als Arbeitsspeicher genutzt werden können. Hinzu kommt, dass das Betriebssystem, die Datenbank aber auch neue Tools wie „Webquery“ temporäre Objekte aufbauen müssen. Ein Teil der Objekte liegt im Hauptspeicher. Sie sollten daher nicht warten, bis eine Meldung des Systems ausgegeben wird, die über den knappen Plattenspeicher informiert, sondern rechtzeitig dafür sorgen, dass ausreichend Plattenplatz auf dem System zur Verfügung steht. Dafür müssen Sie allerdings Ihren Systemstatus von Zeit zu Zeit überprüfen. Im 5250-Umfeld können Sie die Auslastung Ihres Servers und den verfügbaren Plattenplatz jederzeit einsehen. Alles, was Sie dazu machen müssen, ist den Befehl „Mit Systemstatus arbeiten“ (WRKSYSSTS) aufzurufen.

```
                          Mit Systemstatus arbeiten                  EPSSRV
                                                              13.10.09  23:51:09
% CPU benutzt . . . . . :        0,6    Zusatzspeicher:
% DB-Kapazität . . . . . :        0,0      System-ASP . . . . . :    697,8 G
Abgelaufene Zeit . . . . :   04:15:10      % System-ASP benutzt . :    11,1258
Jobs im System . . . . . :        853      Gesamtsumme . . . . . :    697,8 G
% perm. Adressen . . . . :      0,007      Unges. Platz akt.
% temp. Adressen . . . . :      0,038        belegt . . . . . :     7955 M
                                           Max. ungeschützt . . . :     8879 M
Änderungen (falls zulässig) eingeben und die Eingabetaste drücken.


System    Pool-     Reserv.     Max.  -DB-Seiten--  --Nicht-DB--
 Pool   Größe(M)  Größe (M)   Aktiv  fehl.  geles  fehl.  geles
   1     369,47     179,19   +++++    0,0    0,0    0,1    0,5
   2    6379,51     132,42     141    0,0    0,1    1,0    2,9
   3     758,30       0,00     190    0,0    0,0    0,0    0,0
   4      75,82       0,00       6    0,0    0,0    0,0    0,0
```

Der Befehl WRKSYSSTS

Die aktuelle Plattenauslastung finden Sie in dem oberen rechten Anzeigenbereich. Dort ist eine Spalte „ % System ASP genutzt“, welche Auskunft über die aktuelle Plattenauslastung gibt. In unserem Beispiel ist das System mit 11,1258 % belegt – uns steht also noch genügend Plattenplatz zur Verfügung. Das ist natürlich ein sehr niedriger Auslastungswert. Sie sollten darauf achten, dass dieser Prozentwert nicht wesentlich über 70 % steigt. Warten Sie länger, dann kann dies auch Auswirkungen auf die gesamte Performance der Maschine haben. In jedem Fall ist eine Plattenbelegung von mehr als 90 % über einen längeren Zeitraum unbedingt zu vermeiden!

2.9.7
Seite 2

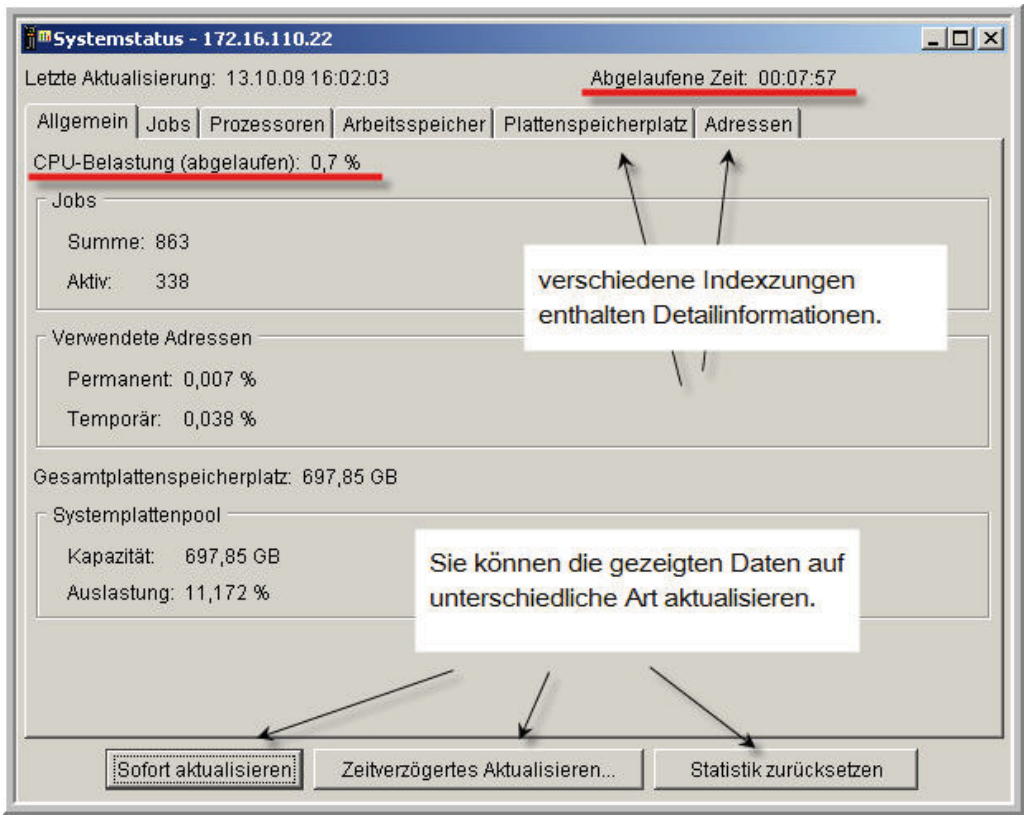
Falls Sie nicht die 5250-Befehle nutzen möchten, sondern lieber mit dem iSeries Navigator arbeiten wollen, so ist auch dies möglich. Der iSeries Navigator hat mit Release V6R1 im Übrigen einen neuen Namen erhalten und heißt jetzt System i Navigator und das grafische Interface hat Konkurrenz bekommen: IBM Systems Director Navigator for i5/OS heißt die neue Alternative. IBM Systems Director Navigator for i5/OS bietet eine benutzerfreundliche Schnittstelle für webfähige System i-Verwaltungstasks im Web und soll mittelfristig den System i Navigator ersetzen. Auch wenn bislang mehr als 800 Funktionen in den IBM Systems Director Navigator integriert wurden, werden bislang nicht alle System i Navigator Funktionen abgebildet. Auch das Laufzeitverhalten dieses Tools lässt derzeit noch Wünsche offen, so dass meiner Meinung nach auch im Release V6R1 der System i Navigator die eigentliche Alternative zur 5250-Oberfläche darstellt. Detaillierte Informationen zum System Director Navigator for i5/OS finden Sie übrigens im Kapitel 7.7.3.

Kommen wir aber wieder auf die Kernfrage zurück: Wie viele Jobs laufen auf dem Server und wie hoch ist die Plattenauslastung? Um sich einen Überblick über die Auslastung des Servers zu verschaffen, markieren wir den Eintrag „Ablaufsteuerung“ im System i Navigator und finden im Kontextmenü den Menüpunkt „Systemstatus“, wie folgende Abbildung zeigt:

Kontextmenü „Ablaufsteuerung“

Anschließend öffnet sich der Dialog „Systemstatus“.

Der Dialog zeigt die für diesen Zeitpunkt geltenden statistischen Informationen Ihres Systems. Auf der Indexzunge „Allgemein“ finden Sie allgemeine Systemstatuswerte, z. B. abgelaufene CPU-Belastung, Anzahl aller Jobs und der aktiven Jobs im System, verwendete permanente und temporäre Adressen, Gesamtplattenspeicherplatz sowie Kapazität und Auslastung des Systemplattenpools. Wir können somit der oberen Abbildung entnehmen, dass am 13.10.2009 um 16:02 Uhr 863 Jobs auf dem Server vorhanden waren, davon 338 aktiv. Aktive Jobs sind Arbeitsaufträge, die bereits ausgeführt werden, aber noch nicht abgeschlossen sind. In einer Jobwarteschlange wartende Jobs und beendete Jobs, die auf das Drucken einer Spool-Datei warten, sind dagegen keine aktiven Jobs. Außerdem können Sie die Plattenauslastung ermit-

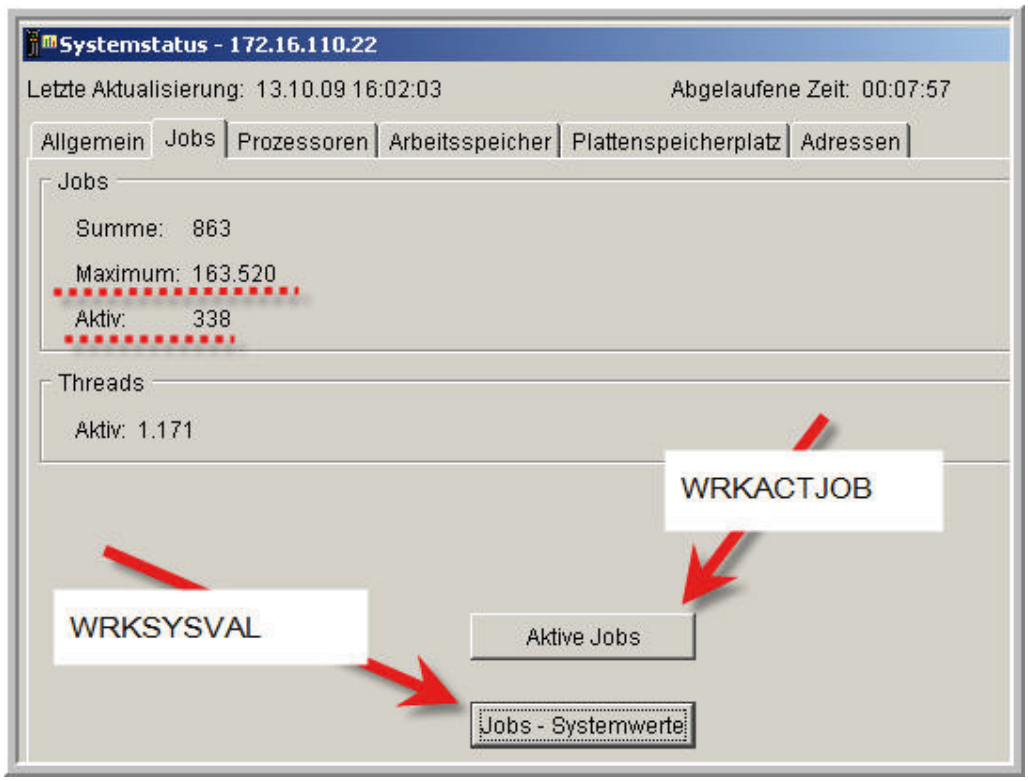


Der Dialog „Systemstatus“

teln. Der dargestellte Server hat 697,85 GB Plattenkapazität, wovon 11,172 % belegt sind. Alle Daten, die Sie sehen können, sind zeitpunktbezogen. Eine Ausnahme bildet hierbei nur das Feld „CPU-Belastung“, das die durchschnittliche Prozessorauslastung eines abgelaufenen Zeitraumes anzeigt. In unserem Beispiel betrug die Prozessorauslastung in den vergangenen sieben Minuten durchschnittlich 0,7%. Natürlich können Sie die statistischen Werte jederzeit aktualisieren. Wenn Sie auf die Schaltfläche „Sofort aktualisieren“ klicken, werden die jeweiligen Werte geändert und spiegeln den aktuellen Status Ihres Systems wieder. Dabei wird auch das Feld „Letzte Aktualisierung“ aktualisiert und zeigt die präzise Zeit an, zu der Sie auf die Schaltfläche „Sofort aktualisieren“ geklickt haben. Anhand des Feldes „Abgelaufene Zeit“ können Sie erkennen wie viel Zeit zwischen dieser Aktualisierungsaktion und dem Öffnen des Dialogs oder dem letzten Klicken auf „Statistik zurücksetzen“ verstrichen ist. Die Aktualisierungszeitpunkte können Sie auch automatisieren, wenn Sie z. B. über mehrere Stunden die Auslastung des Servers verfolgen wollen. Hierzu nutzen Sie den Dialog „Zeitverzögertes Aktualisieren“. In dem sich öffnenden Dialog können Sie angeben in welchen Zeitintervallen (in Minuten) die Statuswerte aktualisiert werden sollen. Ähnlich wie beim Klicken auf „Sofort aktualisieren“ werden anschließend die bereits dargestellten Tasks automatisch in regelmäßigen Intervallen ausgeführt. Falls Sie zwi-

schenzeitlich die Statistikwerte zurücksetzen wollen, klicken Sie auf „Statistik zurücksetzen“. Das Feld „Abgelaufene Zeit“ und das Feld „CPU-Belastung (abgelaufen)“ wird auf 0.0 festgelegt.

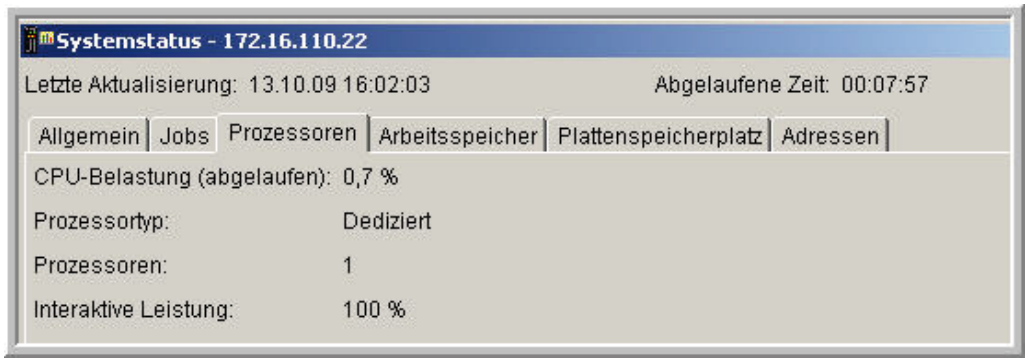
Neben den allgemeinen Informationen finden wir auf der Indexzunge „Jobs“ die Summe der derzeit im System vorhandenen Benutzer- und Systemjobs.



Indexzunge „Jobs“

Ich denke, die Anzeige ist weitest gehend selbsterklärend. Vielleicht noch eine kurze Erläuterung des Begriffs „Thread“. Threads sind unabhängige Arbeitseinheiten in einem System, die einem Job zugeordnet sind. Jeder Job besitzt immer mindestens einen Thread, den so genannten Initial-Thread. Man könnte auch sagen, dass ein Job auf mindestens einen oder auf mehrere Threads bzw. Prozesse verteilt wird, damit ggf. verschiedene Arbeitsschritte gleichzeitig erfolgen können. Beispielsweise könnte ein Thread bestimmte Berechnungen ausführen, während ein anderer auf die Eingabe zu verarbeitender Daten wartet. Erwähnenswert ist in diesem Zusammenhang noch die maximale Anzahl auszuführender Jobs (MAXIMUM). Wenn dieser Maximalwert erreicht wird, können Sie keine weiteren Jobs starten. Dieser Wert wird im Übrigen durch den Systemwert „QMAXJOB“ gesteuert. Hilfreich finde ich im System i Navigator in diesem Zusammenhang, dass Sie alle zugehörigen Systemwerte anzeigen können, wenn Sie auf „Jobs-Systemwerte“ klicken.

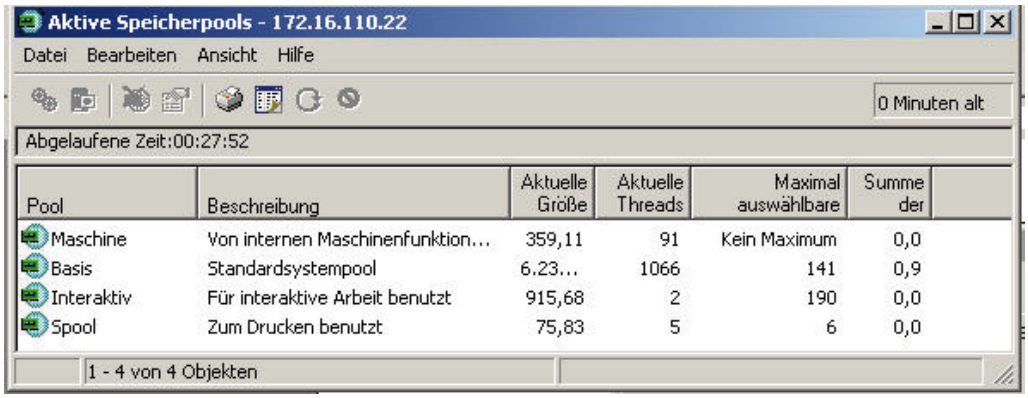
Die Indexzunge „Prozessoren“ zeigt die CPU-Belastung und die Anzahl der Prozessoren im System.



Indexzunge „Prozessoren“

Dargestellt wird zunächst wiederum die durchschnittliche CPU-Belastung im gemessenen Zeitintervall. Bei Mehrprozessorsystemen gibt dieser Prozentsatz die durchschnittliche Auslastung aller Prozessoren an. Arbeiten Sie mit mehreren Prozessoren und logischen Partitionen, erhalten Sie auch hierzu entsprechende Informationen wie beispielsweise den Auslastungsgrad des gemeinsam genutzten Prozessorpools oder die zugeordnete Größe der logischen Partitionen.

Auf der Indexzunge „Arbeitsspeicher“ sehen Sie die Gesamtgröße Ihres Arbeitsspeichers und können von dort direkt auf die aktiven Speicherpools zugreifen, um festzustellen, wie sich der Arbeitsspeicher auf die Pools verteilt.

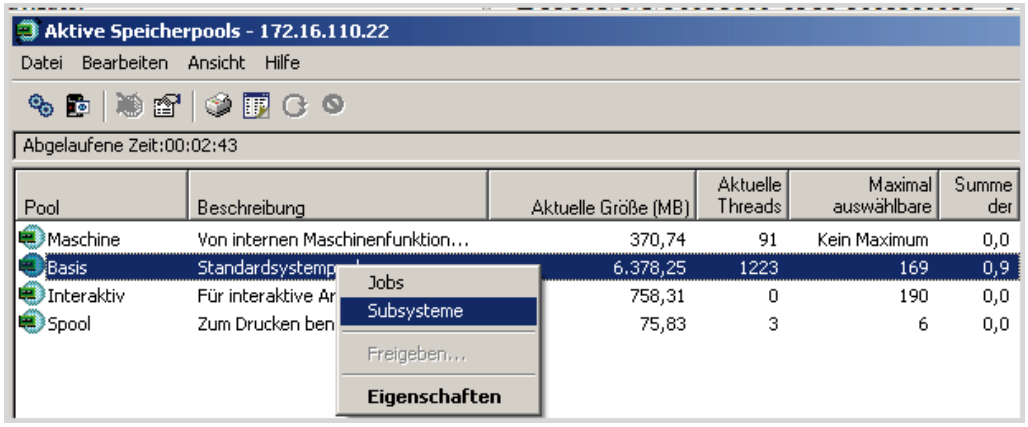


Aktive Speicherpools

Der oberen Grafik kann ich entnehmen, dass der Maschinenpool derzeit 359 MB des Arbeitsspeichers belegt, während der Basispool – also u. a. das Subsystem QBATCH – 6.234 MB Speicher belegt. Der interaktive Pool (Subsystem QINTER) und der Spoolpool teilen sich die restlichen 990 MB des Arbeitsspeichers. Wollen Sie wissen, welche Subsysteme den Pools zugeordnet sind?

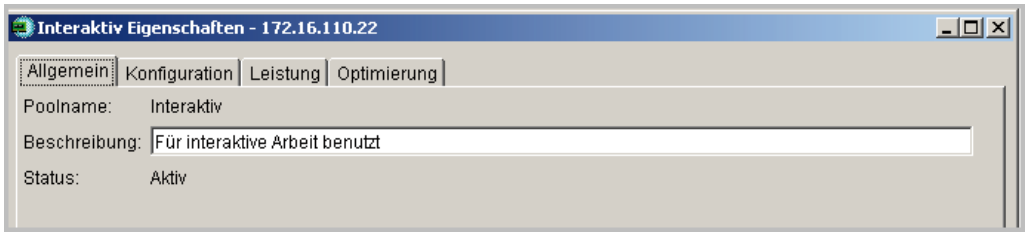
2.9.7
Seite 6

Markieren Sie einen der Pooleinträge und öffnen Sie anschließend das Kontextmenü. Folgende Grafik zeigt, was gemeint ist:



Welche Subsysteme sind dem Pool zugeordnet?

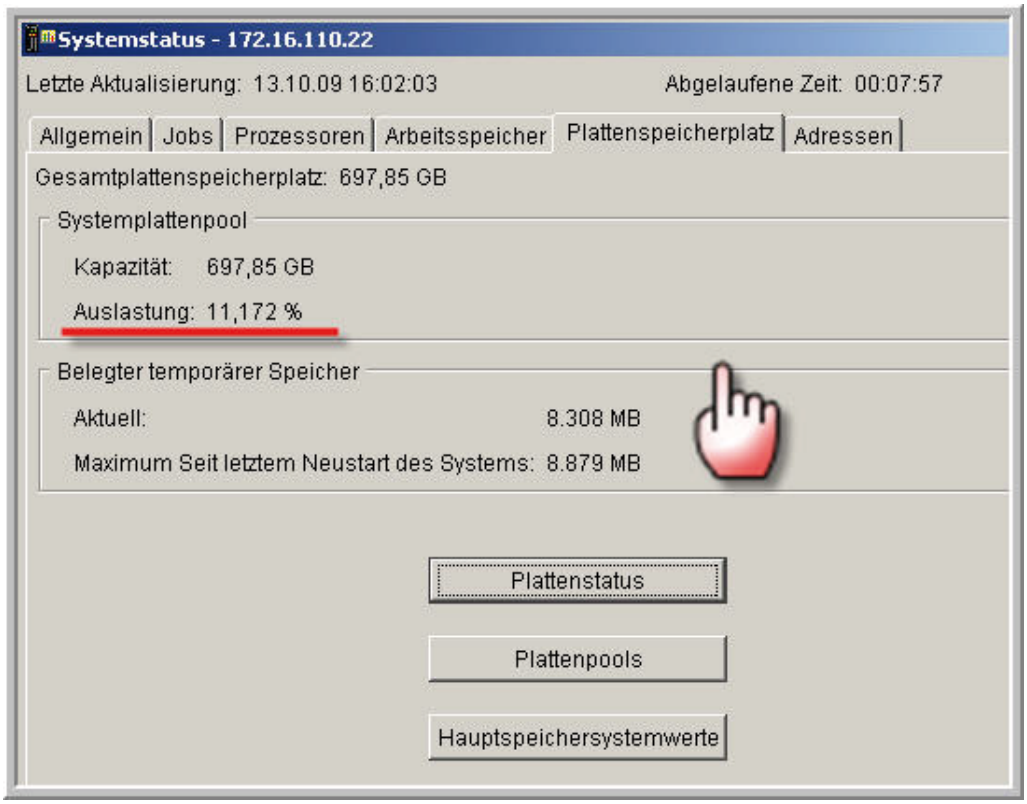
Die Grafik zeigt, dass Sie sich nicht nur die Subsysteme anzeigen können, sondern es auch möglich ist die Jobs zu selektieren, die derzeit im markierten Pool aktiv sind. Und natürlich können Sie sich die Konfiguration und einige Leistungswerte des Speicherpools anzeigen: Hierzu öffnen Sie den Menüpunkt „Eigenschaften“:



Konfiguration und Leistungsdaten des aktiven Pools anzeigen

Seien Sie doch einfach mal neugierig und schauen Sie sich in Ruhe selbst einmal die vielen Verknüpfungsmöglichkeiten an, die der System i Navigator Ihnen zur Verfügung stellt. Es würde den Umfang dieses Kapitels sprengen, wenn ich an dieser Stelle auch noch die vielen Unterregister detailliert erläutern würde.

Kehren wir also wieder zurück zur ersten Ebene und damit zum Register „Plattenspeicherplatz:



Indexzunge „Plattenspeicherplatz“

Auf der Indexzunge „Plattenspeicherplatz“ können Sie die Größe des Plattenspeicherplatzes Ihres Systems anzeigen. Sie können von dieser Indexzunge ausgehend:

- den Status jeder einzelnen Platte ermitteln,
- auf die Plattenpools zugreifen
- und mit allen Systemwerten arbeiten, die die Größe und Verwendung des Plattenspeichers bestimmen.

Sie sehen, dass unser Server 697 GB Plattenspeicher hat, davon sind 11,172 % belegt. Der aktuell für temporäre Objekte im Gebrauch befindliche Hauptspeicher beträgt auf unserem System derzeit 8,3 MB. Maximal wurden seit dem letzten IPL 8.879 MB Hauptspeicher für temporäre Objekte benötigt. Einige der temporären Objekte können aber auch auf den Seitenspeicher – die Festplatte – verschoben werden und entlasten somit den Hauptspeicher. Erreicht Ihre Plattenauslastung den Grenzwert von 95 % wird der Operator benachrichtigt und alle leeren Spooldateien werden gelöscht, wenn sie älter als acht Tage sind. Diese Aktion wird durch verschiedene Systemwerte bestimmt. Der Systemwert QSTGLOWACN bestimmt hierbei die auszuführende Aktion, der Systemwert QSTGLOWLMT legt dabei die Obergrenze fest.

Der Systemwert QSTGLOWLMT

Im Grunde genommen bestimmen Sie mit dem Systemwert die Untergrenze des Zusatzspeicherpools; also den Bereich, der auf der Festplatte mindestens als zusätzlicher Arbeitsspeicher zur Verfügung stehen soll. In unserem Beispiel sollen mindestens 5 % der Platte als Arbeitsspeicher verwendet werden. Das bedeutet im Umkehrschluss, dass nur 95 % der Platte belegt werden dürfen. Wird der Grenzwert überschritten, legen Sie die Maßnahmen mit dem Systemwert QSTGLOWACN fest.

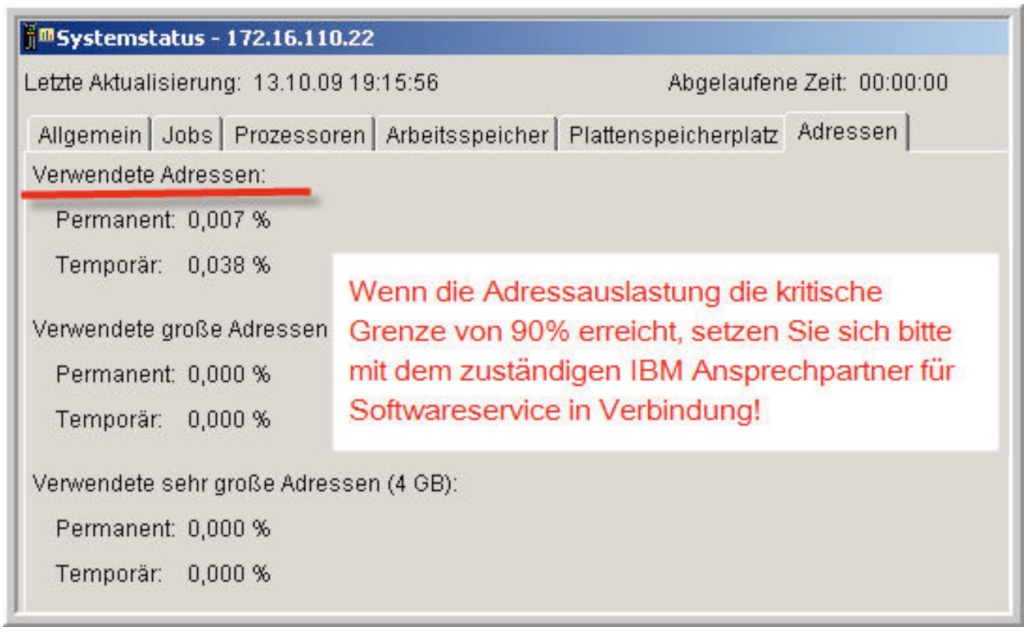
Der Systemwert QSTGLOWACN

Grundsätzlich wird immer eine entsprechende Nachricht an die Nachrichtenwarteschlangen QSYSMSG und QSYSOPR gesendet. Verändern Sie die Defaulteinstellungen Ihres Systems nicht, ist dies die einzige Aktion. Alternativ stehen weitere Aktionen zur Verfügung. Entsprechende Informationen hierzu entnehmen Sie bitte der Bedienerhilfe und den entsprechenden IBM Handbüchern. Grundsätzlich empfiehlt es sich bei stark belasteten Systemen den Grenzwert weiter nach unten zu verlegen, denn falls wirklich einmal 95 % der Platte belegt sind, ist der Zustand bereits äußerst kritisch und es bleibt nur noch wenig Reaktionszeit. Besser Sie lassen sich bereits informieren, wenn 70 oder 80 % der Platte ausgelastet sind. So bleibt genügend Zeit nicht mehr benötigte Objekte zu löschen oder neue zusätzliche Platten zu integrieren.

Tipp:

Ändern Sie die Einstellung des Systemwertes QSTGLOWACN auf den Wert 20. So steht genügend Reaktionszeit für geeignete Maßnahmen zur Verfügung.

Die sechste Indexzunge „Adressen“ zeigt Ihnen, wie die Adressen auf dem System genutzt werden.



Indexzunge „Adressen“

Wenn ein permanentes oder temporäres Objekt auf dem System erstellt wird, wird diesem Objekt von der Plattenspeicherverwaltung eine Adresse zugewiesen. Das System verwendet diese Adresse, um den Bezug zwischen dem Objekt und dem Plattenspeicher herzustellen. Es ist wichtig, den Prozentsatz der im System belegten Adressen regelmäßig zu kontrollieren. Wenn die Zunahme der Adressennutzung dazu führt, dass Sie die kritische Grenze von 90 % erreichen, setzen Sie sich bitte mit dem zuständigen IBM Ansprechpartner für Softwareservice in Verbindung.

Sie sehen, den Systemstatus können Sie sowohl mit dem CL-Befehl WRKSYSSTS als auch über den System i Navigator analysieren. Vorteilhaft im System i Navigator ist allerdings – wie so oft – dass in vielen Fenstern der direkte Zugriff auf weitere Informationen realisiert wurde. Denken Sie nur an die unterschiedlichen Systemwerte und deren Einstellungen!

2.9.7
Seite 10

Plattenbelegung analysieren

Falls Sie feststellen, dass Ihre Platten sich dem Grenzwert nähern, haben Sie zwei Möglichkeiten:

entweder Sie kaufen neue zusätzliche Platten oder räumen auf.

Bevor Sie anfangen nicht mehr benötigte Objekte zu löschen, sollten Sie sich einen entsprechenden Überblick über die Objekte verschaffen, die auf den Platten gespeichert wurden. Hierzu könnten Sie beispielsweise das Menü DISKTASKS aufrufen:

```
DISKTASKS              Aufgaben zum Plattenspeicherplatz
                                                    System:   EPSSRV
Zur Auswahl unten eine Nummer eingeben und die Eingabetaste drücken:

     1. Informationen zum Plattenspeicherplatz erfassen
     2. Informationen zum Plattenspeicherplatz drucken

    10. Mit Bibliotheken arbeiten
    11. Mit Ordnern arbeiten
    12. Mit Objekten eines Eigners arbeiten
```

Das Menü „Disktasks“

Das Menü „Aufgaben zum Plattenspeicherplatz“ (DISKTASKS) ermöglicht in einem ersten Schritt die Analyse der Belegung des Speicherplatzes auf dem System. Der Auswahlpunkt „Informationen zum Plattenspeicherplatz erfassen“ analysiert die Belegung Ihrer Platten. Ein erster oder nur gelegentlicher Aufruf des Menüs läuft daher unter Umständen entsprechend lange. Schließlich müssen alle Platteneinheiten detailliert untersucht werden. In einem weiteren Schritt (Menüpunkt 2) können Sie dann zwischen verschiedenen Berichtstypen wählen:

- Systeminformationsbericht
- Bibliotheks- und Objektbericht
- Ordner- und Dokumentbericht

Ein Systeminformationsbericht enthält beispielsweise eine Übersicht über den gesamten auf dem System verwendeten Plattenspeicherplatz, Hauptspeichergröße, Maschinentyp-Modell und Systemseriennummer. Anschließend wird für die folgenden Objekte der belegte Speicherplatz ermittelt:

- Benutzerbibliotheken
- Ordner und Dokumente
- Die Bibliothek QSYS und andere IBM Bibliotheken
- Lizenzierter interner Code incl. Speicherauszugsbereich, Fehlerprotokolle, Trace-Tabellen, VLIC-Protokolle und Steuerblöcke

2.9.7.1
Seite 2

- Temporärer und freier Speicherplatz
- Interne Objekte, hierzu gehören u. a. zusätzliche Konfigurationsinformationen, Datenbank, Objekte der Leistungsüberwachung, Produktinstallationen, Wiederherstellungsinformationen u. v. m.
- Druckausgabe und Spooling (QSPL)
- Bibliothek für Objekt ersetzen (QRPLOBJ)
- Vom Befehl RCLSTG betroffener Speicher
- Speicherbibliothek zurückfordern (QRCL)

Zusätzlich stehen die Menüpunkte zehn bis zwölf zur Verfügung. Mit Hilfe dieser Auswahlen können Sie alle auf dem System vorhandenen Bibliotheken auflisten und auch mit dem Inhalt der Bibliotheken arbeiten. Der Menüpunkt entspricht somit dem i5/OS Befehl WRKLIB (Work with library). Der Menüpunkt elf entspricht dem Befehl WRKFLR (Work with folder) und listet alle Ordner des Systems auf.

Der letzte Menüpunkt dient dazu mit den Objekten eines Eigners zu arbeiten. Es wird eine Auflistung erzeugt aller Objekte erzeugt, welche diesen Benutzer als Eigner haben. Wenn ein Benutzer als Eigner einer Vielzahl von Objekten auf dem System geführt wird (z. B. QSECOFR oder QDFTOWN), dann kann diese Auswahl unter Umständen sehr lange dauern.

Aufgrund der so ermittelten Daten können Sie anschließend geeignete Maßnahmen einleiten, um Plattenspeicher zurückzufordern.

Plattenplatz zurückgewinnen

- Gelöschte Datensätze in einer Datei belegen so lange zusätzlichen Plattenplatz, bis Sie diesen nicht mit dem Befehl „Physische Teildatei reorganisieren“ (RGZPFM) freigeben. Diesen Befehl sollten Sie regelmäßig verwenden, wenn innerhalb der Dateien Löschoperationen ausgeführt werden und das Dateiobjekt mit dem Attribut „Gelöschte Sätze wieder verwenden“(REUSEDLT) *NO erstellt wurden.
- Löschen Sie auch hin und wieder den Inhalt des Systemprotokolls QHST. Dieses Systemprotokoll wird in einer speziellen Datei QHST in der Bibliothek QSYS gespeichert. Alte Einträge in diesem Protokoll werden zwar zusammengefasst, leider aber nicht von dem System entfernt. Deshalb sollten Sie hin und wieder diese Einträge manuell löschen.
- Löschen von Problemprotokollen. Sobald das System einen Hardwarefehler erkennt, wird dieser in einem Protokoll weggeschrieben. Ähnlich wie das Systemlog (QHST), so müssen Sie auch solche Einträge manuell löschen. Das Löschen dieser Einträge können Sie mit dem Befehl „Probleme bearbeiten“ (WRKPRB) ausführen. Schauen Sie sich die einzelnen Einträge an und entscheiden Sie dann, welche Einträge Sie löschen können. Reorganisieren Sie anschließend alle Dateien in der Bibliothek QSYS, die mit QASX beginnen.
- Löschen Sie alte Spooldateien, Jobprotokolle, Benutzerprofile und Einheitenbeschreibungen, die Sie nicht mehr benötigen.
- Legen Sie alle PTFs permanent auf dem System an.
- Benutzer- und Systemnachrichten werden nicht zwingend automatisch gelöscht.
- Leeren Sie den Inhalt der Bibliotheken QRCL und QRPLOBJ. Diese beiden Bibliotheken haben den Charakter eines Papierkorbes, dessen Inhalt nicht mehr benötigt wird. In der Bibliothek QRPLOBJ befinden sich beispielsweise alte Programmversionen, die dort als Kopie vom Kompiler gespeichert werden, wenn Sie ein HLL-Programm mit der Option REPLACE *YES erneut umwandeln. In der Bibliothek QRCL befinden sich Objekte, die durch den CL-Befehl RCLSTG erstellt wurden. Dazu später mehr.

Auswahl	Objekt	Art	Attribut	Größe	Text
=	Q5EDB1ED28	*PGM	RPGLE	159744	EPSWB603/EPKND00
_	Q5EDB20C4F	*PGM	RPGLE	159744	EPSWB603/EPKND01
_	Q5EDB21E8D	*PGM	RPGLE	221184	EPSWB603/EPKND04

Inhalt der Bibliothek QRPLOBJ

- Löschen Sie nicht mehr benötigte Bibliotheken von Ihrem System. Dabei sollten Sie natürlich wirklich darauf achten, dass Sie nur solche Bibliotheken löschen, deren Inhalt wirklich nicht mehr benötigt wird. Außerdem dürfen Sie keine IBM eigenen Bibliotheken löschen – also: Finger weg von den Bibliotheken, die mit „Q“ beginnen – diese dürfen unter keinen Umständen von dem System entfernt werden!
- Komprimieren Sie solche Objekte, welche eine längere Zeit nicht mehr benutzt worden sind. Dabei geben Sie zunächst die Zeit in Tagen vor, welche die Objekte nicht mehr in Verwendung gewesen sein sollen. Nehmen wir einmal an, Sie wollen alle Objekte auf dem System komprimieren, die mehr als 30 Tage nicht verwendet worden sind:

```
                    Objekt verdichten (CPROBJ)

Auswahl eingeben und Eingabetaste drücken.

Objekt . . . . . > *ALL        Name, generisch*, *ALL
  Bibliothek . . . . . >  EPS005CR  Name, *LIBL, *CURLIB, *ALL...
Objektart . . . . . > *ALL     *ALL, *FILE, *MENU...
     + für weitere Werte
Tage seit letztem Zugriff . . .  30        1-366, *NONE
```

Objekte komprimieren

- Ich verdichte in meinem Beispiel alle Objekte in der Bibliothek „EPS005CR“, die seit 30 Tagen nicht mehr genutzt wurden. Mit dem Befehl „Objekt verdichten“ (CPROBJ) können Programme, Menüs, Bildschirmdateien, Druckerdateien, Module und Serviceprogramme verdichtet werden. Verdichtete Objekte nehmen weniger Speicherplatz in Anspruch. Wird ein verdichtetes Objekt anschließend benutzt oder aufgerufen, wird dem Benutzer zwar automatisch eine nicht verdichtete Version des Objektes zur Verfügung gestellt; nur verzögert sich dadurch natürlich auch der Aufruf des entsprechenden Objektes. Außerdem werden in diesen Fällen nicht verdichtete temporäre Objekte erstellt, die dann natürlich auch Speicher belegen, der erst freigegeben wird, wenn Sie ein IPL durchführen oder beispielsweise die temporären Objekte mittels des CL-Befehls „Temporären Speicher zurückfordern“ (RCLTMPSTG) wieder löschen.
- Mit dem Befehl „Spooldateispeicher zurückfordern“(RCLSPLSTG) wird nicht benutzter Speicher für Spool-Dateien zurückgefordert, der für eine angegebene Anzahl von Tagen nicht benutzt wurde. Spool-Dateien werden mit Datenbankteildateien auf dem System gespeichert. Wird eine Spool-Datei gelöscht, wird der Inhalt der Teildatei gelöscht, aber nicht die Teildatei selbst. Aus diesem Grund kann die Teildatei erneut für die als nächstes erstellte Spool-Datei benutzt werden. So kann der Durchsatz erhöht werden, wenn neue Spoolausgaben erzeugt werden. Aufgrund der

Einstellung des Systemwertes QRCLSPLSTG werden allerdings ohnehin leere Datenbankteildateien automatisch gelöscht, wenn sie mehr als acht Tage nicht genutzt wurden.

Und natürlich können Sie auch den CL-Befehl „Speicher zurückfordern“ (RCLSTG) nutzen. Der Befehl RCLSTG ist vielfältig und soll daher im Folgenden etwas detaillierter besprochen werden

2.9.7.2

Seite 4

Der Befehl „Speicher zurückfordern" (RCLSTG)

Mit dem Befehl „Speicher zurückfordern" (RCLSTG) veranlassen Sie eine Vielzahl verschiedener Prüfungen und in Abhängigkeit der Prüfergebnisse unterschiedliche Aktionen:

- Unvollständig aktualisierte Objekte und Benutzerprofile, die falsch aufgezeichnete Informationen zu Eigentumsrechten für Objekte enthalten, werden wiederhergestellt. Alle Objekte, die nicht mehr zu verwenden sind, werden gelöscht.
- Außerdem werden alle Objekte, die durch beschädigte Berechtigungslisten geschützt sind, wiederhergestellt, und anschließend der Berechtigungsliste QRCLAUTL zugeordnet.
- Wenn der Zugriffspfad für eine geschlüsselte logische Datei beschädigt sein sollte, dann wird dieser Zugriffspfad mit RCLSTG neu aufgebaut.
- Alle Objekte, die keiner Bibliothek zugeordnet werden können, werden in die Bibliothek QRCL gestellt. Diese Bibliothek wird automatisch erstellt, wenn sie noch nicht auf Ihrem System existieren sollte.
- Sie können den Befehl RCLSTG auch verwenden, wenn während eines IPL nicht genügend Speicher zur Verfügung steht, um den Server vollständig zu starten. In diesem Fall erhalten Sie zunächst eine entsprechende Meldung, die auf die Speicherknappheit hinweist und geben daraufhin den Befehl RCLSTG ein, um Speicher entsprechend zurückzufordern.

Der Befehl RCLSTG führt eine Reihe weiterer Funktionen aus und kann daher viel Zeit und Raum in Anspruch nehmen. Dies ist besonders dann der Fall, wenn Sie diesen Befehl nur selten ausführen. Auch die Anzahl und die Art der Objekte im System sowie natürlich das Ausmaß der Beschädigungen sind für die Laufzeit bedeutend. Grundsätzlich muss auch genügend Plattenspeicher zur Verfügung stehen, da andernfalls der Befehl fehlschlägt.

```
                    Speicher zurückfordern (RCLSTG)

Auswahl eingeben und Eingabetaste drücken.

Erforderliche Zeit schätzen . .   *NO          *NO, *YES
Auswählen . . . . . .             *ALL         *ALL, *DBXREF, *DIR
Übergehen . . . . . .             *NONE        *NONE, *DBXREF, *DIR
ASP-Einheit oder -Gruppe . . . .  *SYSBAS      Name, *SYSBAS
```

Der Befehl RCLSTG

Um die voraussichtliche Laufzeit zu planen, können Sie den Parameter „Erforderliche Zeit schätzen" (Estimate) auf den Wert *YES abändern. Bei Angabe von *YES für den Parameter ESTIMATE wird eine Schätzung ausgegeben, die für die Ausführung des Befehls RCLSTG erforderlich ist. Die geschätzte Ausführungszeit wird an das Jobprotokoll gesendet, ohne dass eine der Wiederherstellungsfunktionen ausgeführt wird. Dauert die geplante Zeit zu lange oder Sie befürchten, dass nicht genügend Arbeitsspeicher zur Verfügung steht, können Sie über die Parameter die Befehlsausführung weiter einschränken, indem Sie beispielsweise nur die Wiederherstellung der Systemtabelle QSYS/QADBXREF anfordern oder übergehen. Ebenfalls ist es möglich, nur die Wiederherstellung der Verzeichniseinträge des IFS ROOT-Verzeichnisses anzufordern bzw. diese Funktion zu überspringen.

Sollten Sie trotz der vorhandenen Auswahlmöglichkeiten den Befehl nicht ausführen können, weil nicht genügend Speicher zur Verfügung steht, Sie das System nicht in den eingeschränkten Zustand bringen können oder der Befehl für die Ausführung zu lange benötigen würde, können Sie die Funktionen auch mit einzelnen Befehlen und nacheinander ausführen:

RCLLNK Objektverbindungen zurückfordern
Mit dem Befehl RCLLNK kann eine Verzeichnisstruktur zurückgefordert werden, so dass das angegebene Verzeichnis, sein Inhalt und der Inhalt aller seiner Unterverzeichnisse überprüft wird. Wird SUBTREE(*ALL) angegeben, versucht der Befehl, so viele Objekte wie möglich in der Unterverzeichnisstruktur zurückzufordern. Eine Diagnosenachricht wird für jedes Objekt gesendet, das nicht zurückgefordert werden kann. Zusätzlich wird eine Informationsnachricht gesendet, wenn ein bestimmtes Problem korrigiert wurde, oder eine Diagnosenachricht wird gesendet, wenn ein bestimmtes Problem nicht korrigiert werden kann. Wurden alle Objekte zurückgefordert und alle Probleme korrigiert, wird eine Beendigungsnachricht ausgegeben.

RCLDBXREF DB-Querverweis zurückfordern
Mit dem Befehl RCLDBXREF (DB-Querverweis zurückfordern) können die Querverweiskatalogdaten der Datenbank für eine bestimmte Bibliothek wiederhergestellt werden. Dieser Befehl unterstützt eine Untergruppe der Funktionen, die vom Befehl RCLSTG (Speicher zurückfordern) bereitgestellt werden, mit dem die Querverweiskatalogdaten der Datenbank für das System durch Angabe von SELECT(*DBXREF) wiederhergestellt werden können. Der Unterschied besteht darin, dass es zur Ausführung dieses Befehls nicht erforderlich ist, dass sich das System im Status des eingeschränkten Betriebs befindet. Außerdem kann mit dem Befehl eine bestimmte Bibliothek aus der Gruppe aller Bibliotheken eines Systems zur Wiederherstellung ausgewählt werden.

RCLOBJOWN Objekte eines Eigners zurückfordern
Der Befehl RCLOBJOWN (Objekte eines Eigners zurückfordern) prüft die Objekte, deren Eigner ein bestimmtes Benutzerprofil ist, und stellt sicher, dass alle Objekte, die sich in einer Bibliothek befinden müssen, sich tatsächlich in einer Bibliothek befinden. Befinden sich Objekte dieses Eigners nicht in einer Bibliothek, werden sie in eine Bibliothek eingefügt. Muss das Objekt sich in einer bestimmten Systembibliothek befinden, wird es in diese Bibliothek eingefügt. Kann das Objekt sich in einer beliebigen Benutzerbibliothek befinden und wird es aus dem Zusatzspeicherpool des Systems (ASP 1) oder einem Basisbenutzer-ASP (ASPs 2-32) zurückgefordert, wird das Objekt in die Bibliothek QRCL eingefügt. Wird das Objekt aus einem primären oder sekundären ASP zurückgefordert, wird das Objekt in die Bibliothek QRCLnnnnn eingefügt, wobei nnnnn die Nummer ist, die dem primären ASP vom Betriebssystem zugeordnet wurde.

RCLLIB Bibliothek zurückfordern
Der Befehl RCLLIB (Bibliothek zurückfordern) stellt die internen Objekte einer Bibliothek, die die Objektbeschreibung für alle Objekte der Bibliothek enthalten, sowie das Bibliotheksobjekt wieder her.

RCLSPLSTG Spool-Speicher zurückfordern
Mit dem Befehl RCLSPLSTG (Spool-Speicher zurückfordern) wird nicht benutzter Speicher für Spool-Dateien zurückgefordert, der für eine angegebene Anzahl von Tagen nicht benutzt wurde.

2.9.7.3

Seite 4

Das Menü „CLEANUP“

In Objekten wie Systemprotokollen, Journalempfängern und Nachrichtenwarteschlangen werden Informationen gesammelt und im System gespeichert. Alte Informationen sollten aber in festgelegten Abständen gelöscht werden, damit das System nicht zu voll wird. Der Systemoperator sollte daher die Bereinigungsfunktion mindestens einmal täglich verwenden, um

- die Systemjournale zu löschen.
- die Einträge in Chronik- und Problemprotokollen zu löschen.
- die alten Nachrichten aus den Benutzerprofilnachrichtenwarteschlangen zu löschen.
- die Nachrichten aus der Nachrichtenwarteschlange „QSYSOPR“ und aus allen Arbeitsstationsnachrichtenwarteschlangen zu löschen.
- die Systemausgabe zu reduzieren und daher beispielsweise Jobprotokolle löschen.

Wenn die Bereinigung gestartet ist, übergibt sie eine Gruppe Stapeljobs zum Bereinigen von Objekten, die sich länger als eine bestimmte Anzahl Tage im System befinden. Eigentlich handelt es sich hierbei um einen spezifischen Systemjob, der periodisch nach ihren Vorgaben Bereinigungen auf dem System ausführt. Dieser Job verhindert somit, dass sich auf dem System unnötiger Ballast ansammelt. Objekte, die während des Bereinigungsvorgangs von anderen Benutzern verwendet werden, sind von der Bereinigung ausgeschlossen. Um dies zu vermeiden, sollte der Zeitpunkt für die Bereinigung so festgelegt werden, dass keine Benutzer im System angemeldet sind. Schauen wir uns nun das Menü etwas genauer an:

```
CLEANUP                        Bereinigungsaufgaben
                                                        System:   EPSSRV
Zur Auswahl unten eine Nummer eingeben und die Eingabetaste drücken:

     1. Bereinigungsauswahl ändern
     2. Bereinigung zur geplanten Zeit starten
     3. Bereinigung sofort starten
     4. Bereinigung beenden
```

Das Menü „Cleanup“

2.9.7.4
Seite 2

Schauen wir uns die Menüpunkte kurz an:

1. Bereinigungsauswahl ändern (CHGCLNUP).
Sie können die Vorgabewerte für die Bereinigungsaufgaben anpassen, indem Sie entweder den ersten Menüpunkt „Bereinigungsauswahl ändern" oder den Befehl CHGCLNUP (Change Cleanup) verwenden. Auf den ersten Blick gibt es zwischen Menü und Befehl keine Unterschiede. Allerdings verfügt der Befehl CHGCLNUP über zusätzliche Parameter, die Sie im Menüpunkt vergebens suchen werden. Ich nutze daher lieber den Befehl CHGCLNUP, um den Zeitpunkt der Bereinigung, aber auch die Bereinigungsaufgaben selbst festzulegen:

```
                    Bereinigung ändern (CHGCLNUP)

Auswahl eingeben und Eingabetaste drücken.

Bereinigung gestatten . . . . .   *YES          *SAME, *YES, *NO
Startzeit tägliche Bereinigung    '22:00:00'    Zeit, *SAME, *SCDPWROFF...
Aufbewahren (Anzahl Tage):
  Benutzernachrichten . . . . .   7             1-366, *KEEP, *SAME
  System- und DS-Nachrichten . .  4             1-366, *KEEP, *SAME
  Wichtige Systemnachrichten . .  *KEEP         1-366, *KEEP, *SAME
  Jobprotokolle/Systemausgabe .   7             1-366, *KEEP, *SAME
  Systemjournale/Protokolle . .   30            1-366, *KEEP, *SAME
```

Der Befehl CHGCLNUP

Der Parameter „Bereinigung gestatten" bestimmt grundsätzlich, ob die Bereinigung auf dem System aufgerufen werden kann. Wenn ALWCLNUP(*YES) angegeben ist, kann die Bereinigung mit dem Befehl STRCLNUP (Bereinigung starten) gestartet werden. Außerdem kann die Bereinigungsoperation selbst geändert werden, indem das Programm QEZUSRCLNP mit dem Befehl RTVCLSRC (CL-Quelle abrufen) abgerufen wird, der Quelltext geändert wird und das CL-Programm anschließend neu kompiliert wird (siehe Kapitel 2.9.7.4.1). Bevor wir aber darüber nachdenken, das Cleanup-Programm zu modifizieren, sollten wir erst einmal klären, welche Informationen durch den Cleanup-Job bereinigt werden. Grundsätzlich können

- Benutzernachrichten
- System- und Datenstationsnachrichten
- Wichtige Systemnachrichten (CRITSYSMSG)
- Jobprotokolle und Systemausgabe (SYSPRT)
- Systemjournale und Protokolle (SYSLOG)

bereinigt werden.

„Benutzernachrichten"
gibt an, ob Nachrichten in den Nachrichtenwarteschlangen der Benutzerprofile bereinigt werden sollen. Mit dieser Funktion werden Nachrichten in den Benutzernachrichtenwarteschlangen gelöscht, die länger als die angegebene Anzahl Tage im System gespeichert sind.

System- und DS-Nachrichten
Entscheiden Sie sich auch die System- und Datenstationsnachrichten automatisch bereinigen zu lassen, werden die Nachrichten in der Nachrichtenwarteschlange „QSYSOPR" gelöscht, die länger als die angegebene Anzahl Tage im System gespeichert sind.

Wichtige Systemnachrichten (CRITSYSMSG)
könnten ebenfalls gelöscht werden. Bereinigt wird in diesen Fällen die Nachrichtenwarteschlange „QSYSMSG", sofern Sie die Nachrichtenwarteschlange für derartige Nachrichten nutzen.

Jobprotokolle/Systemausgabe
Mit dem Parameter Jobprotokolle/Systemausgabe steuern Sie, wie lange Jobprotokolle und andere Systemausgaben aufbewahrt werden sollen. Um zu verhindern, dass die Einstellungen für das Bereinigen der Jobprotokolle und Systemnachrichten mit den Ausgaben des Benutzers vermischt werden, werden die Jobprotokolle automatisch in die Ausgabewarteschlange QUSRSYS/QEZJOBLOG seitens des Systems umgestellt. Die Ausgabewarteschlange für die Druckerdateien QPSRVDMP, QPPGMDMP und BASDMP werden auf QUSRSYS/QEZDEBUG umgestellt. Alle Einträge in QEZJOBLOG und QEZDEBUG, die älter als die angegebene Anzahl Tage sind, werden gelöscht. Bei diesem Vorgang werden auch die anstehenden Jobprotokolle für Jobs gelöscht, deren Abschluss schon vor mehr als der angegebenen Anzahl von Tagen durchgeführt wurde, ohne dass entsprechende Benutzernachrichten gelöscht werden.

Systemjournale/Protokolle
Hinter dem Parameter „Systemjournale/Protokolle" verbirgt sich die Löschoperation für Systemjournale, Protokolldateien, Problemprotokolldateien und der ALERT-Datenbank, aber auch die Journalempfänger, die für eines der nachfolgenden Systemjournale benutzt werden und älter als die angegebene Anzahl Tage sind, werden gelöscht.

2.9.7.4
Seite 4

Systemjournal	Bedeutung
QAOSDIAJRN	Journal für DIA-Dateien
QDSNX	Journal für DSNX-Protokolle
QSNADS	Journal für SNADS-Dateien
QSXJRN	Journal für Problemdatenbanken
QPFRADJ	Journal für Leistungsanpassungsdaten
QACGJRN	Journal für Jobabrechnungsdaten (Nur wenn das Journal über die Oberfläche für Anwendungen erstellt wurde)
QX400	OSI Message Services
QCQJMJRN	Journal für Managed System Services
Weitere Informationen enthält die Bedienerführung	

Zusätzlich zu den Journalempfängern werden auch Protokolldateien mit dem Namen QSYS/QHST* gelöscht, wenn sie länger als die in diesem Parameter angegebene Anzahl Tage im System gespeichert sind.

Etwas schwieriger ist die Löschoperation für Journale bzw. deren Journalempfänger. Das System kann die Journalempfänger nur löschen, wenn der Journalempfänger zuvor vom Journal abgehangen wurde. Journalempfänger werden automatisch abgehangen, wenn Sie ein IPL durchführen, aber was, wenn Sie nur gelegentlich ggf. nur ein oder zwei Mal im Jahr ein IPL durchführen können? Öffnen Sie einfach mal die erweiterten Parameter des Befehls CHGCLNUP. Wie eingangs erwähnt, verfügt der Befehl CHGCLNUP über weitere Parameter, die Sie im entsprechenden Menü nicht finden.

```
Jobwarteschlange . . . . . .   QCTL        Name, *SAME
  Bibliothek . . . . . . . .     *LIBL      Name, *LIBL, *CURLIB
Journalempfängergröße  . . . .   0005000     1-1919999, *KEEP, *SAME

                Zusätzliche Parameter

  Bürokalenderfunktionen . . . .   30        1-366, *KEEP, *SAME
Ausführungspriorität . . . . . .   55        1-99, *KEEP, *SAME
```

Erweiterte Parameter des Befehls „CHGCLNUP“

Journalempfängergröße
Nicht nur unsere Datenbankdateien werden in Journalen aufgezeichnet, auch die Systemprotokolldateien unterliegen der Aufzeichnung. Die automatische Bereinigungsfunktion löscht zwar die Journalempfänger der Systemprotokolldateien, aber da die Empfänger nur gelöscht werden, wenn Sie zuvor abgehangen werden, kann hier ein Konflikt entstehen. In diesem Kontext hilft der Parameter Journalempfängergröße. Die Bereinigungsfunktion kann nämlich nicht nur die Empfänger löschen, sondern hängt die Journalempfänger auch automatisch ab, wenn der Grenzwert von 5000 KB erreicht wurde. Alle Journalempfänger, die vor mehr als die für den Parameter Systemjournale/Protokolle (SYSLOG) angegebene Anzahl an Tagen abgehangen wurden, werden anschließend auch automatisch gelöscht.

Problemprotokolldateien und -einträge
Problemprotokolldateien werden gelöscht, wenn sie älter als die bei diesem Parameter angegebene Anzahl Tage sind. Tatsächlich wird der Befehl „Problem löschen“ (DLTPRB) aufgerufen, um die Protokolleinträge zu löschen. Dabei wird die für diesen Parameter angegebene Anzahl Tage für den Parameter Tage (DAYS) des Befehls DLTPRB benutzt. Achten Sie darauf, dass der Systemwert QPRBHLDITV (Problemprotokollaufbewahrungsdauer) entsprechend gesetzt ist. Nehmen wir einmal an, dass die Problemprotokolle nur zehn Tage aufbewahrt werden sollen: Sie ändern daher den Parameter Problemprotokolldateien auf den Wert zehn:

```
CHGCLNUP ALWCLNUP(*YES) SYSLOG(10)
```

Schnell wäre festzustellen, dass Ihre Problemprotokolldateien aber nicht nach zehn, sondern erst nach 30 Tagen gelöscht werden. Der Grund hierfür ist der Systemwert QPRBHLDITV:

```
                              Systemwert anzeigen
_

 Systemwert . . . . . :   QPRBHLDITV
 Beschreibung . . . . . :   Intervall zum Anhalten des Problemprotokolls


 Problemanhalte-
   intervall in Tagen . :   30         0-999
```

Der Systemwert QPRBHLDITV

Per Default wird hier festgelegt, dass Problemprotokolle 30 Tage aufbewahrt werden sollen. Die Systemwerteinstellung übersteuert die Einstellungen des Befehl CHGCLNUP – gelöscht werden die Problemprotokolle nach 30 und nicht wie gewünscht bereits nach zehn Tagen. Sie müssen also, wenn Sie die Aufbewahrungszeiten der Systemwerte verkürzen wollen, daran denken, auch den Systemwert abzuändern. Andernfalls bleiben Ihre Einstellungen im Bereinigungsprogramm für die Problemprotokolle ohne Wirkung! Zusätzlich zu den Einträgen, die aus dem Problemprotokoll gelöscht werden, werden einige Problemprotokolldateien in der Bibliothek QUSRSYS reorganisiert.

Jobwarteschlange
Der Bereinigungsjob wird als Batchjob gestartet und häufig als Schedulerjob im Subsystem QCTL ausgeführt. Falls Sie den Bereinigungsjob in einem anderen Subsystem ausführen wollen, müssen Sie nur den Jobwarteschlangennamen entsprechend abändern.

Ausführungspriorität
Sobald der Job aktiv ist, erhält er die Ausführungspriorität 55. Die Ausführungspriorität ist ein Wert im Bereich von 1 (höchste Priorität) bis 99 (niedrigste Priorität), der die Bedeutung des Jobs bezeichnet, wenn er mit anderen Jobs um Maschinenressourcen konkurriert. Zum Vergleich: Interaktive Jobs haben normalerweise die Ausführungspriorität 20, ein klassischer Batchjob in der Regel die Priorität 50. Allerdings sollten Sie bedenken, dass dieser Wert für die relative Bedeutung des Jobs und nicht für dessen absolute Bedeutung steht. Ein Job mit der Ausführungspriorität 20 ist somit nicht etwa mehr als doppelt so wichtig wie einer mit der Ausführungspriorität 50.

Kommen wir zurück zum Bereinigungsmenü, das wir uns einfach zur Erinnerung erneut ansehen:

```
CLEANUP                        Bereinigungsaufgaben
                                                          System:   EPSSRV
Zur Auswahl unten eine Nummer eingeben und die Eingabetaste drücken:

     1. Bereinigungsauswahl ändern
     2. Bereinigung zur geplanten Zeit starten
     3. Bereinigung sofort starten
     4. Bereinigung beenden
```

Das Menü „Cleanup"

Wenn Sie die Bereinigungsaufgaben und die Zeiträume festgelegt haben, können Sie die Bereinigung starten. Auch jetzt stehen wieder zwei Möglichkeiten zur Auswahl zur Verfügung: Entweder Sie nutzen den Menüpunkt 2 oder den Befehl STRCLNUP OPTION(*SCHED). Beides ermöglicht die automatische Ausführung der Bereinigungsjobs. Dabei verwendet das System die Angaben, welche sie in der Auswahl 1 vorgegeben haben.

Alternativ können Sie die Bereinigung auch sofort starten. Dabei werden die Planungsangaben außer Acht gelassen. Falls Sie zuvor den Job bereits für eine geplante Ausführungszeit aufgerufen haben, müssen Sie die Bereinigungsaufgaben zunächst beenden (Auswahl 4 oder ENDCLNUP) und anschließend mit der Auswahl 3 sofort starten. Auch hier steht ihnen wieder der Alternativbefehl STRCLNUP OPTION(*IMMED) zur Verfügung.

2.9.7.4

Seite 8

Manuelle Cleanup-Funktionen integrieren

Die Cleanup-Programme übernehmen viele Bereinigungsaufgaben für verschiedene Systemobjekte. So werden Benutzer- und Systemnachrichten bereinigt. Auch Joblogs in der Ausgabewarteschlange QEZJOBLOG werden gelöscht, aber was ist mit den übrigen Benutzerspoolfiles, die im System viele Tage liegen bleiben und darauf warten, manuell gelöscht zu werden? Wann werden die physischen Dateien reorganisiert? Und wer kümmert sich um die vielen Objekte – Programme und Query/400-Abfragen, die nicht mehr benötigt werden? Aber auch die vielen Benutzerjournale sollten von Zeit zu Zeit gelöscht werden. Sie können diese manuellen Aufgaben einfach in das Cleanup-Programm integrieren, indem Sie das Programm „QEZUSRCLNP“ nutzen.

Eine Besonderheit von CL-Programmen ist es, dass sie einfach dekompiliert werden können, um den Quellcode von Programmen wiederherzustellen, die nur als Objekt vorhanden sind. Sollten Sie nicht wissen, was gemeint ist, finden Sie die entsprechenden Informationen im Kapitel 2.12.3.1. Diese Eigenschaft nutzen wir jetzt, um eigene Cleanup-Funktionen in das Bereinigungsprogramm zu integrieren.

Versuchen wir folgende Aufgabenstellung zu realisieren. Ich möchte zukünftig die Journalempfänger der Bibliothek EPSW09APP abhängen und anschließend löschen. Normalerweise würden Sie natürlich Ihre abgehangenen Journalempfänger sichern, bevor sie gelöscht werden. Doch in unserer Schulungsumgebung sind die Daten eher irrelevant, da es sich ausschließlich um Testdaten und nicht um Produktionsdaten handelt.

Um die Aufgabe zu realisieren rufe ich in einem ersten Schritt den Quellcode des Programms „QEZUSRCLNP“ mit dem Befehl RTVCLSRC (retrieve cl source) ab.

```
/* Message Bereinigung startet wird ausgegeben */
             QSYS/SNDPGMMSG MSGID(CPI1E91) MSGF(QCPFMSG)  -
                            TOMSGQ(*SYSOPR) MSGTYPE(*INFO)
             QSYS/MONMSG MSGID(CPF2400)

/* Integriert werden eigene Bereinigungsbefehle */




/* Ende der manuellen Bereinigungsaufgaben   */

/* Message Bereinigung endet  wird ausgegeben */
            QSYS/SNDPGMMSG MSGID(CPI1E92) MSGF(QCPFMSG)
                           TOMSGQ(*SYSOPR) MSGTYPE(*INFO)
            QSYS/MONMSG MSGID(CPF2400)

            RETURN
```

Aufbau des Programms QEZUSRCLNUP

Ich habe die Programmquelle bereits ein wenig strukturiert, um sie übersichtlicher zu gestalten. Im Programm wird als erstes die Meldung „Bereinigung wurde gestartet." ausgegeben. Die Meldung gehört noch zum Originalprogramm. Anschließend können Sie Ihre eigenen Bereinigungsbefehle integrieren.

Kommen wir auf die geplante Aufgabenstellung zurück. Beim Arbeiten mit Journalen existieren immer zwei Objekte: das Journal und die Journalempfänger. Die Daten werden in den Journalempfängern gespeichert. Die administrativen Informationen – Name der aufgezeichneten Dateien, aktuell zugeordneter Empfänger usw. – verbleiben im Journal. Die Empfänger verfügen in der Regel über dieselbe Namensstruktur wie die Journale. In den SQL Schemata wird beispielsweise stets das Journal QSQJRN erstellt. Die zugehörigen Empfänger verfügen über dieselben Namen, ergänzt um eine vierstellige Ziffer. Diese Ziffern werden der Reihe nach mit 0001 beginnend vergeben.

Opt	Objekt	Art	Bibliothek
_	QSQJRN2038	*JRNRCV	EPSW09APP
_	QSQJRN	*JRN	EPSW09APP

Das Journal und seine Empfänger

Sie sehen, in der Bibliothek EPSW09APP existiert das Journal QSQJRN und der Empfänger QSQJRN2038. 2.038 Empfänger wurden für das Journal bereits erstellt. Will ich dieses Journal jetzt über das Bereinigungsprogramm manuell verwalten, muss ich

1. einen neuen Empfänger erstellen.
2. den alten Empfänger deaktivieren.
3. den neuen Empfänger aktivieren.
4. den alten Empfänger ggf. sichern und anschließend löschen.

Die Arbeitschritte 1 bis 3 werden mit Hilfe des Befehls CHGJRN (change journal) ausgeführt.

```
                          Journal ändern (CHGJRN)

Auswahl eingeben und Eingabetaste drücken.

Journal  . . . . . . . . . . . . > QSQJRN      Name
  Bibliothek . . . . . . . . . . >   EPSW07CR   Name, *LIBL, *CURLIB
Journalempfänger:
  Journalempfänger . . . . . .     *GEN        Name, *SAME, *GEN
    Bibliothek . . . . . . . .                 Name, *LIBL, *CURLIB
```

Der Befehl CHGJRN

Der Journalempfänger wird vom System erstellt und anschließend automatisch an das angegebene Journal angehängt. Der Journalempfänger wird dabei mit denselben Attributen, demselben Eigner, denselben Berechtigungen, derselben Protokollierungsstufe und in derselben Bibliothek erstellt wie der derzeit angehängte Journalempfänger. Der Name des neuen Empfängers wird zudem vom Namen des derzeitigen Empfängers abgeleitet, indem eine 4-stellige Ziffernfolge an einen Teil des Namens des derzeitigen Empfängers angehängt wird oder indem der Wert eins zu der Nummer des Namens des derzeitigen Empfängers hinzuaddiert wird. Der Name des erstellten und angehängten Empfängers wird anschließend in einer Informationsnachricht zurückgegeben.

```
+ > CHGJRN JRN(EPSW07CR/QSQJRN) JRNRCV(*GEN)
    Journalempfänger QSQJRN1014 in Bibliothek EPSW07CR erstellt.
    Journalempfänger QSQJRN1013 und *N abgehängt.
```

Ergebnis des Befehls CHGJRN

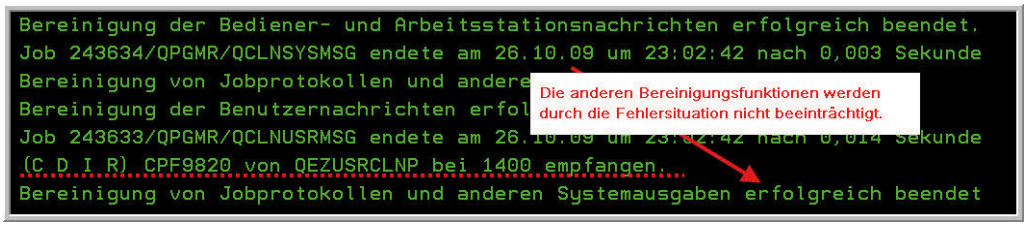
In unserer Schulungsumgebung ist die Sicherung der Journalempfänger nicht relevant; vielmehr will ich dafür sorgen, dass die Journalempfänger nicht auf dem Server liegen bleiben, wenn wir ein IPL durchführen oder die Empfängerschwellen erreicht werden.

Ich werde also die abgehangenen Empfänger in unserer Schulungsbibliothek EPSW09APP löschen, ohne sie zuvor zu sichern. Dies gelingt mir mit den CL-Befehlen „Change journal“ (CHGJRN) und „Delete journal receiver“ (DLTJRNRCV). Beide CL-befehle integriere ich in das Programm „QEZUSRCLNP“, so dass das modifizierte Bereinigungsprogramm anschließend folgendermaßen aussieht:

```
/* ----EPSW09APP-------------------------------------------------*/
        CHGJRN     JRN(EPSW09APP/QSQJRN) JRNRCV(*GEN) +
                     MNGRCV(*SYSTEM) DLTRCV(*NO)
        RCVMSG     MSGTYPE(*INFO) MSG(&MSG) MSGDTA(&DATEN) +
                     MSGID(&MSGID) MSGF(&MSGF) MSGFLIB(&MSGLIB)
        SNDPGMMSG  MSGID(&MSGID) MSGF(&MSGLIB/&MSGF) +
                     MSGDTA(&DATEN) TOMSGQ(*SYSOPR)

        DLTJRNRCV  JRNRCV(EPSW09APP/QSQJRN*) DLTOPT(*IGNINQMSG)
        MONMSG CPF7022
```

Das modifizierte Programm QEZUSRCLNUP

Es soll an dieser Stelle nicht um die verwendeten CL-Befehle zur Verwaltung der Datenbankjournale gehen. Wichtiger erscheint mir in diesem Zusammenhang, dass Sie in das Bereinigungsprogramm beliebige CL-Befehle integrieren können. So lassen sich die unterschiedlichsten Aufgaben realisieren. Sie könnten alle Spooldateien, die nicht mehr benötigt werden, löschen. Physische Dateien könnten reorganisiert werden usw. Beachten müssen Sie in diesem Zusammenhang lediglich die Objektberechtigungen. Das Bereinigungsprogramm QEZUSRCLNUP wird vom Benutzer QPGMR ausgeführt. Wenn der Benutzer QPGMR nicht über die entsprechenden Berechtigungen verfügt, wird das Bereinigungsprogramm entsprechende Fehlernachrichten an das Benutzerprofil QSYSOPR senden.



Fehlersituationen beeinträchtigen die anderen Systembereinigungsfunktionen nicht!

Das im oberen Bild dargestellte Systemprotokoll zeigt deutlich, dass die Bereinigung der Journalreceiver Fehler verursachen, Das sollte nicht passieren – kann aber passieren! Es ist in diesem Fall gar kein Problem, da die übrigen Systembereinigungsfunktionen problemlos ausgeführt werden, obwohl das Bereinigungsprogramm QEZUSRCLNUP eine Fehlernachricht an das Benutzerprofil QSYSOPR sendet.

Globalisierung und IBM i

Das Öffnen der Märkte und die damit verbundene Globalisierung bestimmen heute mehr denn je das Tagesgeschehen. Mit der Einführung des Internets und der damit verbundenen Ausweitung der Kommunikation sind wir zusammen mit einer ausgeweiteten Logistikkette in der Lage, Waren an jeden beliebigen Ort dieser Welt zu versenden.

Doch der Wegfall der Begrenzung des Marktes hat auch Folgen. Wer heute seine Geschäfte ausdehnt, der muss in der Regel auch die Basis für eine einwandfreie Kommunikation schaffen. Das hört sich zunächst ganz einfach an – wenn man aber erst einmal die ersten kleineren Überlegungen anstellt, dann stellt man sehr schnell fest, dass die Ausweitung alles andere als trivial ist.

Kompatibilität, Unterstützung von Zeichensätzen und die Speicherung der Daten in einer für alle Anwendungen verarbeitbaren Form sind nur einige der Punkte, die bei einem globalen Einsatz in Bezug auf die Speicherung und Darstellung der Informationen bedacht werden müssen.

Wie einfach waren die Anfangszeiten der IT. Zentrale Verarbeitung und klar definierte Zeichen, die anzuzeigen, zu drucken oder zu speichern waren. Doch je globaler sich im Laufe der Jahre die Märkte entwickelt haben, desto schneller wuchsen auch die Anforderungen an die IT, die vielfältigen Formen der Speicherung von Daten oder besser von deren Bestandteilen in Form der Zeichen zu bewerkstelligen.

Haben Sie sich einmal Gedanken gemacht, wie beispielsweise unser „Ä“ oder „ß“ in Japan dargestellt werden? Haben Sie einmal französische oder tschechische Daten zur Verarbeitung erhalten? Hier beginnen die Auswirkungen der Globalisierung, die Grenzen der Standard-IT auszuhebeln.

Wenn Daten nur in einer Sprache (gemeint ist hier wirklich die Sprache und nicht etwa eine Programmiersprache!) bereitgestellt werden müssen, dann kann man dies mit den klassischen Zeichensätzen durchführen, die eine besondere Ausrichtung auf die Unterstützung der nationalen Zeichen haben. Sollen aber mehrere unterschiedliche Sprachen mit abweichenden Zeichensätzen verarbeitet werden, dann müssen unterschiedliche Zeichensätze oder Codepages eingesetzt werden.

Zeichensätze sind die Grundlage für die Darstellung der Daten, die mit einem beliebigen IT-System gespeichert werden. So unterschiedlich die einzelnen Zeichensätze auch sein mögen – alle basieren auf der Speicherung der Informationen in Form von numerischen Werten. So wird zum Beispiel der Buchstabe „A“ als Wert 0100 0001 dargestellt.

Ein Zeichensatz (im Englischen sprechen wir von „Coded Character Set" – kurz: CCS) ist folglich eine Darstellung von Ziffern, Buchstaben, Sonderzeichen und Steuerzeichen in einer tabellarischen Form.

Leider haben sich im Laufe der Jahre verschiedene Standards entwickelt, die zum Teil erhebliche Unterschiede aufweisen. Kompatibilität und auch Möglichkeiten der Zeichendarstellungen der Zeichensätze bestimmen neben den Anwendungen, welcher Zeichensatz verwendet werden kann oder verwendet werden muss. Limitierungen, die bei der Entwicklung der älteren Zeichensätze noch nicht bedacht wurden, sind heute häufig Grund für Anpassungen. Die Unterstützung mehrerer Sprachen beispielsweise – heute eine Normalanforderung in den meisten Unternehmen – ist nicht oder nur teilweise möglich.

Nicht unproblematisch sind dann die Fehler bei den möglichen Transformationen, die bei der Umsetzung von einem Zeichensatz in einen anderen Zeichensatz vorgenommen werden müssen. „Wundersame" Zeichen sind dann die Folge, was bis zur Verstümmelung des Ursprungstextes führen kann.

Mit dem Betriebssystem IBM i verwenden wir per Standard EBCDIC. Dabei handelt es sich um einen von IBM Anfang der 60er Jahre für Großrechner entwickelten Zeichensatz. Dieser ist folglich in gewisser Weise auf den Einsatz mit IBM-Systemen beschränkt und bei weitem nicht alle an das System i angeschlossenen Einheiten sind in der Lage, EBCDIC zu verarbeiten. Folglich müssen Umsetzungen der IBM i-Informationen in eine kompatible Form für Endgeräte oder Fremdanwendungen durchgeführt werden.

Denken wir alleine einmal an die Datenübertragung zwischen IBM i und Fremdsystemen. Wie oft treten dabei Probleme auf, weil von einer Codepage in eine andere gewechselt wird oder Verschlüsselungen gewechselt werden müssen? Probleme mit Umlauten, Steuerzeichen und anderen sind häufig an der Tagesordnung.

Diese Problemstellung hat nicht nur IBM dazu bewogen, in Richtung einer einheitlichen Zeichensatzverwendung und einem globalen Denken eine Ebene zu schaffen, mit der alle denkbaren Zeichen dargestellt und auf einem System problemlos verarbeitet werden können: Unicode.

Mit dem Einsatz von Unicode gehören solche Probleme der Vergangenheit an – denn eine Umsetzung wird nicht mehr benötigt. Die Daten liegen schlicht und einfach in dem korrekten Format vor und können somit plattformunabhängig ohne weitere Konvertierung verarbeitet werden.

Zeichensätze

Die Speicherung und die Darstellbarkeit von Buchstaben, Ziffern, Sonderzeichen und Symbolen werden im Umfeld der IT über Zeichensätze realisiert. Diese Zeichensätze stellen fest definierte Regeln dar, nach denen in Bit-Regeln die Zeichen präsentiert werden.

Schauen wir uns nun zum besseren Verständnis die unterschiedlichen Zeichensätze an, die wir in Verbindung mit dem IBM i antreffen können, bzw. mit denen wir Daten unseres Systems austauschen müssen.

2.10.1 Seite 2

ASCII

Mit diesem auf Einzelbytezeichen basierten Codierschema können je nach verwendetem Bit-Einsatz 128 bzw. bis zu 256 Zeichen dargestellt werden. Mit 7 Bits werden alle im englischen Alphabet vorkommenden Buchstaben, Ziffern, Satz- und Steuerzeichen dargestellt. Mit einer 8-Bit-Umsetzung können zusätzliche Sonderzeichen dargestellt werden.

Ursprünglich wurde ASCII von den USA entwickelt. Demzufolge fehlen auch die Zeichen, die im amerikanischen Alphabet nicht vorkommen. 1963 wurde der erste ASCII-Zeichensatz vorgestellt, der 1967 als Standard veröffentlicht und letztmals 1986 überarbeitet wurde.

Der Zeichencode besteht aus 128 Zeichen, wobei 33 nicht druckbar sind. Die 95 druckbaren Zeichen entsprechen den Zeichen, die im amerikanischen Raum zum Einsatz kommen. Die nachfolgende Abbildung zeigt diesen Zeichensatz:

```
 !"#$%&'()*+,-./0123456789:;<=>?
@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_
`abcdefghijklmnopqrstuvwxyz{|}~
```

ASCII

Auch die ASCII-Zeichen werden in einem hexadezimalen Verfahren gespeichert. Dabei kommt die folgende Tabelle zum Einsatz:

Code	...0	...1	...2	...3	...4	...5	...6	...7	...8	...9	...A	...B	...C	...D	...E	...F
0...	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1...	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2...	SP	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3...	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4...	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5...	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6...	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7...	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL

ASCII Hex

Jedes Zeichen wird in ASCII mit 7 Bit dargestellt. Folglich ergeben sich 2 hoch 7 Kombinationen. Dies sind also 128 mögliche Kombinationen.

Wie Sie in der vorhergehenden Abbildung erkennen können, handelt es sich bei den ersten ASCII-Zeichen um Steuerzeichen, die verschiedenste Verwendungen haben. Auf die Steuerzeichen folgen zunächst die Ziffern und letztlich die Buchstaben sowohl in Groß- als auch im Kleinformat.

2.10.1.1
Seite 2

EBCDIC

Dieser von IBM für Großrechner entwickelte Zeichensatz dominiert die IBM-Systeme. EBCDIC ist die Abkürzung für Extended Binary Coded Decimals Interchange Code. Dieser Zeichensatz basiert auf einem 8-Bit-Code, der im Laufe der Jahre aber auch in 16- und 32-Bit Versionen bereitgestellt wurde. Mit EBCDIC lassen sich bis zu 256 Zeichen darstellen.

Leider ist eine direkte Kompatibilität mit ASCII nicht gegeben. Dies liegt an dem heterogenen Aufbau der beiden Zeichensätze. Während in ASCII zuerst die Ziffern codiert sind, finden wir in EBCDIC zunächst die Definition der Buchstaben.

Eine weitere Unterscheidung findet man in der Codepage. IBM bietet 14 verschiedene Codepages, mit denen jeweils die landes- oder besser sprachenspezifischen Anforderungen an die Darstellungen von Zeichen abgedeckt werden. Auf diese Weise lassen sich mit EBCDIC auch zum Beispiel asiatische Schriften abbilden.

Hier eine Übersicht der Codepages:

- 00037 USA, Kanada
- 00273 Deutschland, Österreich
- 00277 Dänemark, Norwegen
- 00278 Finnland, Schweden
- 00280 Italien
- 00284 Spanien, Südamerika
- 00285 Großbritannien
- 00297 Frankreich
- 00423 Griechenland
- 00500 Belgien, Schweiz, Kanada
- 00838 Thai
- .....

Letztlich sind die Codepages eine Notlösung, mit der die Einschränkungen der Zeichensätze mehr oder weniger umgangen wurden. Codepages beinhalten neben den Basiszeichen des EBCDIC-Zeichensatzes auch Sonderzeichen, die in den jeweiligen Sprachen benötigt werden.

Die Codepages sind an die ISO Norm ISO 8859 angelehnt und können mittels einer 8-Bit-Codierung bis zu 256 Zeichen abbilden.

2.10.1.2
Seite 2

Sicher kennen Sie diese durch die Einstellungen auf Ihrem System i. Hier kann man beispielsweise über den Systemwert QCHRID die Codepage festlegen. Diese hat zum Teil erhebliche Auswirkungen, wie die beiden nachfolgenden Codepages zeigen:

Die EBCDIC-Codepage 273, die meist für den deutschsprachigen Raum eingesetzt wird:

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
4_			â	{	à	á	ã	å	ç	ñ	Ä	.	<	(	+	!	4_ ($4_{hex} = 0100_{bin}$)
5_	&	é	ê	ë	è	í	î	ï	ì	~	Ü	$	*	)	;	^	5_ ($5_{hex} = 0101_{bin}$)
6_	-	/	Â	[	À	Á	Ã	Å	Ç	Ñ	ö		%	_	>	?	6_ ($6_{hex} = 0110_{bin}$)
7_	ø	É	Ê	Ë	È	Í	Î	Ï	Ì	`	:	#	§	,	=	„	7_ ($7_{hex} = 0111_{bin}$)
8_	Ø	a	b	c	d	e	f	g	h	i	«	»	ð	ý	þ	±	8_ ($8_{hex} = 1000_{bin}$)
9_	°	j	k	l	m	n	o	p	q	r	ª	tº	æ	¸	Æ	¤	9_ ($9_{hex} = 1001_{bin}$)
A_	µ	ß	s	t	u	v	w	x	y	z	¡	¿	Ð	Ý	Þ	®	A_ ($A_{hex} = 1010_{bin}$)
B_	¢	£	¥	·	©	@	¶	¼	½	¾	¬		¯	¨	´	×	B_ ($B_{hex} = 1011_{bin}$)
C_	ä	A	B	C	D	E	F	G	H	I		ô	¦	ò	ó		C_ ($C_{hex} = 1100_{bin}$)
D_	ü	J	K	L	M	N	O	P	Q	R	¹	û	}	ù	ú	ÿ	D_ ($D_{hex} = 1101_{bin}$)
E_	Ö	÷	S	T	U	V	W	X	Y	Z	²	Ô	\	Ò	Ó	Õ	E_ ($E_{hex} = 1110_{bin}$)
F_	0	1	2	3	4	5	6	7	8	9	³	Û	]	Ù	Ú		F_ ($F_{hex} = 1111_{bin}$)
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	

EBCDIC 273

Für die Codepage 37 (USA, Kanada etc) werden die folgenden hexadezimalen Werte genutzt:

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
4_			â	ä	à	á	ã	å	ç	ñ	¢	.	<	(	+		4_ ($4_{hex} = 0100_{bin}$)
5_	&	é	ê	ë	è	í	î	ï	ì	ß	!	$	*	)	;	¬	5_ ($5_{hex} = 0101_{bin}$)
6_	-	/	Â	Ä	À	Á	Ã	Å	Ç	Ñ	¦		%	_	>	?	6_ ($6_{hex} = 0110_{bin}$)
7_	ø	É	Ê	Ë	È	Í	Î	Ï	Ì	`	:	#	@	,	=	„	7_ ($7_{hex} = 0111_{bin}$)
8_	Ø	a	b	c	d	e	f	g	h	i	«	»	ð	ý	þ	±	8_ ($8_{hex} = 1000_{bin}$)
9_	°	j	k	l	m	n	o	p	q	r	ª	º	æ	¸	Æ	¤	9_ ($9_{hex} = 1001_{bin}$)
A_	µ	~	s	t	u	v	w	x	y	z	¡	¿	Ð	Ý	Þ	®	A_ ($A_{hex} = 1010_{bin}$)
B_	^	£	¥	·	©	§	¶	¼	½	¾	[	]	¯	¨	´	×	B_ ($B_{hex} = 1011_{bin}$)
C_	{	A	B	C	D	E	F	G	H	I		ô	ö	ò	ó		C_ ($C_{hex} = 1100_{bin}$)
D_	}	J	K	L	M	N	O	P	Q	R	¹	û	ü	ù	ú	ÿ	D_ ($D_{hex} = 1101_{bin}$)
E_	\	÷	S	T	U	V	W	X	Y	Z	²	Ô	Ö	Ò	Ó	Õ	E_ ($E_{hex} = 1110_{bin}$)
F_	0	1	2	3	4	5	6	7	8	9	³	Û	Ü	Ù	Ú		F_ ($F_{hex} = 1111_{bin}$)
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	

EBCDIC 37

Schauen Sie sich einmal in den beiden Codepages die Koordinaten 0/C an:

Einmal lautet das Ergebnis „ä“ und einmal „{“ .

Wenn das bei Umstellungen einmal keinen Ärger gibt! (Natürlich gibt es den – aber genau deshalb gibt es auch Umsetzungstabellen.) Wenn diese nun auch noch überall greifen würden, dann wäre das schön. Wir könnten die Abhandlungen zum Thema Globalisierung dann jetzt beenden. Da ich aber einige Seiten mehr zu schreiben habe, lässt das die Vermutung zu, dass eine Umsetzung so ohne weiteres wohl nicht immer möglich ist.

Dass IBM noch maßgebliche Veränderungen an EBCDIC-Zeichensätzen vornimmt, dürfte wenig wahrscheinlich sein – denn zu eingeschränkt ist die Verwendung auf den IBM-Systemen. Die Unterstützung von flexiblen Zeichensätzen erlaubt es den Betreibern dieser Maschinen sowieso, unkomplizierten Datenaustausch mit Fremdsystemen durchführen zu können.

2.10.1.2

Seite 4

UTF-8

Hier handelt es sich um eine Erweiterung des ASCII-Zeichensatzes mit 16-Bit-Unicode Nutzung. Damit können weitaus mehr Zeichen dargestellt werden, als es der „alte“ ASCII-Zeichensatz gestattet. RPG-Anwendungen benötigen bei der Verarbeitung von UTF-8-basierten Daten eine Umsetzung.

Dies ist nur eine kleine Übersicht zum Thema Zeichensätze. Darüber hinaus existieren noch weitere Zeichensätze, die wir hier nicht weiter behandeln wollen.

Beispielszenario
Nehmen wir einmal ein Alltagsbeispiel für die Zeichensätze und die damit verbundenen Probleme:

Währungskennzeichen sind heute im internationalen Handel wichtig und in fast jedem Unternehmen auf Ausdrucken oder Anzeigen zu finden.

Wie wird nun ein solches Währungskennzeichen in den zuvor genannten Zeichensätzen definiert? Schauen wir uns das am Beispiel des Dollarzeichens ($) an:

Bei der Verwendung der EBCDIC-Codepage 273 (die wir z. B. für deutsche Umgebungen nutzen) wird das Dollarzeichen ($) mit dem hexadezimalen Wert 5B dargestellt.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
4_			â	{	à	á	ã	å	ç	ñ	Ä	.	<	(	+	!	4_ ($4_{hex} = 0100_{bin}$)
5_	&	é	ê	ë	è	í	î	ï	ì	~	Ü	$	*	)	;	^	5_ ($5_{hex} = 0101_{bin}$)
6_	-	/	Â	[	À	Á	Ã	Å	Ç	Ñ	ö		%	_	>	?	6_ ($6_{hex} = 0110_{bin}$)
7_	ø	É	Ê	Ë	È	Í	Î	Ï	Ì	`	:	#	§	,	=	„	7_ ($7_{hex} = 0111_{bin}$)
8_	Ø	a	b	c	d	e	f	g	h	i	«	»	ð	ý	þ	±	8_ ($8_{hex} = 1000_{bin}$)
9_	°	j	k	l	m	n	o	p	q	r	ª	º	æ	¸	Æ	¤	9_ ($9_{hex} = 1001_{bin}$)
A_	µ	ß	s	t	u	v	w	x	y	z	¡	¿	Ð	Ý	Þ	®	A_ ($A_{hex} = 1010_{bin}$)
B_	¢	£	¥	·	©	@	¶	¼	½	¾	¬		¯	¨	´	×	B_ ($B_{hex} = 1011_{bin}$)
C_	ä	A	B	C	D	E	F	G	H	I		ô	¦	ò	ó		C_ ($C_{hex} = 1100_{bin}$)
D_	ü	J	K	L	M	N	O	P	Q	R	¹	û	}	ù	ú	ÿ	D_ ($D_{hex} = 1101_{bin}$)
E_	Ö	÷	S	T	U	V	W	X	Y	Z	²	Ô	\	Ò	Ó	Õ	E_ ($E_{hex} = 1110_{bin}$)
F_	0	1	2	3	4	5	6	7	8	9	³	Û	]	Ù	Ú		F_ ($F_{hex} = 1111_{bin}$)
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	

EBCDIC 273

Wird hingegen die Codepage 285 verwendet, dann finden wir als hexadezimalen Wert für das Dollarzeichen ($) 4A. 5B wird hier für das Währungszeichen des GBP verwendet (£).

2.10.1.3
Seite 2

Wie Sie sehen, ist der Einsatz der Codepage mit Vorsicht zu genießen und man muss sehr wohl überlegen, ob man diese Form der Zeichensatzunterstützung weiter betreibt, wenn die Komplexität der Unterstützung verschiedener Sprachen zu groß wird.

Die Verwendung der Zeichensätze über Plattformen oder Anwendungen hinweg ist nicht trivial. Im Idealfall verwenden unterschiedliche Systeme denselben Zeichensatz. Anderenfalls sind unter Umständen Umsetzungen durchzuführen. Diese sind zwar auch kein Hexenwerk, bedürfen aber zusätzlicher Rechenzeit und stellen so gerade bei Massendatenverarbeitungen ein gewisses Hindernis dar.

SBCS
Die meisten Dekodierungsprozesse der alten Generation basieren auf Single-Byte pro Zeichen. Deshalb werden diese auch als „Single Byte Character Sets" (kurz SBCS) bezeichnet. Wie der Name schon sagt, wird jedes Zeichen mit einem Byte dargestellt. Diesen Zeichensätzen ist gemein, dass sie bis zu maximal 256 Zeichen abbilden können. Diese Beschränkung ist auch einer der Gründe, warum wir heute derart viele Zeichensätze finden – denn 256 Zeichen genügen bei weitem nicht, um die alleine für den europäischen Raum notwendigen Zeichen abbilden zu können.

Neben Ziffern und Buchstaben finden wir in SBCS-Zeichensätzen auch Steuerzeichen, die in Anwendungen und mit den Systemen genutzt werden können.

Der für den deutschsprachigen Raum am weitesten verbreitete SBCS-Standard ist ISO 8859-1, den wir unter anderem auch mit dem IBM i einsetzen können.

DBCS
Die Limitierung der maximalen Anzahl der Zeichen war der Grund für die Entwicklung der Double Byte Character Sets (kurz: DBCS). Nehmen wir alleine einmal den asiatischen Raum: Dort existieren mehrere tausend Zeichen und diese müssen natürlich auch mit IT-Systemen verarbeitet werden können. Die Zeichenumsetzung ist auch hier byte-basiert.

Häufig sind DBCS-Zeichensätze in der Lage, die Zeichen der SBCS-Zeichensätze interpretieren zu können – aber das muss leider nicht immer so sein.

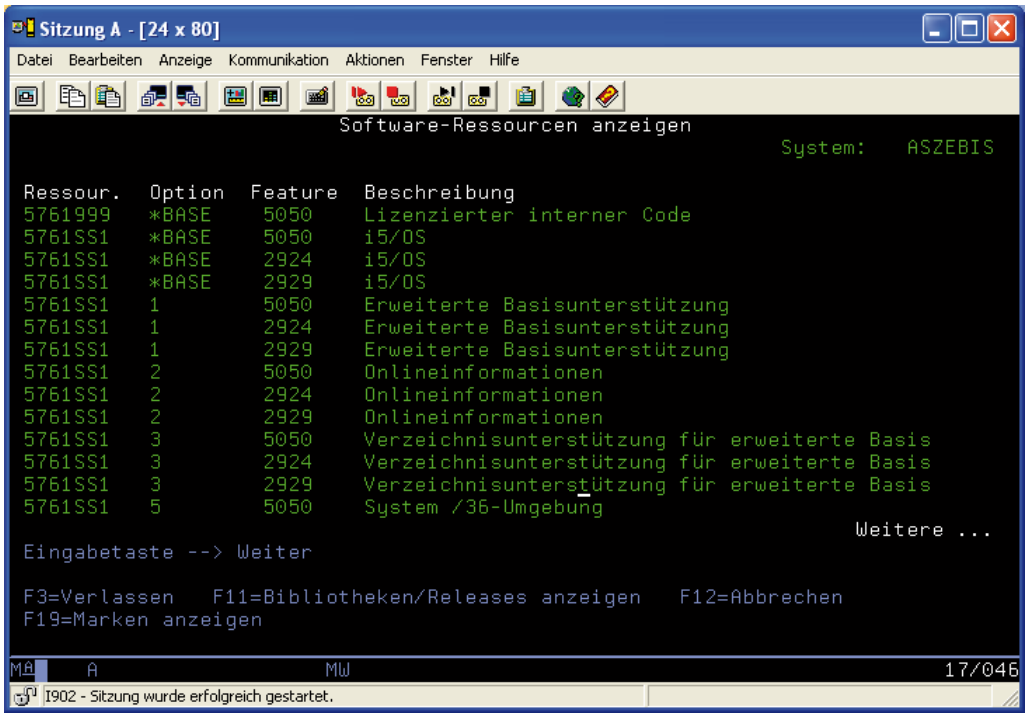
Sprachunterstützung IBM i

Die Unterstützung von verschiedenen Sprachen ist auf dem System i kein Problem. Die Möglichkeit, neben einer Primärsprache beliebige Sekundärsprachen einzusetzen, bildet die Grundlage für den sprachübergreifenden Einsatz eines einzelnen Systems bzw. einer einzelnen Partition.

Systemsprachen
Jedes IBM i muss über eine Primärsprache verfügen. Dabei handelt es sich um die Installation des Betriebssystems und der Lizenzprogramme in einer bestimmten Sprache. Für den deutschen Sprachraum ist dies die Sprachversion 2929. Die Primärsprache bestimmt unter anderem, in welcher Sprache Menüs, Hilfetexte, Befehle und andere Bereiche angezeigt werden.

Werfen wir kurz einen Blick auf die Lizenzprogramme eines IBM i in deutscher Sprache: Dort finden wir das Betriebssystem und auch die meisten Lizenzprogramme mit dem Sprachzusatz 2929.

Die nachfolgende Abbildung zeigt die Darstellung der auf einem System i installierten Lizenzprogramme nebst den installierten Sprachen. Mit dem Befehl DSPSFWRSC können wir diese Informationen anzeigen.



DSPSFWRSC

Wie Sie anhand der Abbildung sehen können, ist zum Beispiel das Basisbetriebssystem 5761-SS1 i5/OS in den Sprachen 2929 und 2924 installiert. Hier ist anzumerken, dass auf diesem System eine Sekundärsprache (nämlich Englisch – also 2924) installiert ist.

Die Primärsprache auf dem System ist hier Deutsch – also 2929.

Anmerkung:

Einige Lizenzprogramme werden von IBM nur in der englischen Version (2924) ausgeliefert und können auch auf einem nicht-englischen System installiert und betrieben werden. Deshalb kann es unter Umständen vorkommen, dass Sie auf einem System einen Mix an installierten Lizenzprogrammen mit unterschiedlichen Sprachen finden.

Nehmen wir einmal an, wir haben ein System, auf dem nicht nur die deutschen Niederlassungen arbeiten, sondern auch die englischen und französischen Niederlassungen. Für letztere gilt, dass diese in der Regel mit den deutschen Menüs und Hilfen nicht wirklich viel anfangen können. Deshalb bietet sich in einem solchen Fall die Installation von zusätzlichen „sekundären" Sprachen auf dem System an. Mit dieser Installation besteht dann die Möglichkeit, von Betriebssystem oder Lizenzprogrammen genutzte Bildschirmausgaben in der jeweiligen Sprache zu nutzen.

ACHTUNG!

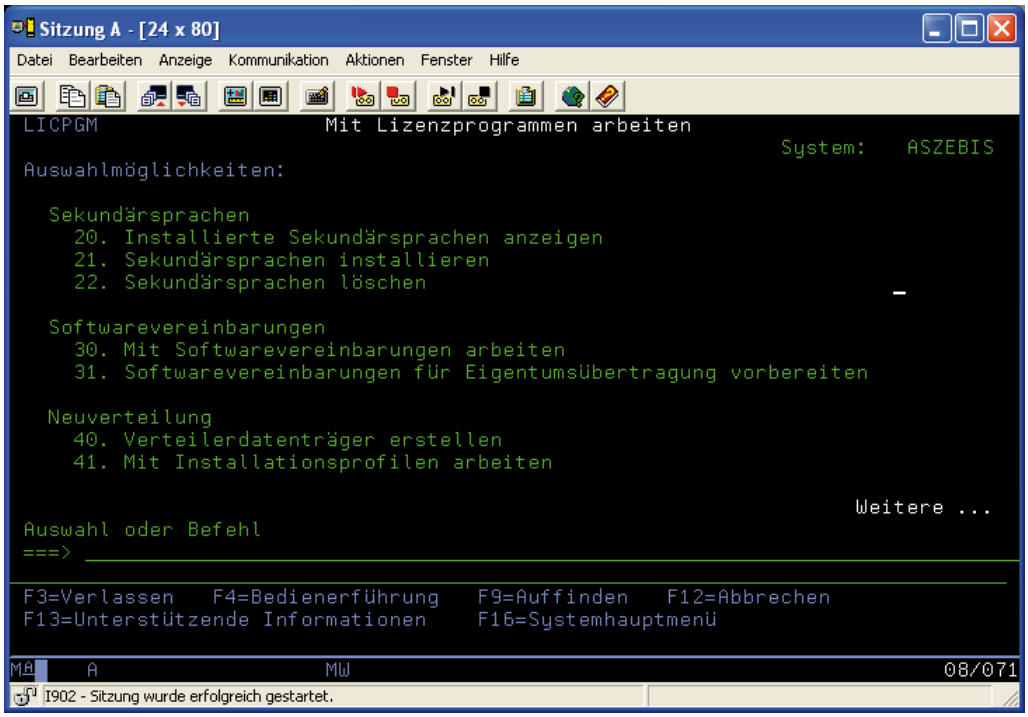
Die Installation einer Sekundärsprache bedeutet nicht, dass etwa eine „On-the-fly"-Übersetzung aller Bereiche durchgeführt wird. Im Gegenteil: Es findet gar keine Übersetzung statt, sondern bestimmte Bereiche werden schlicht in der gewünschten Sprache angezeigt. Davon betroffen sind KEINE eigenen Anwendungen, Bildschirmausgaben, Bedienerhilfen oder Benutzerdaten! Sollen diese in unterschiedlichen Sprachen genutzt werden, dann müssen sie manuell übersetzt bzw. angepasst werden!

Wenn Sie auf einem IBM i eine neue Sprache installieren wollen, dann benötigen Sie dazu die entsprechende Installationssoftware in Form einer CD/DVD oder als Download von IBM. Es gibt nicht, wie bei der Installation der Primärsprache, eine Reihe unterschiedlicher Installationsdatenträger, sondern die sprachenspezifischen Ausprägungen sind auf einem einzelnen Datenträger (z. B. CD) enthalten und können von dort installiert werden.

Die Installation einer Sekundärsprache kann bei einem in Betrieb befindlichen System durchgeführt werden. Eine Beendigung der Subsysteme ist hier nicht zwingend erforderlich.

Trotz allem – eine Systemsicherung kann nie schaden.

Die Installation erfolgt über das Menü GO LICPGM. Hier finden wir unter anderem auch verschiedene Optionen, um mit den Sekundärsprachen zu arbeiten. Wir wollen uns an dieser Stelle mit der Option 21 (Sekundärsprache installieren) beschäftigen.

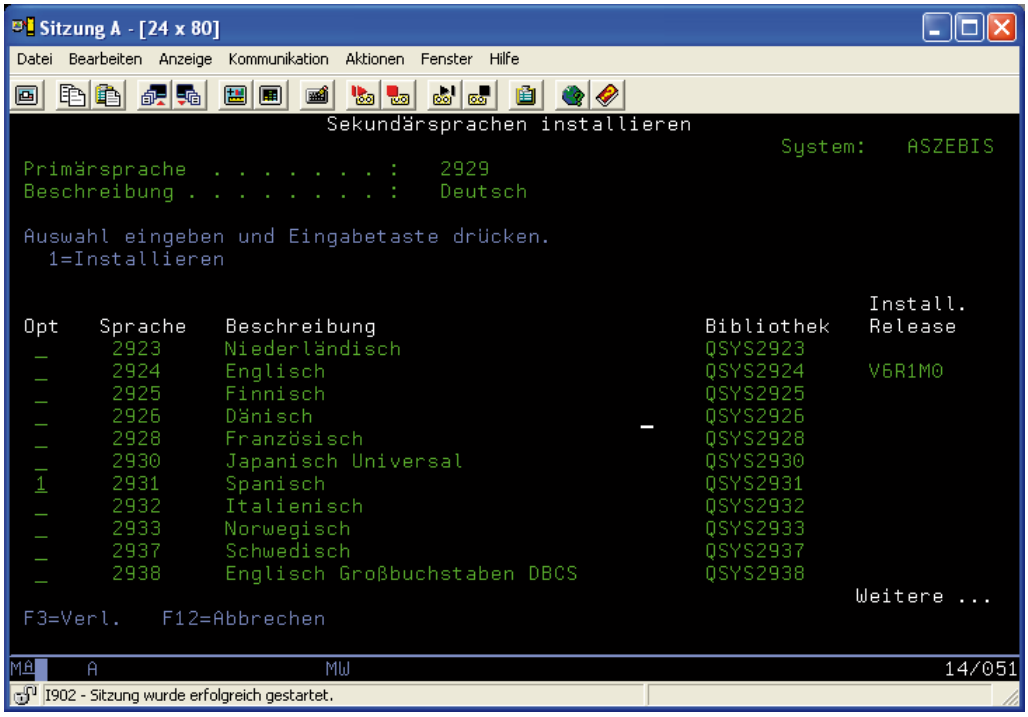


Sekundärsprachen in GO LICPGM

Nach dem Einlegen des Datenträgers mit der zu installierenden Sekundärsprache wählen wir die Option 21 in dem Menü LICPGM.

Damit gelangen wir in eine Übersicht der für das System verfügbaren Sprachen. Ein Beispiel dafür finden Sie in der nachfolgenden Abbildung.

2.10.2
Seite 4



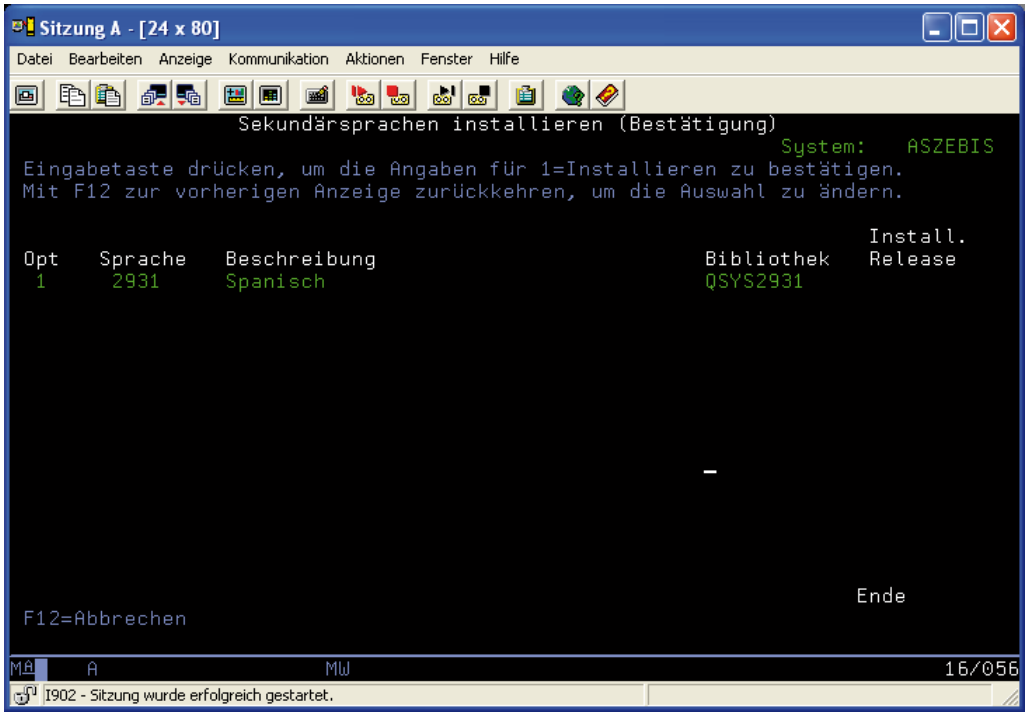
Auswahl Sekundärsprache

In der Übersicht der Sprachen sehen wir unter anderem auch unsere installierte Primärsprache. Diese ist in unserem Beispiel Deutsch (2929).

In der Abbildung sehen Sie zudem, dass bereits eine Sekundärsprache auf dem System installiert ist: Englisch – mit dem Sprachencode 2924.

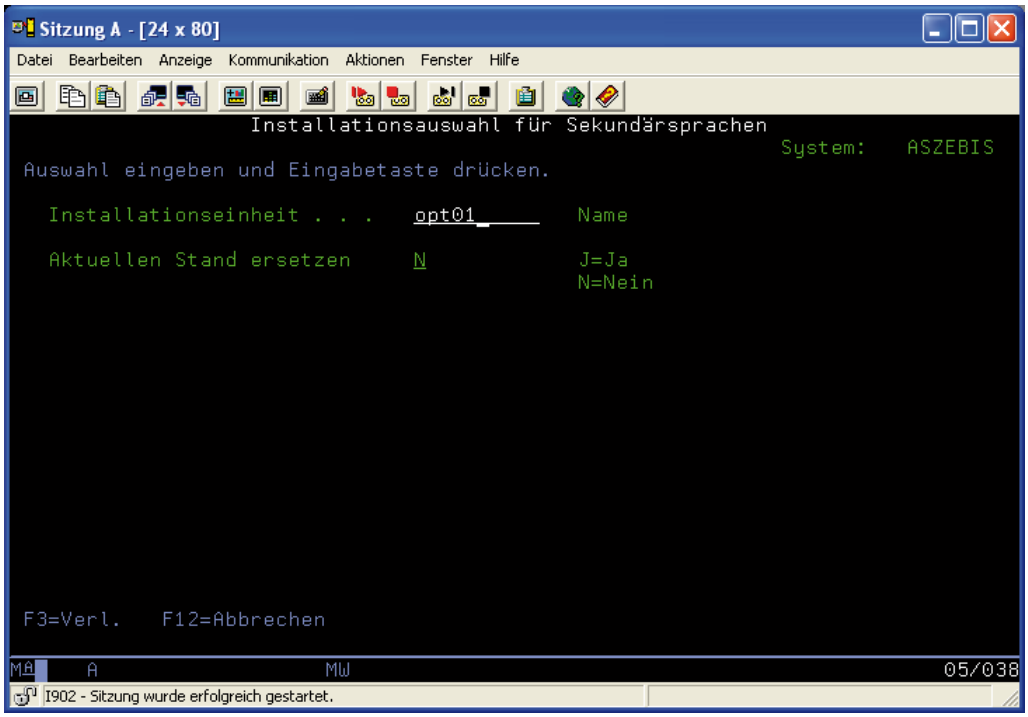
Wie bereits erwähnt, können wir eine beliebige Anzahl von Sekundärsprachen auf dem System installieren. In dem Beispiel soll zusätzlich Spanisch (2931) installiert werden.

In der Folgeanzeige wird dargestellt, dass die Sprache in einer speziellen Sprachbibliothek installiert wird. Diese beginnt mit QSYS, gefolgt von der Sprachen-ID (z. B. 2931).



Sprachenbibliothek

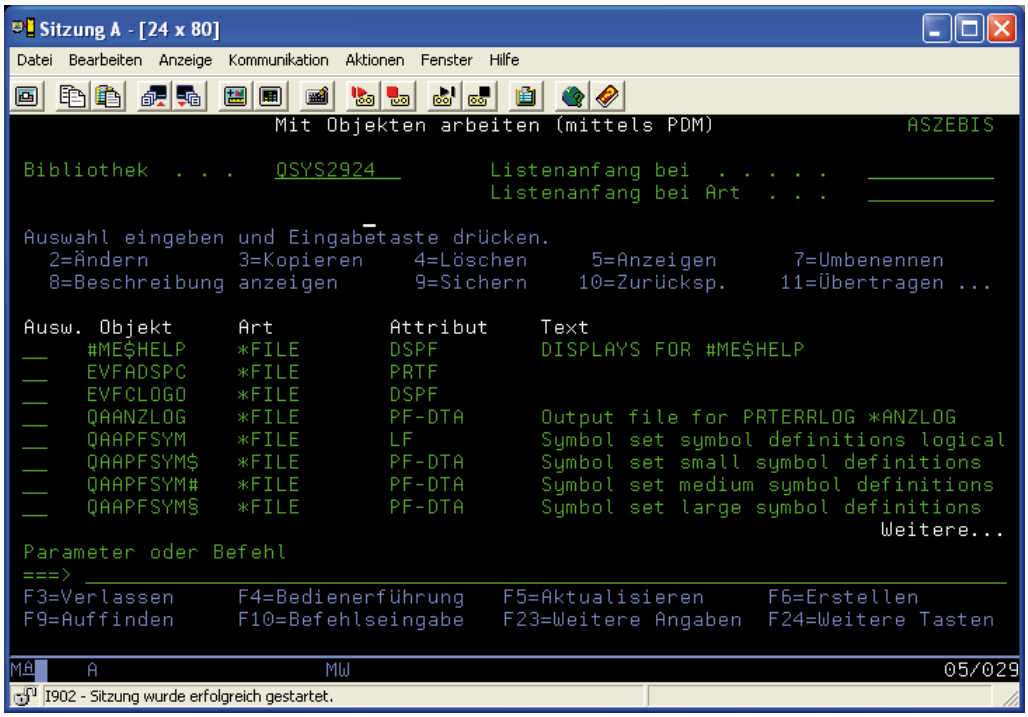
Die letzte Anzeige dient nun zur Eingabe der Installationseinheit und der Entscheidung, ob die Sprache aktualisiert werden soll, wenn diese bereits installiert ist.



Installationsdetails

2.10.2
Seite 6

Nachdem die Sekundärsprache installiert wurde, finden wir diese in der Anzeige von DSPSFWRSC. Außerdem wurde eine Sprachbibliothek QSYS29xx auf dem System angelegt. Darin enthalten sind die sprachabhängigen Komponenten.



Sprachbibliothek QSYS29xx

Anmerkung!

Beachten Sie im Falle eines Releasewechsels, dass neben dem Betriebssystem und den Lizenzprogrammen auch die neue Version der Sprachen installiert wird. Die Sekundärsprachen stehen immer für ein Release zur Verfügung. Deshalb muss ein Releasewechsel auf einem System mit Sekundärsprachen auf jeden Fall auch die neue Version der Sekundärsprachen beinhalten!

Zeichendarstellung und IBM i

Neben der installierten Primärsprache und den optionalen Sekundärsprachen ist die Darstellung der Informationen und Daten in dem für die verschiedenen Sprachen notwendigen korrekten Format notwendig. Für die Darstellung der Zeichen (hier sind in der Regel Alphawerte in Feldern inklusive Sonderzeichen gemeint) stehen auf dem IBM i verschiedene Zeichensätze zur Verfügung, die wahlweise eingesetzt werden können. Kompatibilität spielt hier eine wichtige Rolle, denn nicht selten müssen Daten in verschiedenen Zeichensätzen bereitgestellt werden. Schauen wir uns einmal als Beispiel zwei unterschiedliche EBCDIC-Zeichensätze für die Speicherung der Daten auf dem IBM i an:

Die EBCDIC-Codepage 273, die meist für den deutschsprachigen Raum eingesetzt wird:

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
4_			â	{	à	á	ã	å	ç	ñ	Ä	.	<	(	+	!	4_ ($4_{hex} = 0100_{bin}$)
5_	&	é	ê	ë	è	í	î	ï	ì	~	Ü	$	*	)	;	^	5_ ($5_{hex} = 0101_{bin}$)
6_	-	/	Â	[	À	Á	Ã	Å	Ç	Ñ	ö		%	_	>	?	6_ ($6_{hex} = 0110_{bin}$)
7_	ø	É	Ê	Ë	È	Í	Î	Ï	Ì	`	:	#	§	,	=	„	7_ ($7_{hex} = 0111_{bin}$)
8_	Ø	a	b	c	d	e	f	g	h	i	«	»	ð	ý	þ	±	8_ ($8_{hex} = 1000_{bin}$)
9_	°	j	k	l	m	n	o	p	q	r	ª	º	æ	¸	Æ	¤	9_ ($9_{hex} = 1001_{bin}$)
A_	µ	ß	s	t	u	v	w	x	y	z	¡	¿	Ð	Ý	Þ	®	A_ ($A_{hex} = 1010_{bin}$)
B_	¢	£	¥	·	©	@	¶	¼	½	¾	¬		¯	¨	´	×	B_ ($B_{hex} = 1011_{bin}$)
C_	ä	A	B	C	D	E	F	G	H	I		ô	¦	ò	ó		C_ ($C_{hex} = 1100_{bin}$)
D_	ü	J	K	L	M	N	O	P	Q	R	¹	û	}	ù	ú	ÿ	D_ ($D_{hex} = 1101_{bin}$)
E_	Ö	÷	S	T	U	V	W	X	Y	Z	²	Ô	\	Ò	Ó	Õ	E_ ($E_{hex} = 1110_{bin}$)
F_	0	1	2	3	4	5	6	7	8	9	³	Û	]	Ù	Ú		F_ ($F_{hex} = 1111_{bin}$)
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	

Für die Codepage 37 (USA, Kanada etc.) werden die folgenden hexadezimalen Werte genutzt:

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
4_			â	ä	à	á	ã	å	ç	ñ	¢	.	<	(	+		4_ ($4_{hex} = 0100_{bin}$)
5_	&	é	ê	ë	è	í	î	ï	ì	ß	!	$	*	)	;	¬	5_ ($5_{hex} = 0101_{bin}$)
6_	-	/	Â	Ä	À	Á	Ã	Å	Ç	Ñ	¦		%	_	>	?	6_ ($6_{hex} = 0110_{bin}$)
7_	ø	É	Ê	Ë	È	Í	Î	Ï	Ì	`	:	#	@	,	=	„	7_ ($7_{hex} = 0111_{bin}$)
8_	Ø	a	b	c	d	e	f	g	h	i	«	»	ð	ý	þ	±	8_ ($8_{hex} = 1000_{bin}$)
9_	°	j	k	l	m	n	o	p	q	r	ª	º	æ	¸	Æ	¤	9_ ($9_{hex} = 1001_{bin}$)
A_	µ	~	s	t	u	v	w	x	y	z	¡	¿	Ð	Ý	Þ	®	A_ ($A_{hex} = 1010_{bin}$)
B_	^	£	¥	·	©	§	¶	¼	½	¾	[	]	¯	¨	´	×	B_ ($B_{hex} = 1011_{bin}$)
C_	{	A	B	C	D	E	F	G	H	I		ô	ö	ò	ó		C_ ($C_{hex} = 1100_{bin}$)
D_	}	J	K	L	M	N	O	P	Q	R	¹	û	ü	ù	ú	ÿ	D_ ($D_{hex} = 1101_{bin}$)
E_	\	÷	S	T	U	V	W	X	Y	Z	²	Ô	Ö	Ò	Ó	Õ	E_ ($E_{hex} = 1110_{bin}$)
F_	0	1	2	3	4	5	6	7	8	9	³	Û	Ü	Ù	Ú		F_ ($F_{hex} = 1111_{bin}$)
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	

2.10.2.1
Seite 2

Zeichendarstellungen müssen unter Umständen auch zwischen unterschiedlichen Zeichensätzen kompatibel sein. Dazu bedient man sich eines Standards, der als „Character Data Representation Architecture – kurz: CDRA" bezeichnet wird. CDRA arbeitet basierend auf im System integrierten Komponenten, die dafür Sorge tragen, dass die Umsetzungen korrekt durchgeführt werden können. Auf diese Weise ist es beispielsweise möglich, dass wir auf einem System Jobs laufen haben, welche ein und dieselben Daten mit einer CCSID verarbeiten können, während andere Jobs abweichende CCSIDs nutzen und dabei die Darstellung der Zeichen dennoch korrekt durchgeführt wird.

ACHTUNG!

Bei dem Einsatz der CCSID 65535 findet keine automatische Konvertierung zwischen dieser CCSID und anderen CCSIDs statt! Letztlich erscheint 65535 als gute Lösung für die Darstellung eines weiten Zeichenspektrums – entpuppt sich bei näherem Hinsehen aber leider allzu oft als Problem, wenn es um das Thema Globalisierung und internationale Nutzung geht.

Die Zeichendarstellung ist mit dem IBM i über bestimmte Codierregeln realisiert. Zusätzliche Berechnungen stellen zudem sicher, dass die Zeichen stets in der korrekten Form dargestellt werden. Berechnungen und Umsetzungen können unter Umständen auch bei der Entwicklung von Anwendungen für den Mehrsprachenraum zum Einsatz kommen. Dort müssen dann zusätzlich zu den vom Betriebssystem gelieferten Funktionen Zusatzarbeiten ausgeführt werden, damit die Daten auch wirklich korrekt dargestellt oder ausgegeben werden.

Warum ist es eigentlich wichtig, dass wir die korrekten und auch „passenden" CCSIDs verwenden?

Nehmen wir einmal ein einfaches und beinahe alltägliches Beispiel:

In einer Anwendung werden Rechnungen verarbeitet. Diese können in unterschiedlichen Währungen erstellt werden, beispielsweise auch in US-Dollar. Natürlich muss auf den Rechnungen das Währungszeichen abgedruckt und in den Anzeigeprogrammen entsprechend dargestellt werden.

Wenn wir auf unserem System eine Datei mit den Rechnungsdaten haben, dann muss das Dollarzeichen darin so gespeichert werden, dass es beispielsweise von den Niederlassungen in den USA genauso korrekt verarbeitet werden kann wie von einer Niederlassung in Deutschland oder Dänemark.

Die Besonderheit liegt in den verschiedenen Zeichensätzen und den Hexwerten, die für die Zeichendarstellung zum Einsatz kommen. Bleiben wir bei unserem Beispiel – dem Dollarzeichen und schauen uns dessen Darstellung in den verschiedenen Zeichensätzen einmal an:

Land	CCSID	Code Page	Codedetail	Zeichen
USA	037	65535	X'5B'	$
England	285	65535	X'5B'	£
Dänemark	277	65535	X'5B'	Å

Wie man anhand des Beispiels erkennen kann, würde das Auslesen der gespeicherten Daten beim Einsatz unterschiedlicher CCSIDs zu abweichenden Ergebnissen führen. Während ein Benutzer, dessen Job mit der CCSID 037 ausgeführt wird, das Dollarzeichen korrekt sieht, würde dieses bei einem Job mit der CCSID 285 als Pfundzeichen (£) erscheinen und folglich ein vollkommen falsches Ergebnis suggerieren!

Die in dem System verankerten Umsetzungstabellen für die verschiedenen Zeichensätze stellen sicher, dass die Datenintegrität gewahrt wird.

Bezogen auf unser Beispiel mit der Anzeige des Dollarzeichens würden die Informationen für die Anwender der englischen CCSID die Codedetails X'4A' verwenden – den Hexwerten, die in der CCSID 285 für das $-Zeichen stehen.

Wie Sie anhand dieses Beispiels sehen können, ist der Einsatz von CCSIDs in den verschiedenen Bereichen des IBM i immens wichtig und kann bei falscher oder fehlerhafter Einstellung unter Umständen fehlerhafte Ergebnisse liefern, da die dargestellten Informationen nicht den wirklich gespeicherten entsprechen!

Die Standard-CCSID wird über den Systemwert QCCSID definiert und steht dann, wenn nichts anderes auf Detailebene definiert wird, auf dem gesamten System und allen Jobs zur Verfügung.

Grundsätzlich müssen wir uns in den folgenden Fällen über den Einsatz von abweichenden CCSIDs Gedanken machen:

Unterstützung von verschiedenen Sprachen, für die unterschiedliche Zeichensätze benötigt werden.

- Das bedeutet dann aber unter Umständen auch Anpassungen bei den Eingabegeräten, z. B. den Tastaturen – denn auch diese müssen auf die Landesspezifika abgestimmt sein! Nehmen wir ein einfaches Beispiel: Geben Sie ohne größeren Aufwand mit einer deutschen Tastatur das Währungszeichen für das britische Pfund ein. Sie werden sehen – so einfach ist das gar nicht… Drucker sind in diesem Zusammenhang ein weiteres Thema – denn nicht alle Drucker sind in der Lage, die Vielzahl der

möglichen Druckzeichen auch wirklich korrekt zu Papier zu bringen. Dies gilt insbesondere für ältere Modelle, die hier und da noch im Einsatz sind.

- Die Unterstützung kann auf ein und demselben IBM i benötigt werden oder auch dann erforderlich sein, wenn Daten zwischen verschiedenen Maschinen mit unterschiedlichen Primärsprachen ausgetauscht werden sollen. Als Beispiel: Es soll eine Datenübertragung aus einem Zweigwerk in Tschechien auf die Hauptmaschine in Deutschland stattfinden. In einem solchen Fall muss der Datenaustausch unter Umständen mittels CCSID-Umsetzungen komplettiert werden.

Globalisierung – Definitionsbereiche für IBM i

Multilinguale Einstellungen und notwendige Parametrisierung für globale Anwendungen können mit dem IBM i auf den verschiedenen Ebenen durchgeführt werden.

Die Primärsprachen und Sekundärsprachen haben wir bereits zuvor behandelt.

Auf Systemebene dienen verschiedene Systemwerte der Festlegung des Verhaltens von Jobs und damit verbundenen Aktionen – auch in Bezug auf nationale Einstellungen und Zeichendarstellungen.

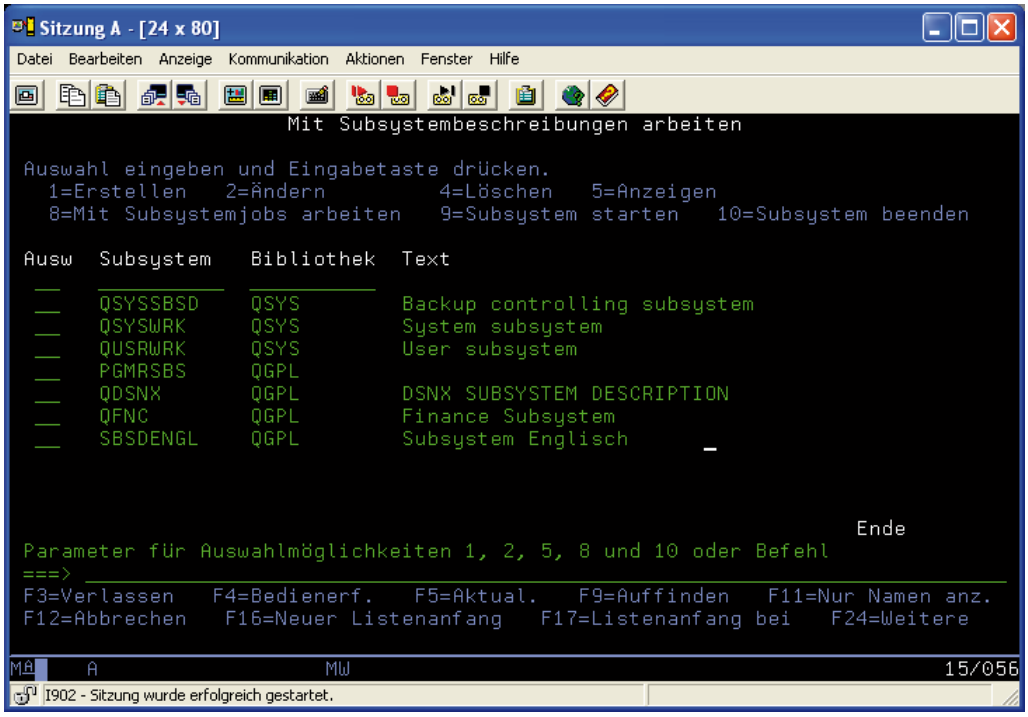
Nachfolgend ein Überblick (Auszug) über die verschiedenen Definitionsebenen für die Globaldefinitionen:

- System
  - Systemwerte

WRKSYSVAL

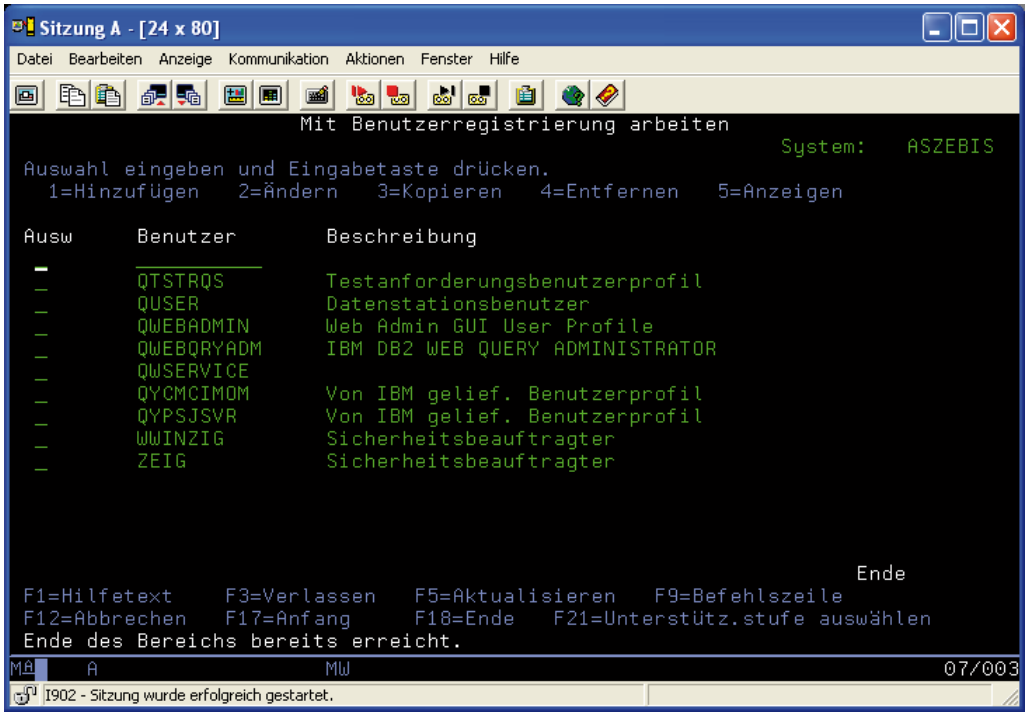
2.10.2.2
Seite 2

- Subsystem
  - Subsystembeschreibungen



WRKSBSD

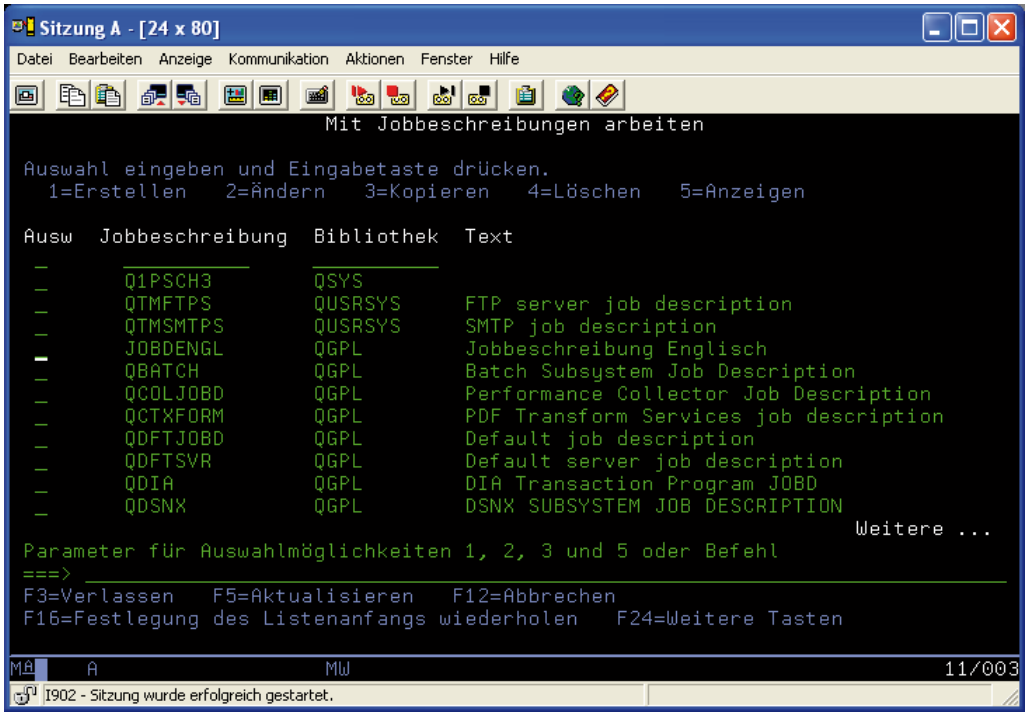
- Benutzerprofil
  - Benutzerprofilbeschreibungen



WRKUSRPRF

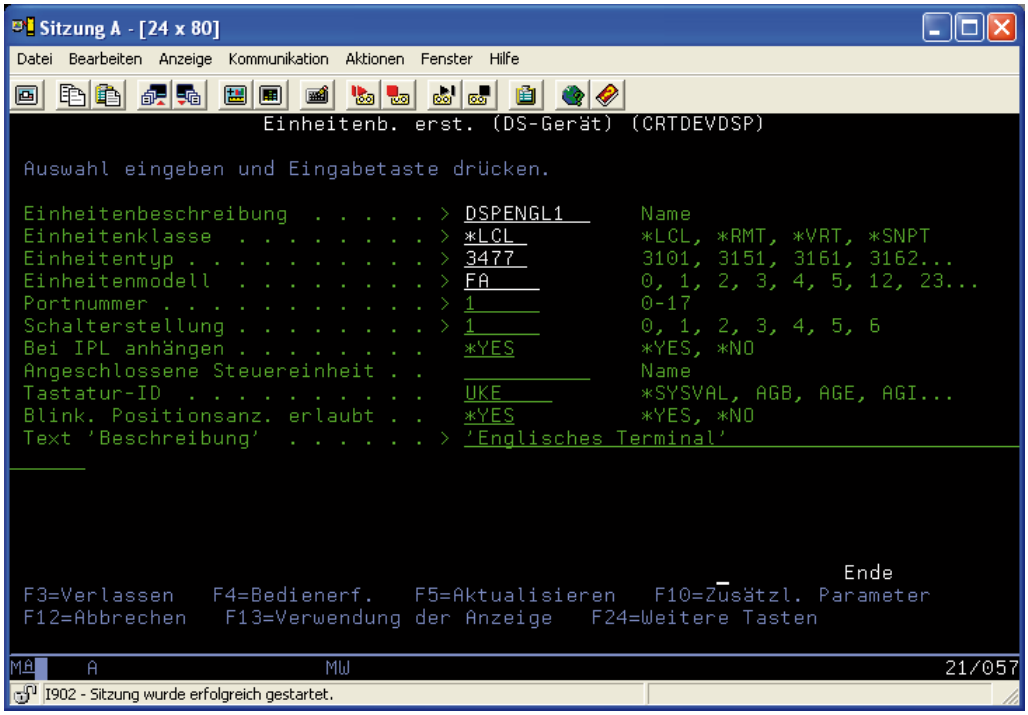
- Jobebene
  - Definition individuell für einen spezifischen Job

2.10.2.2
Seite 4



WRKJOBD

- Einheitenebene
  - Einstellungen für zum Beispiel Drucker oder Eingabegeräte auf Konfigurationsebene. Hier kann unter anderem die Tastaturart für Dialoggeräte angegeben werden.



CRTDEVDSP

Es wäre zu schön, wenn wir die Einstellungen für einen internationalen Einsatz auf einer Ebene vornehmen müssten – aber wohl auch genauso einschränkend. Das Betriebssystem IBM i ist von seiner Auslegung her so flexibel, dass wir die Nutzung in internationalen Umgebungen schnell und flexibel ändern können. Individuelle Einstellungen lassen sich auf tiefster Ebene – zum Beispiel an einem Job – festmachen. Globale Einstellungen hingegen sind auf höchster Ebene wie zum Beispiel über Systemwerte möglich und erfordern keine speziellen Einstellungen, wenn diese systemweit genutzt werden sollen.

2.10.2.2
Seite 6

Systemebene

Auf Systemebene ist sicherlich die Primärsprache als erstes in Bezug auf den internationalen Einsatz zu nennen. Das gesamte Erscheinungsbild, die Texte und Informationen werden in der Primärsprache angezeigt. Optional können zu einer Primärsprache eine oder mehrere Sekundärsprachen installiert werden. Diese liefern dann IBM-Masken, Anzeigen und Hilfetexte in den jeweils installierten Sprachen. Bei dem Einsatz von Sekundärsprachen sind allerdings noch einige weitere Einstellungen vorzunehmen, damit die Sekundärsprache auch korrekt zum Einsatz kommen kann.

Es gibt zudem eine ganze Reihe von Systemwerten, mit denen wir die nationalen und auch internationalen Einstellungen beeinflussen. Zeitzonen, Datumsaufbereitung, Zeichensätze und Landeseinstellungen sind nur einige Beispiele für die Fülle an Definitionsbereichen auf Systemwertebene.

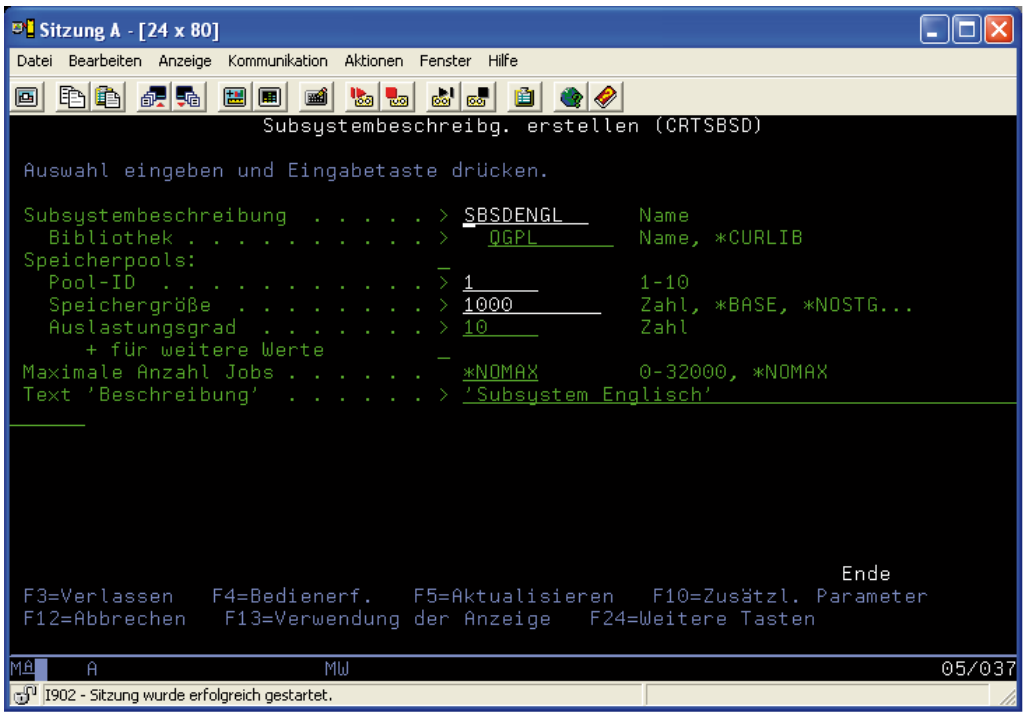
Subsystem
Subsysteme regeln auf dem IBM i die Bewältigung der verschiedensten Anforderungen. Durch geeignete Definitionen von Subsystemen lassen sich auch Anforderungen hinsichtlich eines internationalen Einsatzes der Systeme umsetzen.

Auf der Ebene des Subsystems kann die Sprache für einen Benutzer definiert werden. Zusammen mit installierten Sekundärsprachen lassen sich dann beispielsweise Anzeige und Funktionen in der gewünschten Version darstellen. Dies können wir durch das individuelle Setzen der Anfangsbibliotheksliste auf Subsystemebene erreichen.

Grundsätzlich sollte man situationsbedingt entscheiden, ob eine bestehende Subsystembeschreibung angepasst oder eine neue Subsystembeschreibung erstellt werden soll. Eine neue Subsystembeschreibung bietet sich zum Beispiel bei der Installation und Implementierung einer Sekundärsprache an. Über eine eigene Subsystembeschreibung können wir gezielt die sprachenspezifischen Einstellungen definieren.

2.10.2.2.1
Seite 2

Das Erstellen eines neuen Subsystems erfolgt mit dem Befehl CRTSBSD:



CRTSBSD

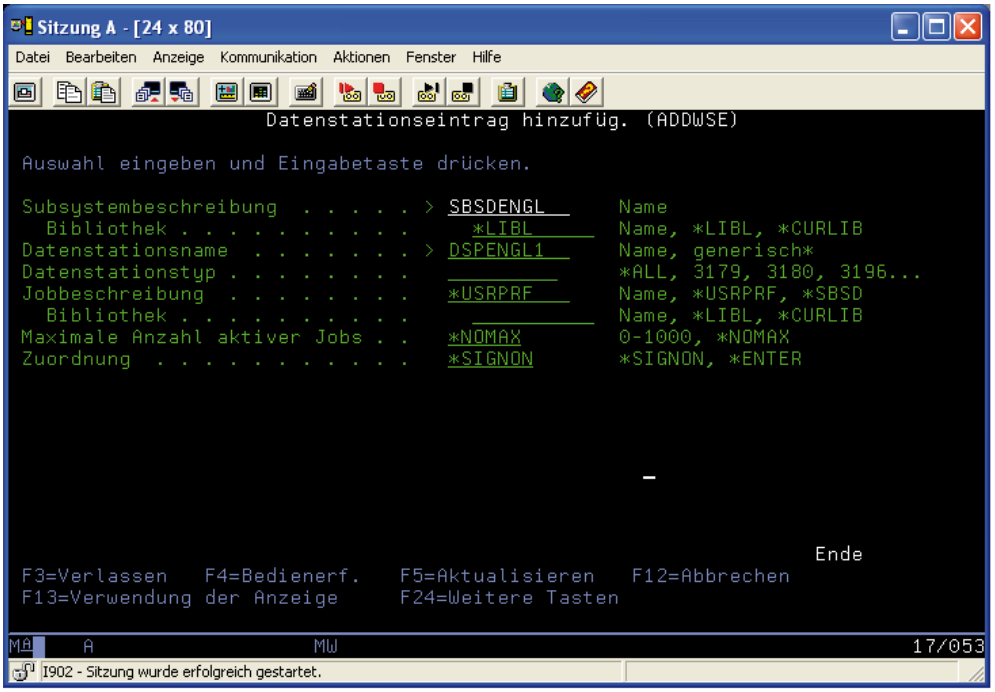
Auf Basis der Subsystemdefinitionen lässt sich beispielsweise die Bibliotheksliste individuell beschreiben und der sprachenabhängige Teil kann an den Anfang der Bibliotheksliste gestellt werden.

Letztlich sind in Bezug auf die internationalen Einstellungen unter anderem die folgenden Bereiche der Subsystembeschreibung anzupassen:

- Workstation Entry (ADDWSE)
  Die Zuordnung von Arbeitsstationen zu einem bestimmten Subsystem kann zum Beispiel Sinn machen, wenn ein Subsystem für eine bestimmte Sprache angelegt wurde und die technischen Anforderungen wie Zeichensatz der Einheit, Tastaturlayout etc. eine solche Zuordnung erfordern.

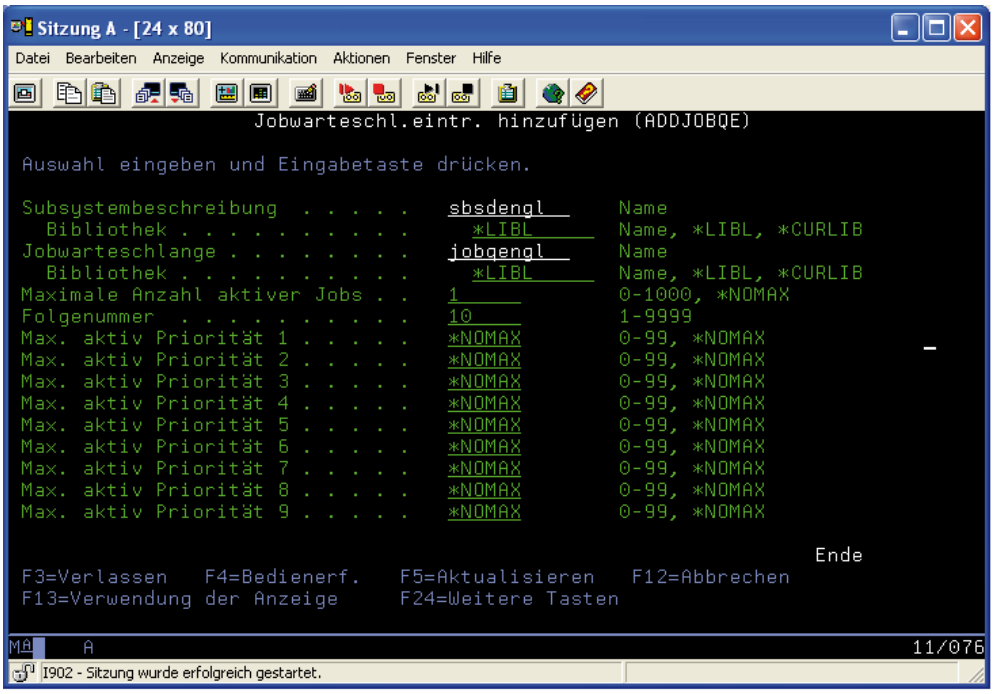
  Grundsätzlich können wir jede fixe Arbeitsstation einem Subsystem fest zuordnen. Damit wird erreicht, dass beim Einschalten der Einheit diese dem definierten Subsystem zugeordnet wird.

  In dem folgenden Beispiel ordnen wir eine Eingabeeinheit „DSPENGL1“ dem Subsystem SBSDENGL zu.



ADDWSE

- Job Queue Entry (ADDJOBQE)



ADDJOBQE

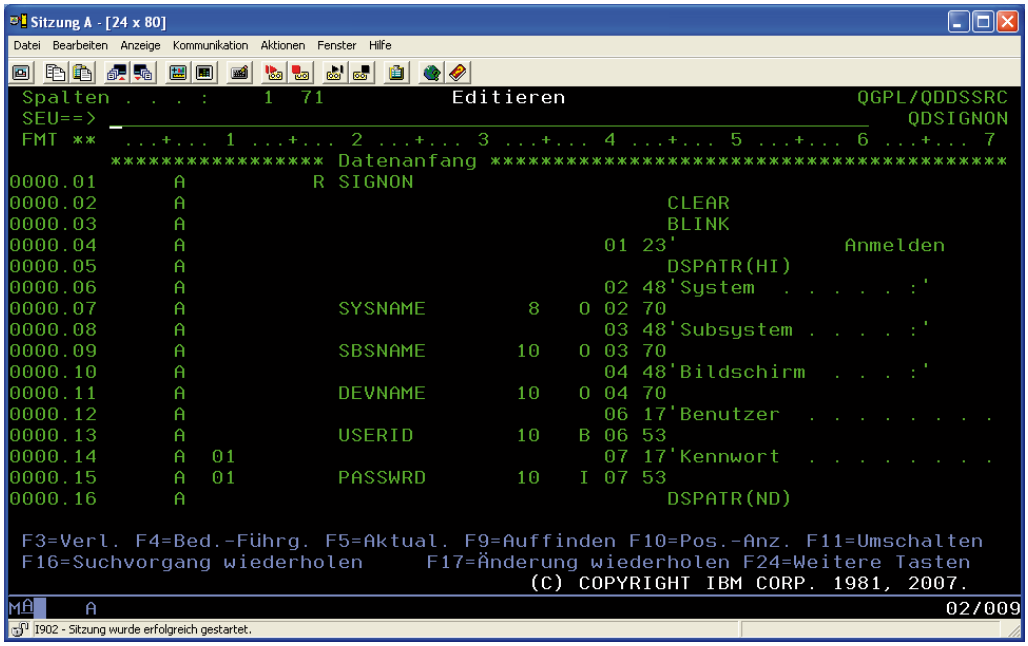
- Communication Entry (ADDCMNE)
- Autostart Entry (ADDAJE)
- Prestart Job Entry (ADDPJE)

2.10.2.2.1
Seite 4

Tipp

Vergessen Sie nicht, auch die Einstellungen für den Anmeldebildschirm anzupassen. Diese Einstellungen können wir ebenfalls in den Subsystembeschreibungen ändern. Hier sollte man natürlich den Login-Bildschirm in der gewünschten Sprache anzeigen, um direkt von Anfang an Irritationen zu vermeiden.

Um den Login-Bildschirm in einer angepassten Sprache darzustellen, kann man die DDS-Beschreibung für den Bildschirm modifizieren. Diese befindet sich in der Regel in der Datei QDDSSRC in der QGPL und trägt den Namen QDSIGNON.



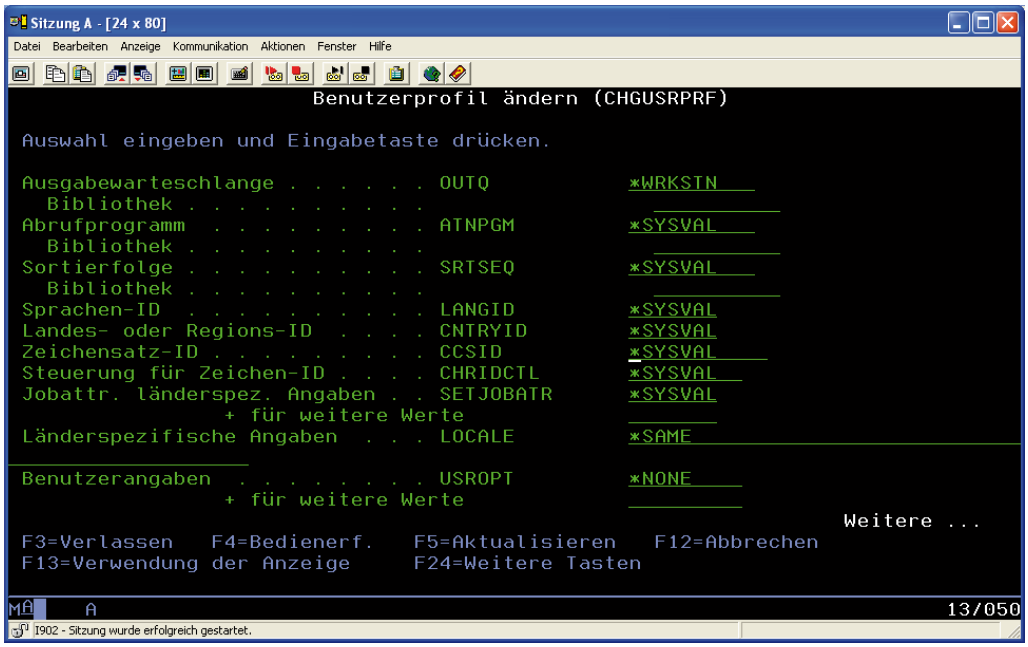
QDSIGNON

Benutzerprofil

Auf einem System können länderspezifische Einstellungen auch auf der Ebene der Benutzerprofile definiert werden. Damit kann man individuell für jeden Benutzer dessen Einstellungen festlegen. Je nach Definition können diese auch von den übergeordneten Ebenen als Standardwerte genutzt werden. Beispiele hierfür sind Systemwerte, die im Normalfall Gültigkeit haben und deren Auswirkung individuell auf der Ebene des Benutzerprofils überschrieben werden können.

Die CCSID lässt sich individuell für jeden Benutzer einstellen. In Abhängigkeit des zu verwendenden Zeichensatzes (Single Byte (kurz: SBCS) oder Double Byte (kurz: DBCS)) kann die CCSID dann auf einen SBCS- oder DBCS-Wert bzw. auf die CCSID 65535 gestellt werden.

Schauen Sie sich zum Anpassen der CCSID auch die folgende Abbildung an, die exemplarisch auch die CCSID in dem Benutzerprofil zeigt. Dieser Wert für die CCSID kann beim Erstellen eines Benutzerprofils (Befehl CRTUSRPRF) oder durch das Ändern (CHGUSRPRF) gesetzt werden.



CHGUSRPRF

Jobebene

Auf Jobebene finden wir die individuellen Jobeinstellungen, welche unter anderem auch die CCSID beinhalten. Diese kann für einen Job auf unterschiedliche Weise gesetzt werden.

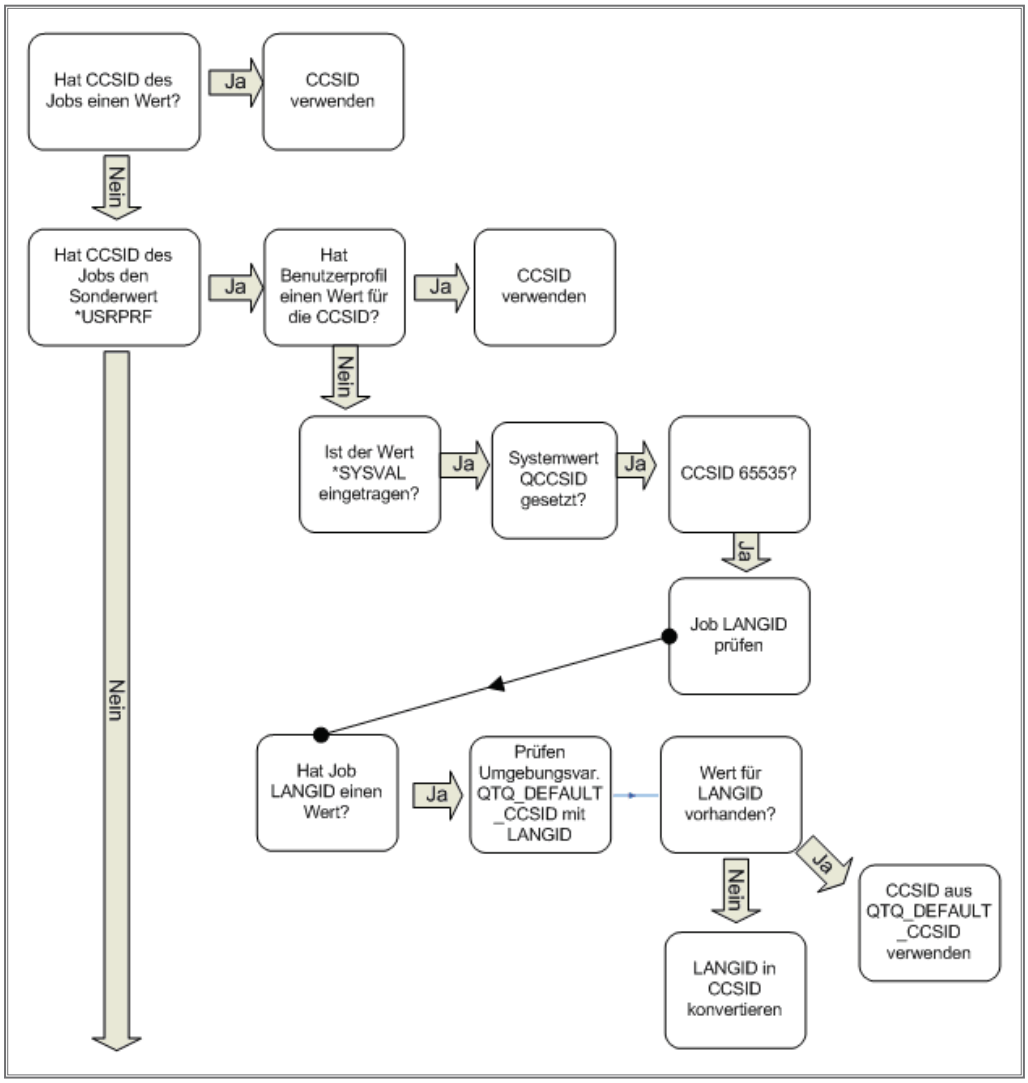
Aber wie wird die CCSID für einen Job ermittelt? Schauen wir uns dazu die folgenden Prüfschritte an:

- Wenn für den Job eine spezifische CCSID angegeben wurde, dann wird diese verwendet.
- Wenn für den Job als CCSID der Sonderwert *USRPRF eingetragen wurde, dann erfolgt eine Prüfung, ob in dem Benutzerprofil, mit dem der Job aufgerufen wurde, eine CCSID angegeben ist. Ist dies der Fall, dann wird diese CCSID für den Job genutzt.
- Ist in dem Benutzerprofil keine CCSID, sondern der Sonderwert *SYSVAL angegeben, dann wird in dem Systemwert QCCSID der Wert ermittelt und für den Job verwendet.

2.10.2.2.1
Seite 6

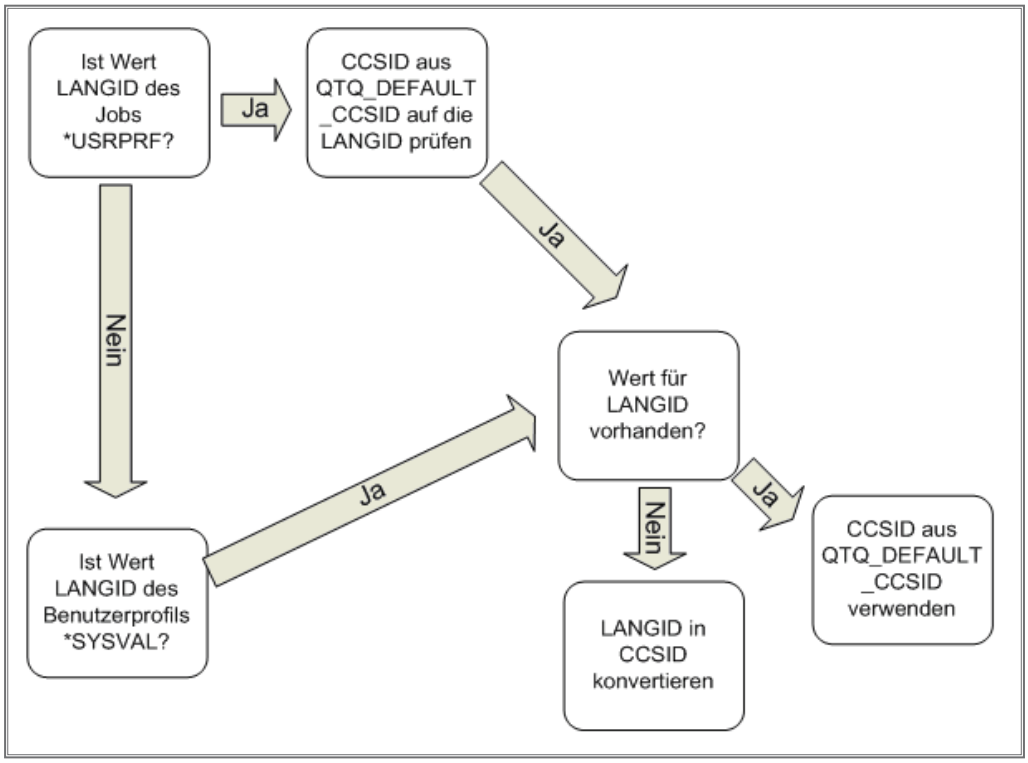
- Ist in QCCSID als Wert 65535 eingegeben, dann wird die Sprach-ID des Jobs geprüft. Ist der Wert für LANGID gefüllt, dann wird die Umgebungsvariable QTQ_DEFAULT_CCSID geprüft und der Wert darin gesucht. Ist die LANGID darin enthalten, dann wird die CCSID des entsprechenden Eintrags aus der QTQ_DEFAULT_CCSID verwendet. Ist kein solcher Eintrag vorhanden, dann setzt das System die LANGID in eine CCSID um.
- Ist der Wert LANGID des Jobs mit *USRPRF vorbelegt, dann wird eine Prüfung der Sprachen-ID des Benutzerprofils durchgeführt.
- Ist die LANGID des Benutzerprofils mit einem Wert gefüllt, dann wird die Umgebungsvariable QTQ_DEFAULT_CCSID geprüft und der Wert darin gesucht. Ist die LANGID darin enthalten, dann wird die CCSID des entsprechenden Eintrags aus der QTQ_DEFAULT_CCSID verwendet. Ist kein solcher Eintrag vorhanden, dann setzt das System die LANGID in eine CCSID um.
- Ist der Wert LANGID des Benutzerprofils mit dem Sonderwert *SYSVAL gefüllt, dann wird die Umgebungsvariable QTQ_DEFAULT_CCSID geprüft und der Wert darin gesucht. Ist die LANGID darin enthalten, dann wird die CCSID des entsprechenden Eintrags aus der QTQ_DEFAULT_CCSID verwendet. Ist kein solcher Eintrag vorhanden, dann setzt das System die LANGID in eine CCSID um.

Die folgende Abbildung zeigt diesen Ablauf nochmals schematisch:



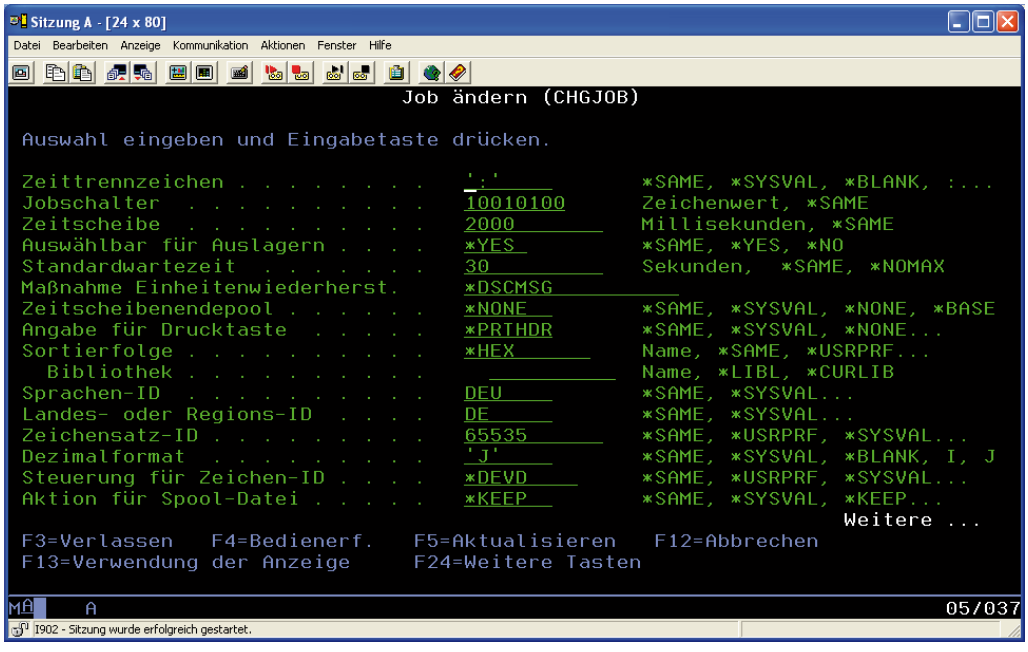
CCSID Findung Teil 1

2.10.2.2.1
Seite 8



CCSID Findung Teil 2

Anpassungen an einen Job lassen sich wahlweise bei der Jobdefinition oder auch nachträglich mit Hilfe des Befehls CHGJOB durchführen. Die nachfolgende Abbildung zeigt einige der Parameter, die für das Ändern eines Jobs verwendet werden können.



CHGJOB

Globale Einstellungen können bei Bedarf auch auf Jobebene angepasst werden. Dies kann beispielsweise für die Ausführung von bestimmten Programmen sinnvoll sein. Unter anderem kann für einen Job die CCSID verändert werden. Dies ist allerdings nicht durchgängig möglich. Der Programmaufruf CALL QCMDEXC kann zwar für den Aufruf von Programmen genutzt werden, leider fehlt hier aber die Möglichkeit der Angabe von Zusatzparametern, welche auch die globalen Einstellungen beeinflussen. Somit kann bei dieser Aufrufvariante die Anpassung des Jobs beispielsweise in Sachen CCSID nicht unmittelbar erfolgen.

Jobs lassen sich mit den Befehlen SBMJOB oder CHGJOB beeinflussen.

Interaktive Jobs verwenden die CCSID des Benutzerprofils, mit dem der Job aufgerufen wird. Bei dem Einsatz eines Batchjobs wird die CCSID des aufrufenden Jobs genutzt. Ausnahme hier ist die gezielte Angabe einer CCSID zusammen mit dem Befehl SBMJOB.

ACHTUNG!

Wenn ein Job mit einer SBCS CCSID ausgeführt wird, dann können mit diesem Job KEINE DBCS-Daten verarbeitet werden! Die DBCS-Unterstützung ist nur dann möglich, wenn eine mixed CCSID verwendet wird.

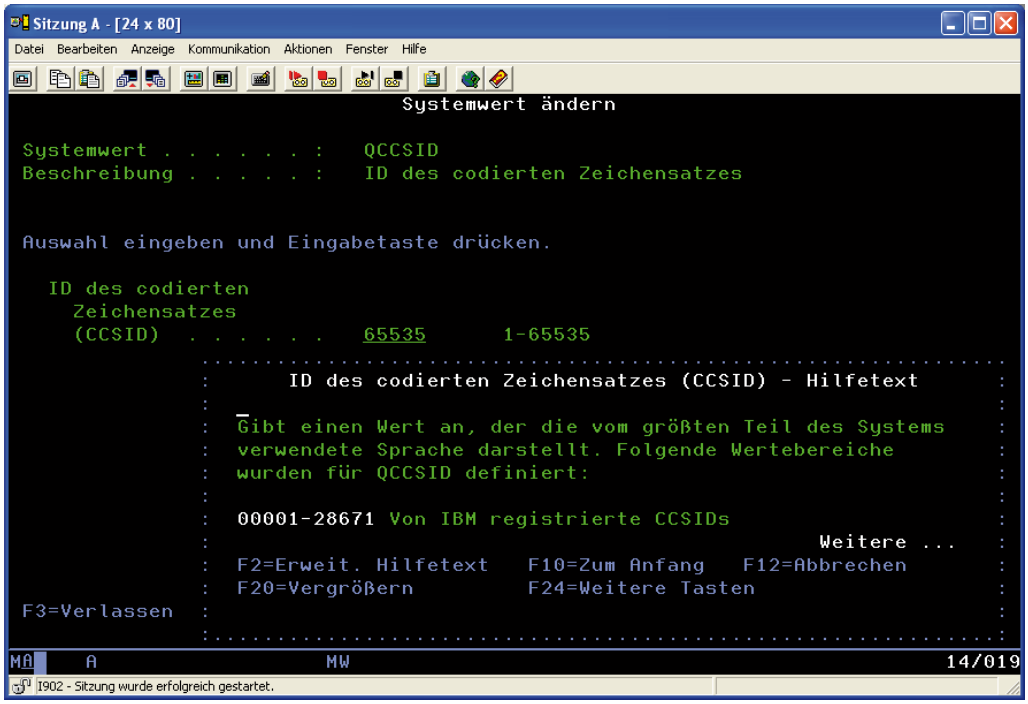
Systemwerte

Für die Globalisierung bzw. die internationalen Einstellungen stehen uns verschiedene Systemwerte zur Verfügung. Diese müssen in der Regel aufeinander abgestimmt gesetzt werden und bestimmen das allgemeine Verhalten des Systems, da sie als Grundeinstellungen quasi als Standardwerte genutzt werden.

Die folgenden Systemwerte können wir für die nationalen Einstellungen definieren:

- QCCSID (ID des codierten Zeichensatzes)
  Mit diesem Systemwert legen wir die für das System allgemein gültige Zeichensatz-ID fest. IBM liefert diesen Wert per Standard mit der Voreinstellung 65535 aus. Damit werden dann allerdings auch bestimmte Zeichenerkennungen nicht durchgeführt. Grundsätzlich sollte man die Verwendung der Standardeinstellung 65535 überdenken und einen besser geeigneten Wert wählen, der zur eingesetzten Sprache am besten passt. Insbesondere der Datenaustausch oder die Kommunikation mit nicht originären IBM i-Anwendungen wie zum Beispiel Java-Anwendungen, WebSphere-Programmen oder Domino-Anwendungen können mit der Einstellung 65535 Probleme bekommen.

2.10.2.2.1
Seite 10

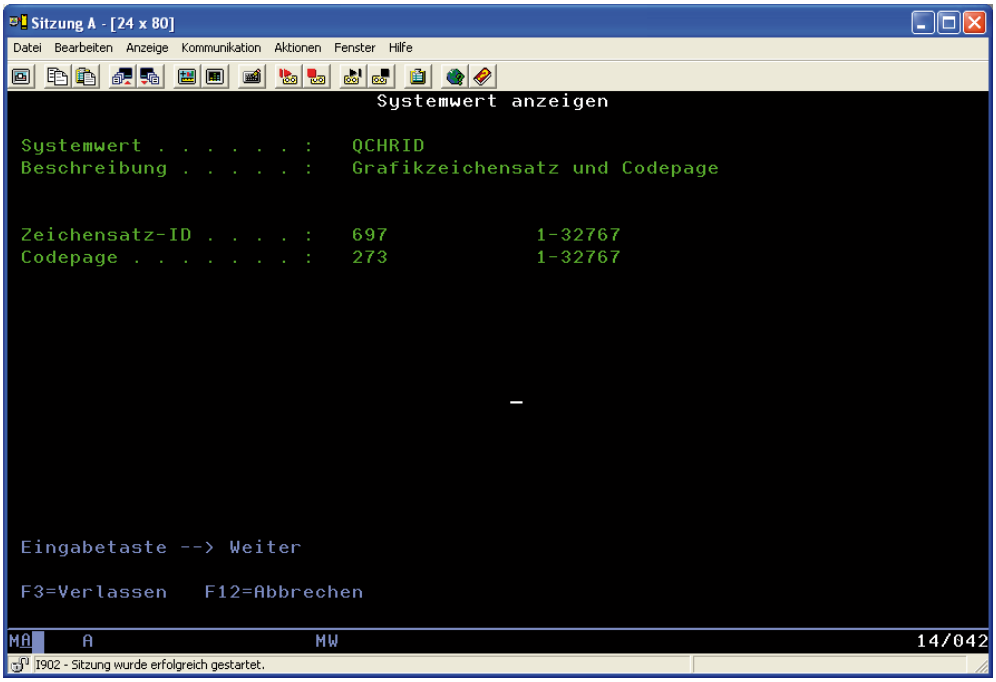


QCCSID

Beachten Sie bei der Wahl der CCSID die unterschiedlichen SBCS- und DBCS-Versionen. Hier ist die Kompatibilität wieder immens wichtig!

Eine Beispieleinstellung für ein deutsches System kann hier sein: 1141(273).

- QCHRID – Grafikzeichensatz und Codepage
  Mit diesem Systemwert legen wir den Zeichensatz und die Codepage fest, mit denen die Bildschirmanzeigen und Druckausgaben versehen werden. Der Wert von QCHRID kann dann geändert werden, wenn in QCCSID der Wert 65535 eingestellt ist. Änderungen sind auch möglich, wenn der Parameter „Codepage“ denselben Wert besitzt wie die QCCSID.



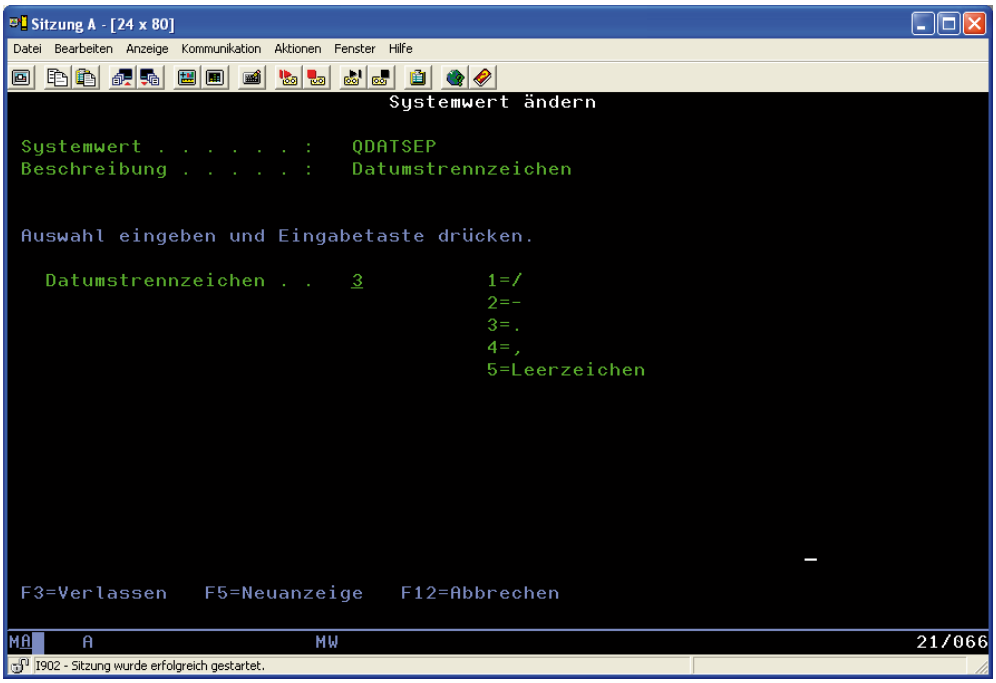
QCHRID

Eine Beispieleinstellung für ein deutsches System kann hier sein: QCHRID 697 1141(273)

- QCHRIDCTL – Steuerung für Zeichen-ID
  Mit diesem Systemwert wird die Zeichen-ID festgelegt, welche für die Darstellung der Daten in Bildschirmanzeigen und Druckausgaben genutzt wird. Hier sind zwei Sonderwerte als Eingabeoptionen möglich:
  – *DEVD
  – *JOBCCSID
- QCNTRYID Landes- oder Regions-ID
  Dieser Systemwert ist zusammen mit dem Wert aus QLANGID für die korrekte Entschlüsselung der Daten und korrekte Verwendung der Sprachspezifika verantwortlich.
  Dieser Wert wird für ein deutsches System mit „DE“ eingestellt.
- QCURSYM Währungssymbol
  In diesem Systemwert legen wir das Währungssymbol fest, das mit den DDS-Schlüsselwörtern EDTWRD oder EDTCDE genutzt werden kann. Gewöhnlich verwendet man hier in Deutschland das Eurozeichen €.
- QDATFMT Datumsformat
  Mit diesem Systemwert legen wir die Darstellung von Datumswerten fest. Für ein deutsches System bietet sich die Darstellung in Form von Tag/Monat/Jahr an – dies ist der Wert DMY.

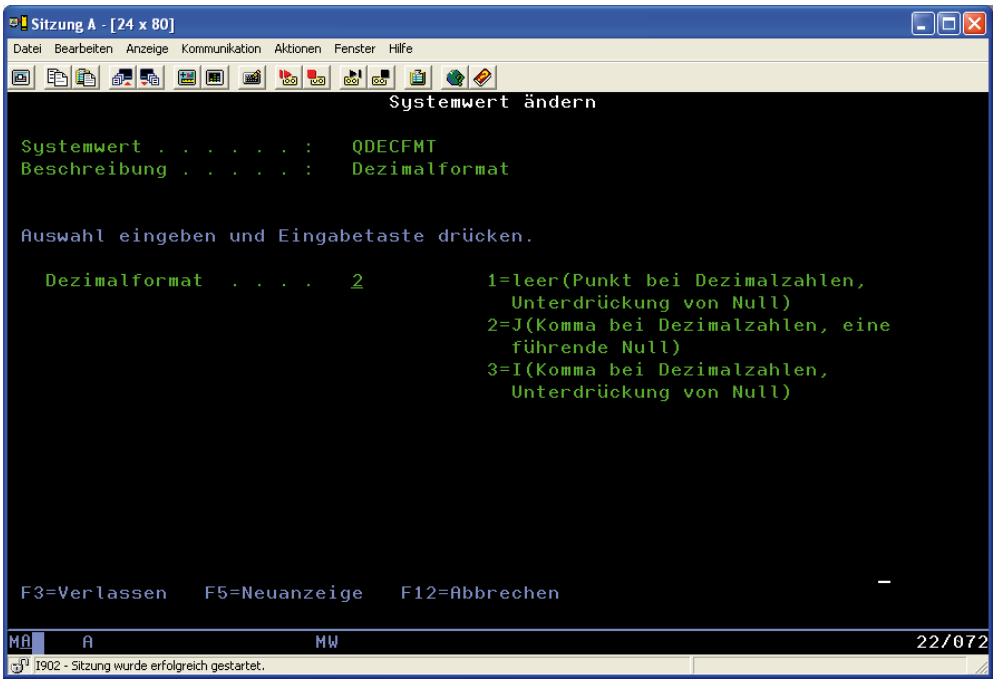
2.10.2.2.1
Seite 12

- QDATSEP Datumstrennzeichen
  Dieser Wert definiert das Trennzeichen innerhalb eines Datumsfelds. Für ein deutsches System nimmt man gewöhnlich den Wert 3, der dem Punkt(.) entspricht. Die folgende Abbildung zeigt die möglichen Werte und deren Bedeutungen:



QDATSEP

- QDECFMT
  Mit diesem Systemwert definieren wir die globalen Dezimaltrennzeichen, die zusammen mit den DDS-Editiercodes zum Einsatz kommen. Die gültigen Werte können Sie in der nachfolgenden Abbildung finden:

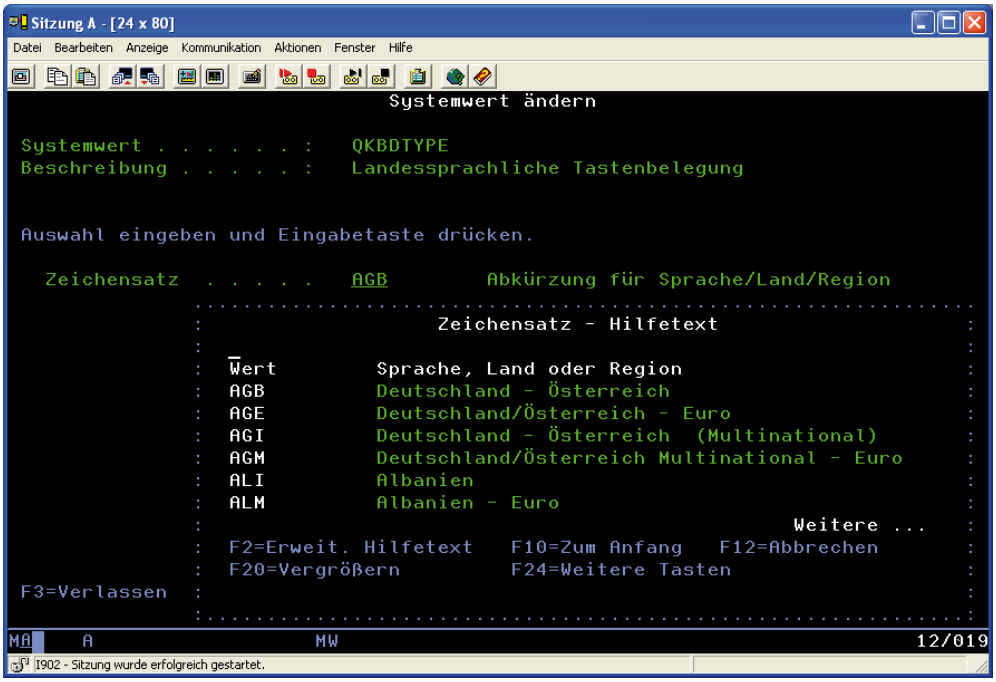


QDECFMT

- QIGC-Anzeiger für installierte DBCS-Version
  Dieser Systemwert ist nicht veränderbar und wird automatisch auf „1“ (installiert) gestellt, wenn eine DBCS-Version auf dem System installiert ist.

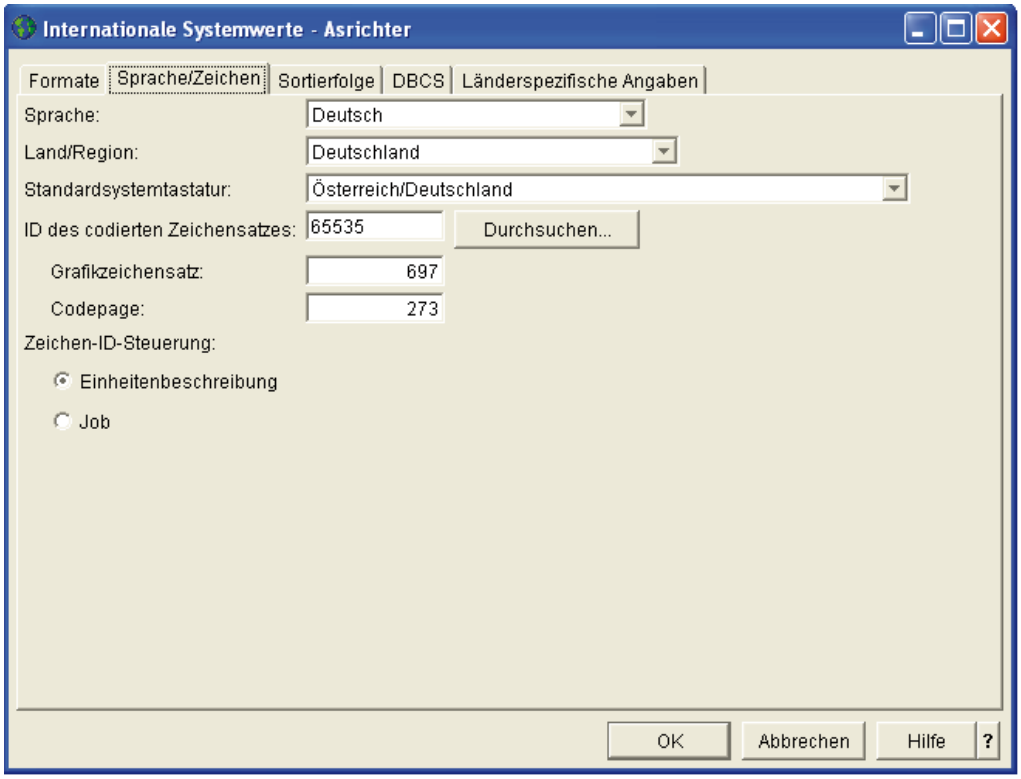
  Seit der Version V5R3 des Betriebssystems sind alle Primärsprachen automatisch als DBCS-fähig ausgeliefert. Deshalb finden wir im Normalfall auf allen neueren Maschinen den Wert „1“ für QICG. Wenn der Wert 0 ist, dann muss in dem Systemwert QCCSID auch ein SBCS-Zeichensatz angegeben werden! Wenn auf einem System eine Primärsprache mit DBCS-Unterstützung installiert ist, dann wird der Wert von QIGC automatisch auf „1“ gesetzt. In diesem Fall sollte in QCCSID eine Angabe für eine mixed CCSID hinterlegt werden. Als Beispiel könnte dort dann 05026 oder 65535 eingetragen werden.
- QIGCCDEFNT DBCS-Schriftarten
  Dieser Systemwert wird dann benutzt, wenn auf einem System Druckausgaben mittels AFP und AFPDS gedruckt werden sollen.
- QKBDTYPE Landessprachliche Tastenbelegung
  Dieser Systemwert definiert den zu verwendenden Zeichensatz für die Tastatureingabe. Für ein deutsches System wird hier AGB, AGE, AGI oder AGM eingestellt.

2.10.2.2.1
Seite 14

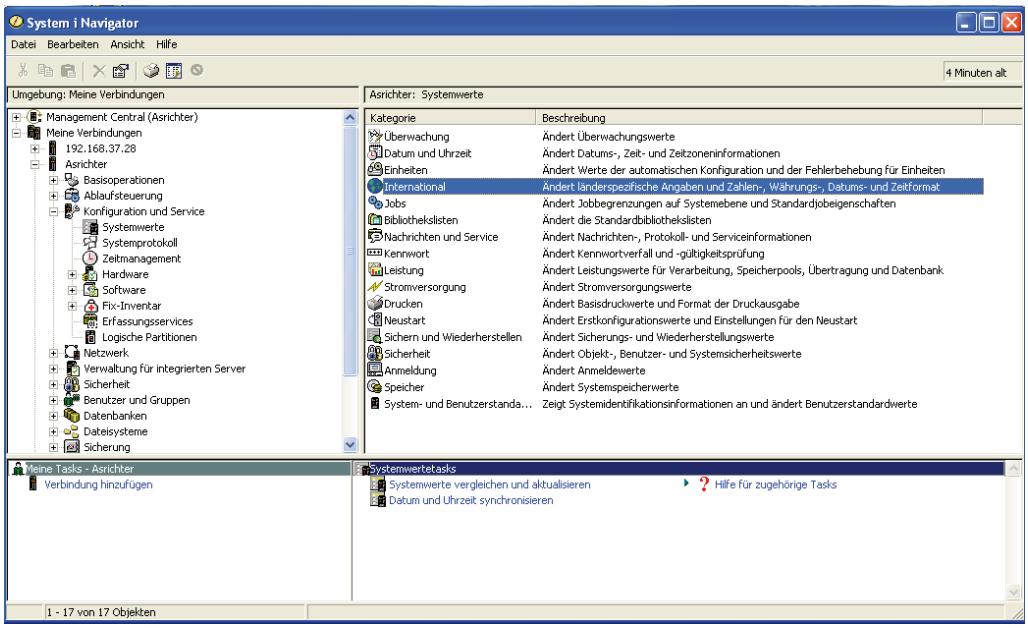


QKBDTYPE

- QLANGID Sprachen-ID
  Mit diesem Wert legen wir die Sprache fest, die Benutzer im Regelfall verwenden. Für ein deutsches System wird hier „DEU“ als Wert angegeben. Dieser Systemwert kommt unabhängig von der ausgewählten CCSID zum Einsatz.
- QSETJOBATR Jobattribute aus länderspezifischen Angaben übernehmen
  Mit diesem Systemwert werden bestimmte Jobattribute während des Startens eines Jobs gesetzt. Dabei können die folgenden Werte definiert werden:
  – *CCSID zu verwendender Zeichensatz
  – *DATFMT zu verwendendes Systemdatumsformat
  – *DATSEP zu verwendendes Trennzeichen für Datumsangaben
  – *TIMSEP zu verwendendes Trennzeichen für Zeitangaben
  – *SRTSEQ zu verwendender Algorithmus für Sortierfolgen
  – *DECFMT zu verwendendes Systemdatumsformat
- QTIMSEP
  Dankenswerterweise sind diese Systemwerte in dem System i Navigator in der Gruppe „International“ zusammengefasst und lassen sich dort verhältnismäßig einfach anpassen.



Systemwerte im System i Navigator



Internationale Systemwerte

2.10.2.2.1

Seite 16

Anwendungen

Bei dem Einsatz in internationalisierten Umgebungen sind die Anwendungen natürlich ebenfalls wichtig. Was würden Sie auf Anhieb sagen, müsste bei der Entwicklung von Anwendungen und deren globalem Einsatz bedacht werden?

Hier sind nur einige wenige Punkte, die es bei der Ausrichtung von Anwendungen für einen globalen Einsatz zu bedenken gilt.

- Anwendungssprachen
- CCSID-Unterstützung
  - Daten müssen in einer umsetzbaren CCSID hinterlegt sein.
  - Einheiten müssen die CCSID unterstützen.
- Anwendungssteuerung und Logik müssen auf internationalen Einsatz ausgelegt sein:
  - So sind z..B. einfache Abfragen mit Ja (J) / Nein (N) unter Umständen anzupassen (YES/Y bzw. NO/N).
- Im Programmcode enthaltene Sonderzeichen
  - Zwar nur wichtig zu Wandlungszeiten – aber was ist mit Sonderzeichen, die in einem anderen Zeichensatz (auf einer anderen Maschine in einer Auslandsniederlassung) nicht unterstützt oder falsch umgesetzt werden?
- Sortierungen
  - Häufig werden die Sortierfunktionen, die Standard-IBM-Anwendungen oder auch individuelle Programme bieten, für die Planung eines internationalen Einsatzes gar nicht berücksichtigt.
- Währungsformate
  - Werden Währungsformate korrekt behandelt?
- Trennzeichen
  - Welche Trennzeichen müssen unterstützt werden?
- ……

Hätten Sie alle diese Punkte auch aufgezählt? Der Vollständigkeit halber sei erwähnt, dass diese Auflistung nur ein Einblick in die verschiedensten Themenbereiche ist, die in Bezug auf die Anwendungsentwicklung und einen internationalen Einsatz berücksichtigt werden müssen.

2.10.2.3
Seite 2

Übersetzungen sind hier gar nicht erst aufgeführt – lassen sie sich doch nur im seltensten Fall mit einem Tool erzeugen. Zu gerne erinnere ich mich daran, wie allen Ernstes ein Teilnehmer auf einer US-Konferenz in einem Vortrag zu Internationalisierung sagte „Die Sekundärsprache Deutsch hat ein Problem – die Übersetzungen funktionieren nicht." Nach langen Diskussionen hatte damals der Referent herausfinden können, dass der Teilnehmer tatsächlich davon ausgegangen war, dass mit der Installation einer Sekundärsprache quasi automatisch alle Inhalte von Dateien übersetzt und dem Zielanwender in dessen Sprache präsentiert würden.

In diesem Abschnitt möchte ich Sie für die Teilbereiche der Anwendungsentwicklung und Datenhaltung sensibilisieren, die wir im Hinblick auf einen multilingualen Einsatz beachten sollten.

Die Merkmale für einen internationalen und multilingualen Einsatz einer Anwendung sind:

- Sprachenunabhängiger Zentralbereich
- Sprachenabhängiger Individualbereich
  - Zeichensätze
  - Konstanten
  - Währungen
  - Datums-/Zeitangaben
  - Aufbereitungs-/Editiercodes
  - Sonderzeichen
  - Steuerzeichen
  - Sortierungen
  - Kulturelle Besonderheiten
  - Unterstützung von Ein-/Ausgabegeräten
  - Integration von Nachrichtendateien (auch Bedienerhilfen)
  - Umsetzungen (z. B. Abkürzungen für Wochentage (Mo, Di, Mi….))
  - Abkürzungen
  - …..

Natürlich lassen sich Anwendungen so gestalten, dass es für die verschiedenen Sprachen individuelle Displayfiles gibt – aber sicher ist dies keine sonderlich geschickte Lösung, denn diese bedarf einer entsprechend umfangreichen Wartung.

Die Grundlage für die Entwicklung von multilingualen Anwendungen ist sicher die korrekte Einstellung der Basis – sprich der Konfiguration der Datenbank und des Betriebssystems mitsamt der Peripherie.

Für ein zentral in Deutschland betriebenes System bieten sich die nachfolgenden Einstellungen als Grundlage für einen globalen Einsatz an:

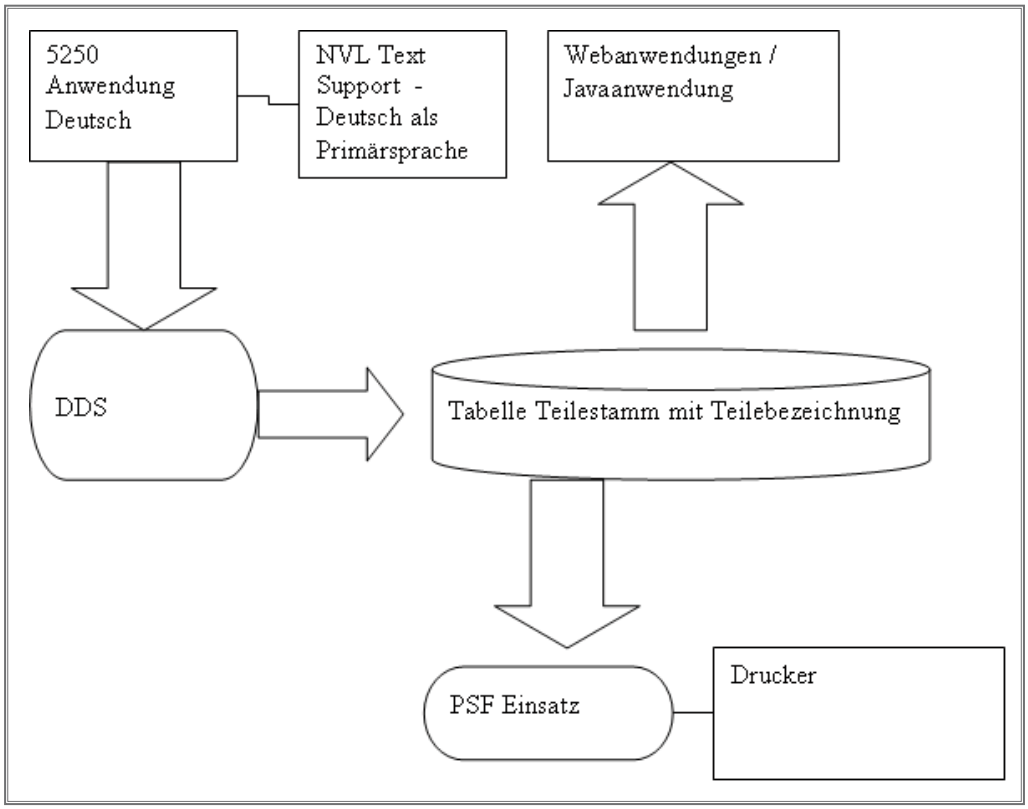
QCHRID	QDECFMT	QKBDTYPE	QCURSYM	QDATSEP	QDATFMT	QTIMSEP
00697 00273	J	AGB	$	.	DMY	:
QCCSID	QCNTRYID	QLOCALE	QLANGID	Internet CCSID	Client Entschl üssel.	
00273	DE	DE_DE	DEU	00819	ISO-8859-1 Latin1	

Empfehlung Systemeinstellungen

Die nachfolgende Skizze zeigt einen typischen Informationsfluss. Dieser ist auf viele Anwendungen übertragbar und kann zum Beispiel auch in einer Auftragsverwaltung vorgefunden werden. Stellen Sie sich einmal vor, dass ein Kunde Büroklammern bestellt. Diese werden in einem Auftragserfassungsprogramm als Artikelnummer eingegeben. Der Artikeltext „Büroklammern“ wird aus dem Teilestamm eingelesen und in der Auftragsposition erfasst. Diese wird in dem Erfassungsprogramm angezeigt. Idealerweise erscheint hier auch wirklich „Büroklammern“ und nicht ein Ersatz, der das für den deutschen Zeichensatz typische „ü“ umsetzt.

Selbiges gilt auch für den Druck der Auftragsbestätigung. Nicht selten findet man auf Ausdrucken fehlerhafte Umsetzungen von Sonderzeichen und/oder Umlauten.

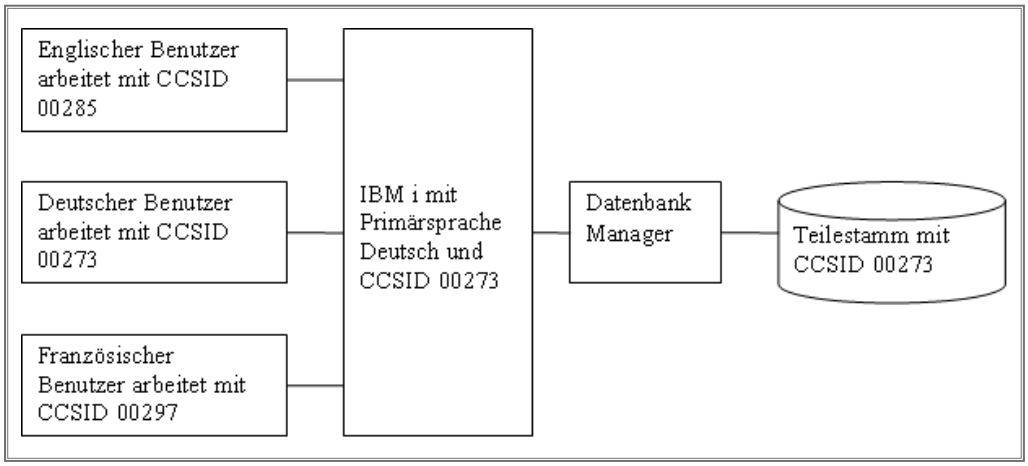
2.10.2.3
Seite 4



Schematischer Ablauf Informationsfluss

Ein weiteres Beispiel finden Sie in der folgenden schematischen Darstellung, welche ein System darstellt, auf dem Anwender aus verschiedenen Ländern arbeiten. Ein Anwender arbeitet in England mit der CCSID 00285, ein Anwender arbeitet in Deutschland mit der CCSID 00273 und ein weiterer Kollege arbeitet von Frankreich aus mit der CCSID 00297.

Das zentrale System steht in Deutschland und wird mit der CCSID 00273 betrieben. Sind die systemseitigen Einstellungen und die Definitionen für die nationalen Einstellungen korrekt, dann sollten die verschiedenen Anwender kein Problem haben, die Daten in ihrer jeweils korrekten Version nicht nur zu sehen, sondern diese auch für alle nutzbar zentral abzulegen und letztlich auch zu drucken.

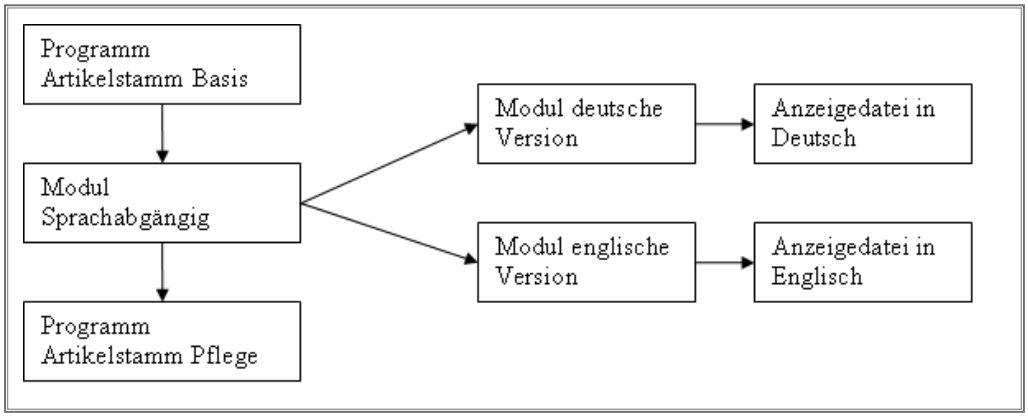


Schematischer Ablauf CCSID-Kompatibilität

Leider würden ohne entsprechende Einstellungen in diesem Beispiel die Eingaben von dem französischen Kollegen dazu führen, dass dessen Daten nicht korrekt abgelegt werden, da die französischen Sonderzeichen nicht in dem Standardzeichensatz der CCSID 00273 enthalten sind und die Datenbankelemente bzw. die Tabelle Teilestamm mit der CCSID 00273 angelegt sind.

Für die Entwicklung von internationalen Anwendungen bietet sich die Differenzierung der Anwendungen in einen sprachenabhängigen und einen sprachenunabhängigen Bereich an.

Meist sind es die Dialog- und Ausgabebereiche, die einer Sprachenabhängigkeit unterliegen. Aber Vorsicht: auch programminterne Steuerungen müssen so ausgelegt sein, dass diese für andere Länder oder vielmehr für andere Sprachen einwandfrei funktionieren. Beispiele für Bereiche, die hier besonders bedacht werden müssen, sind Konstanten, aber auch Listfunktionen oder Sortierfunktionen, die wir beispielsweise auch bei SETLL- und READE-Zugriffen verwenden.



Sprachenspezifische Anwendungsentwicklung

2.10.2.3

Seite 6

Konstanten

Die Verwendung von Konstanten und auch besonders der Gebrauch von Abfragen mit Eingabemöglichkeiten J/N sind weit verbreitet. Aber wie setzt man dies für einen Einsatz beispielsweise mit englischer Sprache um? Dem englischen Kollegen zumuten, dass er an Stelle seines gewohnten Y (Yes) nun ein J eingeben muss – oder doch konsequent den Programmcode ändern?

```
If Dsp.Antwort = 'J';
  ExFmt Dsp01;
Else;
  ExFmt DspNok;
Endif;
```

Beispiel Konstanten im Programmcode

Es dürfte schwierig werden, einem Franzosen oder einem Engländer zu vermitteln, dass er „J“ für „Ja“ (oui oder yes) angeben soll. Hier sollte man in dem Programm auf den Einsatz von Konstanten verzichten und vielmehr eine Variable nutzen, mit der Endanwender ihre Sprachversion für „Ja“ eingeben können.

Auch die nachfolgende Abbildung enthält einen für den internationalen Einsatz nicht zu gebrauchenden Codeausschnitt:

```
   * String Startzeichen Variable
D Wrk_Str_Sta     S               2A   Inz('<%')
   * String Endezeichen Variable
D Wrk_Str_End     S               2A   Inz('%>')

   * String Edit Steuerzeichen Start
D Wrk_Edt_St1     S               1A   Inz('[')
   * String Edit Steuerzeichen Ende
D Wrk_Edt_St2     S               1A   Inz(']')

   * String Inhaber Startzeichen
D Wrk_Inh_St1     S               1A   Inz('{')
   * String Inhaber Endezeichen
D Wrk_Inh_St2     S               1A   Inz('}')
```

Beispiel Sonderzeichen im Programmcode

Sonderzeichen wie in dem Codefragment enthalten, werden nicht mit jeder CCSID unterstützt. Deshalb lässt sich zwar ein solches Programm als Objekt ausführen. Ein Umwandeln dieses Codes beispielsweise auf einer Maschine mit abweichender CCSID würde aber unter Umständen zu Fehlern führen, da die eckigen Klammern anders interpretiert werden.

2.10.2.3.1
Seite 2

Auch das folgende Beispiel ist für einen internationalen Einsatz nicht durchgängig brauchbar – dieser Programmcode könnte auf einer finnischen Maschine wegen abweichender Unterstützung bzw. Darstellung des Dollarzeichens nicht kompiliert werden.

```
     * String Startzeichen Variable
    D Wrk_Str_Sta     S                2A   Inz('<%')
     * String Endezeichen Variable
    D Wrk_Str_End     S                2A   Inz('%>')

     * String Edit Steuerzeichen Start
    D Wrk_Edt_St1     S                1A   Inz('[')
     * String Edit Steuerzeichen Ende
    D Wrk_Edt_St2     S                1A   Inz(']')

     * String Inhaber Startzeichen
    D Wrk_Inh_St1     S                1A   Inz('{')
     * String Inhaber Endezeichen
    D Wrk_Inh_St2     S                1A   Inz('}')
```

Problematik Sonderzeichen im Programmcode

Datum

Die Darstellung und Verarbeitung von Datumswerten ist ebenfalls ein wichtiger Bereich, der bei einer international ausgelegten Anwendung unbedingt bedacht werden muss. Die Vielfältigkeit der verschiedenen Datumsformate findet zwar nur selten in vollem Umfang Anwendung, aber gerade bei der Integration von britischen Anwendern oder US-Anwendern sind Abweichungen vom deutschen Standard zu berücksichtigen.

Hier die vom Betriebssystem und den Entwicklungswerkzeugen unterstützen Datumsformate:

- *MDY (Monat, Tag, Jahr)
- *DMY (Tag, Monat, Jahr)
- *YMD (Jahr, Monat, Tag)
- *JUL (JJ/TTT)
- *ISO (JJJJ-MM-TT)
- *USA (MM/TT/JJJJ)
- *EUR (TT.MM.JJJJ)
- *JIS (JJJJ-MM-TT)

Neben den Datumsformaten sind es zudem die Trennzeichen, die landesspezifische Unterschiede aufweisen. Gültige Trennzeichen sind:

- / (Slash)
- - (Bindestrich)
- . (Punkt)
- , (Komma)
- (blank)

Datumsformate können beispielsweise bei der Erstellung von Dateien definiert werden. In DDS verwenden wir das Schlüsselwort DATFMT und geben als Datentyp „L“ ein – in SQL definieren wir das Feld mit dem Typ DATE.

```
CREATE TABLE DEMO/DATUM (DATUMSFELD DATE )
```

Die Darstellung des Datumsformats richtet sich nach den Vorgaben des Jobs oder Benutzers. Auch in einem Programm lässt sich Einfluss auf die Darstellung von Datumswerten nehmen. Mit der Anweisung DATEDIT in den H-Bestimmungen können wir zum Beispiel das Erscheinungsbild von Datumswerten festlegen, die mit dem Editiercode Y definiert werden.

2.10.2.3.2

Seite 2

```
H DECEDIT('0,') DATEDIT(*DMY.)
```

Beispiel für DATEDIT in RPG H-Anweisung

Mit dem Schlüsselwort DATFMT (ebenfalls in den H-Anweisungen von RPG anzugeben) legen wir das interne Datenformat für Datumswerte fest.

```
H DATEDIT(*DMY.)
H DATFMT(*EUR.)
```

Beispiel für DATFMT zusammen mit DATEDIT

Sortierungen

Insbesondere Sortierungen sind häufig der Grund für die Verwendung von sprachenspezifischen logischen Sichten oder Indizes, die in Programmen eingebunden für die korrekte Verarbeitung der Daten sorgen.

Beispiele für die Verwendung von Sortierungen sind in vielen Bereichen der Maschine zu finden. Seien es SQL-Abfragen, Query-Definitionen oder Anweisungen in Programmen, mit denen wir die Reihenfolge der Informationsermittlung oder Datendarstellung beeinflussen wollen. Aufsteigende oder absteigende Darstellung oder Verarbeitung von Informationen sind in den meisten Anwendungen anzutreffen. Die Selbstverständlichkeit der korrekten Sortierung verdeckt aber die Tatsache, dass intern die für uns lesbaren Informationen byteweise so aufbereitet werden müssen, dass das Ergebnis für uns stimmt.

Hier einige Anweisungen, die in Programmen für Sortierungen genutzt werden können:

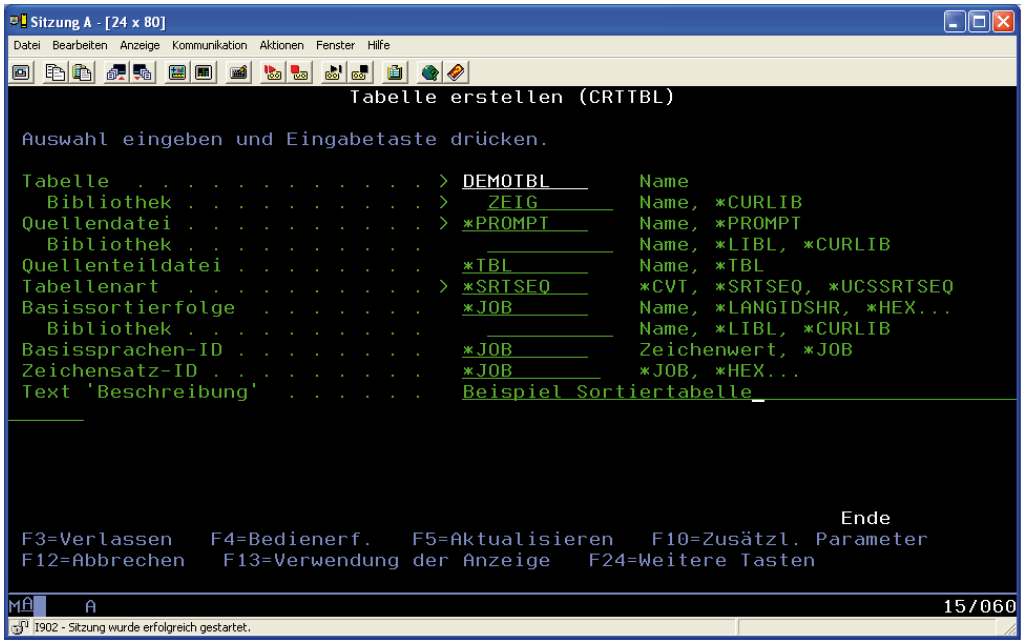
- SORT
- %SORT
- LOOKUP
- %LOOKUP

Sortierungen laufen IBM i intern über Sortiersequenz-Tabellen. Diese spezielle Form von Tabellen ist in der Hauptbibliothek QSYS abgelegt und definiert. Der Hexwert ist letztlich ausschlaggebend dafür, welche Reihenfolge bei Sortierangaben genutzt wird.

Schauen wir uns einmal den Prozess der Anlage und Definition einer eigenen Sortiertabelle an:

Basis dafür bildet der Befehl CRTBL (Tabelle erstellen). Dieser Befehl kann nicht nur für die Erstellung von Sortiertabellen, sondern auch für Umsetzungstabellen verwendet werden.

2.10.2.3.3
Seite 2

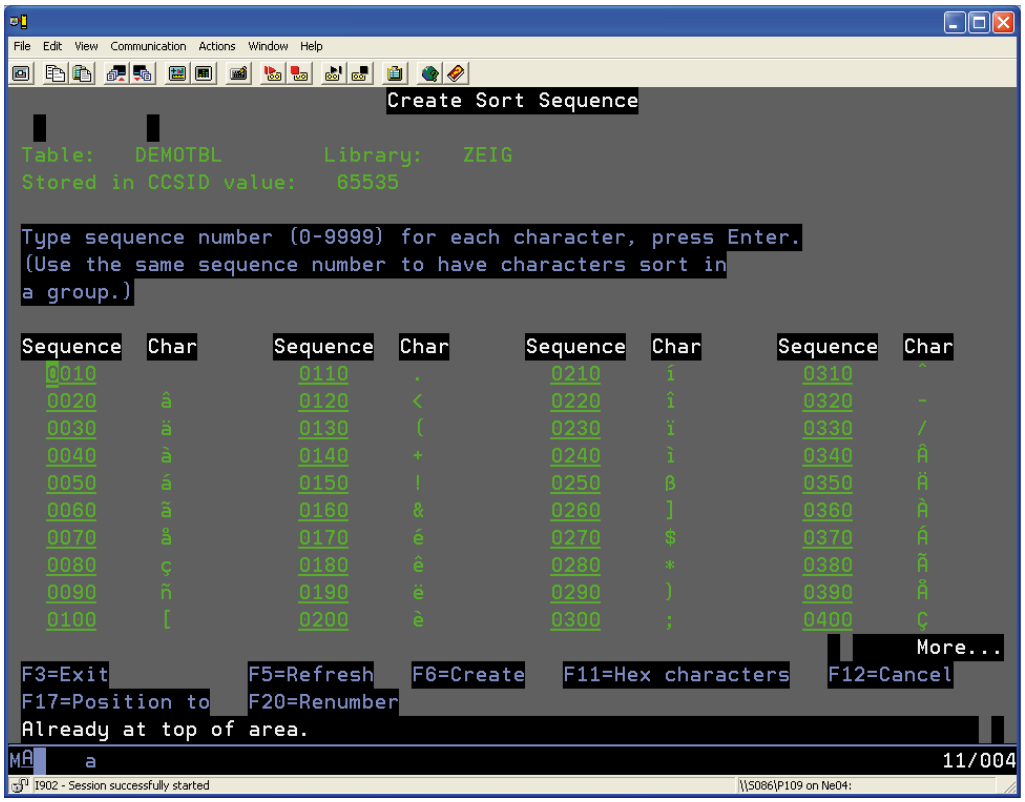


CRTTBL

Geben Sie der Tabelle einen möglichst sprechenden Namen. Auch wenn als Parameter eine Quelldatei angegeben werden kann, so füllen wir diesen Parameter mit dem Sonderwert *PROMPT. Der Teildateityp wird mit *TBL angegeben. Da wir in diesem Beispiel eine Sortiertabelle generieren wollen, verwenden wir als Wert für den Parameter TBLTYPE *SRTSEQ.

Wir werden später noch ein wenig genauer darauf eingehen – aber im Moment verwenden wir als Grundlage für die Sortiervorgaben die Einstellungen des Jobs. Deshalb geben wir in dem Parameter BASESRTSEQ den Wert *JOB ein bzw. übernehmen diesen.

Geben Sie diese Grundinformationen ein und bestätigen Sie die Eingabe. Daraufhin werden wir zu einer neuen Anzeige geleitet:

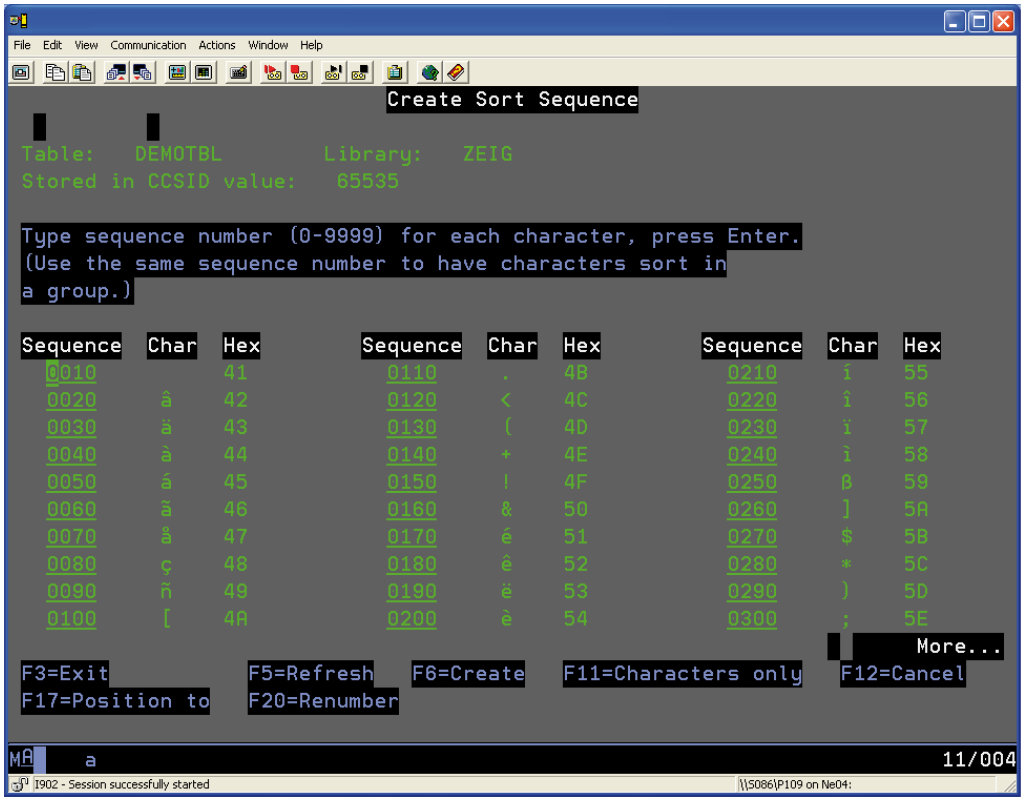


Prompt-Anzeige für CRTTBL

Auch wenn die vorhergehende Anzeige eigentlich selbsterklärend ist: Hier finden wir nichts weiter, als die Sortiervorgaben, nach denen auf dem System die verschiedenen Zeichen berücksichtigt werden. Schauen Sie sich den gesamten Inhalt einmal genauer an und legen Sie Ihr Augenmerk auf die Sortierreihenfolgen, die unter anderem auch genauestens zwischen Groß- und Kleinschreibung differenzieren. Zunächst kommen die Kleinbuchstaben und im weiteren Verlauf erst die Großbuchstaben. Warum ist das so? Die Erläuterung finden Sie hier:

2.10.2.3.3
Seite 4

Mit der Befehlstaste F11 können wir die Anzeige wechseln und uns zu den Zeichen die korrespondierenden Hexwerte anschauen. Diese sind exemplarisch in der folgenden Abbildung enthalten:



Hexwerte in CRTTBL

Diese Darstellung lässt nun in der Tat leicht erkennen, wie die Sortierung durchgeführt wird.

Wünschen Sie hier Anpassungen, dann genügt das Verändern des Bereichs Reihenfolge bzw. Sequence. Aber Vorsicht – diese Einstellungen wirken sich unter Umständen dann später auch auf Standardfunktionen des IBM i aus!

Mit diesem Vorgehen können wir neue Sortiertabellen auf Basis bereits bestehender Tabellen erstellen. Es kann aber auch durchaus Sinn machen, dass man beispielsweise für Matchcodesuchen keine Unterscheidung zwischen Groß- und Kleinbuchstaben macht. Denn welche Reihenfolge wäre bei den folgenden Beispieldaten im Sinne einer Sortierung wirklich korrekt:

- München
- münchen
- Münster
- MÜNCHEN
- MÜnchen
- …

Lassen wir einmal den Umlaut als Besonderheit unseres Zeichensatzes in dem Beispiel unbeachtet und konzentrieren uns lediglich auf die verschiedenen Versionen der Groß-/Kleinschreibung. Sie werden sicher zustimmen, dass für eine Matchcodesuche die Groß-/Kleinschreibung weniger eine Rolle spielt. In einem solchen Falle wäre sicher eine Sortierung rein auf die Buchstaben bezogen und ohne Groß-/Kleinschreibungsberücksichtigung ideal. Auch für ein solches Szenario können wir eine Sortiersequenztabelle erstellen.

Wie würde die Reihenfolge ausschauen, wenn wir die zuvor gezeigten Versionen für „München“ in einer Tabelle erfasst hätten und das Schlüsselfeld „MATCHCODE“ verwenden?

Mit UPDDTA können wir uns die Reihenfolge der Datensätze anzeigen lassen – das Ergebnis bei einer nicht angepassten Sortiervorgabe:

1. münchen

2. München

3. Münster

4. MÜnchen

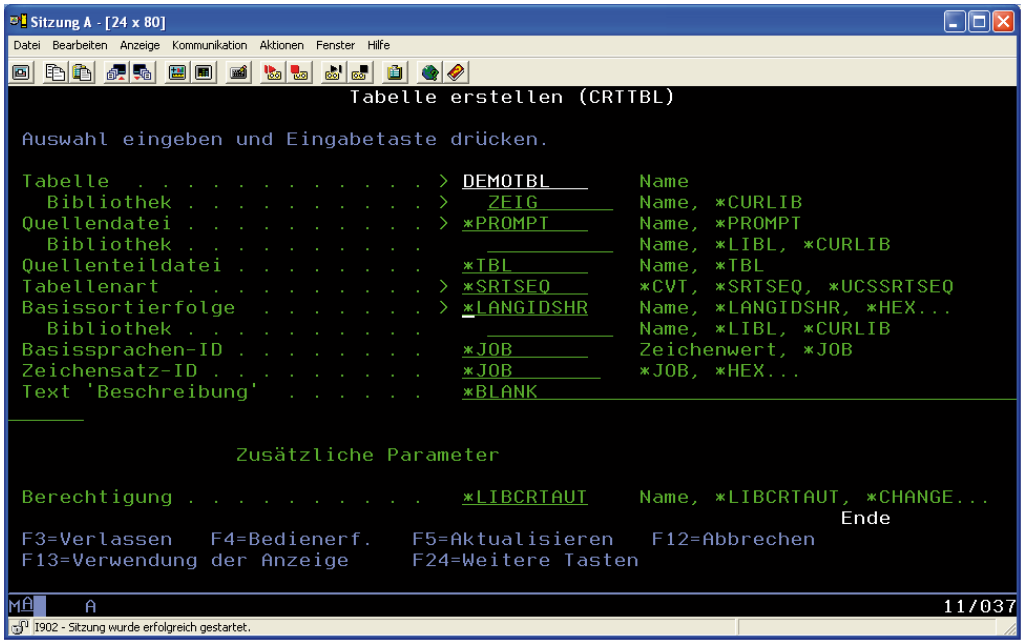
5. MÜNCHEN

Das wäre sicher nicht im Sinne des Erfinders – denn „Münster“ sollte eigentlich am Ende nach allen Versionen der „München“-Eingabe erscheinen.

Dieses können wir durch Anpassungen der Sortiervorgaben erreichen. Dazu stehen uns unterschiedliche Möglichkeiten zur Verfügung, von denen ich nun einige vorstellen werde.

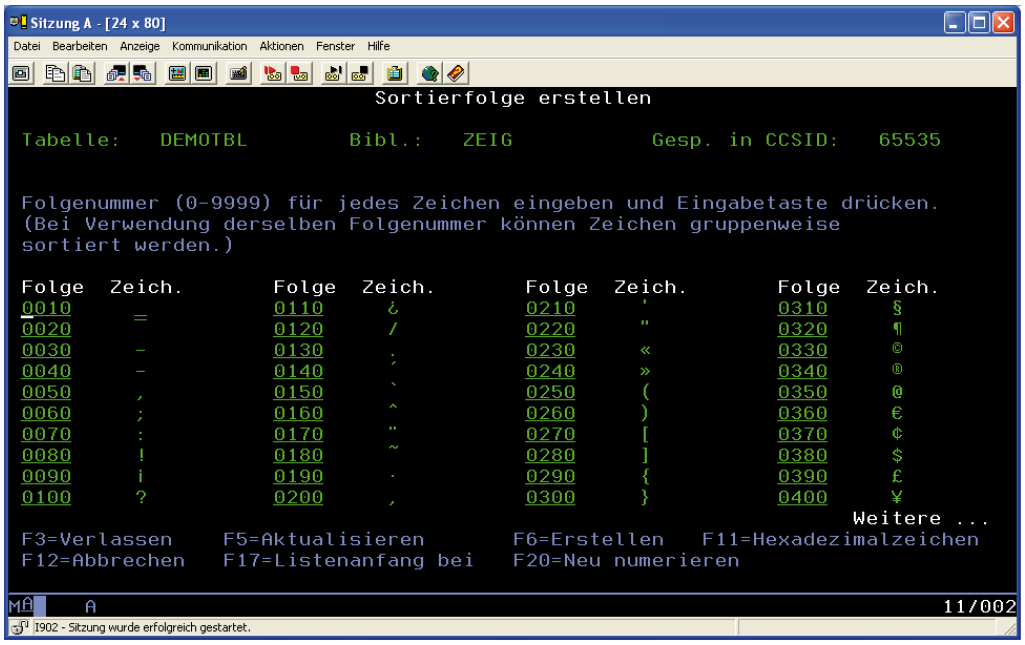
Soll eine vollkommen neue Sortiertabelle erzeugt werden, dann können wir dazu den Sonderwert *LANGIDSHR für den Parameter BASESRTSEQ verwenden. Mit dieser besonderen Einstellung lassen sich auch Sortierfolgen erstellen, die beispielsweise nicht zwischen Groß- und Kleinschreibung differenzieren.

2.10.2.3.3
Seite 6



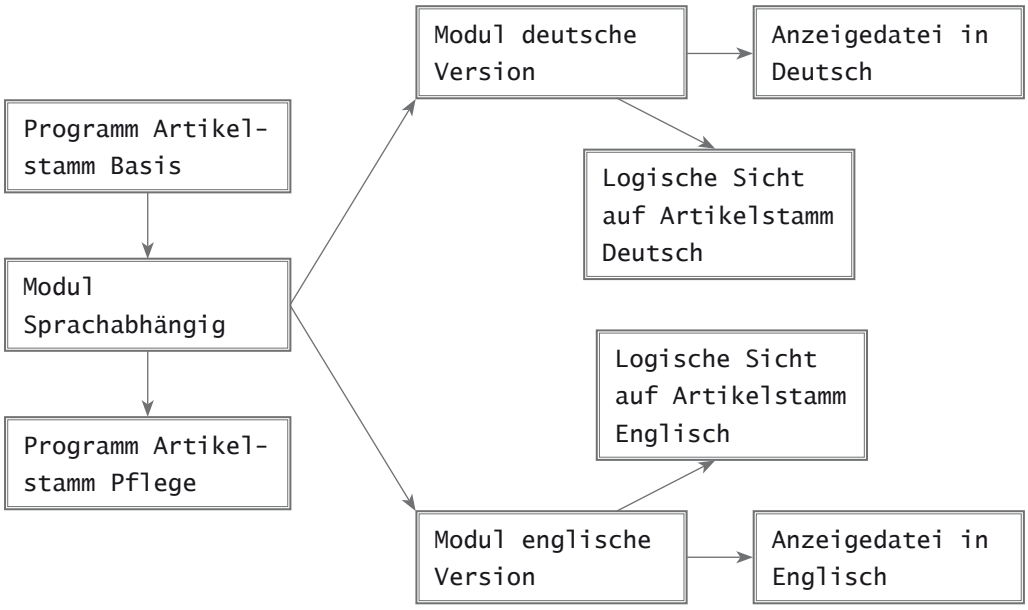
**LANGIDSHR*

Mit dem Sonderwert *LANGIDSHR werden die Sortierreihenfolgen quasi ohne Berücksichtigung der Groß-/Kleinbuchstaben durchgeführt. „Quasi“ bedeutet, dass diese trotzdem in den Sortiervorgaben enthalten sind. In der folgenden Abbildung sehen Sie die Sortierangaben unter anderem für den Buchstaben „M“, den wir in unserem Beispiel (München) verwenden:



*Sortiersequenzen mit Sonderwert *LANGIDSHR*

Basierend auf den Sortiersequenzen können wir beispielsweise für Sprachanforderungen eigene logische Sichten erstellen, mit denen die Sortiervorgaben und -anforderungen individuell abgebildet werden können. Am Beispiel einer multilingualen Anwendung könnte dies dann schematisch dargestellt so ausschauen:



Schematische Darstellung sprachspezifischer logischer Sichten

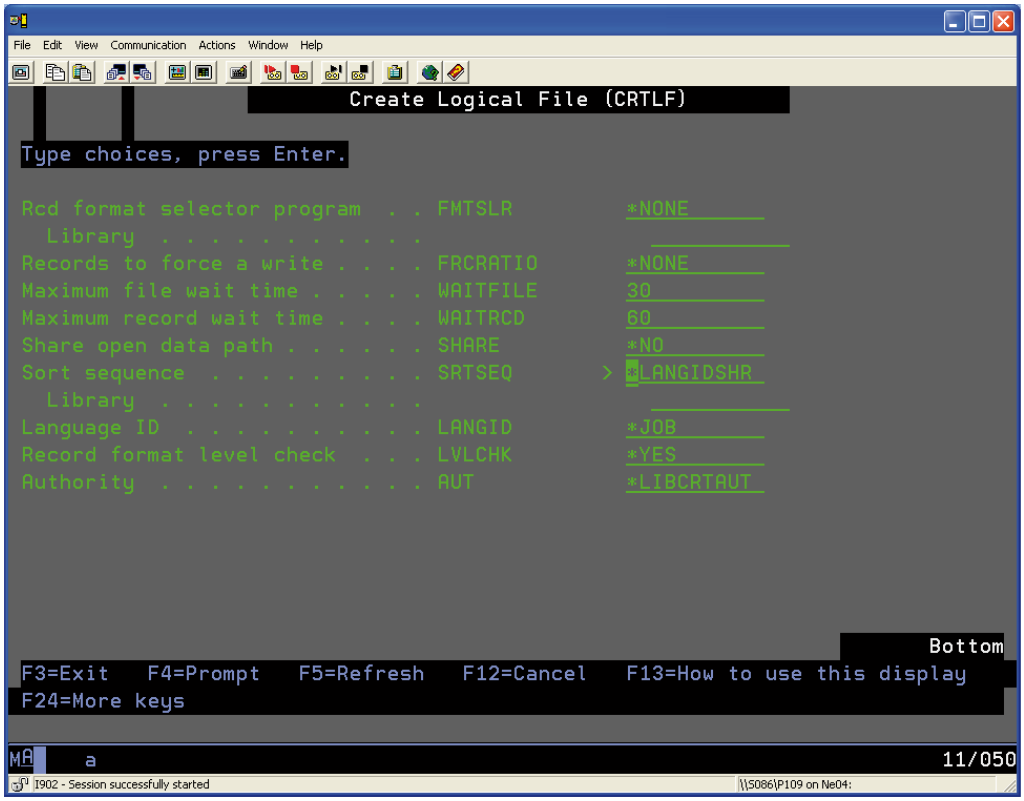
Beim Erstellen einer logischen Sicht können wir die selbst definierten (oder abweichenden) Sortiertabellen angeben. Mit dem Parameter SRTSEQ geben wir bei der Erstellung einer logischen Sicht an, wie die Sortierung zu erfolgen hat. Mögliche Werte sind hier:

- *SRC
  In diesem Fall kommt die Sortiertabelle, die in dem Schlüsselwort ALTSEQ angegeben wird, zum Einsatz.
- *JOB
  Die Sortiervorgaben werden über die Jobdefinitionen ermittelt
- *LANGIDSHR
  Hier wird die Tabelle verwendet, die mit dem im Parameter LANGID definierten Wert korrespondiert.

Mithilfe dieser Funktion können wir gezielt Sortierungen mit länder-/sprachspezifischen Anforderungen realisieren.

2.10.2.3.3
Seite 8

Hier ein Beispiel für den Befehl CRTLF und den Parameter SRTSEQ:



SRTSEQ für CRTLF

Zusammen mit den Sortiertabellen können wir nun in DDS Schlüsselworte verwenden, die Umsetzungen oder Sortierungen beeinflussen.

Wurde eine logische Sicht mit Sortiervorgaben erstellt, dann wirkt sich dies natürlich auch in den Anwendungsprogrammen aus, welche diese Sicht nutzen.

Die Auswirkungen beschränken sich dann aber ausschließlich auf die definierten Schlüsselfelder, für die die Sortiervorgaben in der logischen Sicht getätigt wurden. Folglich kommen diese bei den typischen Lesebefehlen zur Geltung:

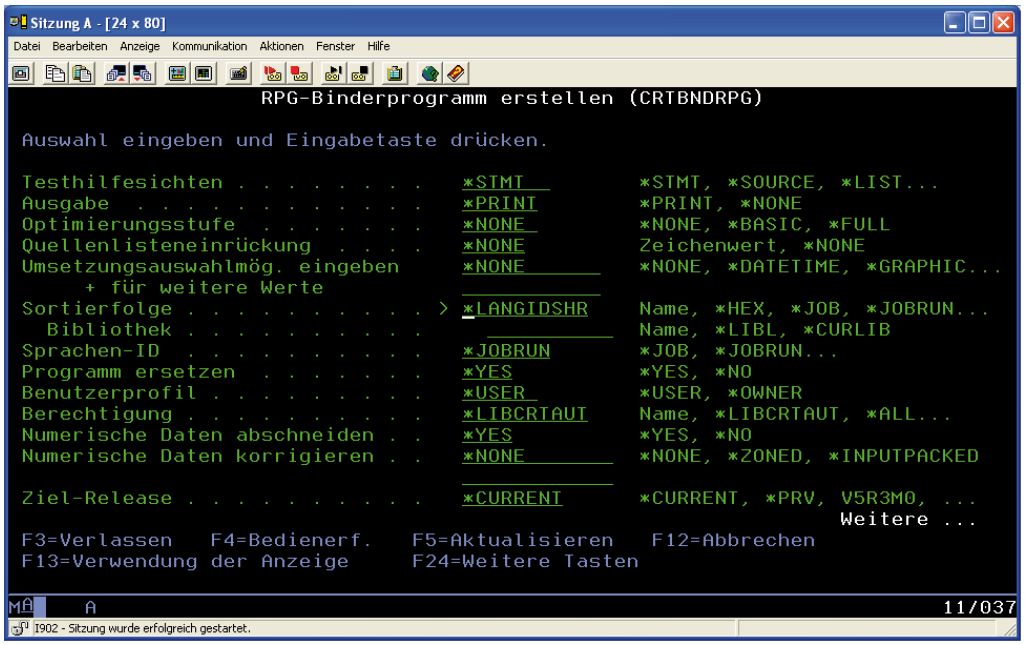
- CHAIN
- SETLL
- SETGT
- READE
- READPE

Andere Sortierungen werden davon nicht beeinflusst!

Anmerkung

Beeinflussungen der Sortierung von Schlüsselfeldern in logischen Sichten können auch auf Feldebene bei der Erstellung der Sichten definiert werden. TRNTBL und ALTSEQ sind Beispiele dafür. Diese hier zu beschreiben würde aber zu weit führen. Deshalb verweise ich an dieser Stelle auf Fachliteratur im Bereich der Anwendungsentwicklung.

Anders verhält es sich, wenn der Parameter SRTSEQ bei der Umwandlung eines RPG- Programms oder -Moduls genutzt wird. Die Verwendung des Schlüsselwortes SRTSEQ bei den Befehlen CRTBNDRPG oder CRTRPGMOD beeinflusst die Sortierung ALLER Felder in dem RPG-Programm bzw. dem RPG-Modul.



SRTSEQ für den Befehl CRTBNDRPG

Sortiersequenzen lassen sich selbstverständlich auch für SQL und embedded SQL verwenden. Hier gelten dieselben Regeln und Auswirkungen, wie sie zuvor für „reines“ RPG erläutert wurden.

Der Befehl CRTSQLRPGI verfügt ebenfalls über den Parameter SRTSEQ, dessen Auswirkungen dieselben sind wie bei dem Befehl CRTBNDRPG.

2.10.2.3.3
Seite 10

Individuelle Sortierungen (und auch solche, bei denen man nicht die lästigen Zusatzangaben bei der Kompilierung durchführen muss) können in einem SQL/RPG-Code auch über die Optionsangaben durchgeführt werden, wie das nachfolgende Beispiel zeigt:

```
/FREE
 EXEC SQL
    SET OPTION SRTSEQ = *LANGIDSHR;
/END-FREE
```

SET OPTION in Embedded SQL

LOCALES

Neben den Grundeinstellungen für das System über die Systemwerte bietet IBM eine Erweiterung für das Betriebssystem IBM i an, mit der wir internationale Einstellungen gezielt definieren und beispielsweise Benutzern zuordnen können.

Diese Funktionen sind über die so genannten Locales abgebildet. Locales finden wir in einer eigenen Bibliothek QSYSLOCALES. Mittels APIs oder ILE-Funktionen haben wir zudem Zugriff auf den Inhalt der Locales-Definitionen und können diese bei Bedarf auch in Anwendungen nutzen. Auf diese Weise können wir zum Beispiel das Erscheinungsbild eines Datumsfeldes beeinflussen, das für einen deutschen Kunden in TT.MM.JJ angezeigt wird, während ein amerikanischer Kollege den Wert im Format MM/TT/JJ sieht.

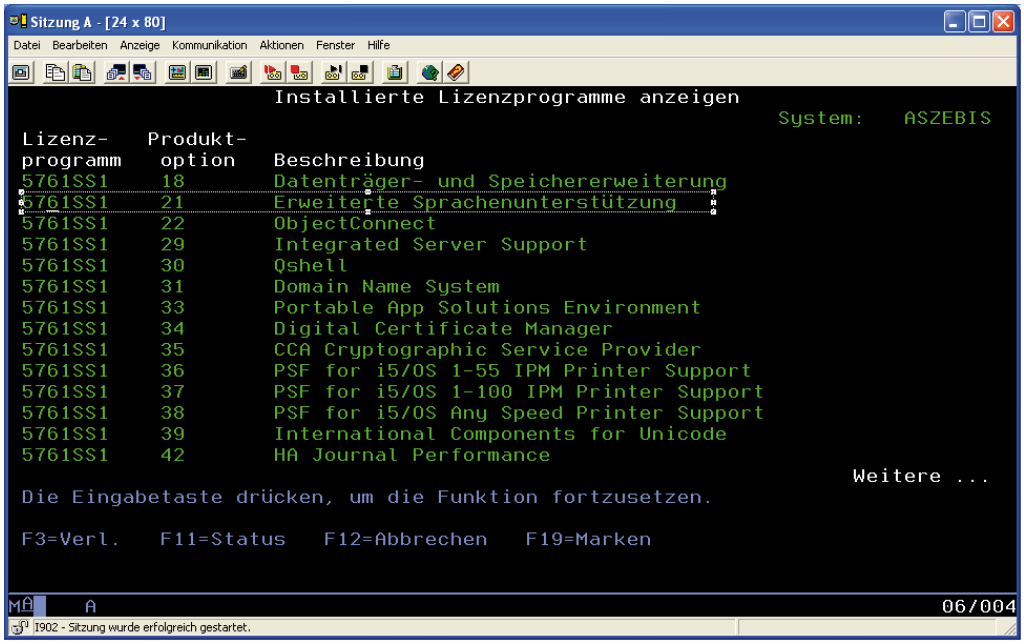
Zeitzonen sind ein beliebtes Beispiel, für das Locales zum Einsatz kommen – aber auch von „Standards" abweichende Definitionen für Zeitformate, Datumsformate oder Aufbereitungsschlüssel.

QSYSLOCALES ist eine Bibliothek, die eigentlich auf keinem System fehlen darf, das für einen internationalen Einsatz genutzt wird. Individuelle Einstellungen für internationalen Einsatz lassen sich durch die folgenden Definitionsbereiche bestimmen:

- Zu nutzender Zeichensatz
- Gegebenenfalls notwendige Konvertierung von Zeichen
- Datumsformate
- Zeitformate
- Zeitzonen Einstellungen
- Aufbereitungsschlüssel für numerische Werte

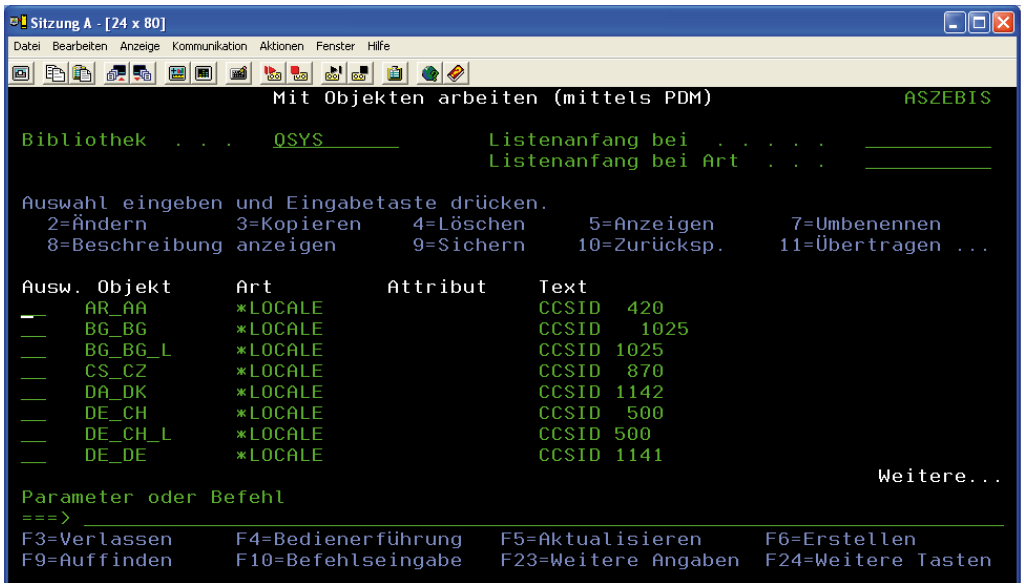
Die Verwendung der QSYSLOCALES ist nur nach der Installation des optionalen Lizenzprogramms 57xx-SS1 Option 21 (National Language Support bzw. Extended NLS Support bzw. „Erweiterte Sprachenunterstützung") möglich. Diese muss auf dem System installiert sein. Prüfen Sie dies mit der Auswahl 10 im Menü GO LICPGM. Dort sollten wir dann einen Eintrag finden, wie er auch in der folgenden Abbildung zu sehen ist:

2.10.3
Seite 2



Erweiterte Sprachenunterstützung

Auf dem IBM i kennen wir im Zusammenhang mit der Internationalisierung einen bestimmten Objekttyp „*LOCALE“. Die folgende Abbildung zeigt einen Ausschnitt der installierten LOCALE-Objekte auf einem IBM i.



*Objekte vom Typ *LOCALE*

Wenn das Lizenzprogramm „Erweiterte Sprachenunterstützung / National Language Support“ installiert ist, dann finden wir auf dem System auch eine Bibliothek mit Namen QSYSLOCALE. In dieser finden wir eine Reihe von Objekten – unter anderem auch Quelldateien, die wir als Grundlage für Anpassungen nutzen.

Auf Basis der Quellcodebeispiele können wir uns individuelle Einstellungen definieren, die dann später beispielsweise auf der Ebene eines Benutzerprofils genutzt werden können, um den Benutzer mit sprachenspezifischen Einstellungen auszustatten.

Die folgende Abbildung zeigt exemplarisch eine solche Quellcodedatei aus der Bibliothek QSYSLOCALE:

DE_DE_E.MBR

```
 Zeile 1       Spalte 1     Einfügen
      ----+----1----+----2----+----3----+----4----+----5----+----6----
000100 comment_char <percent-sign>
000200 escape_char  <slash>
000300
000400 %
000500 % 5716SS1          (C) COPYRIGHT IBM CORP. 1991,1998
000600 % ALL RIGHTS RESERVED.
000700 % US GOVERNMENT USERS RESTRICTED RIGHTS -
000800 % USE, DUPLICATION OR DISCLOSURE RESTRICTED
000900 % BY GSA ADP SCHEDULE CONTRACT WITH IBM CORP.
001000 %
001100 % LICENSED MATERIALS-PROPERTY OF IBM
001200 %
001300 % FILE NAME:       DE_DE
001400 %
001500 % COUNTRY/REGION: GERMANY
001600 %
001700 % LANGUAGE(S):     GERMAN
001800 %
001900 % DESCRIPTION:     LOCALE SOURCE DEFINITION FILE.
002000 %                  WITH EURO SUPPORT, BETA VERSION JULY 1998
002100 %
002200
002300 LC_CTYPE
002400
002500 upper     <A>;<B>;<C>;<D>;<E>;<F>;<G>;<H>;<I>;<J>;<K>;<L>;<M>;/
002600          <N>;<O>;<P>;<Q>;<R>;<S>;<T>;<U>;<V>;<W>;<X>;<Y>;<Z>;/
002700          <A-acute>;<A-grave>;<A-circumflex>;<A-diaresis>;/
002800          <A-tilde>;<A-ring>;<AE>;<C-cedilla>;<Eth>;<E-acute>;/
002900          <E-grave>;<E-circumflex>;<E-diaresis>;<I-acute>;/
003000          <I-grave>;<I-circumflex>;<I-diaresis>;<N-tilde>;/
003100          <O-acute>;<O-grave>;<O-circumflex>;<O-diaresis>;/
003200          <O-tilde>;<O-slash>;<Thorn>;<U-acute>;<U-grave>;/
003300          <U-circumflex>;<U-diaresis>;<Y-acute>
003400
003500 lower     <a>;<b>;<c>;<d>;<e>;<f>;<g>;<h>;<i>;<j>;<k>;<l>;<m>;/
003600          <n>;<o>;<p>;<q>;<r>;<s>;<t>;<u>;<v>;<w>;<x>;<y>;<z>;/
003700          <a-acute>;<a-grave>;<a-circumflex>;<a-diaresis>;/
003800          <a-tilde>;<a-ring>;<ae>;<c-cedilla>;<eth>;<e-acute>;/
003900          <e-grave>;<e-circumflex>;<e-diaresis>;<i-acute>;/
004000          <i-grave>;<i-circumflex>;<i-diaresis>;<n-tilde>;/
004100          <o-acute>;<o-grave>;<o-circumflex>;<o-diaresis>;/
004200          <o-tilde>;<o-slash>;<s-sharp>;<thorn>;<u-acute>;/
```

Auszug aus QSYSLOCALE Quellendatei

Hinweis

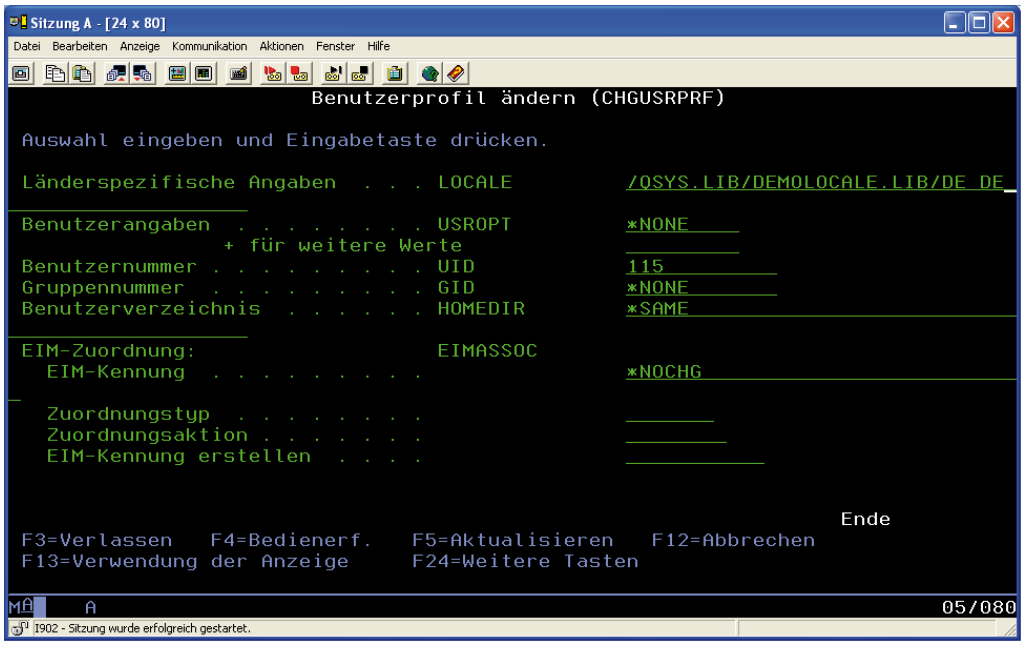
Wundern Sie sich nicht, dass Sie keine Berechtigung für das Editieren der Quellenteildateien haben. Das ist bewusst so gehalten, damit darin nicht versehentlich Änderungen durchgeführt werden. Grundsätzlich soll damit sichergestellt werden, dass die Originalversion der Quellen erhalten bleibt. Wollen Sie Änderungen an den Definitionen vornehmen, dann kopieren Sie die Quelle in einen separaten Bereich. Danach können Sie deren Inhalt dann problemlos anpassen. Wir werden uns damit auch im weiteren Verlauf noch beschäftigen.

Anmerkung

Die letzte wirklich große Änderung an den LOCALE-Quellen erfolgte mit der Einführung des Euro – denn die neue Währung musste in den allgemeinen LOCALE-Definitionen abgebildet werden.

Die Quelldatei QLOCALESRC in der QSYSLOCALE wird grundsätzlich mit der CCSID 37 installiert und verfügt ausschließlich über Lesebe-rechtigungen. Änderungen dürfen an den Inhalten auf keinen Fall gemacht werden. Wenn Anpassungen notwendig sind, dann können wir diese über das Kopieren der Ursprungsquellen realisieren – aber dazu später mehr.

Beispielsweise dient der Parameter *LOCALE in dem Benutzerprofil genau der Angabe der Locale-Definitionen:



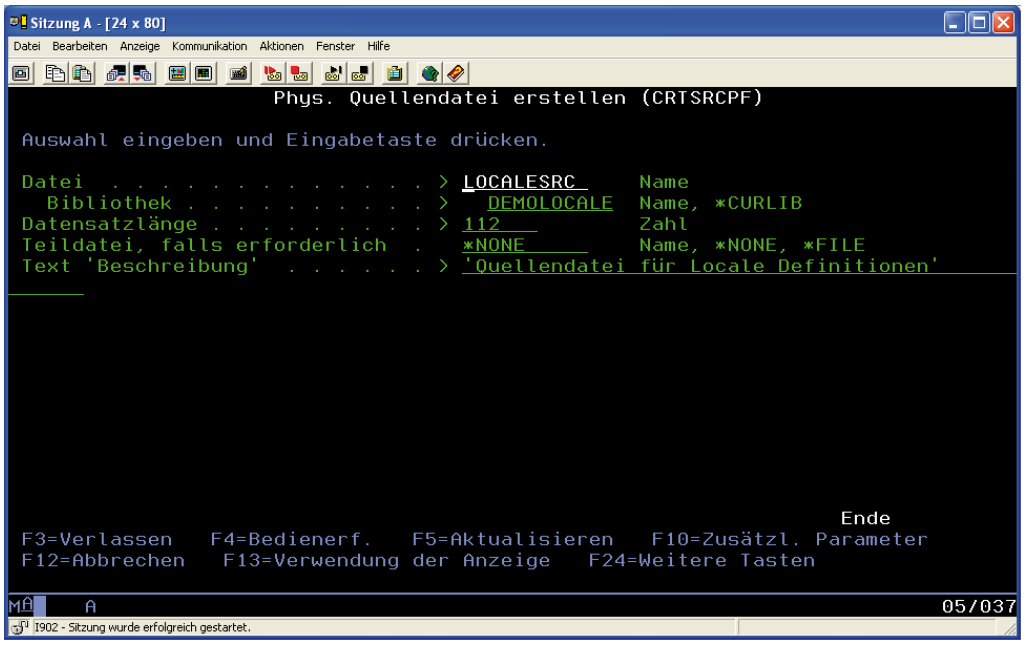
CHGUSRPRF

Die auf diese Weise einem Benutzerprofil zugeordneten Einstellungen gelten sowohl für interaktive Jobs als auch für Batchjobs.

Schauen wir uns nun einmal an, wie wir eigene sprach- bzw. landesspezifische Einstellungen mit Hilfe der Locales definieren können.

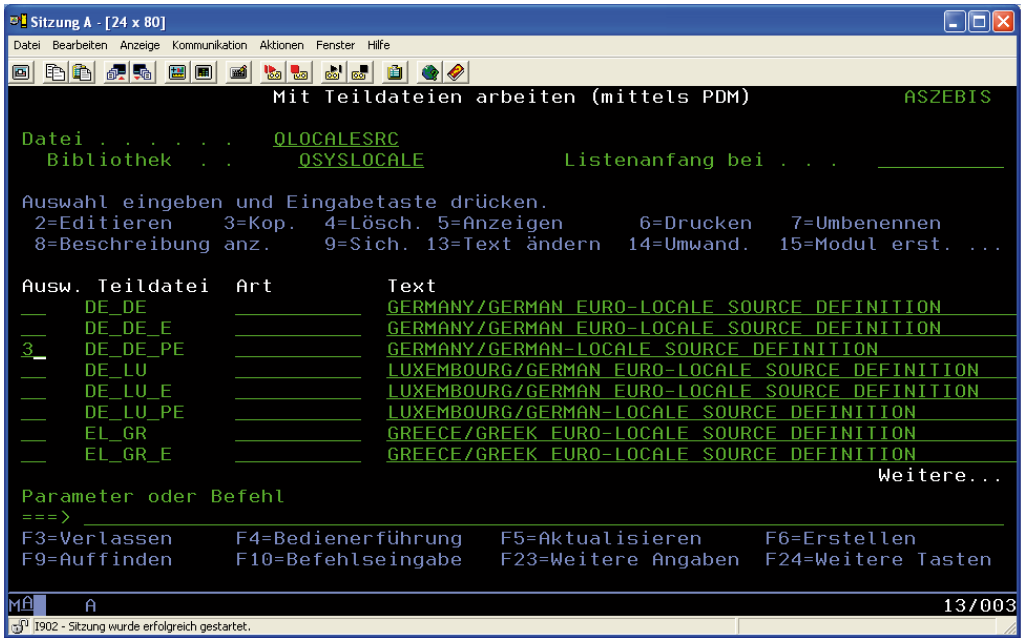
- Stellen Sie sicher, dass die Option 21 des Betriebssystems (57xx-SS1) installiert ist.
  - Sollte diese Option nicht installiert sein, dann können Sie diese jederzeit mit dem Befehl RSTLICPGM installieren. Ein IPL oder ein eingeschränkter Zustand des Systems sind dazu nicht erforderlich.
- Erstellen Sie eine Bibliothek, welche für die individuellen Locale-Einstellungen genutzt wird. In unserem Beispiel erstelle ich die Bibliothek DEMOLOCALE mit dem Befehl CRTLIB DEMOLOCALE.
- In der Bibliothek DEMOLOCALE erstellen wir nun eine neue Quelldatei, welche die Sourcecodes für die einzelnen Locale-Definitionen aufnehmen. Diese Quelldatei nenne ich in dem Beispiel LOCALESRC. Das Erstellen der Quelldatei können Sie mit dem folgenden Befehl durchführen:

  CRTSRCPF DEMOLOCALE/LOCALESRC RCDLEN(112) Text ('Quelldatei für Locale')



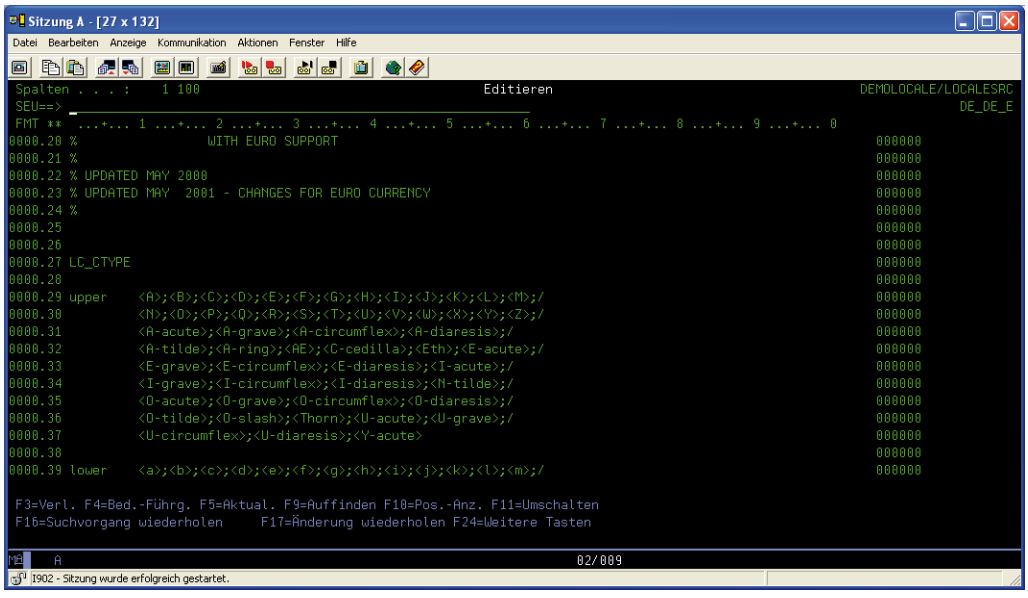
CRTSRCPF

- Nun kopieren wir uns eine der Beispielquelldateien aus der Bibliothek QSYSLOCALE in unsere neue Quelldatei, um diese anzupassen. In unserem Beispiel verwende ich die Quellteildatei DE_DE_E, die sich in der Quelldatei QLOCALESRC befindet. (Achtung! Der Inhalt der QLOCALESRC ist schreibgeschützt und darf nicht verändert werden! Die Nutzung als Kopiergrundlage ist jedoch problemlos möglich!)



Locale-Inhalt kopieren

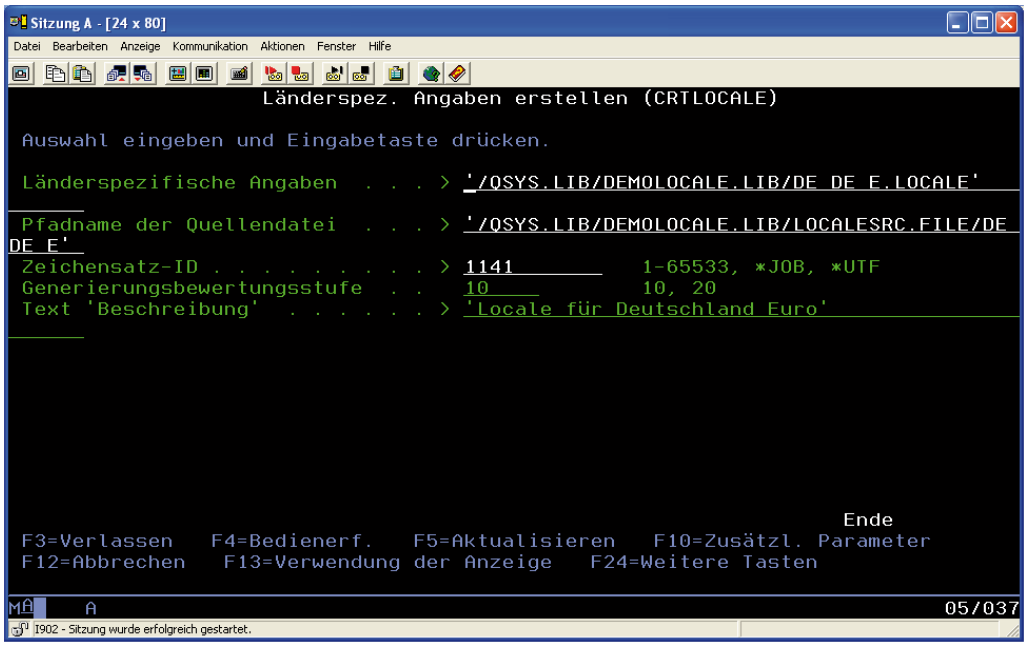
Nachfolgend ein Ausschnitt aus der Quellteildatei DE_DE_E



Quellenteildatei-Inhalte

- Jetzt erstellen wir ein Locale-Objekt basierend auf der neuen Kopie der Quelldatei. Schauen wir uns die Quellteildatei an, dann stellen wir fest, dass diese über keinen Typ verfügt. Folglich können wir diese auch nicht kompilieren. Für das Erstellen eines Locale-Objektes verwenden wir deshalb den Befehl CRTLOCALE. Bei der Erstellung des Locale-Objektes muss die CCSID zwingend mitgegeben werden. Diese muss mit der CCSID übereinstimmen, die in der Quellteildatei zum Einsatz kommt. Für unser Beispiel ist dies die CCSID 1141.

  CRTLOCALE LOCALE('/QSYS.LIB/DEMOLOCALE.LIB/DE_DE_E.LOCALE')
  SRCFILE('/QSYS.LIB/DEMOLOCALE.LIB/LOCALESRC.FILE/DE_DE_E.MBR')
  CCSID(1141) TEXT('Locale für Deutsch_Deutschland_Euro')

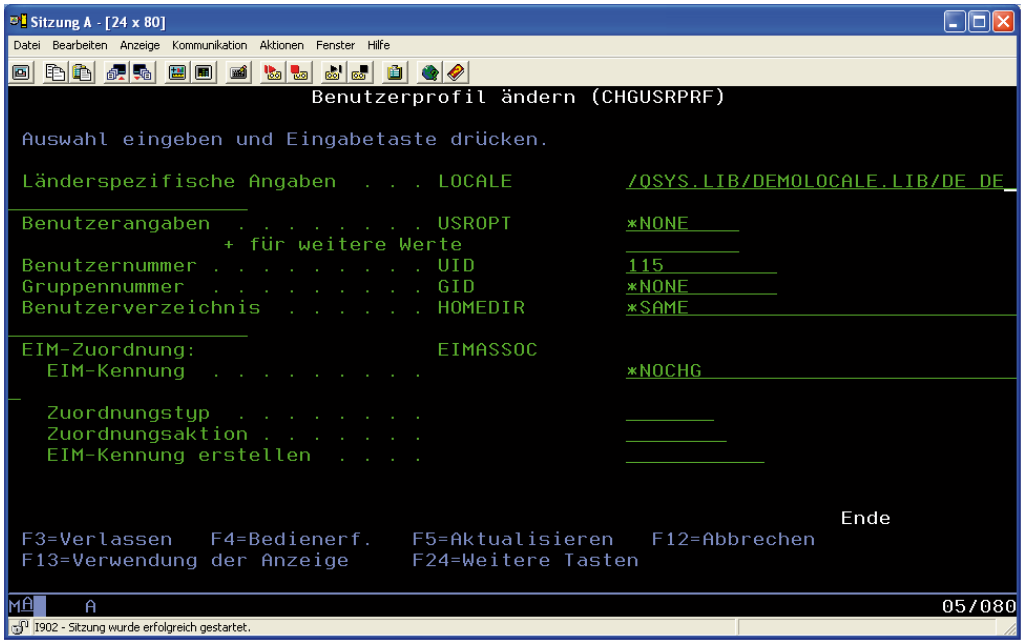


CRTLOCALE

- Nach dem Erstellen der Locale-Definition ordnen wir diese nun einem Benutzer zu. Dazu nehmen wir eine Einstellung in dem Parameter LOCALE des Benutzerprofils vor und geben den Pfad unserer neu erstellen Locale-Definition an:

  CHGUSRPRF Benutzer LOCALE ('/QSYS.LIB/DEMOLOCALE.LIB/DE_DE_E')

2.10.3
Seite 8



CHGUSRPRF

- Weiterhin passen wir in dem Benutzerprofil die Werte an, die aus dem Locale-Objekt verwendet werden sollen. Dazu geben wir die gewünschten Werte in dem Parameter SETJOBATR an:
  - *CCSID
    Die zu verwendende CCSID
  - *DATFMT
    Das gewünschte Datumsformat
  - *DATSEP
    Die gewünschten Datumstrennzeichen
  - *DECFMT
    Die Dezimaltrennzeichen
  - *SRTSEQ
    Die Sortiervorgaben
  - *TIMSEP
    Die Trennzeichen für Zeitwerte

Damit haben wir die Einrichtung einer eigenen Locale-Definition fast abgeschlossen. Wir haben auf Basis einer von IBM zur Verfügung gestellten Definition eine eigene Locale-Beschreibung erstellt. Nur unterscheidet diese sich nicht von der Kopie. Natürlich macht das einfache Kopieren einer vorhandenen Locale-Definition keinen Sinn, denn dann hätten wir auch die Originalversion einsetzen können. Wir können im Gegensatz zu den von IBM bereitgestellten Quellcodes unseren eigens erstellen Quellcode sehr wohl editieren und damit auf unsere spezifischen Anforderungen hin anpassen.

Bei den Anpassungen der Locale-Einstellungen ist natürlich darauf zu achten, dass die Einstellungen so durchgeführt werden, dass die beabsichtigten Ergebnisse auch tatsächlich erreicht werden.

Grundsätzlich darf in einer Locale-Quellendatei nur jeweils ein Eintrag für eine Locale-Kategorie enthalten sein. Das bedeutet, dass wir zum Beispiel keine zwei Bereiche mit LC_CTYPE-Angaben in einer Quelle verwenden dürfen.

Für die Definition von Kategorien in einer Locale-Quelldatei gelten die folgenden Regeln:

- Die Kategorie muss über einen Kategorieheader verfügen.
- Der Kategorieheader muss in Großbuchstaben geschrieben werden.
- Der Kategorieheader muss ein für Locale gültiger Header sein:
  - LC_COLLATE
  - LC_CTYPE
  - LC_MESSAGES
  - LC_MONETARY
  - LC_NUMERIC
  - LC_TIME
  - LC_TOD
- Die in einer Kategorie verwendeten Angaben müssen gültig sein.
  - Schlüsselwörter und Werte müssen passen und definiert sein!
  - Schlüsselwörter können beliebig in Groß-/Kleinbuchstaben angegeben werden.
- Den Abschluss der Kategoriedefinitionen bildet immer eine Ende-Angabe:
  - END + Kategoriename
- Zeichen müssen in eckige Klammern eingefügt werden (<>).
  - Z. B. <E> – dies ist ein symbolischer Name.
    Symbolische Namen sind in der Quellteildatei QLGSYMBOL in QSYSLOCALE/QLOCALESRC definiert.
- Zeichenfolgen müssen in Hochkomma eingeschlossen werden.
  - Z. B. "<E><U><R>"

2.10.3
Seite 10

Schauen Sie sich hierzu auch die folgende Abbildung an, welche die Definition für den Bereich LC_MONETARY beinhaltet:

```
LC_MONETARY

int_curr_symbol    "<E><U><R><space>"
currency_symbol    "<euro>"
mon_decimal_point  "<comma>"
mon_thousands_sep  "<period>"
mon_grouping       3
positive_sign      ""
negative_sign      "<hyphen-minus>"
int_frac_digits    2
frac_digits        2
p_cs_precedes      1
p_sep_by_space     0
n_cs_precedes      1
n_sep_by_space     0
p_sign_posn        1
n_sign_posn        1

END LC_MONETARY
```

LC_MONETARY

Hier noch einige weitere Regeln für das Ändern von Locale-Definitionen:

- Für eine SBCS-Umgebung muss die verwendete CCSID eine EBCDIC Single Byte CCSID sein.
- Für eine DBCS-Umgebung kann die verwendete CCSID eine EBCDIC Single Byte CCSID oder eine mixed-Byte CCSID für DBCS sein.
- Die angegebenen Parameter für DATFMT, DATSEP, DECFMT und TIMSEP müssen gültig sein.
- Sortieranweisungen müssen mit dem Schlüsselwort CPYSYSCOL definiert werden.

Neben diesen allgemeinen Regeln können wir die folgenden Bereiche in den Locale-Definitionen abbilden:

- LC_COLLATE
  - Definition von Zeichen oder Informationen zu Sortierfolgen.
    Ein entsprechender Auszug aus DE_DE_E:

```
000199
000200 LC_COLLATE
000201
000202 order_start
000203
000204 <NUL>
000205 <SOH>
000206 <STX>
000207 <ETX>
000208 <SEL>
000209 <tab>
000210 <RNL>
000211 <DEL>
000212 <GE>
000213 <SPS>
000214 <RPT>
000215 <vertical-tab>
000216 <form-feed>
000217 <carriage-return>
000218 <SO>
000219 <SI>
000220 <DLE>
000221 <DC1>
000222 <DC2>
000223 <DC3>
000224 <RES>
000225 <NL>
000226 <backspace>
000227 <POC>
000228 <CAN>
000229 <EM>
000230 <UBS>
```

LC_COLLATE

2.10.3
Seite 12

- LC-CTYPE
  - Definition von Zeichen-Klassifizierungen und Umsetzung in Groß-/Kleinbuchstaben.

Als Beispiel:

```
000026
000027 LC_CTYPE
000028
000029 upper     <A>;<B>;<C>;<D>;<E>;<F>;<G>;<H>;<I>;<J>;<K>;<L>;<M>;/
000030           <N>;<O>;<P>;<Q>;<R>;<S>;<T>;<U>;<V>;<W>;<X>;<Y>;<Z>;/
000031           <A-acute>;<A-grave>;<A-circumflex>;<A-diaresis>;/
000032           <A-tilde>;<A-ring>;<AE>;<C-cedilla>;<Eth>;<E-acute>;/
000033           <E-grave>;<E-circumflex>;<E-diaresis>;<I-acute>;/
000034           <I-grave>;<I-circumflex>;<I-diaresis>;<N-tilde>;/
000035           <O-acute>;<O-grave>;<O-circumflex>;<O-diaresis>;/
000036           <O-tilde>;<O-slash>;<Thorn>;<U-acute>;<U-grave>;/
000037           <U-circumflex>;<U-diaresis>;<Y-acute>
000038
000039 lower     <a>;<b>;<c>;<d>;<e>;<f>;<g>;<h>;<i>;<j>;<k>;<l>;<m>;/
000040           <n>;<o>;<p>;<q>;<r>;<s>;<t>;<u>;<v>;<w>;<x>;<y>;<z>;/
000041           <a-acute>;<a-grave>;<a-circumflex>;<a-diaresis>;/
000042           <a-tilde>;<a-ring>;<ae>;<c-cedilla>;<eth>;<e-acute>;/
000043           <e-grave>;<e-circumflex>;<e-diaresis>;<i-acute>;/
000044           <i-grave>;<i-circumflex>;<i-diaresis>;<n-tilde>;/
000045           <o-acute>;<o-grave>;<o-circumflex>;<o-diaresis>;/
000046           <o-tilde>;<o-slash>;<s-sharp>;<thorn>;<u-acute>;/
000047           <u-grave>;<u-circumflex>;<u-diaresis>;<y-acute>;/
000048           <y-diaresis>;<feminine>;<masculine>;<mu>
000049
000050 space     <tab>;<newline>;<NL>;<vertical-tab>;<form-feed>;<carriage-return>;/
000051           <space>
```

LC_TYPE

In diesem Bereich sind unter anderem die folgenden Schlüsselwörter gültig:

- Upper – definiert die Großbuchstaben (A-Z)
- Lower – definiert die Kleinbuchstaben (a-z)
- Toupper – definiert die Zuordnung zwischen Groß- und Kleinbuchstaben.

- LC_MESSAGES
  - Definition der Formate für positive und negative Rückmeldungen: Hier ein Auszug aus Locale-Definition für DE_DE_E:

```
000550
000551 LC_MESSAGES
000552
000553 yesexpr "([jJ]|[jJ][aA])"
000554 noexpr  "([nN]|[nN][eE][iI][nN])"
000555 yesstr  "ja:j:J"
000556 nostr   "nein:n:N"
000557
000558 END LC_MESSAGES
```

LC_MESSAGES

In diesem Bereich sind unter anderem die folgenden Schlüsselwörter gültig:

- Yesexpr definiert eine Bestätigung „jJ“
- Noexpr definiert eine negative Prüfung „nN“
- Yesstr Eine vorgegebene Zeichenfolge möglicher Bestätigungen „Ja“
- Nostr Eine vorgegebene Zeichenfolge für Bestätigungen „nein“

- LC_MONETARY
  - Definition der Regeln und Formatierungen für währungsbezogene Daten.
    Hier ein Auszug aus Locale-Definition für DE_DE_E:

```
000466 LC_MONETARY
000467
000468 int_curr_symbol   "<E><U><R><space>"
000469 currency_symbol   "<euro>"
000470 mon_decimal_point "<comma>"
000471 mon_thousands_sep "<period>"
000472 mon_grouping      3
000473 positive_sign     ""
000474 negative_sign     "<hyphen-minus>"
000475 int_frac_digits   2
000476 frac_digits       2
000477 p_cs_precedes     1
000478 p_sep_by_space    0
000479 n_cs_precedes     1
000480 n_sep_by_space    0
000481 p_sign_posn       1
000482 n_sign_posn       1
000483
000484 END LC_MONETARY
```

LC_MONETARY

In diesem Bereich sind unter anderem die folgenden Schlüsselwörter gültig:

- Int_curr_symbol Währungsabkürzung
- Currency-symbol Währungssymbol
- Mon_decimal_point Trennzeichen

2.10.3
Seite 14

- LC_NUMERIC
  - Definition der Regeln und Formatierungen für nicht währungsbezogene numerische Daten. Hier ein Auszug aus Locale-Definition für DE_DE_E:

```
000487 LC_NUMERIC
000488
000489 decimal_point      "<comma>"
000490 thousands_sep      "<period>"
000491 grouping           3
000492
000493 END LC_NUMERIC
```

LC_NUMERIC

In diesem Bereich sind unter anderem die folgenden Schlüsselwörter gültig:

- Decimal_point Trennzeichen für Dezimalen
- Thousands_sep Trennzeichen für 1000er Werte

LC_TIME

- Definition der Regeln und Formatierungen für Zeit- und Datumswerte. Hier ein Auszug aus Locale-Definition für DE_DE_E:

```
000496 LC_TIME
000497
000498 abday    "<S><o>";/
000499           "<M><o>";/
000500           "<D><i>";/
000501           "<M><i>";/
000502           "<D><o><n>";/
000503           "<F><r>";/
000504           "<S><a>"
000505
000506 day      "<S><o><n><n><t><a><g>";/
000507           "<M><o><n><t><a><g>";/
000508           "<D><i><e><n><s><t><a><g>";/
000509           "<M><i><t><t><w><o><c><h>";/
000510           "<D><o><n><n><e><r><s><t><a><g>";/
000511           "<F><r><e><i><t><a><g>";/
000512           "<S><a><m><s><t><a><g>"
000513
000514 abmon    "<J><a><n>";/
000515           "<F><e><b>";/
000516           "<M><a-diaresis><r>";/
000517           "<A><p><r>";/
000518           "<M><a><i>";/
000519           "<J><u><n>";/
000520           "<J><u><l>";/
000521           "<A><u><g>";/
000522           "<S><e><p>";/
000523           "<O><k><t>";/
000524           "<N><o><v>";/
000525           "<D><e><z>"
```

LC_TIME

In diesem Bereich sind unter anderem die folgenden Schlüsselwörter gültig:

- Abday — Abkürzungen für die Wochentage
- Day — Vollständiger Name des Wochentags
- Abmon — Abkürzung für den Monat
- Mon — Vollständiger Name des Monats

2.10.3
Seite 16

- LC_TOD
  - Definition der Regeln für Zeitumstellungen und Zeitzonen. Hier ein Auszug aus Locale-Definition für DE_DE_E:

```
000560 LC_TOD
000561
000562 tzdiff 0
000563 tname ""
000564 dstname ""
000565 dststart 0,0,0,0
000566 dstend   0,0,0,0
000567 dstshift 0
000568
000569 END LC_TOD
```

LC_TOD

In diesem Bereich sind unter anderem die folgenden Schlüsselwörter gültig:

- Tzdiff — Integer Wert für Zeitzonendifferenz (Differenz zwischen Zeitzone und Greenwich)
- Tname — Name der Zeitzone
- Dststart — Startdatum und Startzeit für die Zeitumstellung

Änderungen an den Locale-Quellcodes können im Rahmen der Vorgaben und gültigen Einträge vorgenommen werden.

Beim Editieren sind Leerzeilen zulässig. Kommentare sind auch hier natürlich sinnvoll und beginnen in jeder Kommentarzeile mit dem Prozentzeichen (%).

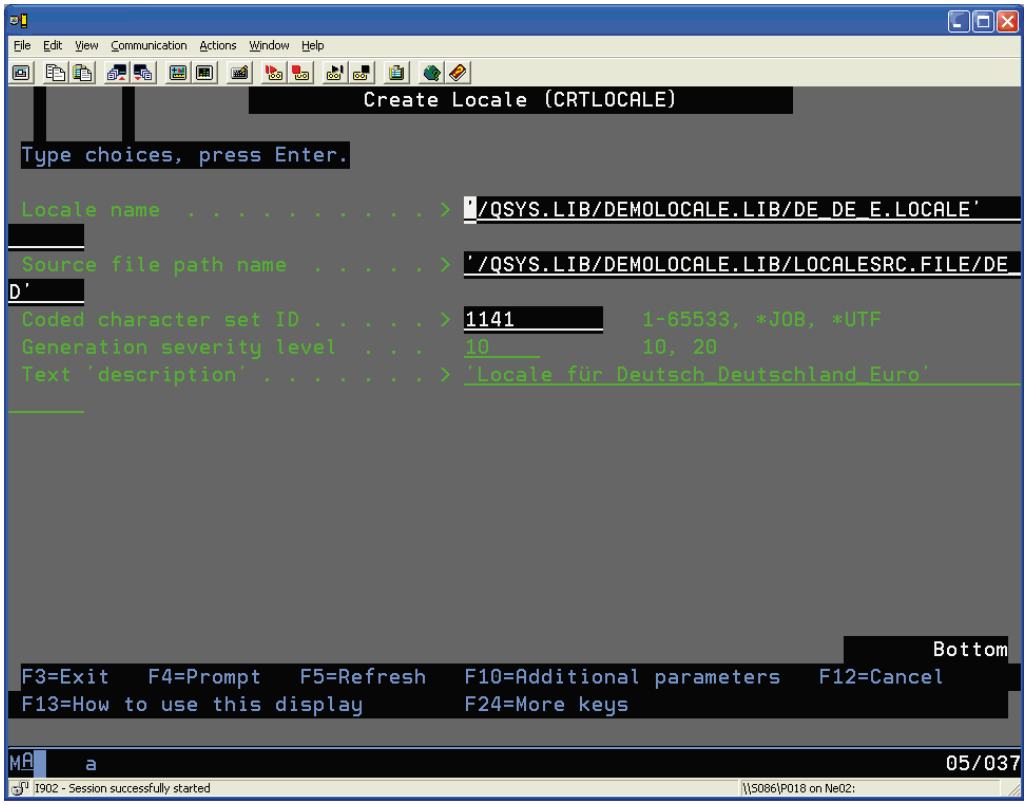
Nehmen wir als Beispiel einer Änderung die Anpassung der Zeitzonen und der Einrichtung der Umstellung auf Sommerzeit. Dazu passen wir die Eintragungen im Abschnitt „LC_TOD“ in unserer Quellenteildatei DE_DE_E in LOCALESRC an und ersetzen den entsprechenden LC_TOD-Teil mit den folgenden Codezeilen:

```
LC_TOD
% Zeitdifferenz zur UTC angegeben in Minuten
tzdiff 60

% Zeitzone
tname "<C><E><T>"

% Name der Zeitzone
dstname "<C><E><T>"

% Beginn der Sommerzeit
% Monat 3 (März), 4. Woche, 7. Tag (Sonntag), 7200 Sekunden
(02:00 Uhr)
dststart 3,4,7,7200

% Ende der Sommerzeit
% Monat 10 (Okt), 4. Woche, 7. Tag (Sonntag),10800 Sekunden
(03:00 Uhr)
dstend 10,4,7,10800

% Zeitverschiebung in Sekunden
dstshift 3600

END LC_TOD
```

Nachdem die Änderungen gespeichert wurden, muss das Locale-Objekt neu erstellt werden. Dazu sind die nachfolgenden Teilschritte erforderlich:

- DLTLOCALE (‘/QSYS.LIB.LIB/DEMOLOCALE.LIB/DE_DE_E’)
- CRTLOCALE LOCALE(‘/QSYS.LIB.LIB/DEMOLOCALE.LIB /DE_DE_E. LOCALE’)
- SRCFILE(‘/QSYS.LIB/DEMOLOCALE.LIB/LOCALESRC.FILE/DE_ DE_E.MBR’)
  CCSID(1141) TEXT(‘Locale für Deutsch_Deutschland_Euro Demo Zeiteinstellungen’)

2.10.3
Seite 18



CRTLOCALE

Damit sind die notwendigen Anpassungen für die Locale abgeschlossen und diese kann zusammen mit einem Benutzerprofil oder mit der Verwendung von APIs in beliebigen Anwendungen genutzt werden.

Das Beispiel der Zeitzonen ist nur eines von einem immens breiten Einsatzspektrum der Locale-Objekte. Insbesondere im Bereich der Globalisierung spielen diese eine große Rolle.

Grundsätzlich ist der Einsatz von Locale nicht nur auf die Ebene der Benutzer beschränkt. Vielmehr können wir Locale auch dazu nutzen, um systemweite Definitionen festzulegen. Allerdings ist dies nicht der empfohlene Weg, da auf Systemebene andere Möglichkeiten für die Beeinflussung der globalen Einstellungen existieren.

Die Nutzung von Locales ist außerdem mit dem folgenden API in Anwendungsprogrammen möglich:

- Retrieve Locale Information (QLGRTVLC, QlgRetrieveLocaleInformation)
  Mit diesem API können wir den Inhalt der angegebenen Locale-Definition abrufen und in einer Anwendung nutzen.

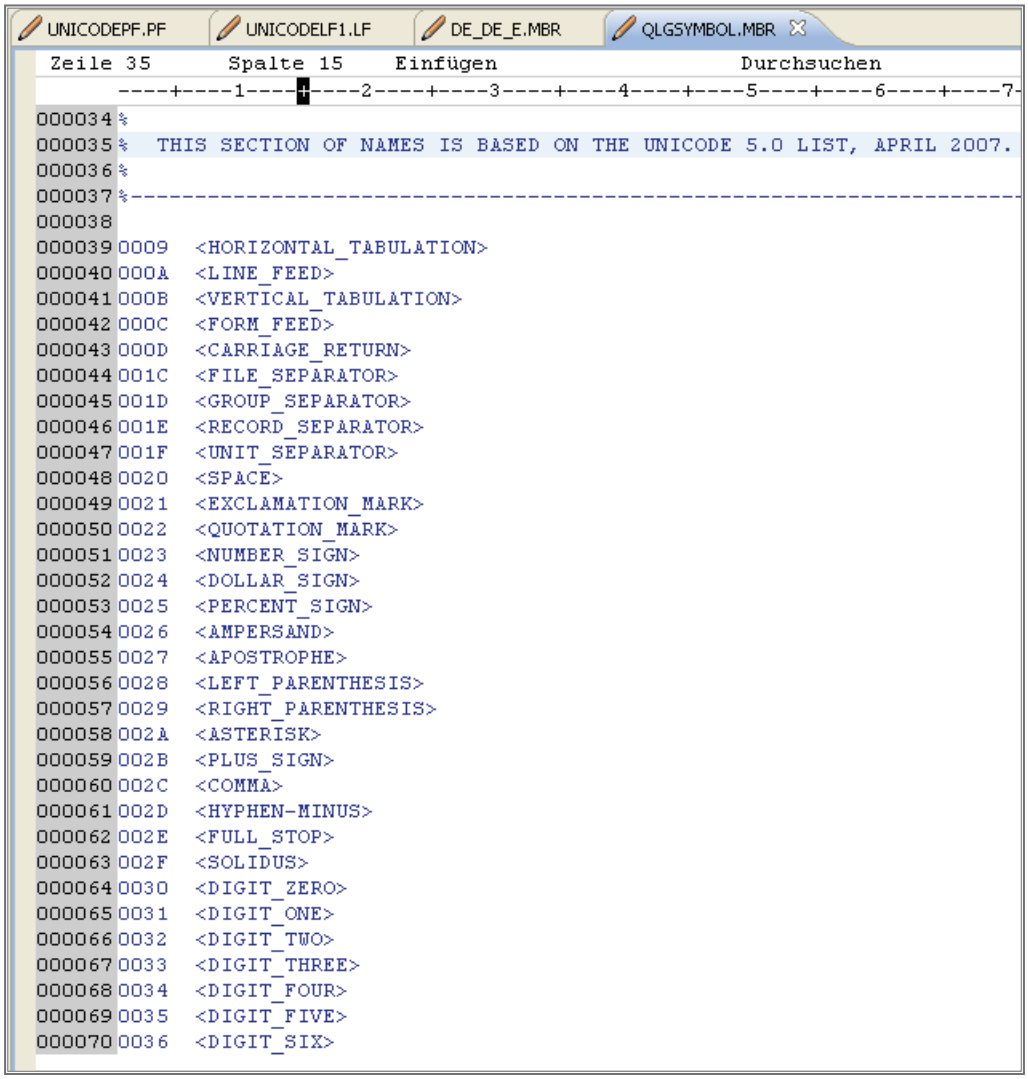
Für die Verwendung von Locales finden wir zwei Systemwerte, welche den Einsatz von Locales beeinflussen:

- QLOCALE
  Dieser Systemwert legt die Locale-Objekte fest. Der Standardwert ist *NONE. Mögliche Werte sind hier:
  – *C
  – *POSIX
  – Lokaler Pfad
- QSETJOBATR
  Mit diesem Wert wird das Setzen der Job-Attribute zur Startzeit eines Jobs beeinflusst. Als Standardwert ist hier *NONE eingestellt. Zusammen mit der Einstellung in QLOCALE können die folgenden Job-Attribute aus dem Locale-Bereich zur Startzeit eines Jobs übernommen werden:
  – *CCSID
  – *DATFMT
  – *DATSEP
  – *SRTSEQ
  – *TIMSEP

2.10.3
Seite 20

QLGSYMBOL

Gültige Symbole, die wir auch zusammen mit den QLOCALESRC-Quellen verwenden, sind in der Quellteildatei QLGSYMBOL on QSYSLOCALE/QLOCALESRC enthalten. Einen exemplarischen Auszug zu dieser Quelldatei finden Sie in der nachfolgenden Abbildung:



Gültige Symbole

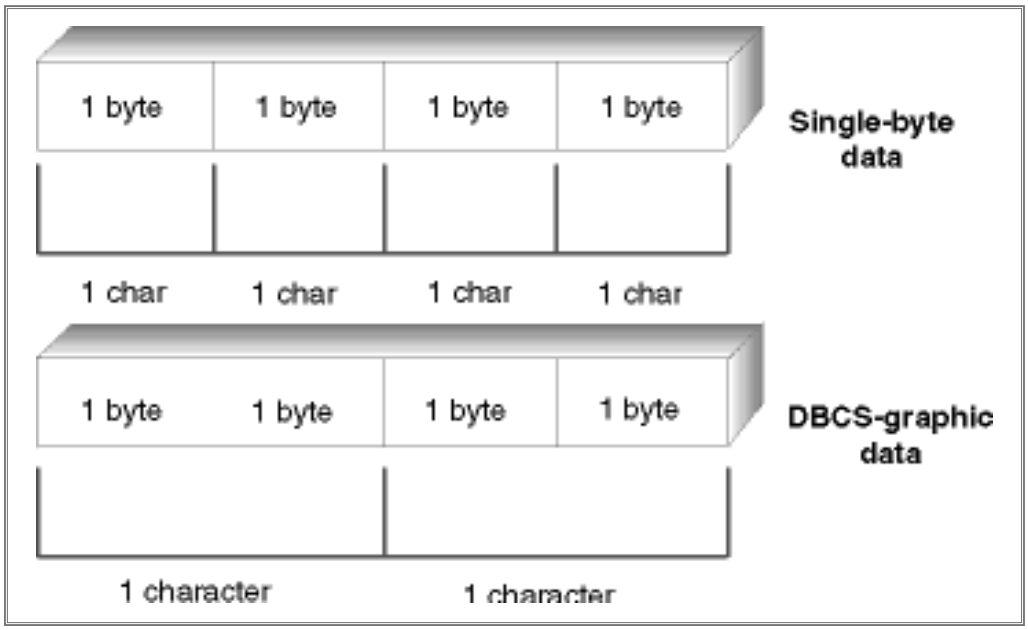
Wie Sie hier sehen können, lassen sich die verschiedensten Symbole definieren. Auch diese Einstellungen können wir bei Bedarf individuell anpassen und für den globalen Einsatz unserer IBM i-Anwendungen nutzen.

DBCS und System i

Die Frühzeiten der IT waren wenig global – und so waren die US-amerikanischen IT-Unternehmen einst der Auffassung, mit einem auf ihre Verhältnisse angepassten Zeichensatz sehr gut auskommen zu können. Aber die Zeiten haben sich bekanntlich gewandelt und der Siegeszug der IT-Systeme ist längst zu einer weltweiten Erfolgsgeschichte geworden, die sich nicht auf die westliche Hemisphäre und deren Zeichensätze beschränkt.

Die Unterstützung der darstellbaren Zeichen ist somit seit Einführung unterschiedlicher IT-Systeme von unterschiedlichen Herstellern in aller Welt eine immer wiederkehrende Herausforderung, der sich zunehmend mehr Unternehmen stellen müssen. Der Grund hierfür liegt zu einem Großteil in der angesprochenen Globalisierung. Die meisten Handelsgrenzen sind gefallen und viel Geschäftsverkehr erfolgt auch mit solchen Kunden oder Geschäftspartnern, deren Sprache Zeichen beinhalten, die wir nicht als „Standard“ auf unseren IT-Systemen verfügbar haben.

Während in den nichtasiatischen Bereichen der SDBC (Single Byte Character Set) -Zeichensatz die größte Verbreitung gefunden hat, sind die in großer Anzahl und Variation existenten asiatischen Zeichen im uns bekannten Basiszeichensatz nicht abbildbar. Deshalb wurde ein erweiterter Zeichensatz auf der Basis von DBCS (Double Byte Character Set) geschaffen. Mit DBCS fällt die Beschränkung, die bei SBCS 128 bzw. 256 Zeichen beträgt, weg. Mit DBCS können somit auch die asiatischen Zeichen gespeichert werden, deren Abbildung in SBCS nicht möglich ist. Wie der Name schon sagt, werden beim Einsatz von DBCS die Zeichen mit 2 Bytes dargestellt – im Gegensatz zur 1-Byte-Darstellung im SBCS-Zeichensatz.



Quelle: IBM

Mit DBCS sind bis zu 65536 Zeichen abbildbar. Dass diese Begrenzung mittlerweile auch ausgeschöpft ist, und Unicode für den allumfassenden Zeichenbedarf genutzt werden kann, soll hier lediglich erwähnt werden.

IBM unterstützt auf der IBM i zwei unterschiedliche DBCS-Codeschemata:

- Erstes Byte
  Hex 41 bis Hex FE
- Zweites Byte
  Hex 41 bis Hex FE
- Double Byte blank
  Hex 4040

In DBCS kommt eine Erweiterung von EBCDIC zum Einsatz, mit der die Double-Byte-Darstellung der Zeichen ermöglicht wird. Diese Erweiterung benutzt zwei Steuerzeichen:

- Shift-Out (SO)
  X'oE'
- Shift-In (SI)
  X'oF'

Alle DBCS-Zeichen, die für die Darstellung auf einer Arbeitsstation oder für die Ausgabe auf einem Drucker zur Verfügung stehen, werden als „Basis-DBCS-Zeichen" bezeichnet und von den angebundenen Einheiten hardwareseitig unterstützt.

Derzeit werden auf dem System i die folgenden DBCS-Zeichensätze unterstützt:

- Japanisch
- Koreanisch
- Einfaches Chinesisch
- Traditionelles Chinesisch

Sollten Sie DBCS zusammen mit System i Access verwenden und einsetzen wollen, dann stehen zusätzlich die folgenden DBCS Codepages zur Verfügung:

- Republic of Korea National-Standardzeichensatz (KS)
- Taiwan Industry Standard-Grafikzeichensatz (Big5)
- China National Standard-Grafikzeichensatz (GB)

Aufgrund der großen Anzahl an DBCS-Zeichen müssen wir dem System gezielte Informationen darüber geben, welche der Zeichen verarbeitet werden sollen.

Bei der Nutzung von Basis-DBCS-Zeichensätzen sind die unterstützten Zeichen in den verwendeten Einheiten (der Hardware) bekannt. Die jeweils unterstützen Zeichen variieren folglich in Abhängigkeit von der Hardware (z. B. Drucker) und der Sprachunterstützung.

Erweiterte DBCS-Zeichen können von den Endgeräten (z. B. den Druckern) nicht alleine umgesetzt werden. Hier ist eine zusätzliche Unterstützung durch das Betriebssystem notwendig. Die Erweiterungen definiert man über das Schlüsselwort IGCEXNCHR(*YES) auf Basis von Druckdateien und/oder Displayfiles.

In diesem Zusammenhang ist die Kenntnis des sogenannten „Last Code-Punkts“ nützlich. Beim letzten Codepunkt handelt es sich um die definierte Trennung zwischen den Basis-DBCS-Zeichen und den erweiterten DBCS-Zeichen. Die Angaben zum letzten Codepunkt variieren in der Regel von Einheit zu Einheit. Er wird von IBM i dazu verwendet, um zu lokalisieren, ob die zu druckenden DBCS-Zeichen von der Einheit selbst unterstützt werden oder mit zusätzlichem Aufwand (Rechenleistung) vom System i bereitgestellt werden müssen. Die Zeichen müssen dann natürlich auf dem System i existieren!

Wenn keine erweiterten DBCS-Zeichen vom System oder von den Endgeräten unterstützt werden, tritt Folgendes ein:

- Die Zeichen werden in der Datenbank (in Abhängigkeit von der Definition der Tabelle/Datei) korrekt gespeichert, aber leider falsch dargestellt/gedruckt.
- Es werden nur Basis-DBCS-Zeichen angezeigt oder gedruckt.
- In Bildschirmdateien werden nichtdarstellbare Zeichen ausgegeben.
- In Druckausgaben werden an Stelle der DBCS-Zeichen nichtdruckbare Zeichen ausgegeben.

Grundsätzlich ist das System i in den neueren Betriebssystemversionen DBCS-fähig – zumindest ab Version V5R3. Im DBCS-Zeichensatz werden neben den Standardzeichen der einzelnen Länder auch die typischen Sonderzeichen abgebildet, wie z. B. Umlaute, Akzentzeichen etc.– sowie die Zeichen einiger asiatischer Sprachen, wie z. B. Japanisch, Chinesisch und Koreanisch.

Die folgenden Sprachen unterstützen DBCS auch vor V5R3:

- 2938 Englisch Uppercase
- 2962 Japanisch Kanji DBCS
- 2984 Englisch Upper/Lowercase DBCS
- 2986 Koreanisch DBCS
- 2987 Traditionelles Chinesisch DBCS
- 2989 Einfaches Chinesisch DBCS

Diese Sprachen können bei Bedarf als Primär- oder Sekundärsprache auf dem System i installiert werden und machen jeweils die in ihnen vorkommenden spezifischen DBCS-Zeichen verfügbar.

In Abhängigkeit von den Anforderungen hinsichtlich des Einsatzes von DBCS sind ggf. zusätzliche Lizenzprogramme auf dem System i erforderlich. 57xx-AP1 (Advanced DBCS Printer Support) ist eines der optionalen Lizenzprogramme, das für den DBCS-Einsatz in Frage kommen kann.

57xx-AP1 (Advanced DBCS Printer Support)
Mit diesem Lizenzprogramm werden auf dem System i zusätzliche DBCS-Druckfunktionen bereitgestellt. Advanced DBCS Printer Support besteht aus den folgenden Hauptbereichen:

- Advanced Printer Writer (APW)
- Advanced Page Printer Writer (APPW)
- Kanji Printer Function (KPF)
- Printer Form Description / Symbol Migration Aid
- Printer Function Control (PFC)

DBCS-Arbeitsstationen und Drucker müssen in der Lage sein, die DBCS-Zeichen darstellen und auch ausdrucken zu können. Dabei kommt es nicht darauf an, dass es sich bei den Einheiten um reine DBCS-Einheiten handelt. Natürlich bieten sich solche an – aber es ist auch möglich, solche Einheiten zu verwenden, die beispielsweise erst durch eine Zusatzsoftware oder ein Zusatzfeature DBCS-fähig gemacht wurden.

Das Einstellen von DBCS-Zeichen ist stark mit der verwendeten Eingabeeinheit verbunden. Durch die enorme Anzahl der mit DBCS abbildbaren Zeichen ist es natürlich auch klar, dass die Zeichen in ihrem gesamten Umfang nicht von einer Tastatur aus eingegeben werden können – die Anzahl möglicher Zeichen ist schlichtweg zu groß!

Durch die Double Byte Charakter-Speicherung ist jedes DBCS-Zeichen 2 Bytes lang. Sollen DBCS-Grafikfelder eingesetzt werden, dann wird deren Länge über die Anzahl dazu notwendiger DBCS-Zeichen angegeben.

Was ist beim Einsatz von DBCS zu beachten? So unproblematisch die eigentliche Implementierung von DBCS in IBM i auch ist – der Teufel steckt wie immer im Detail. Sortierreihenfolgen, Bildung und Anwendung von Substrings oder auch der Einsatz zusammen mit Eingabe- und Ausgabeeinheiten oder „klassischen“ Anwendungen wie dem guten alten Query müssen beachtet werden.

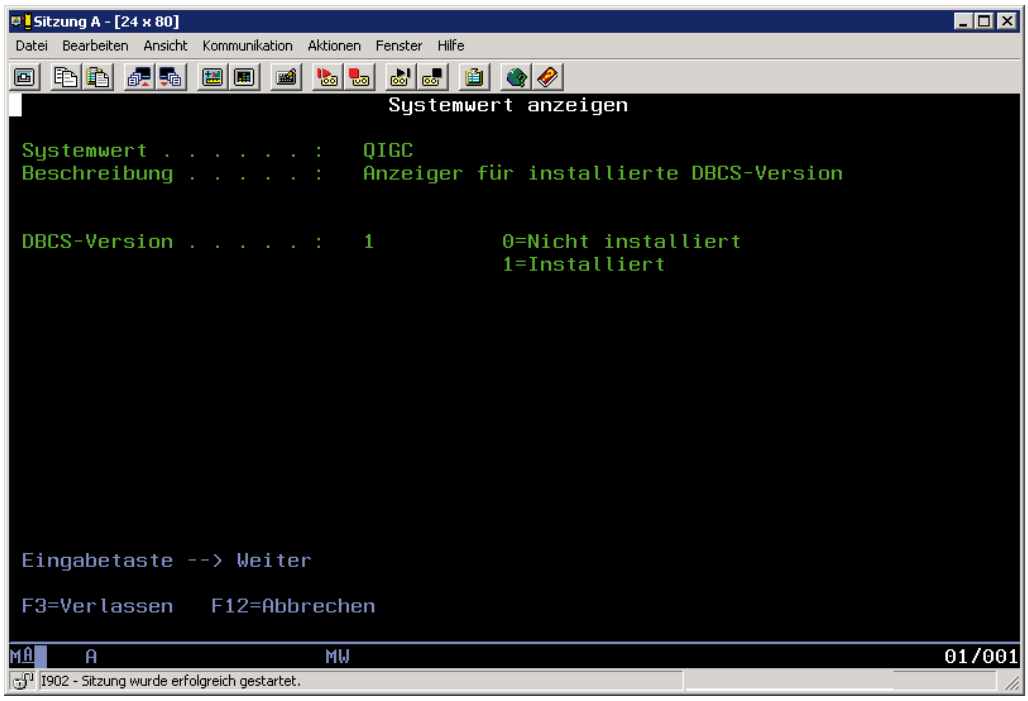
IBM bietet mit IBM i DBCS-Unterstützung für folgende Bereiche:

- Datenbankdateien
- entsprechend ausgelegte Display Files
- entsprechend ausgelegte Printer Files
- HLL-Sprachen
- Nachrichtentexte
- Texte von Objektbeschreibungen

Die nachfolgenden Bereiche sind nicht DBCS-fähig:

- IBM i-Objektnamen
- Befehlsnamen
- Variablen in HLL (z. B. RPG oder Cobol) und CL

Eine der Definitionsebenen für den Einsatz von DBCS ist der Systemwert QIGC. Er legt fest, ob das System grundsätzlich DBCS-Unterstützung bietet:



SYSVAL QIGC

Dieser Systemwert wird automatisch mit der Installation der Primärsprache gesetzt. Ist die Primärsprache DBCS-tauglich, dann wird der Systemwert automatisch auf „1“ gesetzt – anderenfalls ist der Wert mit 0 vorbelegt. Dieser Systemwert hat eigentlich mit dem Erscheinen von Version V5R3 an Bedeutung verloren – denn ab diesem Release sind alle Primärsprachen auf dem System i von Haus aus DBCS-fähig.

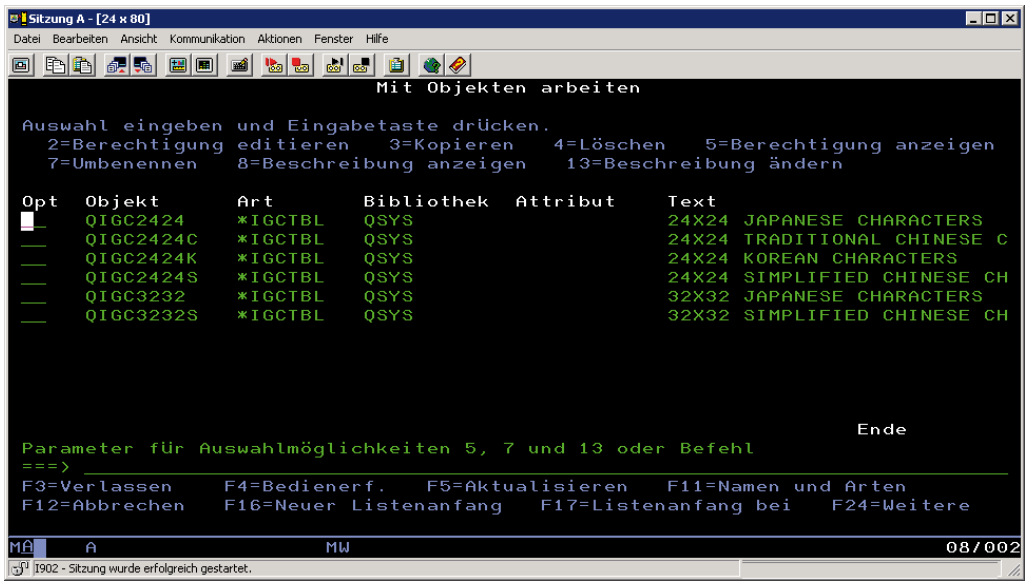
Bevor Sie sich Gedanken über einen Einsatz von DBCS machen und Anwendungen darauf auslegen wollen, sollten Sie die folgenden Punkte beachten. Sie umgehen damit Hürden, an denen zuvor schon so manches DBCS-Projekt gescheitert ist.

- **Alle** Einheiten, die DBCS unterstützen sollen, müssen auch DBCS-fähig sein.
- Stellen Sie sicher, dass DBCS-Daten ausschließlich in DBCS-fähigen Einheiten verarbeitet und nicht etwa gesplittet werden!
- Stellen sie ggf. eine DBCS-Umsetzung für die Fälle sicher, bei denen DBCS-Umsetzungen erforderlich sind.
- Beachten Sie, dass unter Umständen DBCS-Nachrichten für den Einsatz in Programmen genutzt werden können oder müssen.

DBCS-Zeichensatztabellen

Mit der Bereitstellung von DBCS hat IBM verschiedene Zeichensatztabellen in IBM i integriert. In Abhängigkeit von der jeweiligen landesspezifischen Unterstützung in IBM i können die DBCS-Zeichensatztabellen variieren. Diese Zeichensatztabellen werden dazu verwendet, um die Zeichen korrekt anzuzeigen oder auch zu drucken.

Die Zeichensatztabellen werden im Zuge der Installation des Systems IBM i auf dem System mitinstalliert und finden sich als Objekte des Typs *IGCTBL wieder. Die folgende Abbildung zeigt eine Auflistung der per Standard auf einem deutschen System installierten DBCS-Zeichensatztabellen.



DBCS-Zeichensatztabellen

Folgende Zeichensatztabellen stehen z. B. mit IBM i 7.1 zur Verfügung:

- QIGC2424
  - Japanische DBCS-Zeichensatztabelle
  - 24 x 24 Dot Matrix
  - Verwendung u. a. mit japanischen Anzeigen oder Druckausgaben
- QIGC2424C
  - Traditionelles Chinesisch
  - 24 x 24 Dot Matrix
  - Einsatz zusammen mit Druckern 5227 Modell 3 oder 5327 Modell 3
- QIGC2424K
  - Koreanisch
  - 24 x 24 Dot Matrix
  - Einsatz zusammen mit Druckern 5227 Modell 2 oder 5327 Modell 2

2.10.4.1
Seite 2

- QIGC2424S
  - Einfaches Chinesisch
  - 24 x 24 Dot Matrix
  - Einsatz zusammen mit Druckern 5227 Modell 5 oder 5327 Modell 5

QIGC3232
  - Japanisch
  - 32 x 32 Dot Matrix
  - Einsatz zusammen mit Druckern 5583 oder 5337 Modell 1

Sollte der von Ihnen gewünschte DBCS-Zeichensatz nicht in der Standardauslieferung der IBM i enthalten sein, dann können Sie bei Bedarf auch einen individuellen Zeichensatz einsetzen bzw. einer DBCS-Zeichensatztabelle benutzerdefinierte Zeichen hinzufügen.

Dateien und DBCS

Bevor wir DBCS nutzen können, müssen die einzelnen Komponenten, die bei der Verarbeitung der DBCS-Daten ins Spiel kommen, DBCS-fähig sein. Dazu gehören zum Beispiel DDS-Beschreibungen oder besondere DBCS-Spezifikationen, die auf Einheitenebene anzugeben sind.

Nachfolgend liste ich Ihnen die verschiedenen Dateiebenen auf, die DBCS-Unterstützung bieten.

2.10.4.2
Seite 2

DDS

DDS unterstützt derzeit die folgenden DBCS-Varianten:

- Reine DBCS-Felder (DBCS Only)
  Unterstützen reine DBCS-Zeichen. Die in diesen Bereichen verwendeten DBCS-Zeichen müssen in Shift-Out (SO) und Shift-In (SI) -Zeichen eingebettet sein. Z. B. sss (SO) D1D2D (SI)
- DBCS Open-Felder
  Diese Felder unterstützen sowohl Single-Byte- als auch Double-Byte-Zeichen. Auch hier gilt, dass die DBCS-Zeichen in Shift-Out- und Shift-In-Zeichen eingebettet sein müssen.
- DBCS Either-Felder
  Diese Form der DBCS-Felder unterstützt entweder Single-Byte- oder Double-Byte-Zeichen. Es ist jedoch nicht möglich, beide Varianten gleichzeitig zu verwenden. Auch hier gelten die Shift-Out/Shift-In-Regeln!
- DBCS Grafikfelder
  Unterstützen ausschließlich Double-Byte-Zeichen. Hier werden **keine** Shift-Out- und Shift-In-Angaben verwendet!

DDS mit DBSC und Druckdateien (PRTF)

In Druckdateien (PRTF) können DBCS-Daten ebenfalls über die Datenart definiert werden. DSPF unterstützen die folgenden DBCS-Datenarten:

- DBCS Open-Felder
  Datenart O in Spalte 35
- DBCS-Grafikfelder
  Datenart G in Spalte 35

Bereits beim Erstellen der Druckdateien kann der Entwickler vorgeben, ob und wie DBCS mit der Druckdatei verwendet werden soll. DDS-Schlüsselwörter regeln hier die DBCS-Einbindung.

Es kommt auf die verwendeten Schlüsselwörter und deren Koexistenz mit Parametern auf System- oder Druckdatei-Ebene an, ob und wie das System DBCS unterstützt. Wenn Sie z. B. DBCS-Felder in einer Druckdatei beschreiben, auf Druckdateiebene aber den Parameter IGCDTA(*YES) nicht angeben, dann behandelt das System die Datei dennoch als DBCS-fähige Datei.

2.10.4.2.1
Seite 2

Nachfolgend finden Sie einige der wesentlichen DBCS-relevanten Schlüsselwörter.

CHRSIZ (Zeichengröße)
Dieses Schlüsselwort kann wahlweise auf Satzformat- oder Feldebene angegeben werden und definiert die Größe des zu druckenden Bereichs bzw. der zu druckenden Zeichen.

Das Format für CHRSIZ ist: CHRSIZ(Breite (Höhe))

Breite ist ein erforderlicher Parameter, die Höhe kann optional angegeben werden – das System verwendet den Standardwert 1, wenn keine Höhe vorgegeben wurde.

Gültige Werte für CHRSIZ-Breite und -Höhe sind 1 und 2.

Die folgende Abbildung zeigt ein Beispiel:

```
A           R KOPF
A             FLDDBO         50O         POSITION(4.1 2.0)


A             FLDDBG         50G         POSITION(4.1 2.0)
A                                        FONTNAME('Monotype Sans -
A                                        Duospace WT SC' +
A                                        (*POINTSIZE 9.0))
A                                        CCSID(13488 *NOCONVERT)

 * Verwendung von CHRSIZ
A           R POSITION                     CHRSIZ(2 1)
A             FELD01         23O       2OSPACEA(2)
A             FELD02         80A       2OSPACEA(2) CHRSIZ(1 1)
A             FELD03         41O       2OSPACEA(2)
A             FELD04         45O       2OSPACEA(2) CHRSIZ(1 2)
A             FELD05         40A       2OSPACEA(2) CHRSIZ(1)
A             FELD06         25A       2OSPACEA(2) CHRSIZ(2 2)
```

CHRSIZ

In dem gezeigten Beispiel wird das Feld FELD04 doppelt so hoch gedruckt. Das Feld FELD06 wird doppelt so hoch und doppelt so breit ausgegeben.

CHRSIZ

- kann sowohl für DBCS als auch für alphanumerische Daten genutzt werden.
- kann zusammen mit dem Rotieren des Felds genutzt werden.
- kann nicht zusammen mit BARCODE, COLOR oder LPI eingesetzt werden.

IGCALTTYP (alternativer Datentyp)
Dieses Schlüsselwort steht auf Feldebene zum Einsatz bereit. Mit ihm lassen sich alphanumerische Feldinhalte in DBCS-Felder des Typs O umsetzen. Dieses Schlüsselwort kann nur in Verbindung mit der Angabe „IGCDTA(*YES)“ verwendet werden. Diese Angabe muss auf Druckdateiebene z. B. durch OVRPRTF gesetzt werden.

IGCALTTYP

- kann nur zusammen mit alphanumerischen Zeichen genutzt werden.
- sollte nicht für DBCS-Zeichen angegeben werden.
- kann nicht mit dem Einheitentyp *AFPDS genutzt werden.
- kann nicht zusammen mit einem der folgenden Schlüsselwörter genutzt werden:
  - BLKOLD
  - CPI
  - CVTDTA
  - IGCANKCHV
  - TRNSCV

Die folgende Abbildung zeigt die Verwendung von IGCALTTYP:

```
A            R KOPF
A              FLDDBO         50O         POSITION(4.1 2.0)


A              FLDDBG         50G         POSITION(4.1 2.0)
A                                         FONTNAME('Monotype Sans -
A                                         Duospace WT SC' +
A                                         (*POINTSIZE 9.0))
A                                         CCSID(13488 *NOCONVERT)

 * Verwendung von CHRSIZ
A            R POSITION                     CHRSIZ(2 1)
A              FELD01         23O       20SPACEA(2)
A              FELD02         80A       20SPACEA(2) CHRSIZ(1 1)
A              FELD03         41O       20SPACEA(2)
A              FELD04         45O       20SPACEA(2) CHRSIZ(1 2)
A              FELD05         40A       20SPACEA(2) CHRSIZ(1)
A              FELD06         25A       20SPACEA(2) CHRSIZ(2 2)

 * Verwendung von IGCALTTYP
A            R POSITION2                    SKIPB(3)
A              FELD10         40        47SPACEA(2) UNDERLINE
A  30          FELD11         40A       47SPACEA(2) UNDERLINE IGCALTTYP
```

IGCALTTYP

Das Feld „FELD11“ kann in diesem Beispiel DBCS-Daten aufnehmen und drucken.

2.10.4.2.1
Seite 4

IGCANKCNV (Umsetzung alphanumerisch in DBCS)
Dieses Schlüsselwort kann nur auf Feldebene definiert werden, um alphanumerische Zeichen in die zugehörigen DBCS-Zeichen zu konvertieren. Der Einsatz ist hier auf den japanischen Zeichensatz beschränkt. Dieses Schlüsselwort kann **nicht** für DBCS-Grafikfelder (G) definiert werden!

Achten Sie zudem darauf, dass nicht aus Versehen DBCS-Inhalte mit IGCANKCNV umgesetzt werden sollen! Das „doppelte" Umsetzen kann zu Problemen und unerwünschten Ausgaben führen!

```
A          R KOPF
A            FLDDBO        50O         POSITION(4.1 2.0)


A            FLDDBG        50G         POSITION(4.1 2.0)
A                                      FONTNAME('Monotype Sans -
A                                      Duospace WT SC' +
A                                      (*POINTSIZE 9.0))
A                                      CCSID(13488 *NOCONVERT)

 * Verwendung von CHRSIZ
A          R POSITION                    CHRSIZ(2 1)
A            FELD01        23O       20SPACEA(2)
A            FELD02        80A       20SPACEA(2) CHRSIZ(1 1)
A            FELD03        41O       20SPACEA(2)
A            FELD04        45O       20SPACEA(2) CHRSIZ(1 2)
A            FELD05        40A       20SPACEA(2) CHRSIZ(1)
A            FELD06        25A       20SPACEA(2) CHRSIZ(2 2)

 * Verwendung von IGCALTTYP
A          R POSITION2                   SKIPB(3)
A            FELD10        40        47SPACEA(2) UNDERLINE
A  30        FELD11        40A       47SPACEA(2) UNDERLINE IGCALTTYP

 * Verwendung von IGCANKCNV
A          R POSITION3                   CHRSIZ(2) SKIPB(3)
A            FELD20        40O       20SPACEA(2)
A            FELD21        80A       20SPACEA(2) CHRSIZ(1)
A            FELD22        20A       20SPACEA(2) IGCANKCNV
```

IGCANKCNV

In diesem Beispiel wird der Inhalt des Felds FELD22 umgesetzt.

Ähnlich die bei anderen DBCS-relevanten Schlüsselwörtern ist ein Einsatz zusammen mit COLOR, BARCODE oder LPI nicht möglich!

Die folgenden Umsetzungen können durchgeführt werden:

- alphanumerische Zeichenketten in eine DBCS-Zeichenkette
- englische alphanumerische Zeichen in DBCS-alphanumerische Zeichen
- alphanumerisch Katakana in DBCS Hiragana und DBCS Katakana
- DBCS-Code in zugehörige DBCS-Zeichen

IGCCDEFNT (DBCS-codierter Zeichensatz)
Dieses Schlüsselwort kann wahlweise auf Feld- oder Datensatzebene definiert werden und dient der Angabe des DBCS-codierten Fonts für das Drucken von benannten oder konstanten Feldern.

IGCCDEFNT kann nur im Zusammenhang mit *AFPDS für den Einheitentyp der Druckdatei genutzt werden! Damit verbunden ist auch die Notwendigkeit, dass das Lizenzprogramm PSF (57xx-PSF) auf dem System installiert ist!

Das Format für dieses Schlüsselwort ist:
IGCCDEFNT([library-name/ | &library-name-field/]
coded-font-name | &coded-font-name-field
[(*POINTSIZE height-value | &height-value-field
width-value | &width-value-field)])

Als Code-Parameter muss ein auf dem System existierender DBCS-Font angegeben werden:

```
 * Verwendung von CHRSIZ
A          R POSITION                  CHRSIZ(2 1)
A            FELD01        23O       20SPACEA(2)
A            FELD02        80A       20SPACEA(2) CHRSIZ(1 1)
A            FELD03        41O       20SPACEA(2)
A            FELD04        45O       20SPACEA(2) CHRSIZ(1 2)
A            FELD05        40A       20SPACEA(2) CHRSIZ(1)
A            FELD06        25A       20SPACEA(2) CHRSIZ(2 2)

 * Verwendung von IGCALTTYP
A          R POSITION2                 SKIPB(3)
A            FELD10        40        47SPACEA(2) UNDERLINE
A  30        FELD11        40A       47SPACEA(2) UNDERLINE IGCALTTYP

 * Verwendung von IGCANKCNV
A          R POSITION3                 CHRSIZ(2) SKIPB(3)
A            FELD20        40O       20SPACEA(2)
A            FELD21        80A       20SPACEA(2) CHRSIZ(1)
A            FELD22        20A       20SPACEA(2) IGCANKCNV

 * Verwendung von IGCCDEFNT
A          R POSITION4
A            FELD30        240     2 14IGCCDEFNT(XOG16B)
A            FELD31        240     3 14IGCCDEFNT(ZEIG/XOG16C +
A                                      (*POINTSIZE 10.0))
```

IGCCDEFNT

2.10.4.2.1
Seite 6

IGCCHRRTT (DBCS-Zeichenrotation)
Dieses Schlüsselwort kann wahlweise auf Feld- oder Datensatzebene definiert werden. Mit IGCCHRRTT lassen sich DBCS-Zeichen 90 Grad gegen den Uhrzeigersinn drehen. Gedrehte DBCS-Zeichen werden vertikal ausgedruckt. Die Unterstützung beschränkt sich auf reine DBCS-Daten – alphanumerische Daten werden nicht gedreht!

Dieses Schlüsselwort kann nur in Verbindung mit 5553-Druckern oder der Angabe AFP(*YES) eingesetzt werden.

IGCCHRRTT kann nur für DBCS-Felder (O oder G) oder mit DBCS in konstanten Feldern genutzt werden.

Die folgende Abbildung zeigt ein Beispiel:

```
A            FELDO2         80A       2OSPACEA(2) CHRSIZ(1 1)
A            FELDO3         41O       2OSPACEA(2)
A            FELDO4         45O       2OSPACEA(2) CHRSIZ(1 2)
A            FELDO5         40A       2OSPACEA(2) CHRSIZ(1)
A            FELDO6         25A       2OSPACEA(2) CHRSIZ(2 2)

 * Verwendung von IGCALTTYP
A          R POSITION2                   SKIPB(3)
A            FELD10         40        47SPACEA(2) UNDERLINE
A  30        FELD11         40A       47SPACEA(2) UNDERLINE IGCALTTYP

 * Verwendung von IGCANKCNV
A          R POSITION3                   CHRSIZ(2) SKIPB(3)
A            FELD20         40O       2OSPACEA(2)
A            FELD21         80A       2OSPACEA(2) CHRSIZ(1)
A            FELD22         20A       2OSPACEA(2) IGCANKCNV

 * Verwendung von IGCCDEFNT
A          R POSITION4
A            FELD30         24O     2 14IGCCDEFNT(XOG16B)
A            FELD31         24O     3 14IGCCDEFNT(ZEIG/XOG16C +
A                                      (*POINTSIZE 10.0))

 * Verwendung von IGCCHRRTT
A          R POISTION5                   SKIPB(3)
A                                        IGCCHRRTT
A            FELD40         40O       47SPACEA(2) UNDERLINE
A
```

IGCCHRRTT

Bei der Definition der DDS-Schlüsselwörter sollten Sie Folgendes beachten:

- Die Angaben auf DDS-Ebene und Druckdateiebene sollten aufeinander abgestimmt sein. Hier gilt u. a. die Prüfung, ob DBCS-Definitionen in der DDS-Beschreibung enthalten sind und ob IGCDTA(*YES) definiert wurde.
- DBCS-Informationen werden in einer größeren Größe gedruckt als alphanumerische Inhalte. Auch wenn CPI nicht zusammen mit DBCS-Angaben getätigt werden kann, wirken sich die CPI-Definitionen natürlich auf der Druckdateiebene auf die Position der DBCS-Druckzeichen aus!

DDS mit DBCS und Bildschirmdateien (DSPF)

In Bildschirmdateien (DSPF) können DBCS-Daten durch die Angabe der Datenart O oder G definiert werden.

- DBCS Open-Felder
  - Datenart O in Spalte 35
- DBCS-Grafikfelder
  - Datenart G in Spalte 35
  - Länge von 1 – 16383 Zeichen
- DBCS Only
  - Datenart J in Spalte 35
- DBCS Either
  - Datenart E in Spalte 35

Die folgende Abbildung zeigt ein Beispiel für DBCS-Definitionen auf Feldebene:

```
A*****************************************************************
A*
A*%%EC
A                                       DSPSIZ(24 80 *DS3)
A                                       CHGINPDFT(HI CS UL)
A                                       PRINT
A                                       CA03(03 'VERLASSEN')
A*
A*----------------------------*
A          R RECORD1
A            FELD01        80A  I 23  2IGCALTTYP
A                                   1 19' ERMITTELN, PRÜFEN - VERZEICHNIS/D-
A                                       ATEI-NAME '
A                                       DSPATR(RI)
A                                       DSPATR(UL)
A                                   1 72DATE
A                                       EDTCDE(Y)
A                                   2  2USER
```

DBCS DSPF-Angaben

In Bildschirmdateien können wir zudem das Schlüsselwort IGCCNV (DBCS-Umsetzung) einsetzen. Mit diesem Schlüsselwort wird vorgegeben, dass eine DBCS-Umsetzung durchgeführt werden soll.

IGCCNV kommt im Zusammenhang mit IGCALTTYP zum Einsatz. Dieses Schlüsselwort benötigt zwei Parameter:

2.10.4.2.1
Seite 8

IGCCNV(CFxx-Zeilenummer)
Der erste Parameter CFxx definiert die Befehlstaste, mit der die DBCS-Umsetzung gestartet werden soll. Dabei können die bekannten 24 Befehlstasten verwendet werde.

Mit dem zweiten Parameter (Zeilennummer) legen wir die Zeile in der DSPF fest, in der die Umsetzungszeile dargestellt werden soll. Bei Eingabe des alphanumerischen Werts in dieser Zeile wird nach dem Drücken der Befehlstaste der DBCS-Wert in dieser Zeile ausgegeben.

```
A*
A*%%EC
A                                      DSPSIZ(24 80 *DS3)
A                                      CHGINPDFT(HI CS UL)
A                                      PRINT
A                                      CA03(03 'VERLASSEN')
A
A                                      IGCCNV(CF24 24)
A*
A*-------------------------*
A          R RECORD1

A            FELD01        80A  I 23  2IGCALTTYP

A                                    1 19' ERMITTELN, PRÜFEN - VERZEICHNIS/D-
A                                      ATEI-NAME '
```

IGCCNV

ICF-Dateien

ICF unterstützen lediglich die DBCS-Datenart O (DBCS Open).

Sobald DBCS-Daten mit einer Datei zum Einsatz kommen sollen, muss dies auf Dateiebene festgelegt werden. Dazu steht das Schlüsselwort IGCALTTYP zur Verfügung. Alternativ zu diesem Schlüsselwort kann auch der Parameter IGCDTA(*YES) verwendet werden.

```
A*********************************************************************
A*
A*%%EC
A                                      DSPSIZ(24 80 *DS3)
A                                      CHGINPDFT(HI CS UL)
A                                      PRINT
A                                      CA03(03 'VERLASSEN')
A*
A*----------------------------*
A          R RECORD1
A            FELD01        80A  I 23  2IGCALTTYP
A                                    1 19' ERMITTELN, PRÜFEN - VERZEICHNIS/D-
A                                       ATEI-NAME '
A                                       DSPATR(RI)
A                                       DSPATR(UL)
A                                    1 72DATE
A                                       EDTCDE(Y)
A                                    2  2USER
```

DBCS DSPF-Angaben

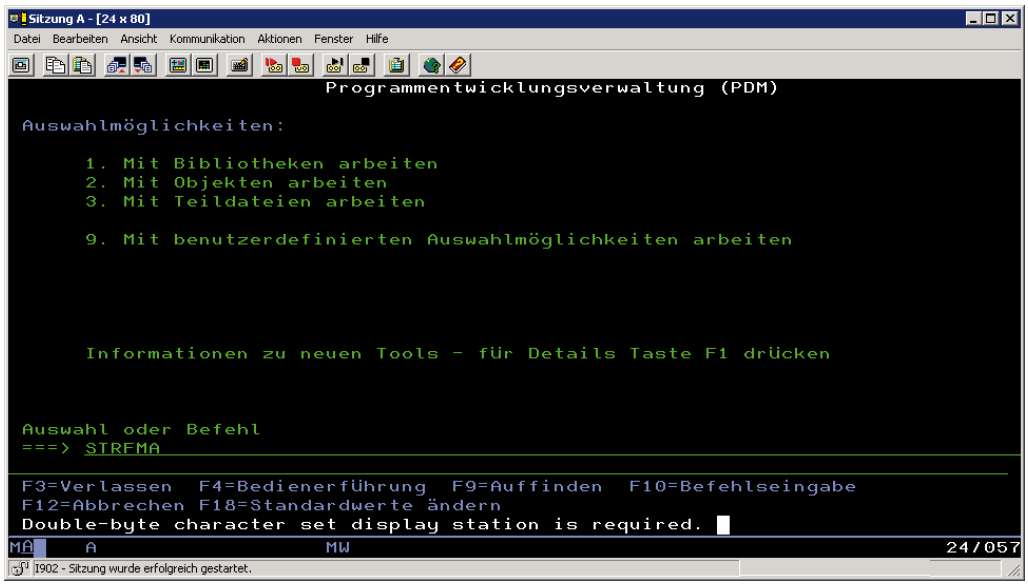
Sobald IGCALTTYP angegeben wird, kann das betreffende Feld (in unserem Beispiel das Feld „FELD01“) DBCS-Daten aufnehmen.

2.10.4.2.2
Seite 2

Eingabeeinheiten und DBCS

Die durchgängige Unterstützung von DBCS von der Eingabe über die Datenhaltung bis hin zur Ausgabe ist eine der Grundvoraussetzungen für dessen Einsatz.

Eine Arbeitsstation oder eine System i Access-Sitzung muss DBCS-fähig sein, wenn DBCS-Daten angezeigt oder eingegeben werden sollen. Ist eine solche Einheit nicht DBCS-fähig, dann werden entweder die DBCS-Daten fehlerhaft dargestellt oder den Anwender erhält die Information, dass eine Verarbeitung der DBCS-Daten nicht möglich ist. Für letzteren Fall finden Sie nachfolgend eine Beispieldarstellung:



Arbeitsstation unterstützt kein DBCS

Für die DBCS-Darstellung mit System i Access eignen sich z. B. die Sprachversionen MRI2938 und MRI 2924.

2.10.4.3
Seite 2

Nachfolgend finden Sie eine Liste einiger der unterstützten DBCS-fähigen Anzeigeeinheiten:

Einheitentyp	Typ-Modell	Einheitenfeature/ Letzter Code Punkt
Japanische Arbeitsstationen		
5295-001 Display Station	5555-B01	2424J4 55FE
5295-002 Display Station	5555-B01	2424J4 68FE
InfoWindow 3477-J Display Station	5555-B01, C01	2424J4 68FE
Personal System/55* running 5250PC	5555-B01	2424J4 68FE
Personal System/55 running graphics-capable 5250PC	5555-G01	2424J4 68FE
Personal System/55 running graphics-capable 5250PC	5555-G02	2424J4 68FE
Personal System/55 running 5250PC/2	5555-E01	2424J0
Personal System/55 running Client Access/400 by OS/400 System	5555-B01	2424J0
3270-type Display Stations*	3279-0	2424J0
Koreanische Arbeitsstationen		
5250-Type Display Stations	5555-B01	2424K0
3270-type Display Stations*	3279-0	2424K0
Traditionelles Chinesisch: Arbeitsstationen		
5250-Type Display Stations	5555-B01	2424C0
3270-type Display Stations*	3279-0	2424C0
Einfaches Chinesisch: Arbeitsstationen		
5250-Type Display Stations	5555-B01	2424S0
3270-type Display Stations*	3279-0	2424S0

Drucken und DBCS

Wie bereits erwähnt, müssen die unterschiedlichen Ebenen der Datenverarbeitung in der Lage sein, DBCS zu unterstützen. Das gilt z. B. auch für die Drucker, die die „Sonderzeichen" zu Papier bringen müssen.

Es kommt letztlich auch darauf an, welche der verschiedenen Druckoptionen beim Druck von DBCS-Daten genutzt wird, also ob es sich um native IBM i-Ausdrucke oder um AFP-basierte Drucke handelt.

Da heute die Kommunikation zu einem Großteil über TCP/IP abgewickelt wird, werden die DBCS-Anbindungen der Drucker und Einheiten natürlich auch über TCP/IP unterstützt.

Beim Einsatz der Drucker kann man grob die folgende Unterteilung verwenden, wenn es darum geht, DBCS-Zeichen zu drucken:

- DBCS SCS-Drucker
- AFP-Drucker mit DBCS Support (IPDS)
  - Nicht alle IPDS-Drucker unterstützen DBCS.
  - IPDS-Drucker müssen AFP DBCS-Zeichen von IBM i herunterladen (Entsprechend großer Speicher im Drucker erforderlich!)
- Drucker mit Anschluss an System i Access (ASCII-Drucker)

Die verschiedenen Druckeranbindungen definieren letztlich die Aktionen, die auch beim Drucken von DBCS beachtet werden müssen bzw. dort Anwendung finden.

SCS-Drucker
SCS-Spooldateien müssen über die Angabe ICGDTA(*YES) verfügen. Damit erkennt der Druckprozess, dass DBCS-Daten zu drucken sind. Dieses Attribut wird in der Druckdatei angegeben (kann z. B. auch mit OVRPRTF gesetzt werden. Die Daten wird an den Print Function Manager übertragen und anschließend an den Drucker gesendet.

AFP-Drucker (IPDS)
Hier werden SCS, AFPDS, LINE oder AFPDSLINE unterstützt. Eine SCS-Spooldatei wird in AFPDS umgesetzt und über PSF/400 in IPDS transferiert. PSF zeichnet hier auch dafür verantwortlich, die AFP DBCS-Zeichen abzurufen und einzubinden.

Nicht alle IPDS-Drucker sind von Haus aus DBCS-fähig. Es kommt auf den individuellen Fall an, ob der betreffende Drucker DBCS-fähig ist. IBM-Webseiten geben hier eine brauchbare Auskunft. Das begründet sich u. a. damit, dass nur einige IPDS-Drucker die AFP DBCS-Zeichensätze herunterladen und umsetzen können. Dieses Verfahren ist ein Muss beim Drucken von DBCS mit AFP.

2.10.4.4
Seite 2

ASCII-Drucker
Die SCS-Spooldatei verfügt über die Angabe ICGDTA(*YES). Auch hier erkennt der Druckprozess, dass DBCS-Daten gedruckt werden sollen. Die DBCS-Umsetzung erfolgt hier durch die Emulation, die die ASCII-Zeichen entsprechend umsetzt.

In Abhängigkeit von den verwendeten Druckspezifika sind für DBCS-Drucke zusätzliche Lizenzprogramme erforderlich.

Wenn Sie ADO zusammen mit DBCS verwenden wollen, dann muss auf Ihrem System die DBCS Font/400-Unterstützung installiert sein. Diese wird von IBM in Form des Lizenzprogramms 5769FN1 bereitgestellt.

- 5769FN1 *BASE Advanced Function Printing DBCS Fonts/400
- 5769FN1 1 AFP DBCS Fonts – Japanese
  – Bibliothek QFNT61
- 5769FN1 2 AFP DBCS Fonts – Korean
  – Bibliothek QFNT62
- 5769FN1 3 AFP DBCS Fonts – Traditional Chinese
  – Bibliothek QFNT63
- 5769FN1 4 AFP DBCS Fonts – Simplified Chinese
  – Bibliothek QFNT64
- 5769FN1 5 AFP DBCS Fonts – Thai
  – Bibliothek QFNT65

Wie Sie in der Auflistung oben erkennen können, werden die verschiedenen DBCS-Zeichensätze in unterschiedlichen Bibliotheken auf dem System IBM i gespeichert. Alle Optionen werden zusammen mit dem Lizenzprogramm 57xx-FN1 ausgeliefert.

Aber bleiben wir bei der Standardversion des Druckens und schauen uns zunächst einmal die Druckdatei an.

Drucker

Um DBCS-Daten auf Papier bringen zu können, muss der Drucker selbst natürlich in der Lage sein, die gewünschten Daten zu drucken.

2.10.4.4.1
Seite 2

Druckereinheitenbeschreibung (DEVPRT)

Damit die Druckerhardware vom System IBM i bedient werden kann, wird eine Druckereinheitenbeschreibung benötigt. Sie kann mit dem Befehl CRTDEVPRT erstellt oder mit CHGDEVPRT geändert werden.

Einer der Parameter, die bei dem Erstellen einer Druckereinheitenbeschreibung angegeben werden können und die für DBCS-Drucke relevant sind, ist der Parameter IGCFEAT. Dieser Parameter kann nur für DBCS-unterstützende Einheiten angegeben werden.

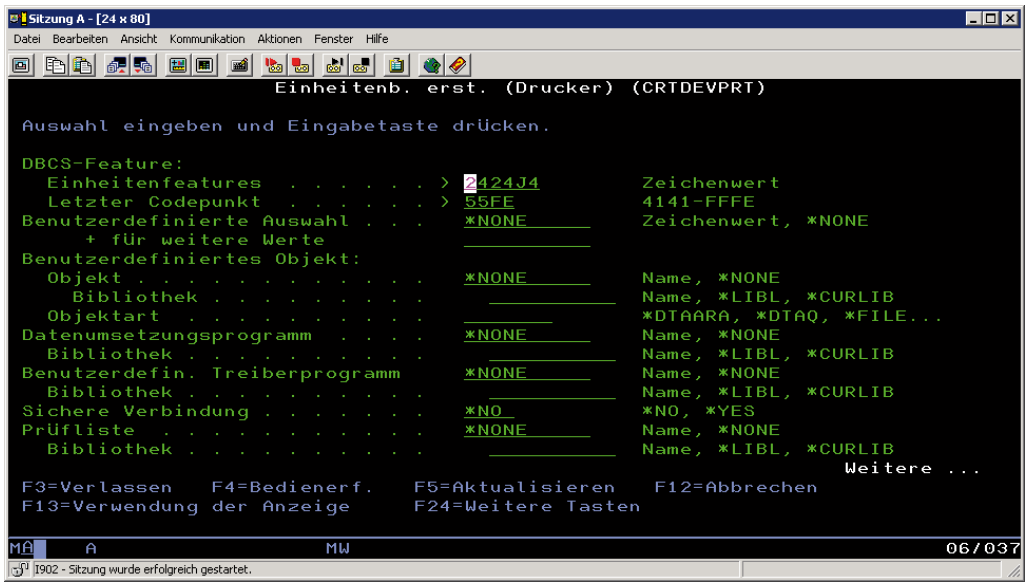
IGCFEAT definiert die zu verwendende DBCS-Tabelle. Damit verbunden ist auch die Definition des letzten Codepunkts, der in jeder DBCS-Zeichensatztabelle angegeben ist. Beim letzten Codepunkt handelt es sich um die definierte Trennung zwischen den Basis-DBCS-Zeichen und den erweiterten DBCS-Zeichen. Die Angaben zum letzten Codepunkt variieren in der Regel von Einheit zu Einheit. Der letzte Codepunkt wird vom System i dazu verwendet, um zu lokalisieren, ob das zu druckende DBCS-Zeichen von der Einheit selbst unterstützt wird oder mit zusätzlichem Aufwand (Rechenleistung) vom System i bereitgestellt werden muss. Das Zeichen muss dann natürlich auf dem System i existieren!

Die Angabemöglichkeiten für IGCFEAT sind:

- SSSS
  - Anzahl der Matrix-Punkte für die Zeichen. Damit wird quasi die Auflösung definiert.
  - 2424 bedeutet beispielweise: 24 Matrix-Punkte in der Höhe und 24 Matrix-Punkte in der Breite.
- L
  - Sprachcode. Derzeit werden unterstützt:
    C = Chinesisch Traditionell
    J = Japanisch
    K = Koreanisch
    S = Chinesisch Einfach
- R
  - Relative Puffer-Größe. Gültige Werte sind:
    0
    1
    2
    4
- Als Wert für den letzten Codepunkt kann der vierstellige Codepunkt des letzten DBCS-Zeichens angegeben werden. Die Angabe dieses Parameters ist optional.

2.10.4.4.2
Seite 2

Nachfolgend finden Sie ein Beispiel für die Definition von IGCFEAT für einen japanischen Einsatz:



IGCFEAT

Druckdatei

Wie zuvor erwähnt, können wir mit einigen Anpassungen eine Druckdatei DBCS-fähig machen.

Ich möchte an dieser Stelle nicht auf die Details einer Druckdatei hinsichtlich deren Erstellung bzw. Verwaltung eingehen, sondern mich bewusst auf die DBCS-Spezifika beschränken.

Zunächst sei erwähnt, dass man die von IBM bereitgestellten StandardDruckdateien nicht in den IBM-eigenen Bibliotheken ändern sollte. Sollten Sie diese anpassen wollen, so empfiehlt sich vorheriges Kopieren und anschließendes Anpassen der Kopie. Die DBCS-Einstellungen können individuell für einzelne Druckdateien oder global für alle Druckdateien genutzt werden. Letzteres ist jedoch nur dann zu empfehlen, wenn wirklich alle Einheiten, die an das System angeschlossen sind, DBCS-fähig sind!

Zunächst muss die Länge eines Felds oder Druckelements in einer Druckdatei definiert werden. Bei der Verwendung von DBCS-Daten ist zu beachten, dass die Länge eines DBCS-Grafikfelds zwischen 1 und 32767 (in älteren OS/400-Releaseversionen 16383) Zeichen betragen kann. Trotz der beachtlichen Länge ist selbstverständlich auch hier die maximale Druckbreite der Druckdatei bzw. des Druckers ein begrenzender Faktor!

Bedenken wir, dass ein Zeichen 2 Bytes lang sein kann (DBCS-Grafikzeichen entsprechend länger), dann müssen wir die Länge in der Druckdatei entsprechend berücksichtigen.

Gehen wir einmal von einem DBCS-Feld mit 3 Zeichen aus, das 1 Shift-out- und 1 Shift-in- Zeichen beinhaltet und über eine Datenlänge von 8 Bytes verfügt: 3 Zeichen * 2 Bytes + Shift-in und Shift out = 8

In einer Druckdatei kennzeichnen wir ein Feld als DBSC-Feld durch die gewünschte Vorgabe in Spalte 35. Hier müssen wir zunächst zwischen DBSC-Zeichen und DBCS-Grafikzeichen unterscheiden. Während für reine DBCS-Zeichen in Spalte 35 ein „O“ anzugeben ist, müssen wir DBCS-Grafikzeichen mit einem „G“ in Spalte 35 definieren.

Auf diese Weise können sowohl DBCS- als auch alphanumerische Daten in ein und demselben Feld abgebildet werden.

2.10.4.4.3
Seite 2

Wenn DBCS in einer Druckdatei (DDS) verwendet wird, dann können die folgenden DDS-Schlüsselwörter verwendet werden:

- BARCODE
- BLKFOLD
- CDEFNT
- CHRID
- COLOR
- CPI
- CVTDTA
- DATE
- DATFMT
- DATSEP
- DFT
- DLTEDT
- EDTCDE
- EDTWRD
- FLTFIXEC
- FLTPCN
- FNTCHRSET
- FONT
- IGHLIGHT
  MSGCON
- PAGNBR
- TIME
- TIMFMT
- TIMSEP
- TRNSPY

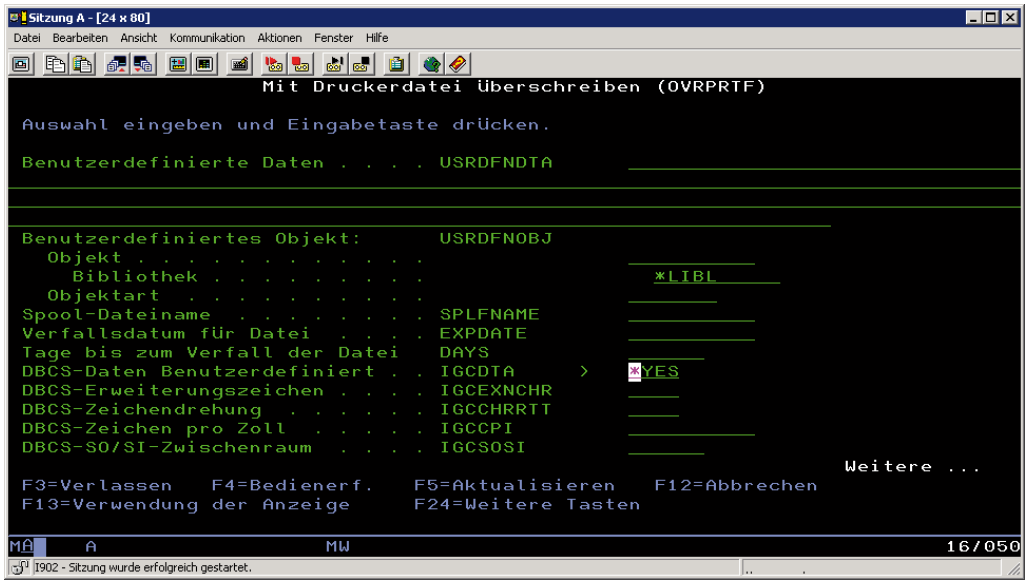
Druckdateien und DBCS

Neben der Kennzeichnung der DBCS-Felder in der Druckdatei können auf globaler Ebene die Definitionsvorgaben für das DBCS-Drucken vorgenommen werden. Dazu finden sich auf Ebene der Drucker- und Anzeigedateien unterschiedliche Parameter, die zum Teil in Koexistenz oder alternativ zu den DDS-Schlüsselwörtern zum Einsatz kommen können.

ICGDTA

Der Parameter ICGDTA(*YES) auf Druckdateiebene definiert, dass die Druckdatei DBCS unterstützt.

OVRPRTF FILE(printerfilename) IGCDTA(*YES)



OVRPRTF

ICGDTA steht z. B. auch bei dem Erstellen einer Druckdatei (CRTPRTF) oder bei der Änderung einet Druckdatei (CHGPRTF) als optionaler Parameter zur Verfügung.

Nachfolgend ein Beispiel für das Erstellen einer DBCS-fähigen Druckdatei:

```
CRTPF FILE(QTEMP/TEMPFILE) RCDLEN(134) +
        MBR(*FILE) IGCDTA(*YES) +
        SHARE(*NO) + LVLCHK(*NO)
```

2.10.4.4.3
Seite 4

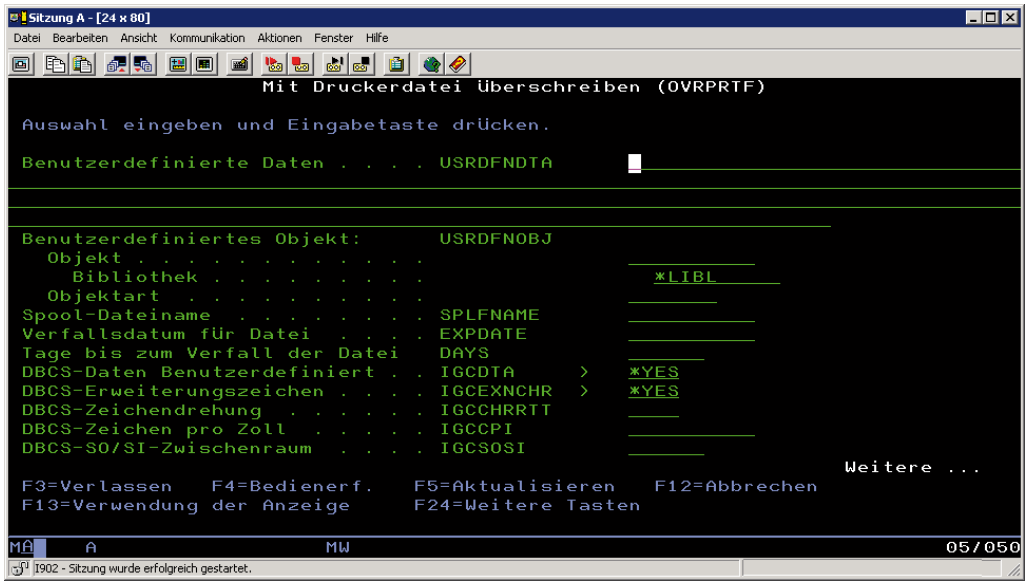
IGCEXNCHR

Für das Drucken von erweiterten DBCS-Zeichen muss der Parameter IGCEXNCHR(*YES) angegeben werden. Wird er verwendet, dann müssen unter Umständen zusätzliche Lizenzprogrammoptionen auf dem System i installiert werden.

Wollen Sie z. B. japanische Druckausgaben auf dem Drucker 5583 erzeugen, dann muss die Kanji-Druckfunktion vom Advanced DBCS Printer Support-Lizenzprogramm installiert sein (57xx-AP1). Neben dem 5583-Drucker wird u. a. auch der japanische 5337-Seitendrucker unterstützt.

Der Parameter IGCEXNCHR kann für Druckdateien (PRTF) oder Anzeigedateien (DSPF) eingesetzt werden und steht u. a. mit den folgenden Befehlen bereit:

- Erstellen der Druckdatei (CRTPRTF)
- Ändern der Druckdatei (CHGPRTF)
- Überschreiben der Druckdatei (OVRPRTF)
- Erstellen einer Anzeigedatei (CRTDSPF)
- Ändern einer Anzeigedatei (CHGDSPF)
- Überschreiben einer Anzeigedatei (OVRDSPF)



IGCEXNCHR

IGCCCHRRTT
Mit IGCCHRRTT lassen sich DBCS-Zeichen 90 Grad gegen den Uhrzeigersinn drehen.

Gedrehte DBCS-Zeichen werden vertikal ausgedruckt. Die Unterstützung beschränkt sich auf reine DBCS-Daten – alphanumerische Daten werden nicht gedreht!

Gültige Werte sind:

- *YES
  DBCS-Zeichen werden beim Drucken um 90 Grad gegen den Uhrzeigersinn gedreht.
- *NO
  DBCS-Zeichen werden beim Drucken nicht gedreht.

Dieses Schlüsselwort kann nur in Verbindung mit 5553-Druckern oder der Angabe AFP(*YES) eingesetzt werden.

IGCCHRRTT kann nur für DBCS-Felder (O oder G) oder mit DBCS in konstanten Feldern genutzt werden.

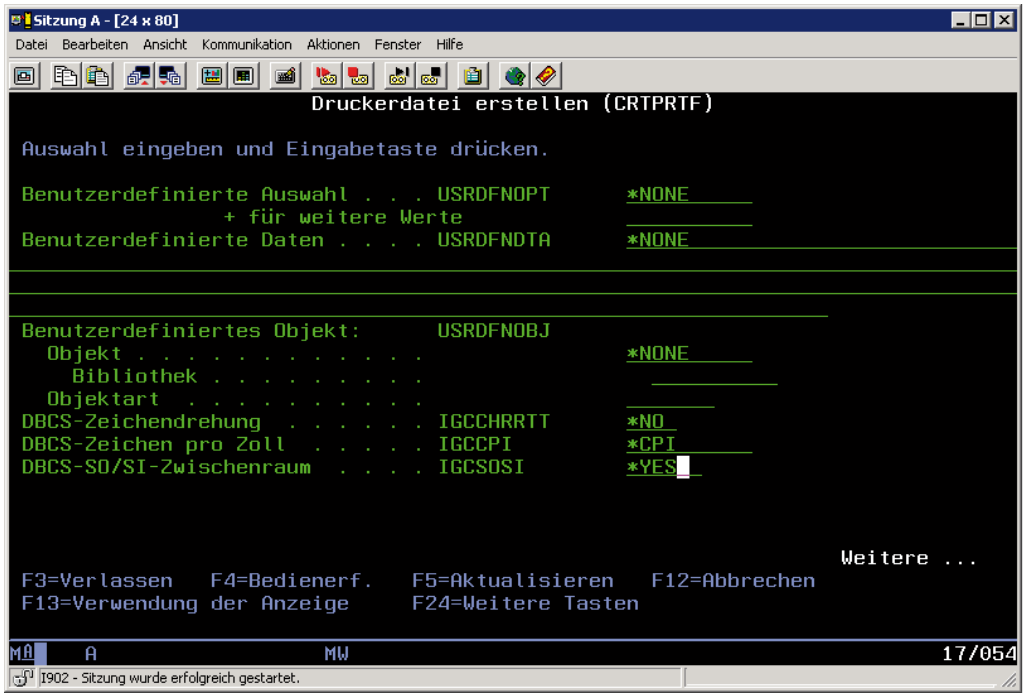
IGCCHRRTT ist neben dem Einsatz auf Druckdatei- oder Anzeigedateiebene auch als Schlüsselwort in DDS verfügbar. Wird IGCCRRTT sowohl auf DDS als auch auf Dateiebene definiert, dann werden die in der DDS enthaltenen IGCCHRRTT-Einträge ignoriert.

IGCSOSI
Mit IGCSOSI wird bestimmt, ob die DBCS-Steuerzeichen gedruckt werden sollen.

Gültige Werte sind:

- *YES
  – DBCS-Steuerzeichen werden als Leerzeichen gedruckt.
- *NO
  – Steuerzeichen werden für den Druck entfernt – in diesem Fall wird auch kein „Ersatzzeichen“ gedruckt.
- *RIGHT
  – Kein Druck von DBCS-Startzeichen.
  – DBCS-Endezeichen werden durch jeweils zwei Leerstellen ersetzt.

2.10.4.4.3
Seite 6



IGCSOSI

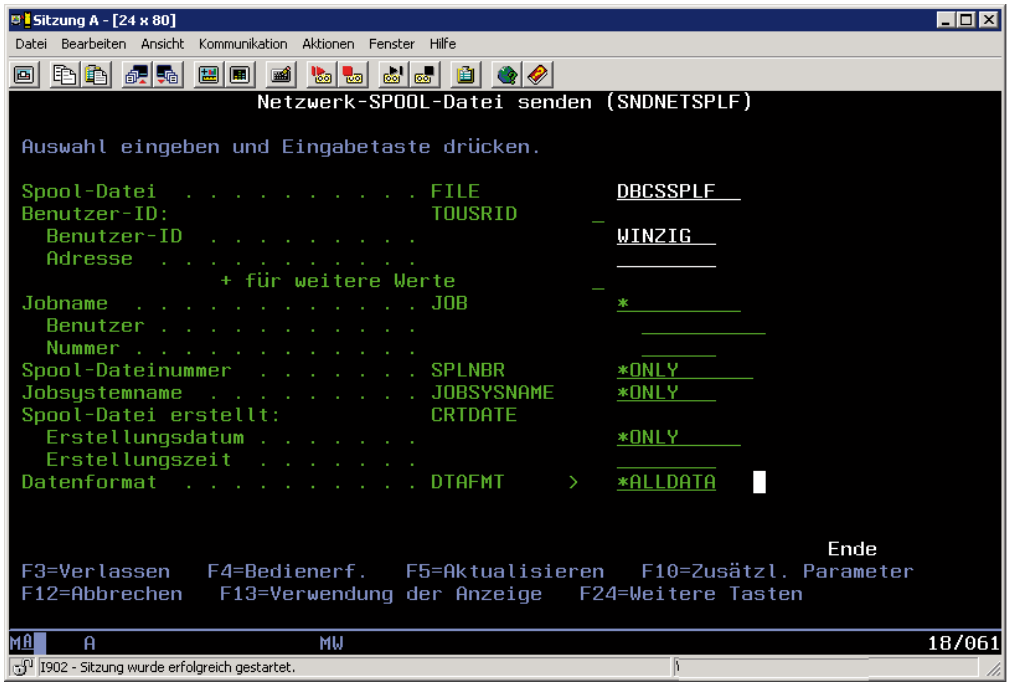
ACHTUNG!

Bei der Konvertierung von Druckausgaben in PDF-Dateien sind zusätzliche Schritte/Einstellungen zu beachten. Per Standard werden SBCS-Zeichensätze in Windows ANSI umgesetzt. Dabei werden die Standard-PDF-Fonts verwendet. Diese Umsetzungsmethode eignet sich für die gängigen SBCS-Zeichensätze, wie die für Westeuropa, USA, Südamerika und Latin-1.

Bei der Umsetzung von DBCS werden andere PDF-Zeichensätze benötigt. Informationen zu diesen Zeichenumsetzungen findet man auf den Webseiten von Adobe.

Die Schlüsselwörter *CODEPAGE oder *IGCCODEPAGE können für die gezielte Umsetzung eingesetzt werden.

Ein weiterer Punkt, der beim Einsatz von DBCS und Druckausgaben von Bedeutung sein kann, ist die Möglichkeit, Spooldateien zu senden. Vielleicht kennen Sie den Befehl SNDNETSPLF. Mit diesem Befehl lassen sich Spooldateien von einem System i auf ein anderes übertragen. Auch hier wird DBCS unterstützt – aber nur dann, wenn der Wert *ALLDATA für den Parameter DTAFMT angegeben wird. Ein Beispiel zeigt die folgende Abbildung.


Sitzung A - [24 x 80]
Datei Bearbeiten Ansicht Kommunikation Aktionen Fenster Hilfe
Netzwerk-SPOOL-Datei senden (SNDNETSPLF)
Auswahl eingeben und Eingabetaste drücken.
Spool-Datei . . . . . FILE DBCSSPLF
Benutzer-ID: TOUSRID
Benutzer-ID . . . . . WINZIG
Adresse . . . . . .
+ für weitere Werte
Jobname . . . . . JOB *
Benutzer . . . . . .
Nummer . . . . . .
Spool-Dateinummer . . . . . SPLNBR *ONLY
Jobsystemname . . . . . JOBSYSNAME *ONLY
Spool-Datei erstellt: CRTDATE
Erstellungsdatum . . . . . *ONLY
Erstellungszeit . . . . . .
Datenformat . . . . . DTAFMT > *ALLDATA
Ende
F3=Verlassen F4=Bedienerf. F5=Aktualisieren F10=Zusätzl. Parameter
F12=Abbrechen F13=Verwendung der Anzeige F24=Weitere Tasten
MA A MW 18/061
I902 - Sitzung wurde erfolgreich gestartet.

SNDNETSPLF

2.10.4.4.3

Seite 8

Performancemanagement V6R1 und i7.1

Nicht selten stellt man sich die Frage: „Was macht der Server eigentlich gerade?“ „Warum dauert das so lang?“ Gerade noch liefen die Applikationen schnell und reibungslos und plötzlich scheint das System einzuschlafen. Manchmal passiert es auch, dass neue Applikationen installiert werden und plötzlich verändert sich das Laufzeitverhalten des Servers. Schön wäre es, feststellen zu können, welche Ressourcen die neuen Programme benötigen, um entsprechende Maßnahmen einzuleiten oder präventiv tätig werden zu können.

Ein anderes Phänomen, das Sie vielleicht auch kennen, sind die sogenannten „Spitzenzeiten“. Den ganzen Tag läuft alles reibungslos. Doch immer zwischen 11:00 und 12:00 Uhr steigen die Antwortzeiten in einem unerträglichen Maß. „Welche Jobs beanspruchen in dieser Zeit überproportional Systemressourcen?“ Wollen Sie Antworten auf Ihre Fragen, erhalten Sie häufig die Empfehlung, die aktuelle Hardware auszutauschen. Befriedigt Sie diese Antwort?

Sicherlich wird der Tag kommen, wo Hardware ausgetauscht werden muss. Trendanalysen können hier helfen. Sie zeigen auf, wann der Server vermutlich erweitert oder eventuell ausgetauscht werden muss. Investitionen lassen sich so besser planen.

Aber was machen Sie, wenn die Hardware ausgetauscht wurde und ihr Server wider Erwarten nur genauso schnell ist wie zuvor? Oder wenn der Server – trotz neuer Hardware – sogar langsamer wird? Die vielfältigen Konfigurationsmöglichkeiten, die wir heute aufgrund von Virtualisierung und dem Einsatz diverser Storagesysteme kennen, bergen gewisse Tücken.

Die laufende Überwachung der Systemperformance ist heute wichtiger denn je. Noch nie hat sich das Tagesgeschäft so schnell verändert wie zurzeit: Neue Standorte werden angebunden, neue Applikationen installiert, bestehende Applikationen verändert, native Datenbankzugriffe werden auf SQL umgestellt, Webportale erhalten Zugriff auf den System-i-Server. Aber wenn sich die Softwareanforderungen derartig schnell ändern, muss das System ggf. ebenfalls angepasst werden. Die Zeiten, in denen man in regelmäßigen Zyklen Hardware einfach unabhängig von den Applikationen austauschen konnte, sind vorbei.

Die Leistungsüberwachung der Server bzw. der Partitionen ist ein komplexer Prozess, der zunächst einmal voraussetzt, dass Sie grundsätzlich das Zusammenspiel zwischen Software und Hardware kennen. Als Hilfestellung hat IBM mit Release V6.1 vielfältige Analysetools integriert, die integraler Bestandteil der Betriebssystemlizenz sind. Diese Tools erleichtern das Performancemanagement des System-i-Servers erheblich.

In erster Linie geht es hierbei um die Überwachung eines bestehenden Systems, damit Ihre Anwender reibungslos arbeiten können, denn sie haben sicherlich eine gewisse Erwartungshaltung. Sie erwarten, dass das System verfügbar ist und ihre Eingaben in einem angemessenen Zeitraum verarbeitet werden. Diese Anforderung sollten Sie in messbaren Zahlen ausdrücken können:

- Wie hoch soll die durchschnittliche Antwortzeit der interaktiven Eingaben sein?
- Wie viele interaktive Transaktionen erwarten Sie pro Stunde oder an einem Tag?
- Wie viele Batchjobs laufen? Und wie hoch sollte deren Antwortzeit sein?

Sie benötigen also ein Anforderungsprofil für Ihren Server. Derartige Anforderungsprofile erstellt man aus unterschiedlichen Gründen:

- Als Vergleichsbasis, wenn Performanceprobleme auftreten
- Als Vergleichsbasis – wenn das System verändert wird – bzgl. neuer Applikationen, Releasewechsel oder Hardware
- Als Ausgangsbasis für die Planung eines neuen Servers

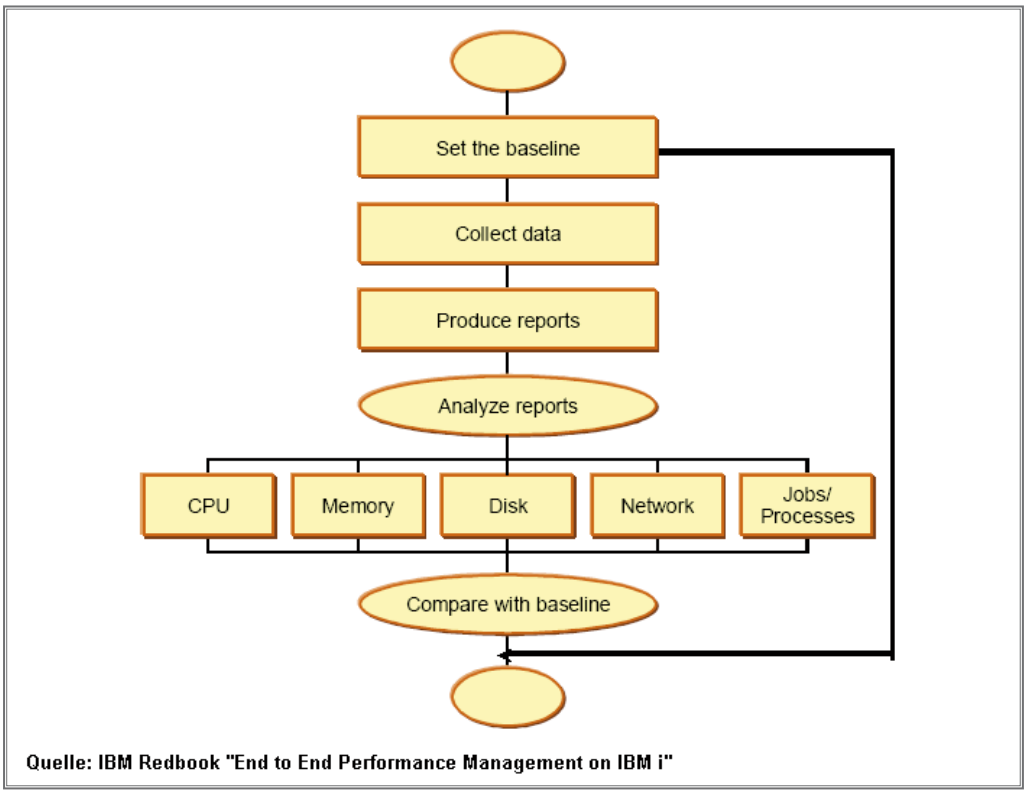
Das Basisprofil müssen Sie sich allerdings selbst erarbeiten. Sie können es weder kaufen noch von einem anderen Unternehmen ableiten, da jeder Server bzgl. der Applikationen und der Clientanzahl unterschiedlich ist. Und auch die Erwartungshaltung der Anwender bzgl. der Antwortzeiten variiert von Unternehmen zu Unternehmen.

Und so könnte Ihr Anforderungsprofil aussehen:

- Durchschnittliche Antwortzeit 0,75 Sekunden
- Tagesabschlüsse (Batch) sollen in vier Stunden abgeschlossen sein
- Durchschnittliche CPU-Auslastung im Tagesgeschäft sollte unter 75 % liegen
- Plattenzugriffe unter 70 %
- Ca. 20.000 interaktive Transaktionen am Tag
- Systemverfügbarkeit 24 x 7

Treten anschließend Probleme auf, können Sie aktuelle Leistungsdaten erheben und verschiedene Reports oder grafische Auswertungen ziehen, die es erlauben, CPU, Hauptspeicher, Plattenzugriffe, Netzwerk und/oder Jobs bzw. Programme zu analysieren. Als Vergleichsdaten haben Sie immer die Basisauswertung, so dass Sie schnell erkennen können, welche Leistungswerte sich im System verändert haben.

Die folgende Grafik verdeutlicht diesen Analyseprozess:



Quelle: IBM Redbook "End to End Performance Management on IBM i"

Der Prozess „Leistungsüberwachung“

2.11
Seite 4

Die Erfassungsservices

Nachdem Sie Ihre Anforderungen formuliert haben, müssen Sie die Leistungswerte Ihres Servers überwachen. Grundsätzlich stehen hierfür zwei Möglichkeiten zur Verfügung:

- Verwendung entsprechender CL-Befehle
- Verwendung der Erfassungsservices

Für die grundlegende Performance-Analyse bietet sich die Leistungserfassung an, die Sie über System i Navigator, die Weboberfläche „IBM Systems Director Navigator for i5/OS“ oder das 5250-Interface starten können. Dieser Erfassungsservice erlaubt uns die kontinuierliche und automatische Datenerfassung, ohne das System spürbar zu belasten. IBM empfiehlt, die Leistungsdaten eines System-i-Servers an sieben Tagen über 24 Stunden zu erfassen. So können Sie schnell reagieren, wenn unerwartete Probleme auftauchen und haben stets genügend Analysedaten zur Verfügung. Diese Daten kann im Übrigen auch IBM nutzen, wenn Sie das Problem nicht selbst lokalisieren können.

Der Erfassungsservice sammelt Daten in zyklischen Intervallen von 15 Sekunden bis zu einer Stunde. Während jedes Intervalls werden Informationen über die Verwendung der Systemressourcen gesammelt, einschließlich der Plattenzugriffe, IOPs, Busse, Pools und Netzwerkverbindungen. Allerdings sammeln die Erfassungsservices keine Stack-Informationen, keine Informationen über Aktivierungsgruppen, Sperrkonflikte oder SQL-Statements. Muss die Analyse in diesen Bereichen erfolgen, stehen weiterführende Analysetools wie „Jobwatcher“ und „Diskwatcher“ zur Verfügung.

Die Leistungsdaten der Erfassungsservices werden in „Management Collection Objects (*MGTCOL)“ gespeichert. Die Größenbegrenzung dieser Objekte liegt mittlerweile im Teraspace-Bereich, so dass Sie problemlos auch große Datenmengen speichern können. Außerdem ist die Auswertung der Daten aufgrund der Management Collection-Objekte unabhängig vom Release des jeweiligen Servers, denn Sie können die Objekte problemlos auf andere Server übertragen. Langwierige Datenbankkonvertierungen gehören damit der Vergangenheit an. Erst wenn Sie Reports erstellen wollen, um die Daten zu analysieren, müssen Sie die Datenbankdateien mit dem CL-Befehl CRTPFRDTA (Create Performance Data) erstellen. Seit V6R1 können die Datenbankdateien allerdings auch automatisch während der Erfassung erstellt werden. Doch dazu später mehr. Der Erfassungsservice benötigt trotz paralleler Datenbankerstellung wenig Systemressourcen, so dass Sie die Erfassungsservices immer als Hintergrundjob mitlaufen lassen können, ohne die Leistung Ihres Servers negativ zu beeinflussen.

2.11.1
Seite 2

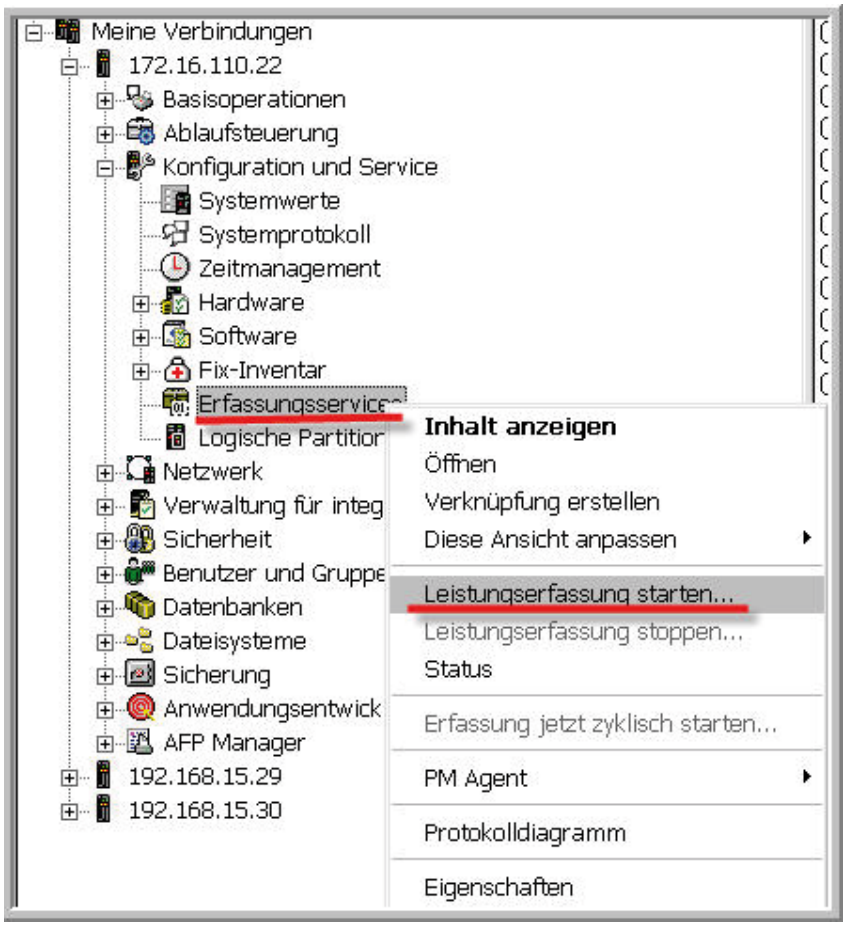
Erfassungsservice konfigurieren – System i Navigator

Die Erfassungsservices sind bereits seitens IBM vorkonfiguriert. Aber Sie können die Konfiguration Ihren Bedürfnissen entsprechend anpassen. Diese Konfiguration beinhaltet u. a.:

- Festlegung der Bibliothek, in der die Analysedaten gespeichert werden
- Festlegung der zyklischen Wiederholungen
- Auswahl der Erhebungsdaten anhand von Analyseprofilen
- Festlegung der Aufbewahrungszeiten
- usw.

Für die Konfiguration der Erfassungsservices stehen verschiedene Interfaces zur Verfügung, die sich teilweise ein wenig in ihren Möglichkeiten unterscheiden:

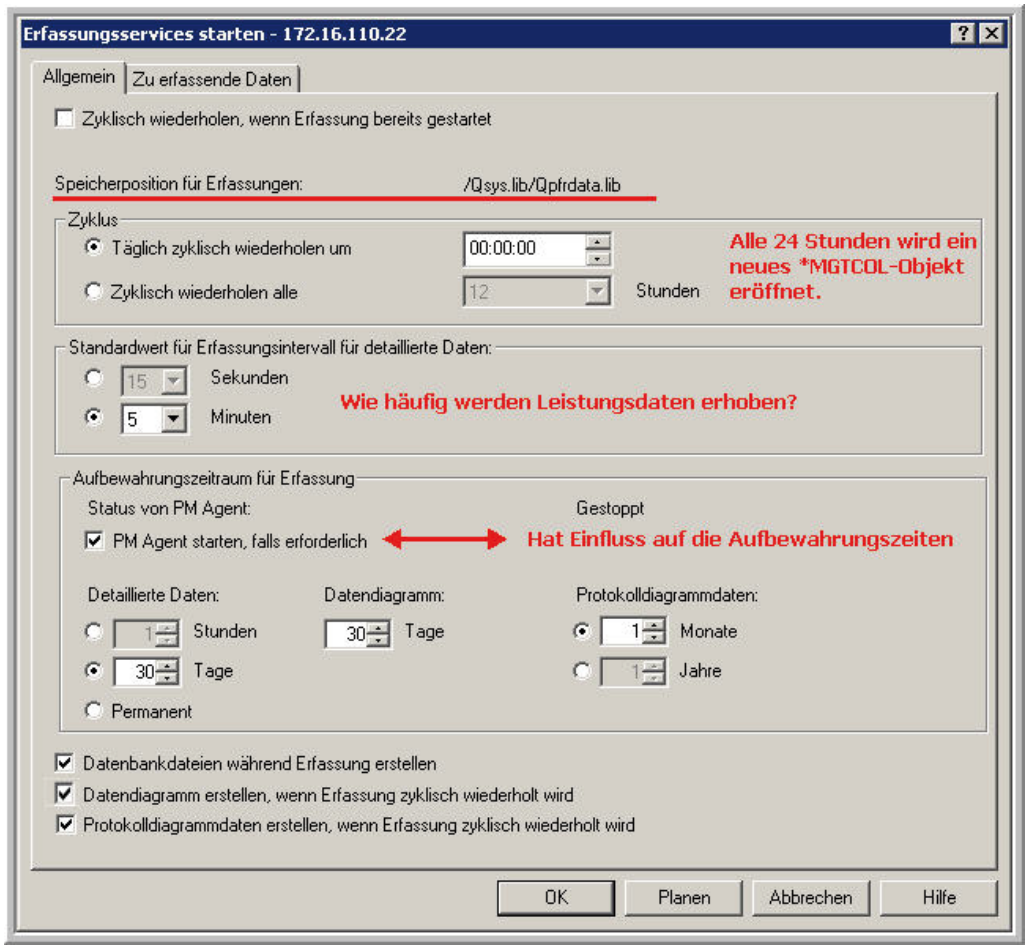
Konfigurieren und starten wir als Erstes die Erfassungsservices über den System i Navigator. Öffnen Sie dazu die Verbindung zu Ihrem System und wählen Sie im Kontextmenü des Eintrags „Konfiguration und Service“ ➜ „Erfassungsservice“ ➜ „Leistungserfassung starten“.



„Leistungserfassung starten“

2.11.1.1
Seite 2

Im dann folgenden Dialog hinterlegen wir allgemein gültige Eigenschaften.



„Erfassungsservices starten" – Allgemein

Hierzu gehören die zyklischen Wiederholungen der Erfassungsservices, das Erfassungsintervall und die Aufbewahrungszeit. Wenn eine Erfassung zyklisch wiederholt wird, werden keine weiteren Daten zum Management Collection Object, dem Erfassungsobjekt, hinzugefügt. In meinem Beispiel wird stets um Mitternacht ein neues Erfassungsobjekt gestartet. So ist sichergestellt, dass die einzelnen Erfassungen nicht zu umfangreich werden. Alternativ hätte ich das Erfassungsobjekt auch in Stundenintervallen neu starten können.

Der nächste Eintrag „Standardwert für Erfassungsintervall für detaillierte Daten" enthält das Zeitintervall für die jeweiligen Messungen. Hier haben Sie die Wahl zwischen Sekunden und Minuten. Der Standardwert beträgt 15 Sekunden. Diesen Wert können Sie durchaus erhöhen. Ohne konkrete Probleme ist ein Erfassungsintervall von fünf Minuten in der Regel ausrei-

chend. Der hier hinterlegte Richtwert gilt ohnehin nur dann, wenn einer Erfassungskategorie – die wir auf der zweiten Seite noch wählen werden – keine spezifische Intervallgröße seitens IBM zugeordnet wurde. Nur wenn kein Konflikt zwischen den spezifischen Intervallgrößen einzelner Kategorien und dem Standardintervall vorliegt, wird das Standardintervall verwendet. Ich entscheide mich an dieser Stelle für ein Standardintervall von fünf Minuten.

Abschließend müssen wir noch bestimmen, wie lange die Erfassungsobjekte (*MGTCOL) in der Standardbibliothek QPFRDATA für Analysen zur Verfügung stehen sollen. Anhand der hinterlegten Aufbewahrungszeit kann anschließend ein Verfalldatum ermittelt werden. „Abgelaufene" Auswertungen werden vom Server-Job QYPSPFRCOL automatisch vom System entfernt, wenn ein neuer Erfassungszyklus startet.

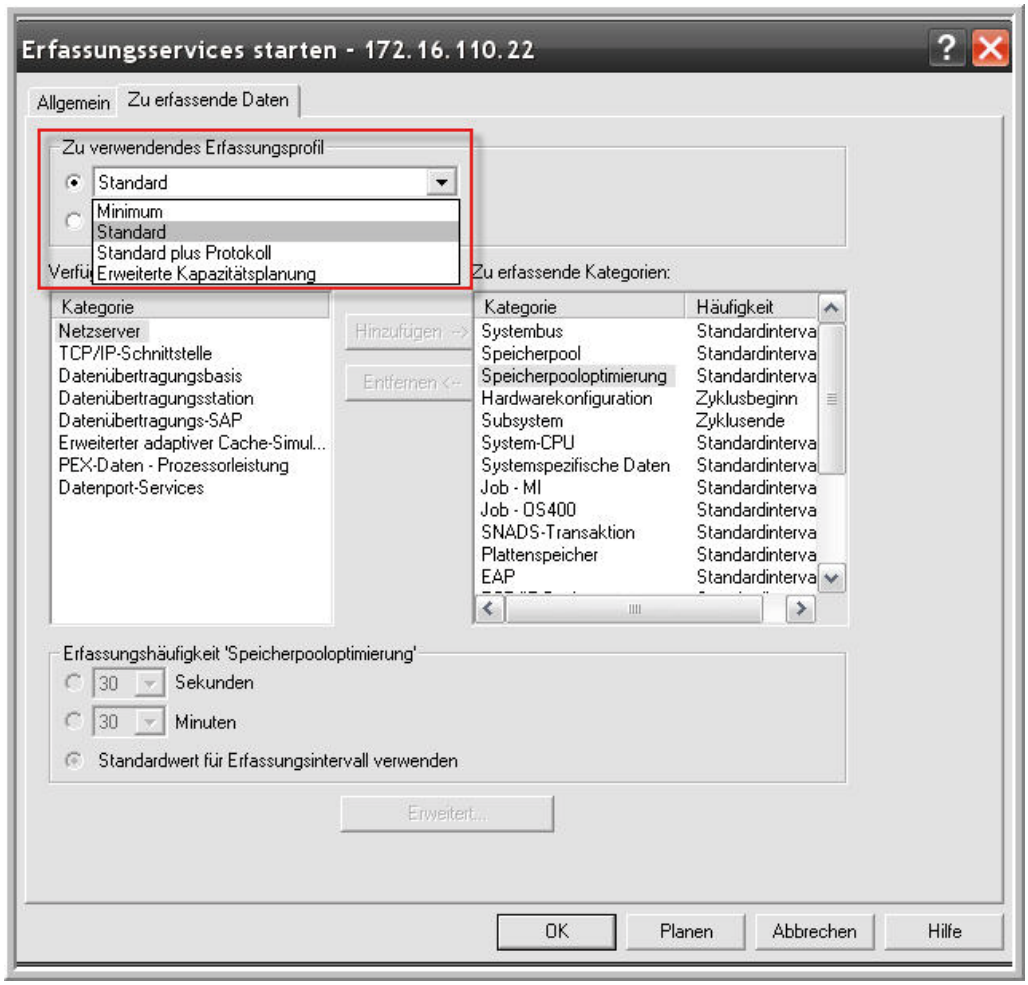
Achten Sie darauf, dass für die Datendiagramme und Protokolldiagrammdaten individuelle Aufbewahrungszeiträume gewählt werden. Datendiagramme werden benötigt, um Analysedaten grafisch darzustellen. Die maximalen Aufbewahrungszeiten sind davon abhängig, ob Sie erlauben, dass PM Agent bei Bedarf auf Ihrem Server gestartet wird. PM Agent ist ein IBM-Tool oder besser gesagt eine kostenlose Dienstleistung der IBM. Der PM Agent fasst Leistungs- und Kapazitätsinformationen aus dem Server zusammen. Sobald die Daten von den Erfassungsservices erfasst wurden, verarbeitet PM Agent die Daten und bereitet sie für die Übertragung an IBM zur späteren Analyse und Berichtsgenerierung vor. Vor Release i7.1 war es zwingend erforderlich, dass PM Agent gestartet war, wenn Datendiagramme länger als sieben Tage aufbewahrt werden sollten. Seit Release i7.1 ist es ausreichend die Zustimmung zum Start von PM Agent zu erteilen, falls dies erforderlich sein sollte.

Mit dem Eintrag „Protokolldiagrammdaten" bestimmen Sie, wie lange Detailinformationen verfügbar sind. Diese Detailinformationen können Sie in den grafischen Auswertungen anzeigen und erhalten so zu spezifischen Datenpunkten Detailinformationen. Protokollinformationen lassen sich nur archivieren, wenn PM Agent gestartet werden darf.

Jetzt müssen Sie noch festlegen, ob die Datenbankdateien, die für diverse PT1-Reports benötigt werden, sowie die Datendiagramme und Protokolldiagrammdaten bereits während der Leistungserfassung erstellt werden sollen. Die Erstellung beeinflusst nicht die Performance Ihres Systems. Allerdings benötigen die Dateien Festplattenplatz. Die Informationen können auch später jederzeit erstellt werden. Sie müssen daher abwägen, ob Sie bereit sind, ein wenig Wartezeit in Kauf zu nehmen, um Plattenplatz zu sparen.

2.11.1.1
Seite 4

Wir wechseln jetzt auf die zweite Registerseite „Zu erfassende Daten“.



Das Register „Zu erfassende Daten“

Anhand der Abbildung oben können Sie erkennen, dass es bereits definierte Datenprofile gibt. Mittels dieser Profile können Sie die benötigten Erfassungswerte sehr einfach selektieren.

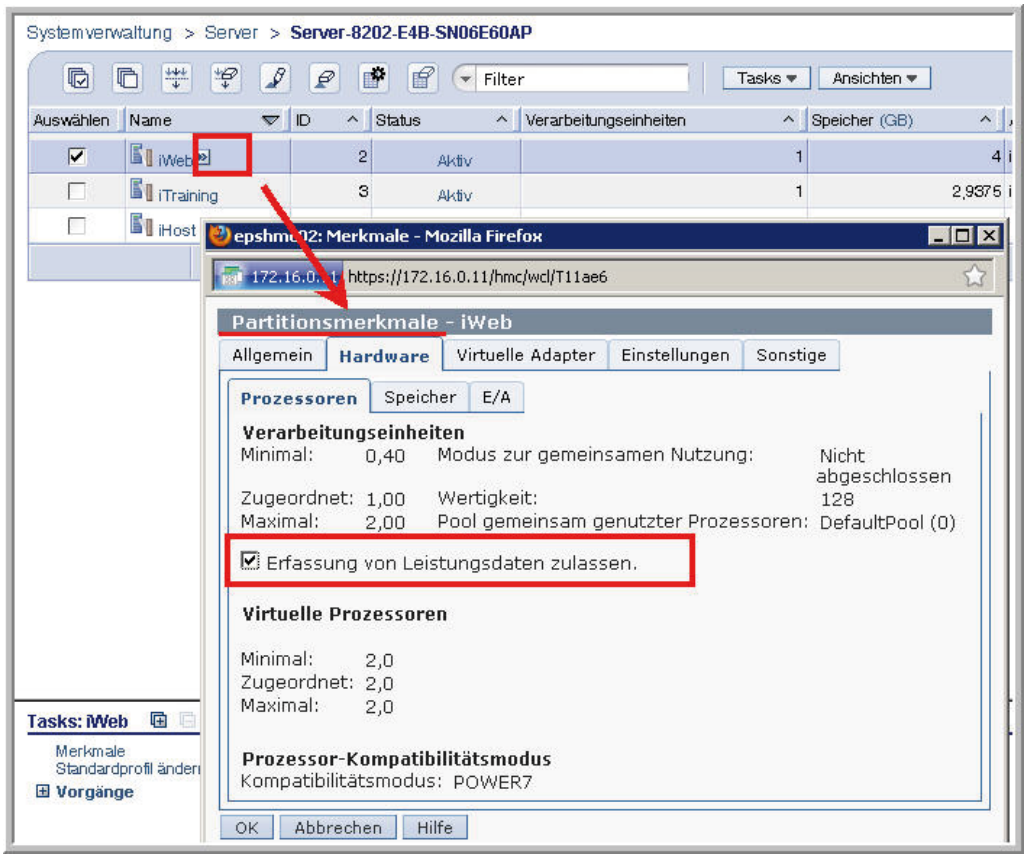
Die Mindestanforderungen für eine spätere Analyse sind erfüllt, wenn Sie sich für das Profil „Minimum“ entscheiden. Dabei werden Daten über die Auslastung der Systembusse, der Speicherpools und der CPU erhoben. Zusätzlich erhalten Sie Informationen über die aktiven Jobs Ihres Systems, die Auslastung des Plattenspeichers und die Verwendung der E/A-Prozessoren.

Das „Standardprofil“ geht über die Erfassung dieser Daten hinaus, indem zusätzlich Informationen bzgl. Speicherpooloptimierung, Subsystem, SNADS-Transaktionen, lokale Antwortzeiten, APPN, SNA, Basisinformationen zu TCP/IP und Benutzertransaktionen erfasst werden. Wenn Sie einen Domino-Server oder das Lizenzprogramm „IBM HTTP Server für i5/OS“

auf dem System installiert haben, werden auch dafür Leistungsdaten erhoben. Das Standardprofil enthält somit alle notwendigen Datenkategorien, um die späteren Auswertungen mit den System-i-Performance-Tools durchzuführen.

Benötigen Sie zusätzlich zu den Standardkategorien Informationen über Ihre logischen Partitionen, den Netzwerkserver (Netserver) oder die TCP/IP-Schnittstelle, müssen Sie das „Standard plus Protokoll“ wählen. Manchmal müssen jedoch gewisse Voraussetzungen erfüllt sein, damit die Daten auch korrekt erfasst werden:

Sie können beispielsweise problemlos die Leistungsdaten partitionierter Systeme erfassen. Die LPAR-Daten werden in den Datenbankdateien QAPMLPARH gespeichert. Wenn Sie diese Daten mit den Inhalten der Datei QAPMSYSPRC, die die Systemprozessordaten enthält, kombinieren, erhalten Sie eine Auswertung der verschiedenen LPARs. Alternativ steht als PT1-Auswertung der Bericht „Physical System“ zur Verfügung. Voraussetzung für die Erfassung der LPAR-Informationen ist, dass Sie die Leistungserfassung für die LPARs erlauben. Hierzu müssen Sie in den LPAR-Eigenschaften (HMC) allerdings explizit die Leistungserfassung gestatten:

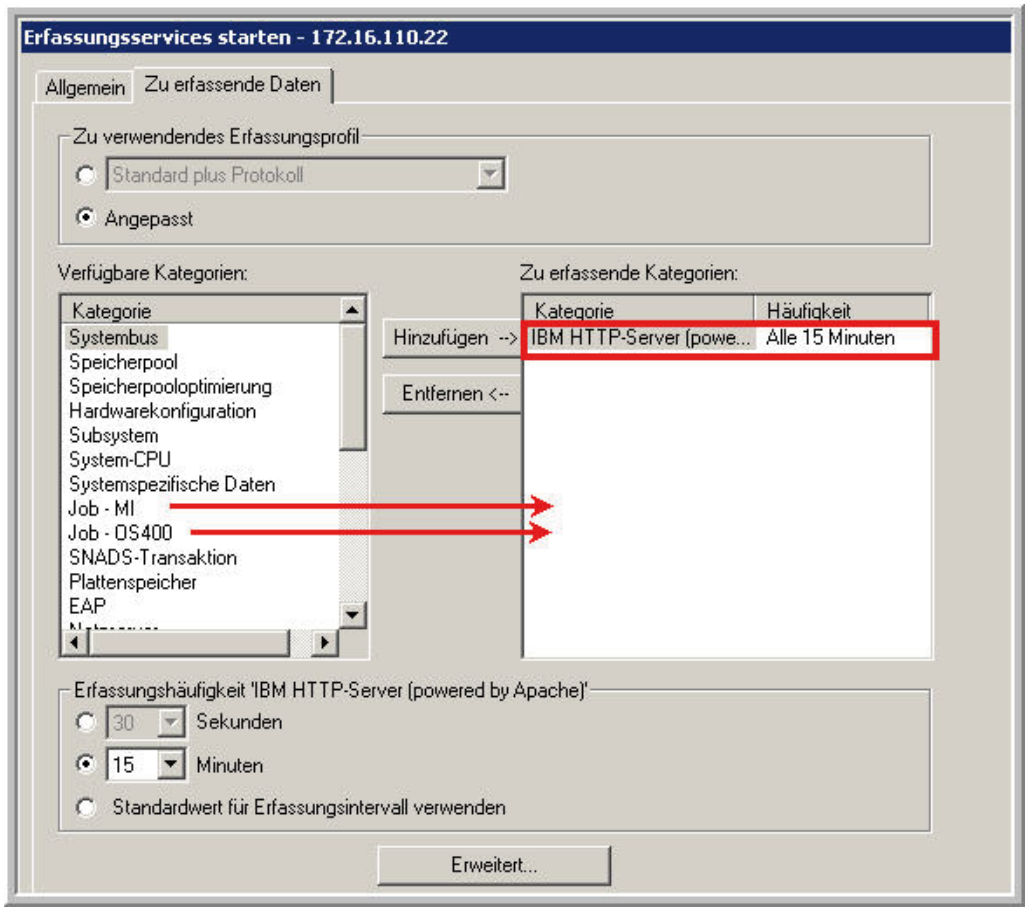


Leistungsdaten für LPARs zulassen

2.11.1.1
Seite 6

Das letzte Standardprofil „Erweiterte Kapazitätsplanung“ ist nicht oder nur in Ausnahmen zu empfehlen: Es enthält alle Informationen des „Standard plus Protokolls“ und zusätzlich Daten für den Performance Explorer (PEX). Der Performance Explorer wird eingesetzt, wenn Sie sehr detaillierte Informationen über ein spezielles Programm oder eine bestimmte Systemressource benötigen. Für die globale Analyse sind diese Daten zu detailliert und daher nicht zu empfehlen.

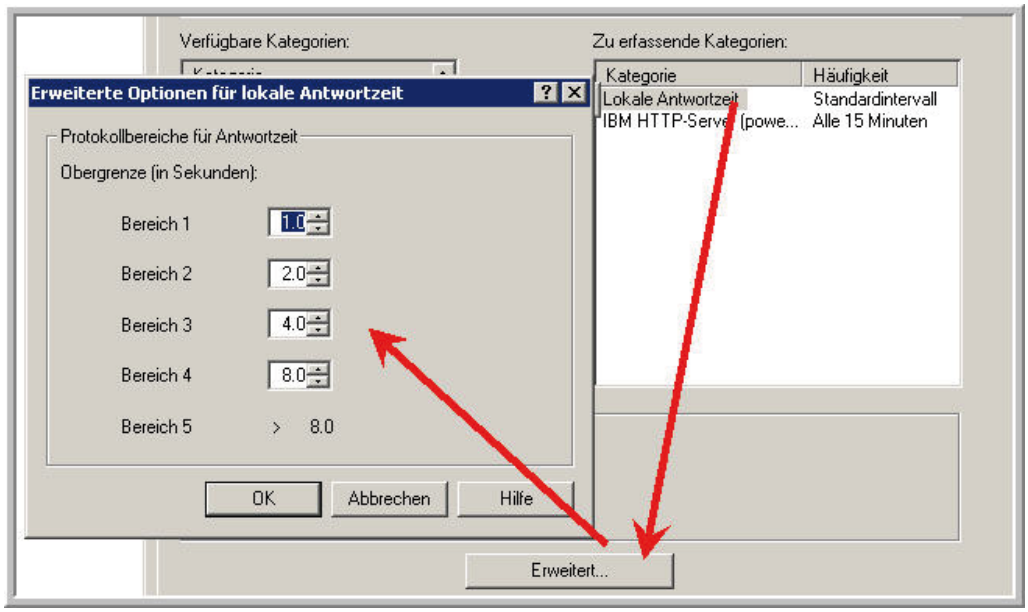
Und falls keines der Profile Ihren Vorstellungen entspricht, können Sie bei Bedarf auch ein benutzerdefiniertes Profil erstellen und die zu erfassenden Kategorien aus einer Liste der verfügbaren Datenkategorien individuell auswählen:



Benutzerdefinierte Profile

Als Erstes fällt auf, dass es hier die vorselektierte Kategorie IBM HTTP-Server gibt. Die Erfassungsservices können Performancedaten der Websphere Application Server-Jobs erfassen. Die meisten Informationen kommen dabei von „Performance Monitoring Infrastructure (PMI)“. PMI ist das zentrale Monitoring Tool für den Websphere Applikation Server und soll beim Tuning des Applikation Servers helfen. Gut ist, dass diese Leistungsdaten innerhalb der Erfassungsservices verfügbar sind, so dass Sie sich zunächst einmal nicht mit unterschiedlichen Analysetools befassen müssen. Allerdings könnte es erforderlich sein, dass Sie in der IBM Web Adminstration for i Console die PMI-Check-Box aktivieren müssen, damit Sie entsprechende Leistungsdaten erhalten.

Grundsätzlich können Sie jede verfügbare Kategorie für Ihr angepasstes Profil wählen und dabei sogar für die verschiedenen Kategorien individuelle Zeitintervalle auswählen. Außerdem stehen für einige Kategorien erweiterte Parameter zur Verfügung. Wählen Sie einfach mal die Kategorie „Lokale Antwortzeit“.



Benutzerdefinierte Profile – Erweiterte Parameter

Wenn Sie diese Kategorie auswählen, können Sie auf den Button „Erweitert“ klicken und weitere Einstellungen vornehmen. Diese Möglichkeit existiert für einige, aber nicht für alle Kategorien. Schauen Sie sich die verschiedenen Kategorien und deren erweiterte Einstellungen an!

Eines müssen Sie beim Zusammenstellen der individuellen Kategorien jedoch bedenken: Die Kategorien „Job – MI“ und „Jobs – Operating System“ sind Voraussetzung für die Anzeige der Protokolldiagrammdaten!

2.11.1.1
Seite 8

Abschließend eine kurze Übersicht der zur Verfügung stehenden Standardkategorien:

- System Bus
- Memory Pool and Pool Tuning
- Hardware Configuration
- Subsystem
- System CPU
- System level data
- Jobs – MI Tasks und Threads
- Jobs – Operating System
- Disk Storage
- IOP
- Local Response Time
- Communication (Base, Station, SAP)
- APPN
- ARM
- SNA
- SNADS Transaction
- TCP/IP Base
- TCP/IP Interface
- Integrated xSeries Server (IxS)
- Extended Adaptive Cache
- User-defined Transactions
- Domino
- HTTP Server (Powered by Apache)
- Data Port Services
- LPAR
- WAS

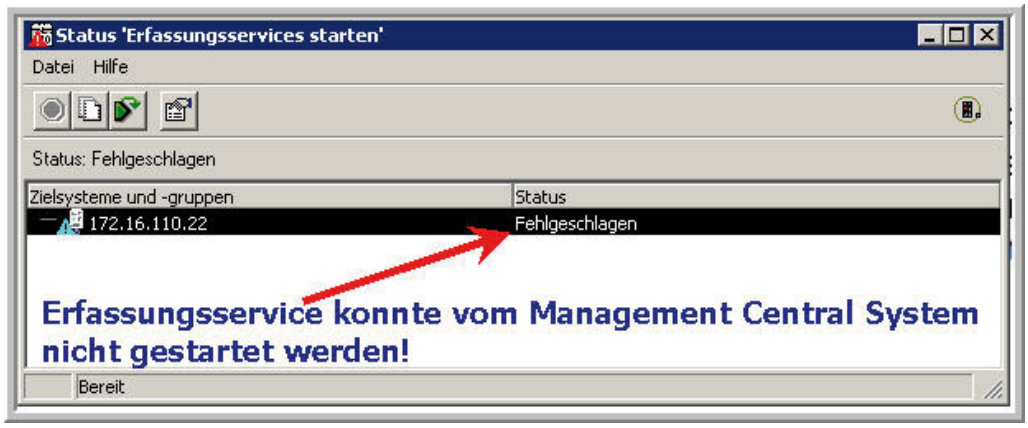
Neu in V6R1 hinzugekommen ist die Analyse der JVM.

Und mit Release i7.1 können auch Informationen zu folgenden Punkten erfasst werden:

- Removable Storage
- External Storage
- System Internal Data

Da wir die Auswertungsmöglichkeiten bislang gar nicht kennen, entscheide ich mich zunächst für das Profil „Standard Plus Protokoll“, welches von IBM empfohlen wird, und bestätige meine Eingaben.

Vermutlich werden Sie anschließend darüber informiert, dass zum Starten der Erfassungsservices eine Verbindung zum zentralen System hergestellt werden muss. Bestätigen Sie auch diese Eingabe. Falls Sie jetzt die folgende Anzeige sehen, konnte der Erfassungsservice nicht erfolgreich gestartet werden.

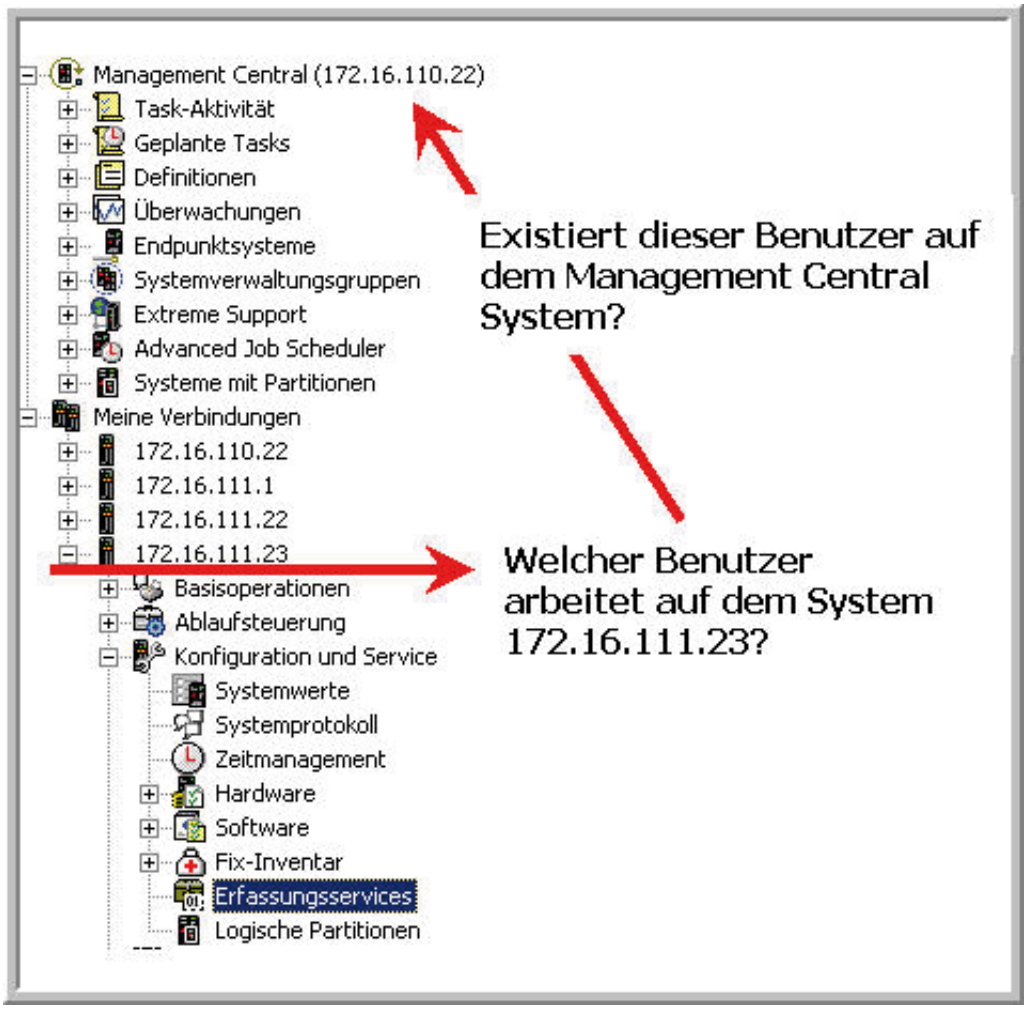


Erfassungsservice fehlgeschlagen

2.11.1.1
Seite 10

Dafür könnte es folgenden Grund geben:

Da der Erfassungsservice über das Management Central System gestartet wird, könnte es Berechtigungsprobleme geben. Ist das Benutzerprofil, welches den Service startet, für das Management Central System berechtigt?



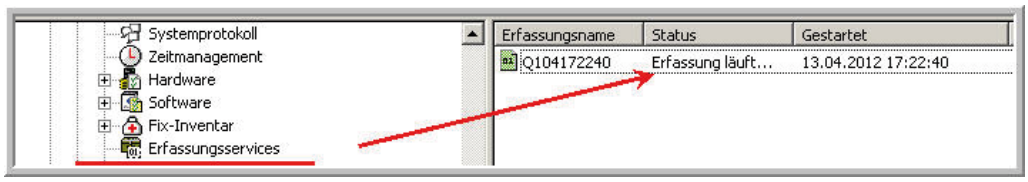
Berechtigungsprobleme durch unterschiedliche Benutzerprofile

Wenn die Erfassungsservices hingegen korrekt gestartet wurden, müssen Sie den Server-Job QYPSPFRCOL im Subsystem QSYSWRK sehen. Dieser Job sammelt nicht nur die Erfassungsdaten, sondern bereinigt auch die gesammelten Informationen in der Bibliothek QPFRDATA, wenn deren Aufbewahrungszeit abgelaufen ist.

Job	Benutzer	Art	-----Status-----	Funktion
QYPSJSVR	QYPSJSVR	STAPEL	AKTIV	PGM-jvmStartPa
QYPSPFRCOL	QSYS	STAPEL	AKTIV	PGM-QYPSPFRCOL
QYUSALRMD	QSYS	BATCHI	AKTIV	PGM-QRMCCTEXEC

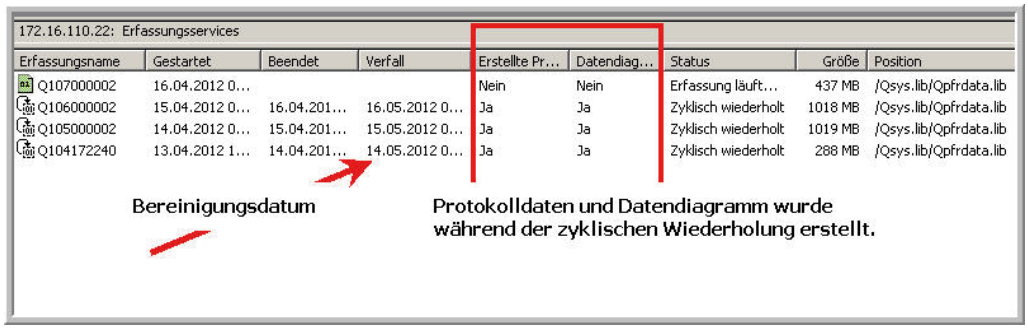
Der Serverjob QYPSPFRCOL

Auch im System i Navigator können Sie erkennen, ob die Erfassungsservices erfolgreich gestartet wurden:



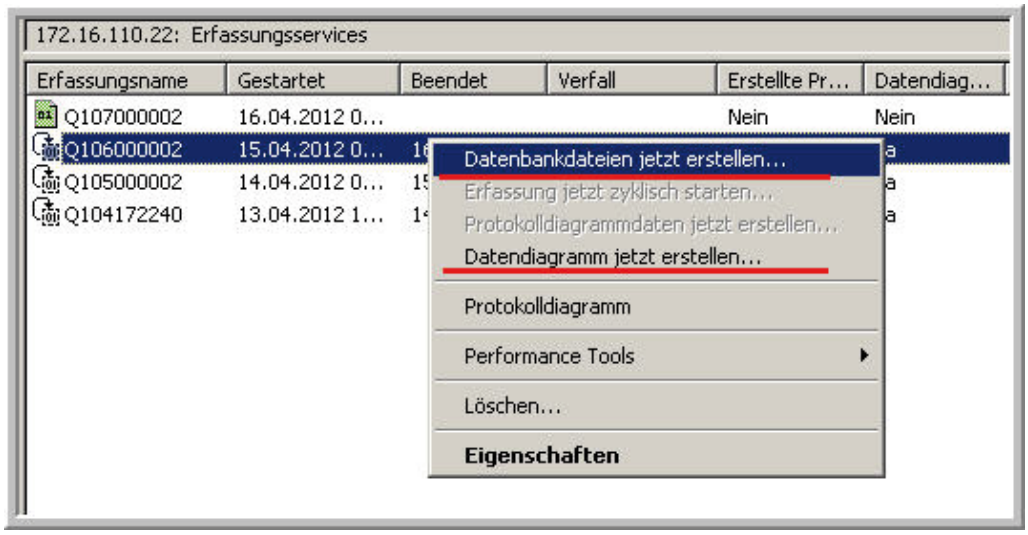
Die laufenden Erfassungsservices

Im Laufe der Zeit sammeln sich hier die *MGTCOL-Objekte:



*Übersicht der vorhandenen *MGTCOL-Objekte*

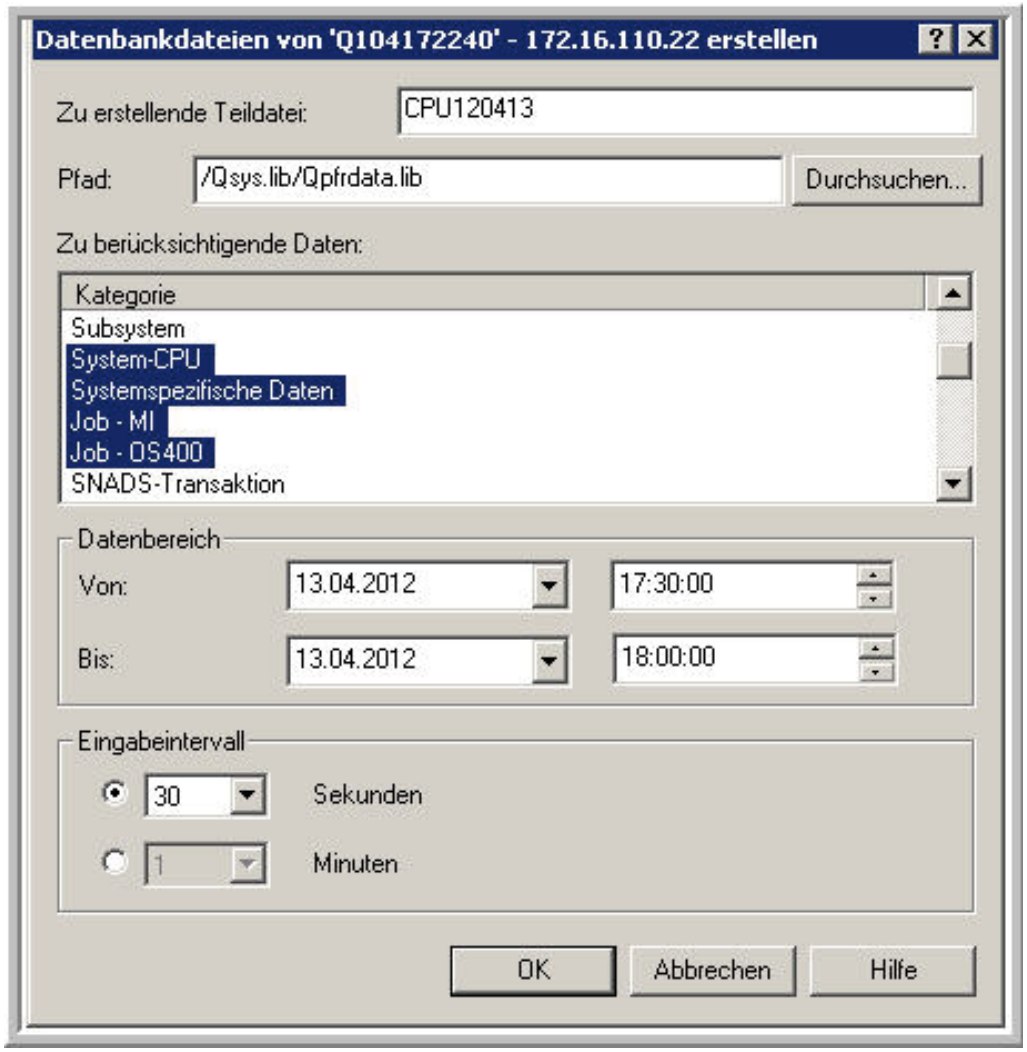
Falls Sie sich zuvor entschlossen haben, die Datenbankdateien und Protokolldaten nicht automatisch zu generieren, können Sie jetzt anhand der Erfassungsservices (*MGTCOL) jederzeit die gewünschten Informationen erstellen:



*Mit *MGTCOL-Objekten arbeiten*

Ich entscheide mich in der vorangegangenen Abbildung, nachträglich Datenbankdateien zu erstellen und erzeuge im folgenden Beispiel die Teildatei

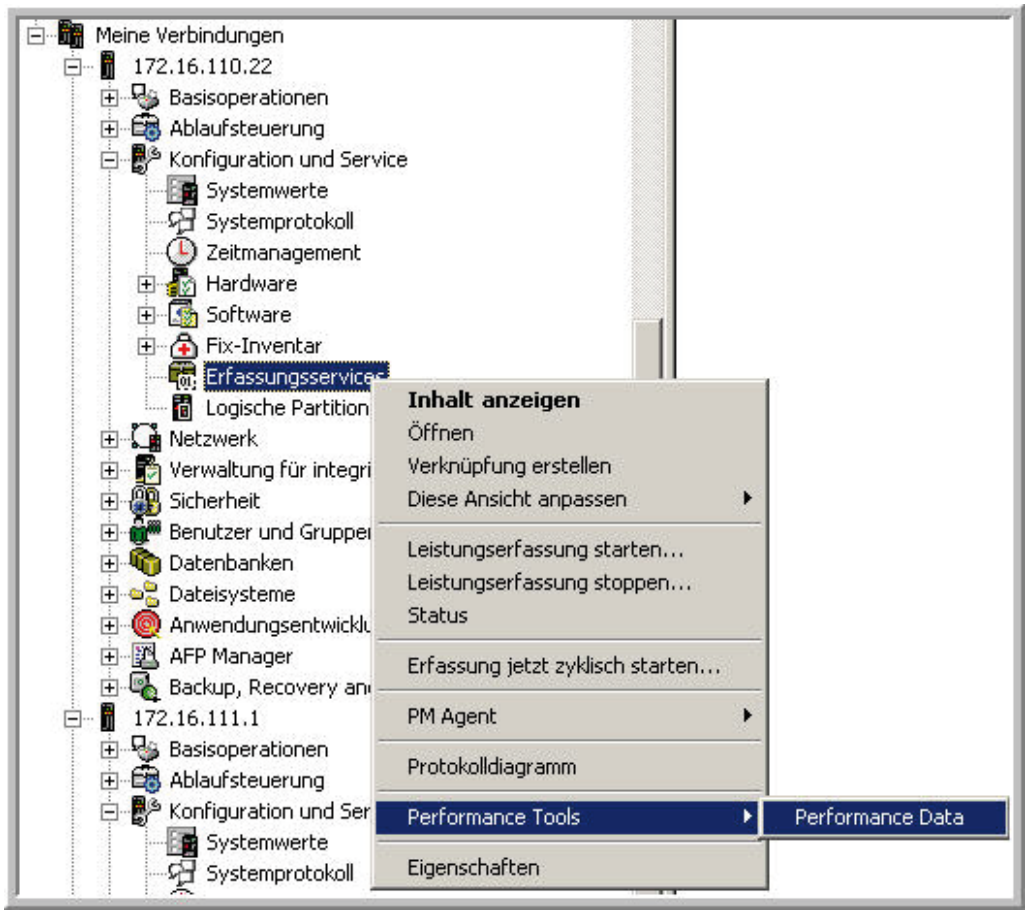
„CPU120413“. Dabei beschränke ich mich auf die Analyse der CPU-Auslastung für den Zeitraum von 17:30 Uhr bis 18:00 Uhr.



Datenbankdateien erstellen

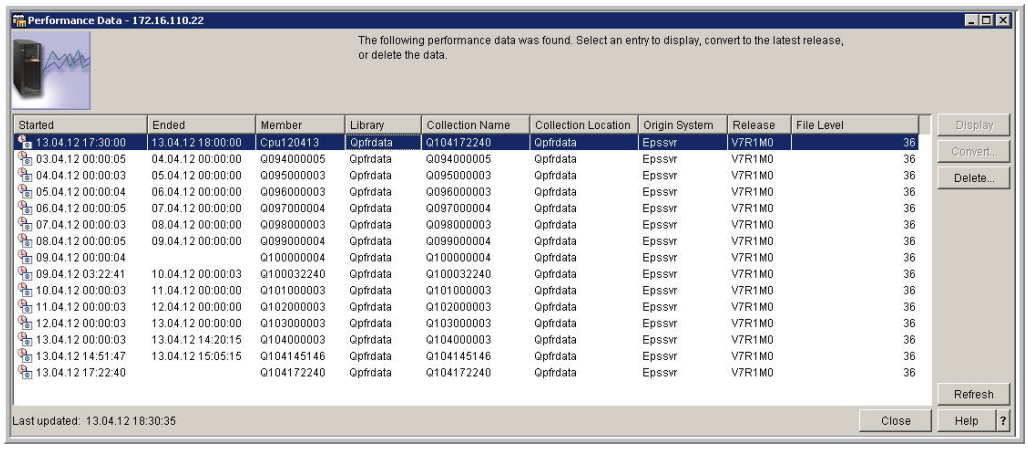
Gespeichert werden die Datenbankinformationen in Teildateien der entsprechenden physischen Dateien. Da die Datenbankdateien mehrfach erzeugt werden können, kann man auf diesem Weg sehr gut relevante Zeiten oder spezifische Probleme analysieren. Des Weiteren können auch nur einzelne Kategorien ausgewählt werden. Es kann daher sinnvoll sein, den Teildateinamen sprechend zu wählen. Bei Angabe des gleichen Teildateinamens wird eine bestehende Teildatei gleichen Namens überschrieben.

Welche Datenbankdateien und welche Teildateien bereits existieren, lässt sich schnell überprüfen. Nutzen Sie im System i Navigator den Pfad „Erfassungsservices“ ➜ „Performance Tools“ ➜ „Performance Data“:



Datenbankdateien anzeigen

So erhalten Sie eine Übersicht aller Teildateien:

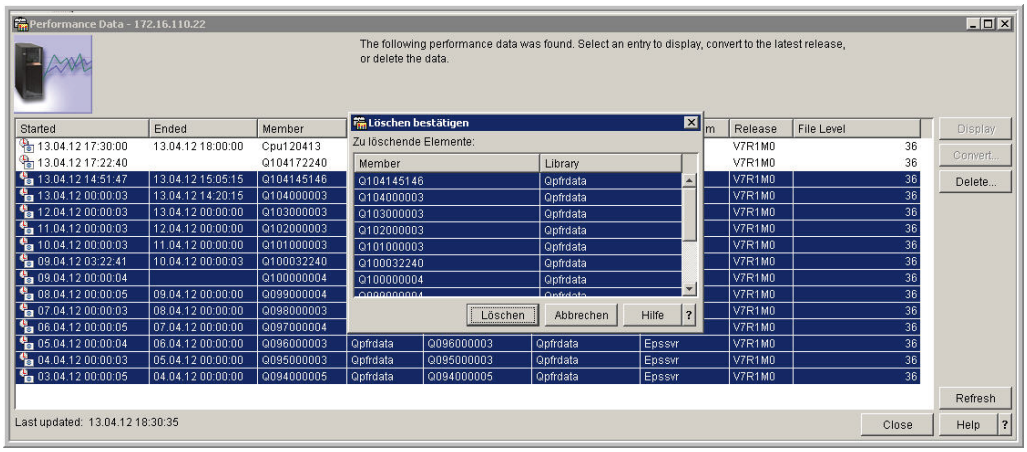


Datenbankdateien anzeigen (2)

Achten Sie auf die letzte Spalte der Abbildung. Anhand des „File Level“ können Sie erkennen, ob Sie die Daten sofort analysieren (= Display) können, oder ob die Datenbankdateien von einem vorhergehenden Release stammen. Wenn der Level kleiner oder gleich dem Wert 18 ist, müssen Sie die Daten-

bankdateien zunächst auf das aktuelle Release umsetzen (= Button Convert), um die Auswertungen anzuzeigen. Ist der Level hingegen größer, können Sie die Daten sofort analysieren.

Diese Oberfläche können Sie auch nutzen, um beispielsweise nicht mehr benötigte Datenbankdateien zu löschen. Markieren Sie hierfür einfach die entsprechenden Einträge und klicken Sie auf den Button „Delete“. Nach einer erneuten Bestätigung werden die Datenbankdateien gelöscht.



Datenbankdateien löschen

Die Datenbankteildateien werden nicht automatisch bereinigt, wenn die *MGTCOL-Objekte gelöscht werden.

Erfassungsservice konfigurieren – IBM Systems Director Navigator for i

Alternativ zum System i Navigator können Sie auch IBM Systems Director Navigator for i verwenden. Obwohl IBM Systems Director Navigator for i Teil der Betriebssystemlizenz (5761SS1/5770SS1 Option 3) ist, habe ich die Erfahrung gemacht, dass einige der dafür notwendigen Komponenten nicht auf den Servern verfügbar waren. Dies lag in der Regel daran, dass sie bislang nicht installiert wurden. Sie sollten daher überprüfen, ob folgende Basislizenzen auf Ihrem Server vorhanden sind:

Release V6R1
IBM HTTP Server for i (5761DG1)
Qshell (5761SS1 Option 30)
Portable App Solutions Env (PASE) (5761SS1 Option 33)
IBM Developer Kit for Java (5761JV1)
J2SE 5.0 32 bit (5761JV1 Option 8)
IBM Toolbox for Java (5761JC1)
Host Servers (5761SS1 Option 12)

Release i7.1
IBM HTTP Server for i 5770DG1
Qshell 5760SS1 Option 30
Portable App Solutions Env (PASE) 5760SS1 Option 33
IBM Developer Kit for Java 5761JV1
J2SE 5.0 32 bit 5761JV1 Option 8
Host Servers 5770SS1 Option 12

Erst wenn die Voraussetzungen gegeben sind, werden Sie sich auf der Weboberfläche anmelden können. Eigentlich existiert dieses Webinterface bereits eine ganze Weile; ist aber dennoch bislang wenig im Einsatz. Das mag an den nicht wirklich überzeugenden Antwortzeiten liegen. Allerdings musste ich feststellen, dass mit zunehmenden Gebrauch einerseits die Antwortzeiten deutlich besser wurden, andererseits insbesondere im Bereich „Leistungsanalyse“ die Oberfläche einiges an Komfort zu bieten hat, was dem Navigator leider fehlt. Schade, dass IBM sich entschlossen hat, mit zwei unterschiedlichen Interfaces zu arbeiten, anstatt konsequent ein Interface einzusetzen.

2.11.1.2
Seite 2

Die Weboberfläche wird über den Port 2001 aufgerufen:

IBM Systems Director Navigator for i

Wahrscheinlich erhalten auch Sie einen Sicherheitshinweis Ihres Browsers, dass der Verbindung nicht vertraut werden darf. Sie haben dann zwei Möglichkeiten:

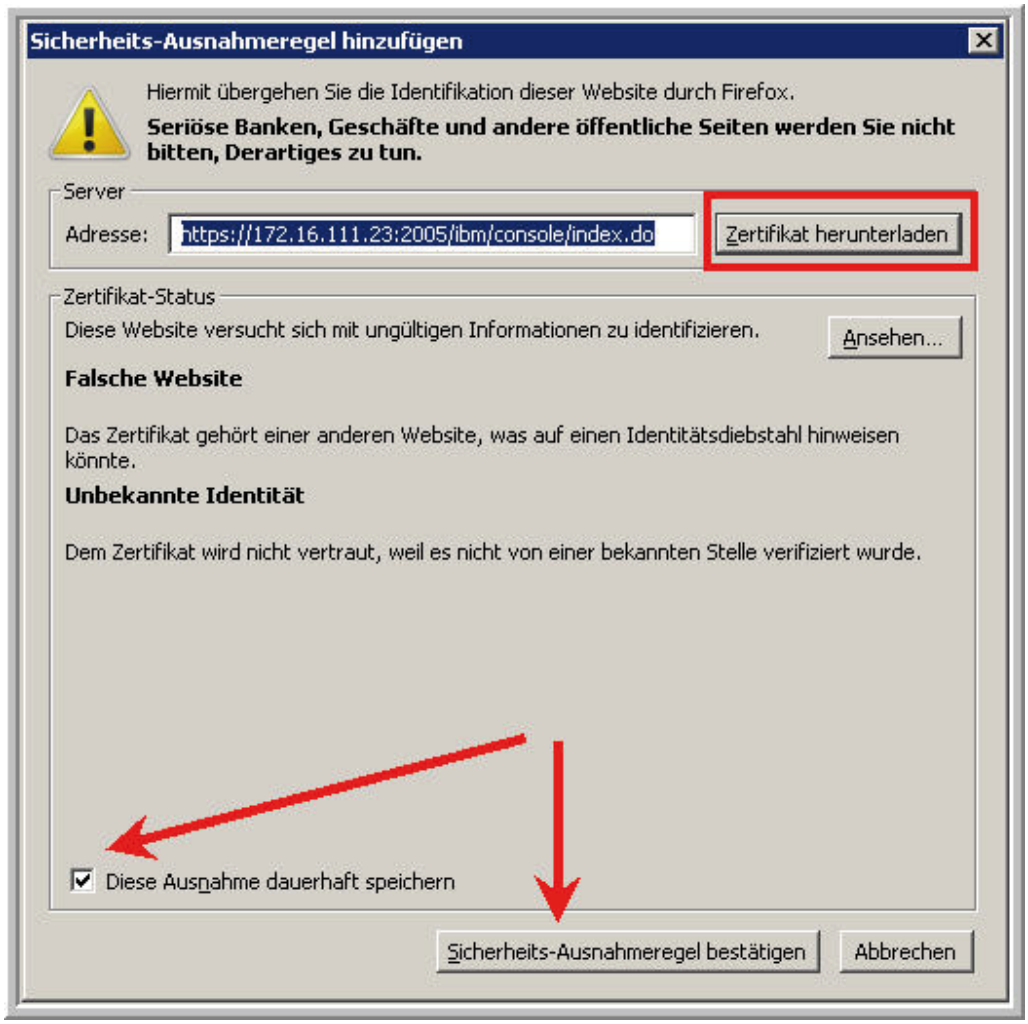
Entweder Sie ignorieren den Hinweis und setzen den Ladevorgang fort oder Sie laden das Zertifikat herunter.

Vermutlich machen Sie es wie ich. Viele Male habe ich den Hinweis ignoriert und das Laden einfach fortgesetzt. Doch irgendwann hat mich der Hinweis gestört und ich habe das Zertifikat geladen und als Ausnahmeregelung bestätigt. Hierfür klicken Sie im Firefox-Browser auf den Button „Ausnahmen hinzufügen“, nachdem Sie bestätigt haben, das Risiko zu kennen. Die folgende Grafik zeigt, was gemeint ist:

Ausnahmen hinzufügen

2.11.1.2
Seite 4

Anschließend erscheint ein weiterer Dialog:



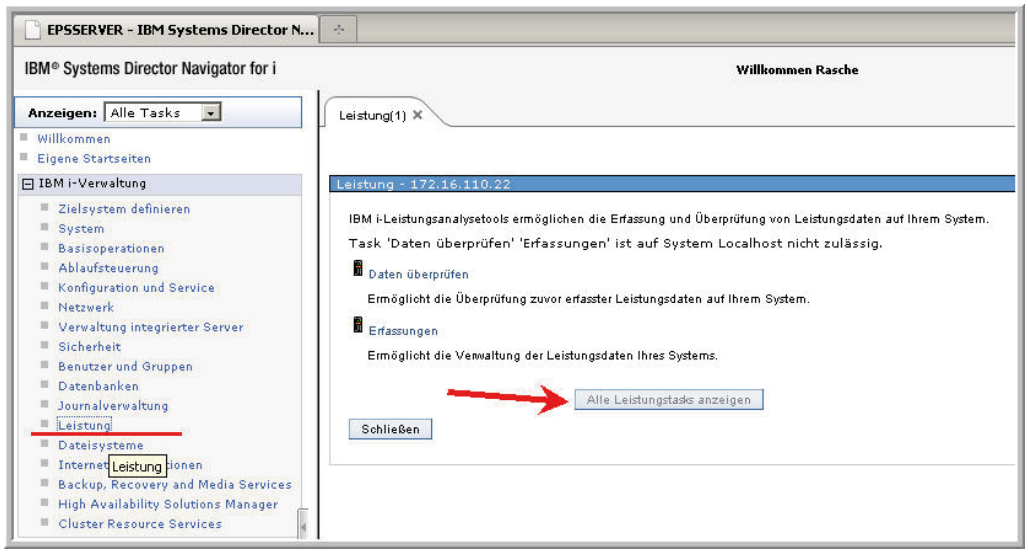
Sicherheits-Ausnahmeregel hinzufügen

Auf dieser Seite müssen Sie als Erstes das Zertifikat herunterladen und anschließend mit dem Button „Sicherheits-Ausnahmeregel bestätigen“ selbiges bestätigen. Achten Sie darauf, dass der Haken für die dauerhafte Speicherung gesetzt ist.

Ab sofort wird Ihr Firefox-Browser sich nicht mehr beklagen, wenn er den IBM Systems Director Navigator for i öffnen muss.

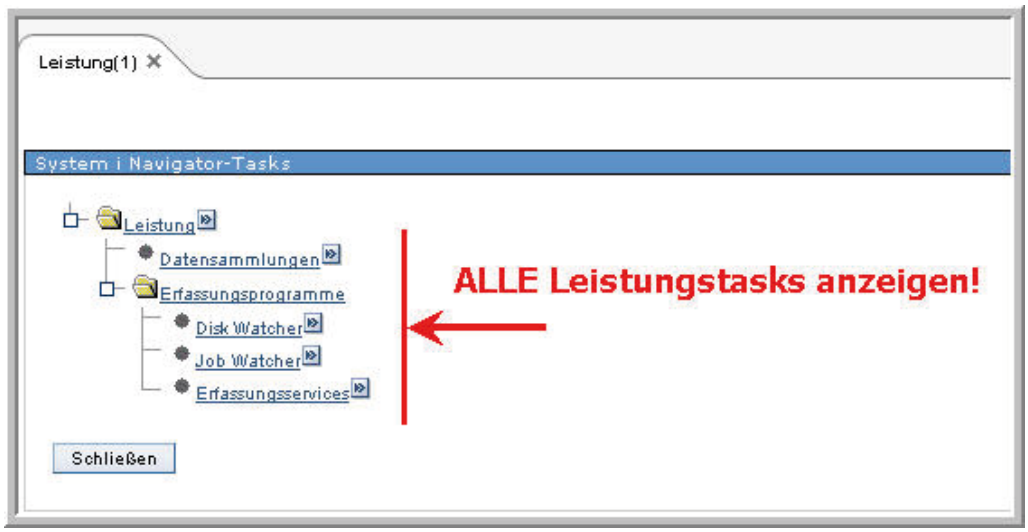
Wenn diese Hürde überwunden ist, erhalten Sie den Anmeldedialog. Hier melden Sie sich mit Ihrem Benutzernamen an. Der Anmeldevorgang kann ein wenig dauern – also nicht nervös werden.

Ist auch diese Hürde genommen, öffnen Sie den Eintrag „IBM i-Verwaltung ➜ Leistung“ wie in folgender Abbildung gezeigt:



Das Register „Leistung“

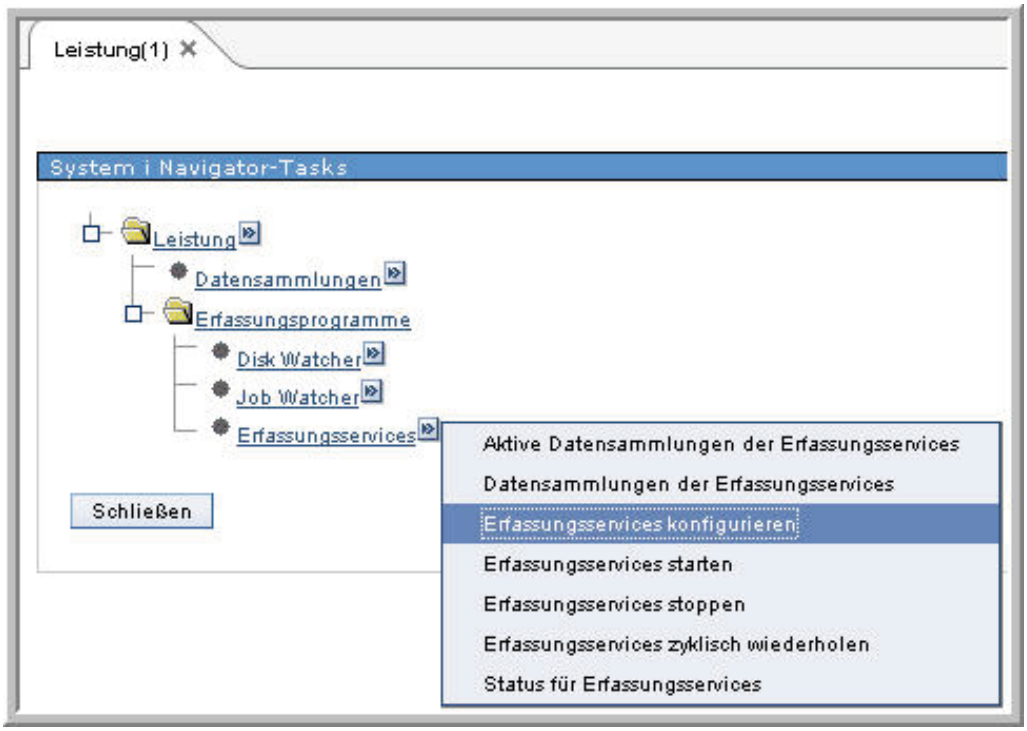
Bevor Sie jetzt Erfassungsservices starten, sollten Sie erst einmal auf den Button „Alle Leistungstasks anzeigen“ klicken. Daraufhin verändert sich nämlich die Anzeige folgendermaßen:



Alle Leistungstasks anzeigen

Öffnen Sie jetzt das Kontextmenü der „Erfassungsservices“, indem Sie auf den kleinen Pfeil klicken:

2.11.1.2
Seite 6



Das Kontextmenü der Erfassungsservices

Im IBM Systems Director Navigator for i steht Ihnen anders als im System i Navigator gezeigt, der Menüpunkt „Erfassungsservices konfigurieren“ separat zur Verfügung. Alternativ können Sie aber auch in diesem Interface die Erfassungsservices während des Startens konfigurieren, wie wir es im System Navigator gesehen haben.

Schauen wir uns auch hier den Dialog „Erfassungsservices konfigurieren“ an. Die Konfiguration unterteilt sich in drei Register:

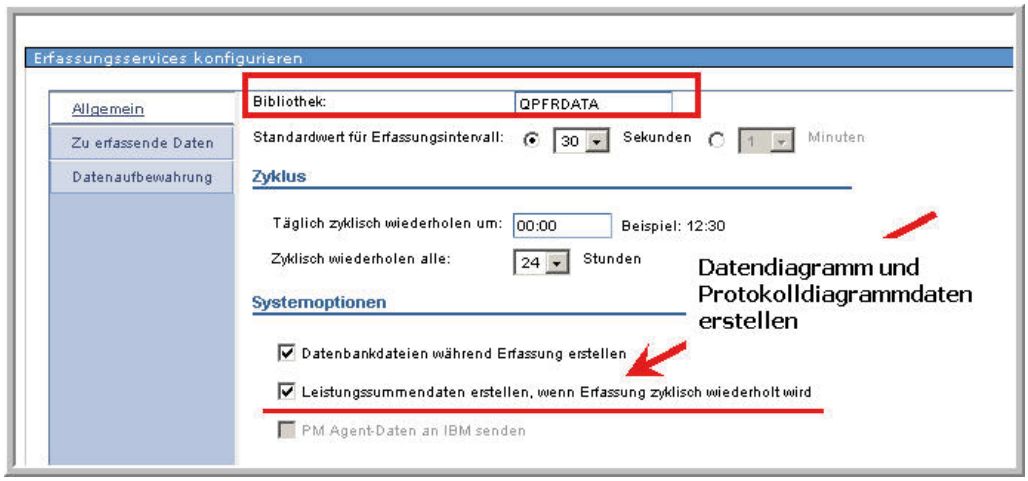
Allgemein:
Allgemeine Angaben zur Bibliothek, dem Wiederholungszyklus, der Datenbank und den Diagrammerstellungsoptionen

Zu erfassende Daten:
Auswahl der Datenprofile

Datenaufbewahrung:
Aufbewahrungszeit für *MGTCOL-Objekte und Datendiagramme

Schauen wir uns im Folgenden die Unterschiede zum System i Navigator an:

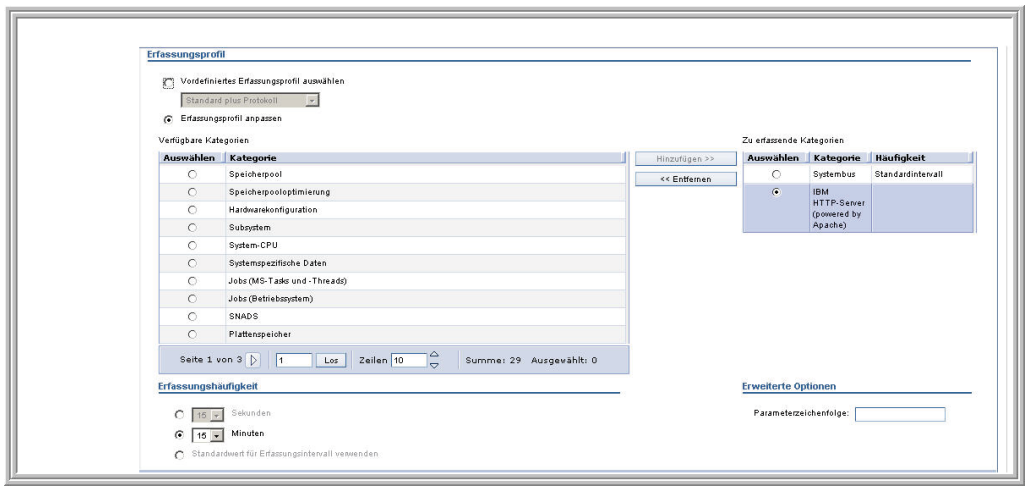
Anders als im Navigator können Sie in dieser Oberfläche eine eigene Bibliothek für die *MGTCOL-Objekte verwenden. Bedenken Sie hierbei aber, dass die Bereinigungsservices nur für die Standardbibliothek gelten.



Erfassungsservice konfigurieren (1) – IBM Director Navigator

Alle weiteren Attribute sind Ihnen durch das vorangegangene Kapitel bereits bekannt.

Auf der nächsten Seite „Zu erfassende Daten“ finden Sie einerseits die bereits erläuterten Standarddatenprofile, andererseits können Sie auch hier wieder individuelle Profile konfigurieren. Der Dialog für individuelle Datenprofile ist ähnlich aufgebaut wie im System i Navigator.



Erfassungsservice konfigurieren (2) – IBM Director Navigator

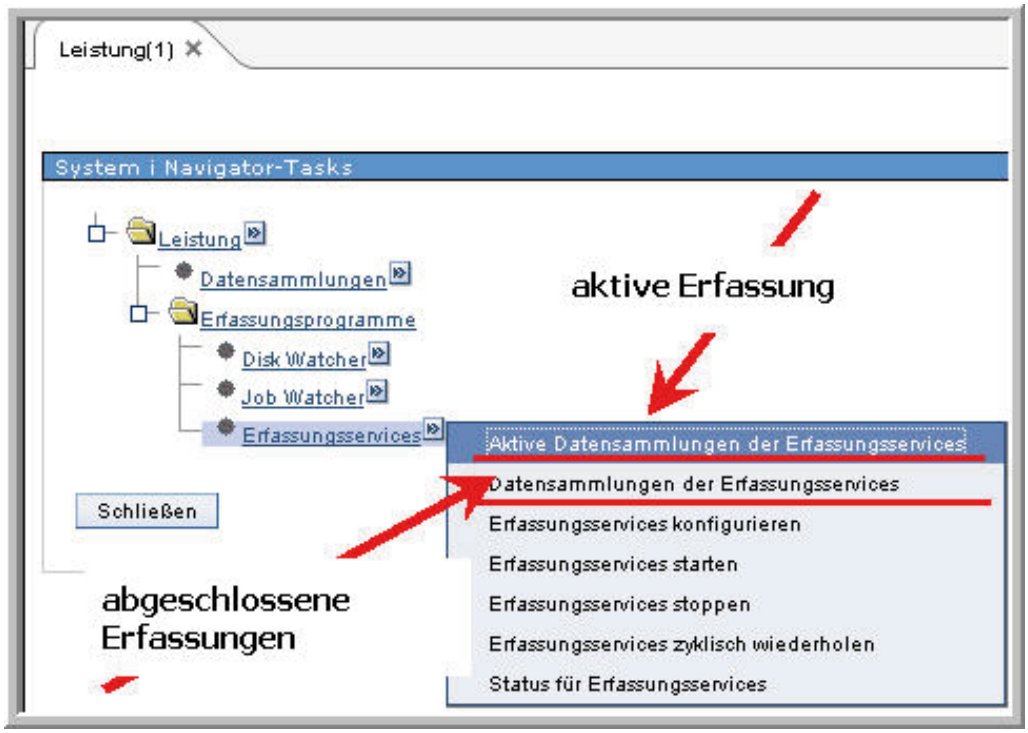
2.11.1.2
Seite 8

Und auch auf der dritten Seite finden wir keine neuen Informationen.

Erfassungsservice konfigurieren (2) – IBM Director Navigator

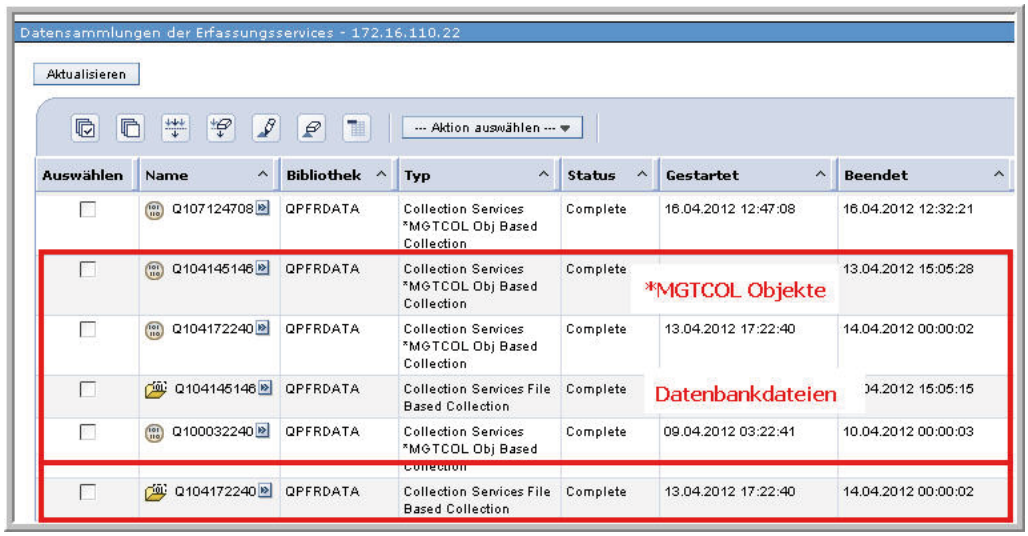
Es fällt lediglich auf, dass Sie in diesem Interface nicht festlegen können, wie lange die Protokolldiagrammdaten gespeichert werden. Dies ist nur mit dem System i Navigator möglich. Bestätigen Sie abschließend den Dialog, um die Erfassungsservices zu starten.

Wie im System i Navigator können Sie sich aktive und abgeschlossene Erfassungsservices anzeigen lassen.



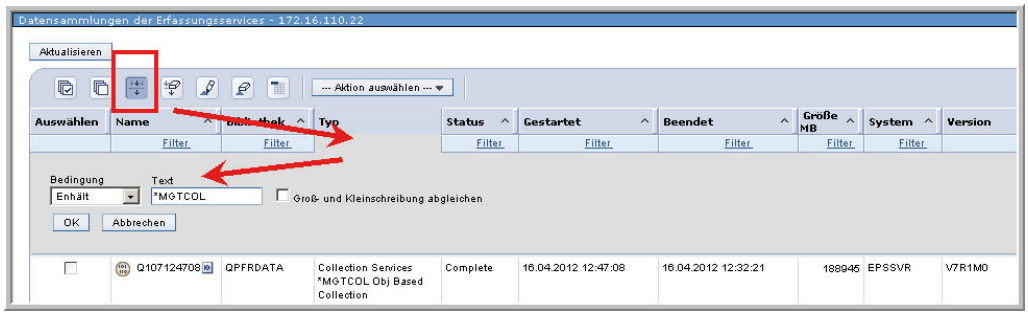
Erfassungsdaten anzeigen

Anders als beim System i Navigator wird hier zwischen abgeschlossenen und aktiven Erfassungen unterschieden. Schauen wir uns die abgeschlossenen Datensammlungen an.



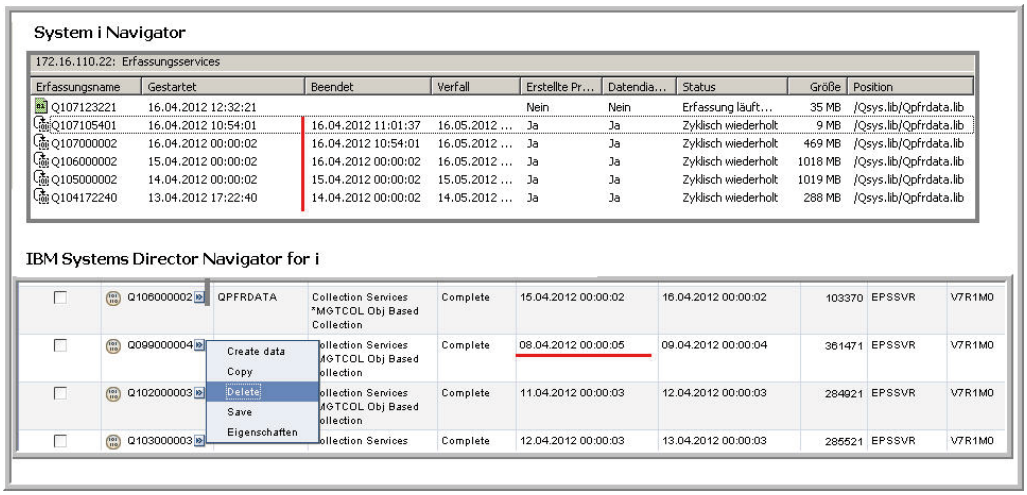
Erfassungsdaten anzeigen (2)

Die Weboberfläche zeigt die *MGTCOL-Objekte und die Datenbankdateien in einer Übersicht an. Das ist zunächst einmal nicht weiter störend, da wir über Filter getrennte Sichten aufbauen können.



Erfassungsdaten filtern

Was allerdings derzeit ein Problem ist: Wenn Sie sowohl mit dem System i Navigator als auch mit der Weboberfläche arbeiten, werden in der Weboberfläche nicht existierende Daten angezeigt.



Die Weboberfläche zeigt nicht vorhandene Daten

Diese Tatsache kann schnell zu Verwirrung führen, wenn Sie versuchen, mit solchen Objekten zu arbeiten oder Sie zu löschen. Hoffentlich handelt es sich hierbei nur um einen Fehler, der bald korrigiert wird. Für den Moment sollten Sie sich aber für eine Oberfläche entscheiden, um unnötigen Ärger zu vermeiden.

Abschließend eine letzte Anmerkung zum Webinterface. Sie müssen sich korrekt abmelden, wenn Sie Ihre Arbeit beenden:

IBM Systems Director Navigator for i beenden

Schließen Sie die Webseite einfach ohne auf „Abmeldung“ zu klicken, bleibt Ihr Job aktiv und bis V6R1 ist eine erneute Anmeldung unter dem Benutzernamen nicht mehr möglich.

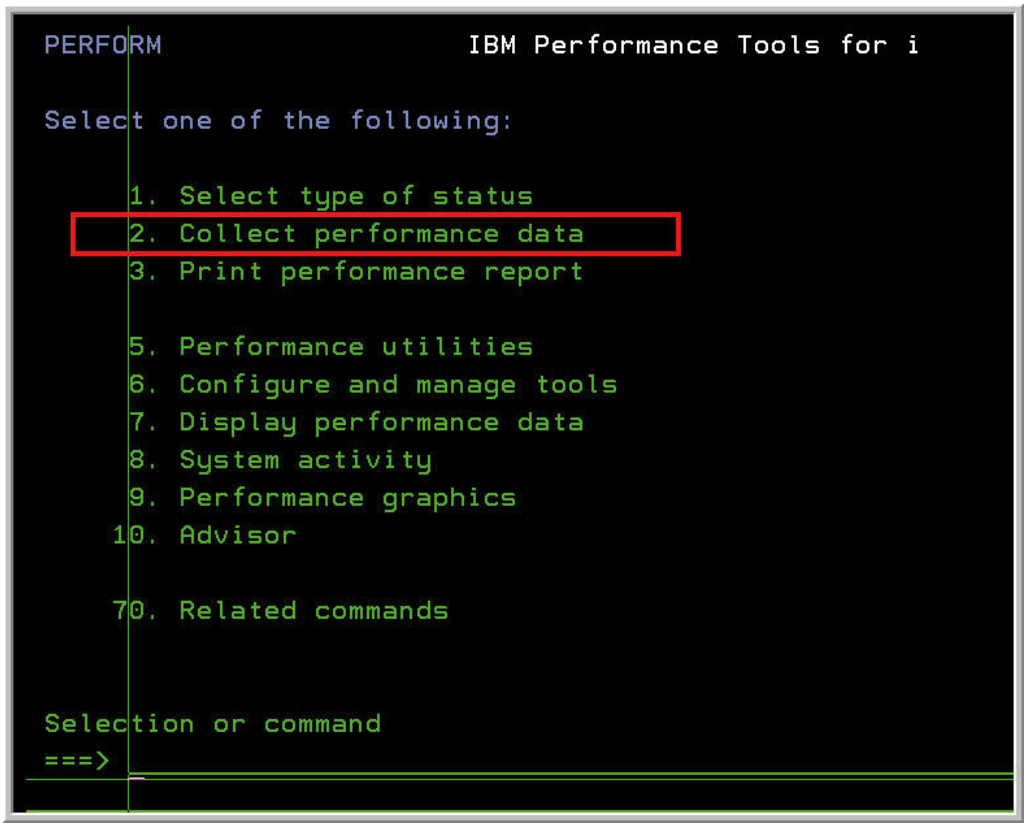
Erfassungsservice konfigurieren – 5250-Interface

Bislang habe ich ausschließlich grafische Oberflächen für den Start der Erfassungsservices eingesetzt, alternativ stehen aber auch entsprechende 5250-Befehle zur Verfügung.

CL-Befehl	Bedeutung
STRPFRCOL	Erfassungsservice starten
ENDPFRCOL	Erfassungsservice beenden
ANZPFRDTA	Analyse der Erfassungsdaten
DSPPFRDTA	Analysedaten anzeigen
CFGPFRCOL	Erfassungsservice konfigurieren
CHKPFRCOL	Überprüfung der Erfassungsservices
CVTPFRCOL	Erfassungsservices konvertieren
Neu i6.1	
DLTPFRCOL	Erfassungsservices löschen
SAVPFRCOL	Erfassungsservices sichern
RSTPFRCOL	Erfassungsservices zurückspeichern

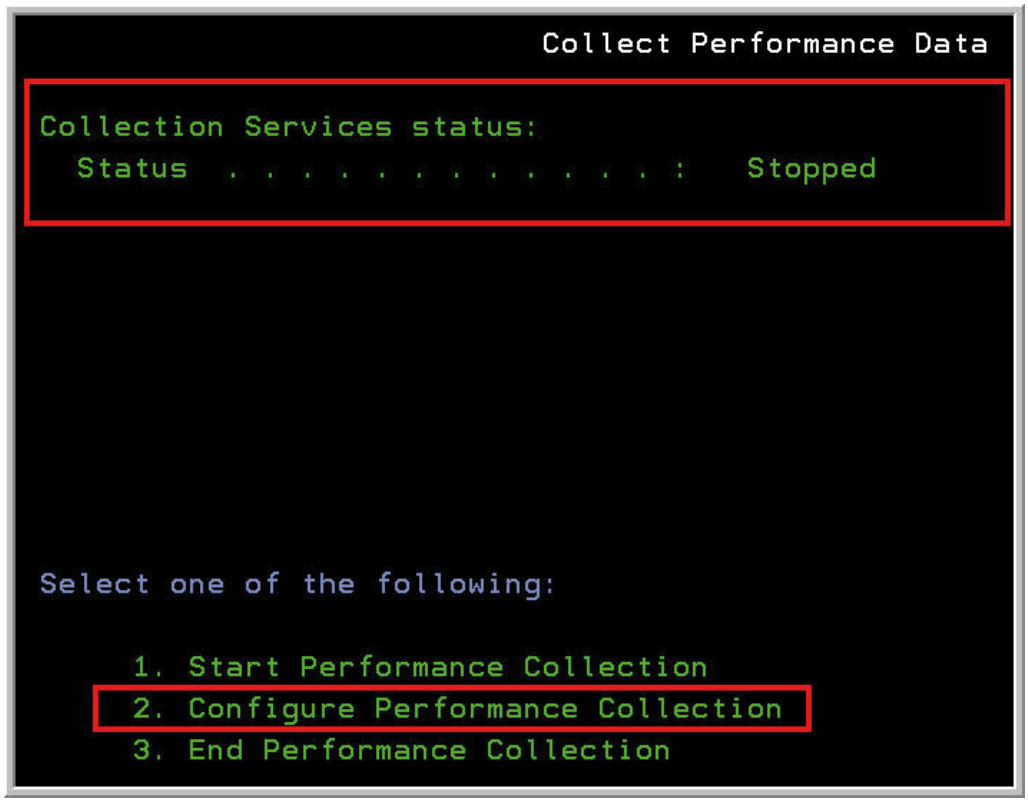
Alternativ können Sie auch das Menü „Perform“ verwenden. Das „Performance Tool for i Menü“ fast alle Tätigkeiten zusammen, erlaubt Ihnen die Erfassungsservices zu starten und stellt Ihnen zusätzlich diverse Auswertungsmöglichkeiten zur Verfügung.

2.11.1.3
Seite 2



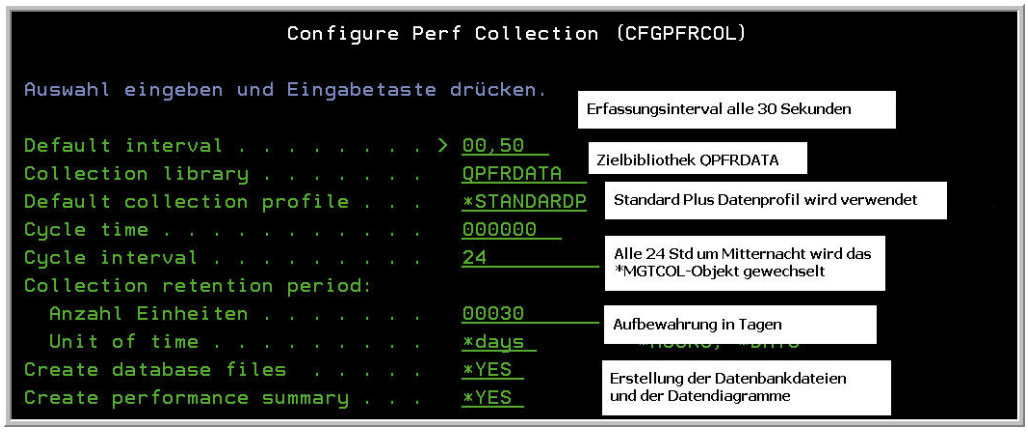
Das Menü PERFORM

Um die Erfassungsservices in diesem Interface zu starten, müssen Sie den Menüpunkt 2 auswählen.



Erfassungsservice verwalten

Als Erstes sehen Sie den aktuellen Status des Erfassungsservice. Auf unserem Server sind die Erfassungsservices aktuell gestoppt. Wenn ich jetzt Auswahl 2 wähle, erscheint ein neuer Dialog, der es mir erlaubt, die Erfassungsservices zu konfigurieren:



Erfassungsservice konfigurieren (2)

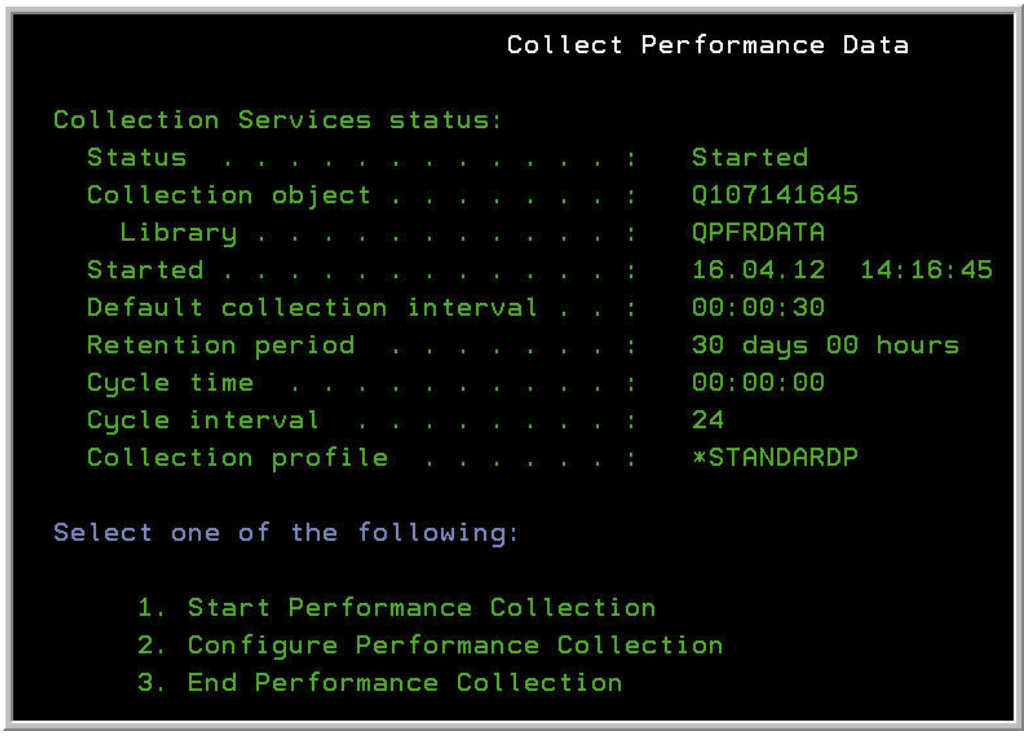
Vieles an den Informationen kennen Sie bereits, weshalb sie nicht weiter erläutert werden.

2.11.1.3
Seite 4

In einem zweiten Schritt müssen wir jetzt die Erfassungsservices starten. Dies kann über den Menüpunkt 1 oder den CL-Befehl STRPFRCOL erfolgen:

Erfassungsservice starten

Anders als in der grafischen Oberfläche müssen wir hier zwei separate Befehle ausführen; der Befehl STRPFRCOL erlaubt uns nicht die gleichzeitige Konfiguration der Erfassungsservices.



Statusabfrage der Erfassungsservices

Job Watcher-Analysen

Job Watcher wurde entwickelt, um einzelne oder alle Jobs oder Tasks eines Servers detailliert zu untersuchen. Die Informationen werden nahezu in Echtzeit gesammelt, so dass Sie mit der Datenanalyse umgehend beginnen können. Gesammelt werden die Daten von einem Serverjob. Dieser Job speichert die Daten in verschiedenen Datenbankdateien, die dann wiederum mit dem Webinterface IBM Systems Director Navigator for i ausgewertet werden können. Es handelt sich hierbei zwar um historische Momentaufnahmen – also einen Snapshot – und nicht um eine Ablaufverfolgung (Trace). Vielleicht sind bei einer Ablaufverfolgung die Daten ein wenig exakter und ggf. enthalten Sie auch die ein oder andere zusätzliche Information, aber dies kompensiert nicht den Seiteneffekt, den eine Traceverfolgung auf die Gesamtperformance Ihres Systems hätte. Insbesondere dann nicht, wenn das System stark ausgelastet ist.

Job Watcher stellt Ihnen sicherlich einige Informationen zur Verfügung, die Sie auch mit anderen Analysetools erhalten. Der große Vorteil gegenüber den anderen Analysen ist die Möglichkeit zur Auswertung der Wartezeiten in Kombination mit Stack-Informationen und der Darstellung der detaillierten Konfliktsituationen. So ist es beispielsweise möglich, negative Konfliktsituationen wie Satzsperren problemlos zu analysieren. Zudem berücksichtigt Job Watcher SQL-Zugriffe und Aktivierungsgruppen.

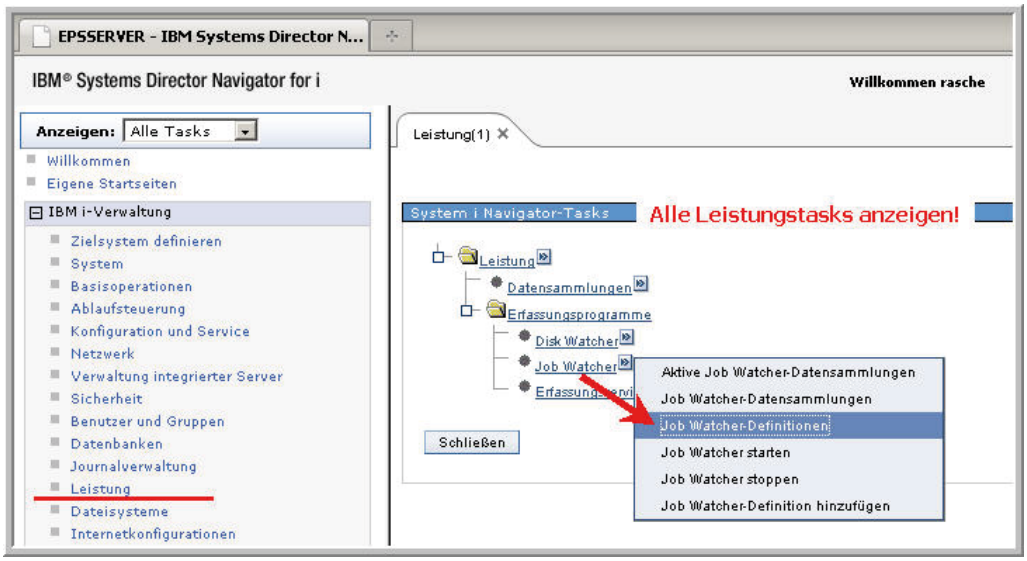
Job Watcher können Sie unterschiedlich einsetzen:

- Sie können die Auswertungen präventiv starten, um eine Vergleichsbasis für eventuelle Performanceprobleme zu haben. In diesem Fall starten Sie Job Watcher für ausgewählte Zeiträume morgens, mittags, nachmittags und abends, wenn vermehrt Batch-Jobs ausgeführt werden.
- Aber Sie können die Auswertungen auch umgehend starten, wenn einzelne Programme oder Jobs Performanceprobleme haben. Die Analyse wird Ihnen helfen, die Ursache für das schlechte Laufzeitverhalten zu identifizieren.

2.11.2
Seite 2

Job Watcher mit Standardefinitionen starten

Job Watcher können Sie nur über die Weboberfläche IBM Systems Director Navigator for i oder mit entsprechenden CL-Befehlen konfigurieren. Starten wir also wieder die Weboberfläche, melden uns an und öffnen den Eintrag „Leistung“. Wenn soweit alles geglückt ist, sollten Sie folgenden Bildschirm sehen:



Job Watcher – Definitionen (1)

Unterhalb der „Erfassungsprogramme“ müssten Sie den Eintrag „Job Watcher“ finden. Dass der Eintrag bei Ihnen nicht vorhanden ist, kann verschiedene Gründe haben:

- Haben Sie vergessen „Alle Leistungstasks anzeigen“ zu wählen?
- Wurde Job Watcher auf dem Server überhaupt installiert?
  Prüfen Sie, ob das Feature 5103 des Lizenzprogramms 5770PT1 installiert ist. Das Features ist Bestandteil Ihrer Betriebssystemlizenz.
- Wenn Sie weiterhin Probleme haben, sollten Sie nochmals prüfen, ob die Systemvoraussetzungen erfüllt sind:

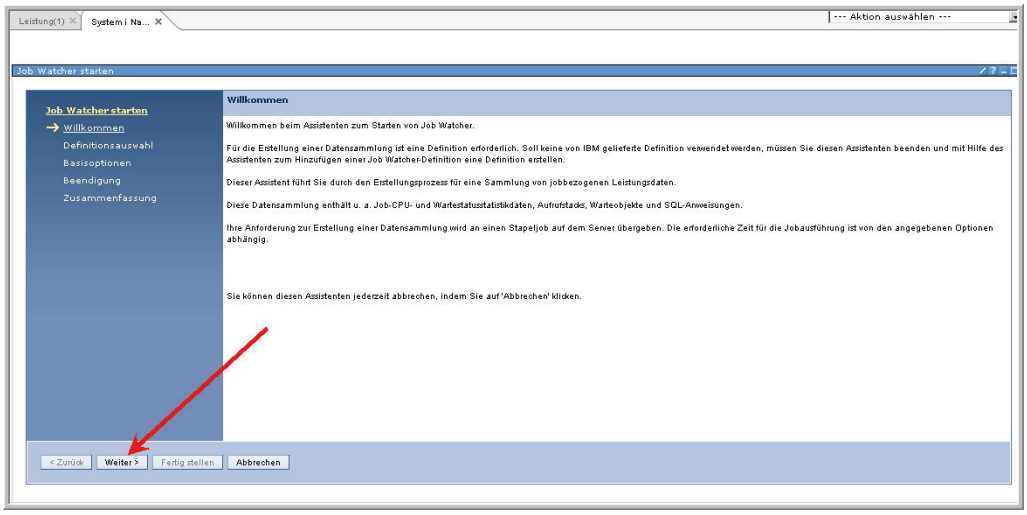
Release V6R1
IBM HTTP Server for i (5761DG1)
Qshell (5761SS1 Option 30)
Portable App Solutions Env (PASE) (5761SS1 Option 33)
IBM Developer Kit for Java (5761JV1)
J2SE 5.0 32 bit (5761JV1 Option 8)
IBM Toolbox for Java (5761JC1)
Host Servers (5761SS1 Option 12)

2.11.2.1
Seite 2

Release i7.1
IBM HTTP Server for i 5770DG1
Qshell 5770SS1 Option 30
Portable App Solutions Env (PASE) 5770SS1 Option 33
IBM Developer Kit for Java 5761JV1
J2SE 5.0 32 bit 5761JV1 Option 8
Host Servers 5770SS1 Option 12

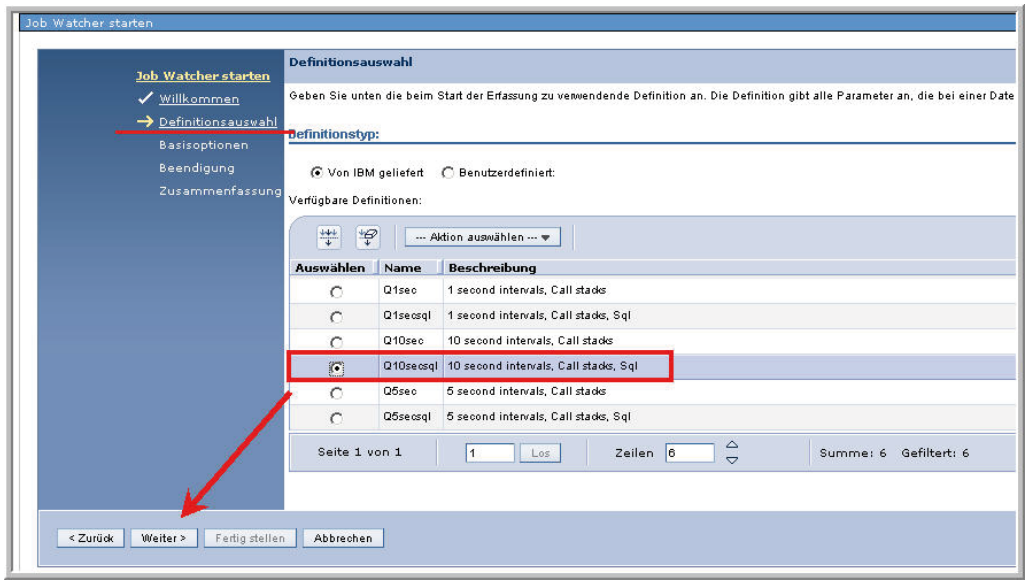
Wenn Sie dann den Eintrag "Job Watcher" sehen, können Sie über das Kontextmenü verschiedene Aufgaben wahrnehmen.

Sie können beispielsweise eigene Job Watcher-Definitionen erstellen und zur Analyse nutzen oder die bestehende Standardkonfiguration verwenden. Falls Sie sich für die letztgenannte Möglichkeit entscheiden, wählen Sie – wie ich es getan habe – den Eintrag "Job Watcher starten".



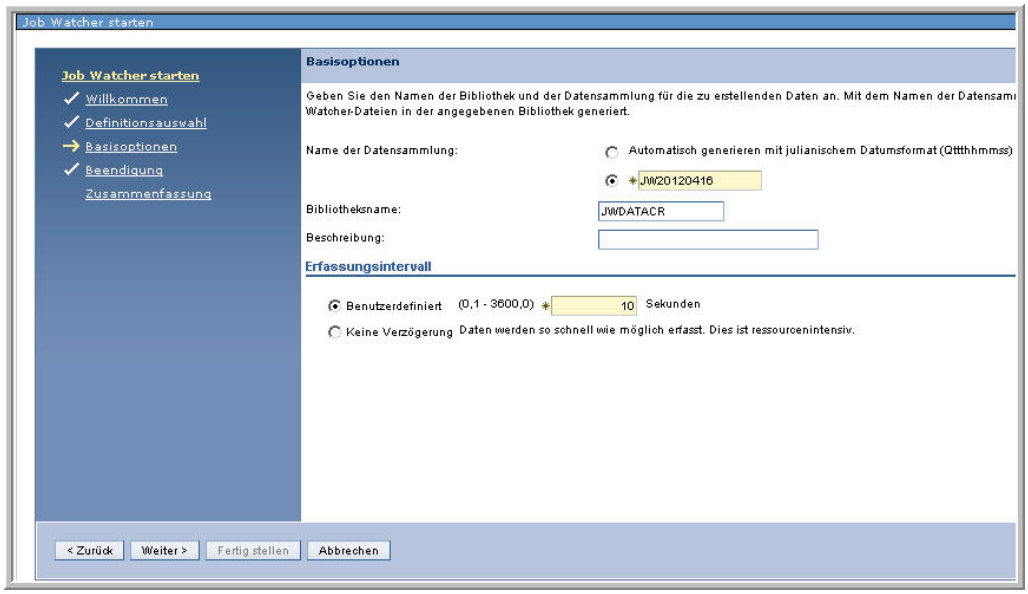
Job Watcher starten (1)

Den Willkommensbildschirm können Sie einfach mit „Weiter“ bestätigen. Anschließend erhalten Sie den Bildschirm zur „Definitionsauswahl“:



Job Watcher starten – Definitionsauswahl

Für welches Erfassungsintervall Sie sich entscheiden, bleibt Ihnen überlassen. Sie haben die Wahl zwischen Intervallen von einer, fünf oder zehn Sekunden. Darüber hinaus müssen Sie entscheiden, ob Sie die SQL-Informationen zusätzlich zum Call Stack der verschiedenen Jobs benötigen oder eher nicht. Ich habe im Falle konkreter Probleme gute Erfahrungen mit einem Sekunden-Intervall gemacht und entscheide mich meistens zusätzlich für die SQL-Erfassung. Anschließend bestätigen Sie den Dialog mit „Weiter“.



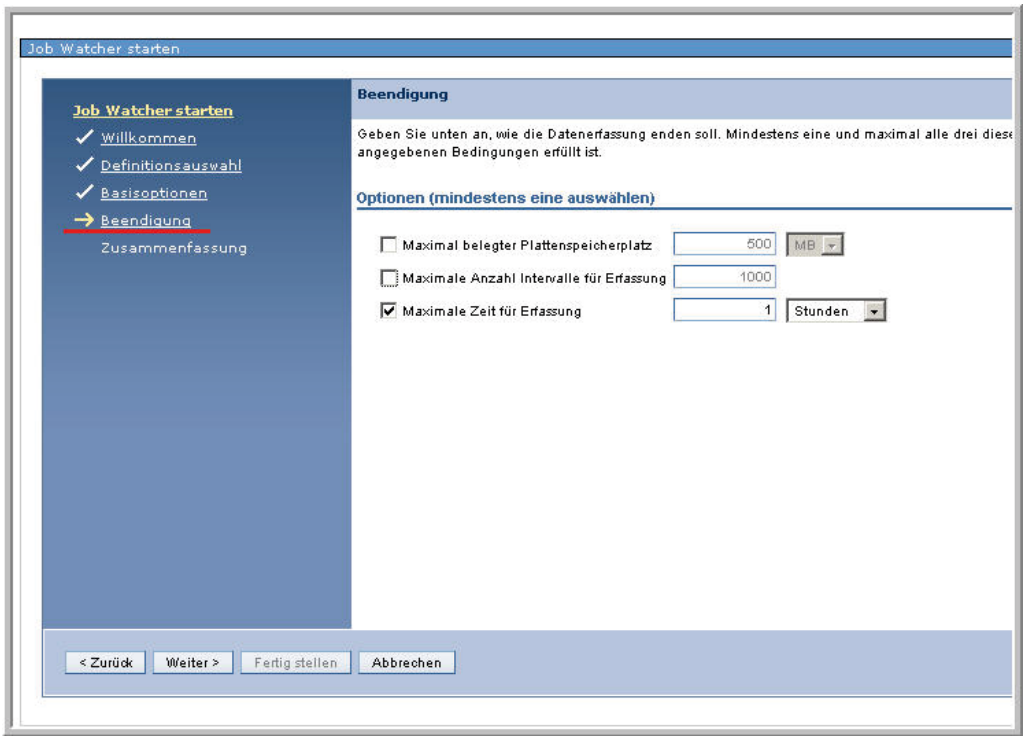
Job Watcher starten – Basisoptionen

2.11.2.1
Seite 4

Im diesem Bildschirm hinterlegen Sie einige Basisoptionen. Als Erstes muss der Name der Datensammlung festgelegt werden. Die Analysedaten werden als Teildateien in den Tabellen der Datensammlung gespeichert. Solange Sie nicht mit SQL auf die Analysedaten direkt zugreifen wollen, ist der Name der Teildatei eher nebensächlich. Ich habe jedoch die Erfahrung gemacht, dass es zur späteren Identifizierung nützlich sein kann, eigene Teildateinamen (z. B. Tjjjjmmtt) zu verwenden, die das Datum der Erfassung beinhalten.

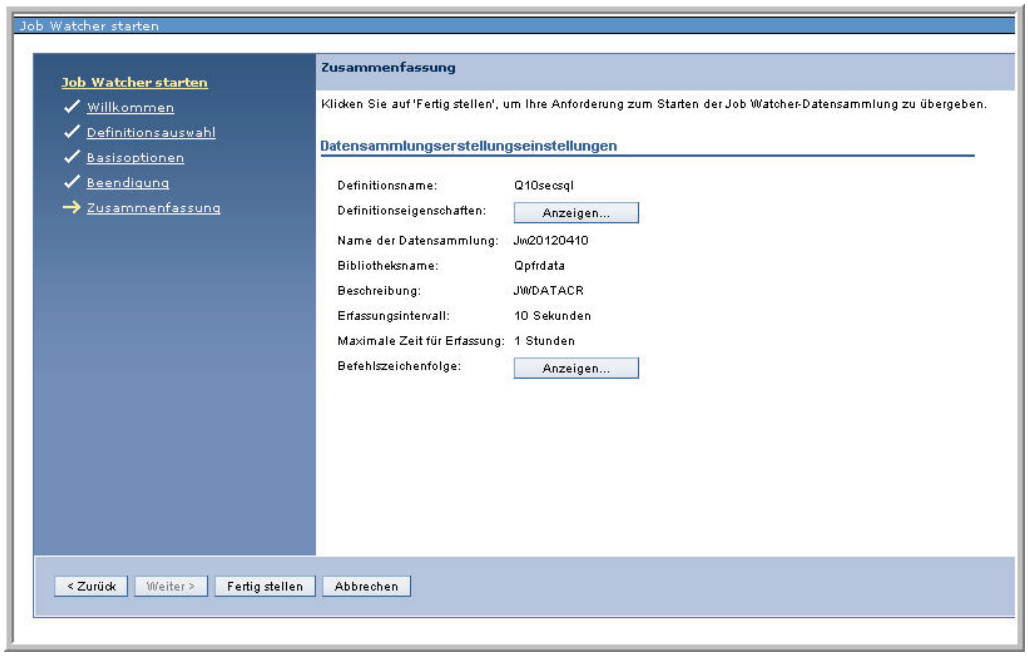
Gespeichert werden die Daten in meinem Beispiel in der Bibliothek QPFRDATA. Häufig ist es sinnvoller, eine eigene Bibliothek für die Job Watcher-Daten zu verwenden, insbesondere dann, wenn die Erfassungsservices gestartet wurden. Ich korrigiere daher noch einmal meine Eingaben und verwende anstelle der Bibliothek QPFRDATA die Bibliothek JWDATACR.

Zu guter Letzt müssen wir hier auch das Erfassungsintervall festlegen. Ich entscheide mich für ein Intervall von 30 Sekunden. Sie können die Daten auch nahezu in Echtzeit erfassen lassen. Aber Vorsicht! Diese Erfassungen benötigen schnell viel Platz auf Ihrer Festplatte und können das System durchaus belasten! Erst jetzt bestätige ich meine Eingabe mit „Weiter“.



Job Watcher starten – Beendigung

Abschließend bestimmen Sie den Zeitraum der Erfassung oder anders gesagt, wann die Datenerfassung wieder beendet werden soll? Anders als die Erfassungsservices starten wir Job Watcher nicht dauerhaft, sondern für ausgesuchte Analysezeiträume. Wir benötigen daher ein oder mehrere Kriterien für die Beendigung. In meinem Beispiel endet die Erfassung nach einer Stunde. Mit „Weiter“ bestätige ich auch diesen Dialog.



Job Watcher starten – Zusammenfassung

In der Zusammenfassung finden wir alle vorgenommenen Einstellungen und haben die Möglichkeit, uns die Befehlszeichenfolge anzusehen:

```
QSYS/SBMJOB
CMD(CALL PGM(QSYS/QPYSTRCLTR)
PARM(('QSYS/STRJW DFN(Q10SECSQL)
COL(JW20120416) LIB(JWDATACR)
TEXT('''') RPLDTA(*YES) COLITV(10.0)
ENDCOL((*NBRSEC 3600) )') 114 'QIBM_JW_JWDATACR/JW20120416'))
JOB(QPYJWCOL) JOBD(QGPL/QDFTSVR) JOBQ(*JOBD)
SYSLIBL(*SYSVAL) CURLIB(*CRTDFT) INLLIBL(*JOBD) DSPSBMJOB(*NO)
MSGQ(*NONE) SRTSEQ(*SYSVAL) LANGID(*SYSVAL) CNTRYID(*SYSVAL)
CCSID(*SYSVAL) INLASPGRP(*JOBD) ALWMLTTHD(*NO)
```

Anhand der Befehlszeichenfolge ist zu erkennen, dass das Programm QPYSTRCLTR aufgerufen wird. Der auf diesem Weg gestartete Job heißt QPYJWCOL, und verwendet wird hierfür die Jobbeschreibung QDFTSVR. Wenn Sie diese Eingabe mit „Fertig stellen“ bestätigen, müssten Sie anschließend den Job QPYJWCOL im Subsystem QBATCH sehen.

2.11.2.1
Seite 6

Job Watcher ist gestartet

Der Job läuft, wenn Sie nicht eingreifen, im Subsystem QBATCH. Wenn Sie den Job Watcher-Job in einem anderen Subsystem starten wollen, haben Sie eigentlich nur zwei Möglichkeiten, um einzugreifen:

- Sie halten die Jobwarteschlange QBATCH an und verschieben manuell den Job in eine andere Jobwarteschlange, beispielsweise in die Jobwarteschlange QPGMR.
- Sie verwenden die gezeigte Befehlszeichenfolge, um mit Hilfe eines Scheduler-Jobs oder über CL-Programme eigene Aufrufe zu generieren. Dazu mehr im nächsten Kapitel.

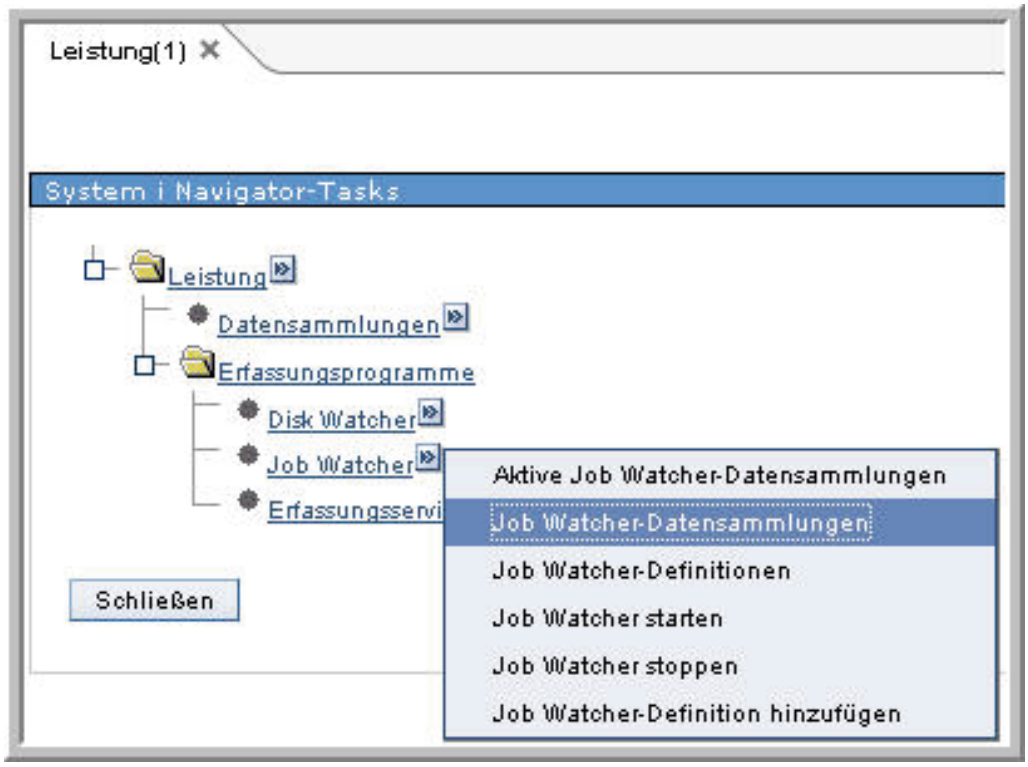
Prüfen wir abschließend noch die Datenlage. Alle Job Watcher-Dateien finde ich in der Bibliothek JWDATACR.

Die erfassten Daten finde ich für mein Beispiel in verschiedenen Tabellen der Bibliothek JWDATACR:

Die Prüfung

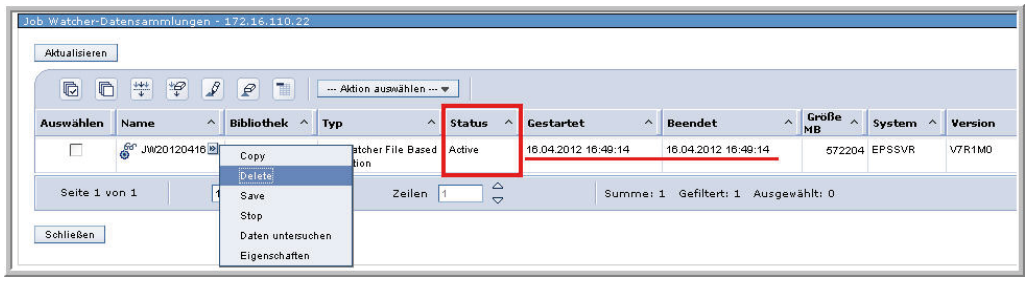
Wie geplant werden unsere Daten in der Bibliothek JWDATACR gesammelt. Alle Job Watcher-Datendateien beginnen im Übrigen mit der Zeichenfolge QAPYJW. Auswerten können wir die Daten später mit der Weboberfläche. Sie können aber auch mit SQL auf die Informationen zugreifen. Dies ist allerdings nur dann zu empfehlen, wenn Sie sehr genau mit den Inhalten der Dateien vertraut sind und wissen, welche Informationen Sie abrufen müssen, um entsprechende Ergebnisse zu erhalten. Viel einfacher ist die Verwaltung

der Datensammlung und auch die spätere Auswertung mit der Weboberfläche. System i Navigator kennt bislang keine Möglichkeiten mit den Job Watcher-Daten zu arbeiten. Nutzen wir also die Weboberfläche:



Job Watcher-Datensammlungen verwalten

Wie in der oberen Abbildung gezeigt, nutzen Sie im Kontextmenü den Eintrag „Job Watcher-Datensammlungen“. Daraufhin öffnet sich der folgende Dialog:



Job Watcher-Datensammlungen verwalten (2)

Hier haben Sie die Möglichkeit, mit den JW-Teildateien zu arbeiten. Sie können Auswertungen sichern oder löschen, aber auch aktive Job Watcher-Auswertungen stoppen. Auffällig ist auch, dass Sie trotz aktiver Erfassung bereits die Daten analysieren könnten. Sie müssen also nicht auf das Ende der Erfassung warten!

2.11.2.1

Seite 8

Job Watcher-Datensammlungen analysieren

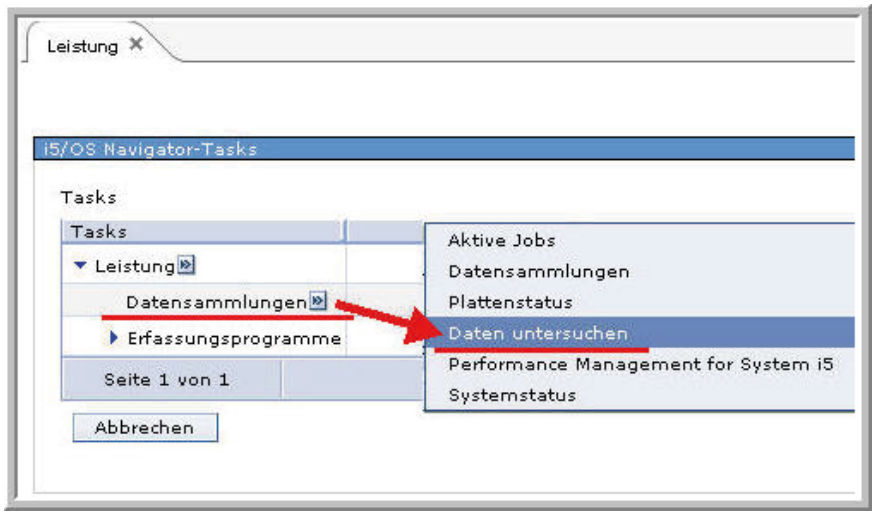
In den vorangegangenen Kapiteln haben wir Analysedaten auf verschiedene Arten mit dem Job Watcher erfasst. Nun wollen wir diese Daten analysieren und uns dabei v. a. die im Job Watcher zur Verfügung stehenden Funktionen anschauen. Wie werden die einzelnen Charts aufgerufen? Wie analysieren wir scheinbar interessante Zeiträume?

Im Vordergrund steht dabei nicht die Interpretation der Daten. Dies würde den Rahmen des Kapitels sprengen. Hinzu kommt, dass eine Interpretation neben persönlicher Erfahrung auch eine genaue Kenntnis des jeweiligen Servers erfordert. Sie können die Daten halt nur in Abhängigkeit der vorhandenen Hardware und vor allem unter Berücksichtigung der installierten Programme interpretieren. Generelle Richtlinien und allgemeine Formeln helfen heute angesichts immer komplexer werdender Applikationen nicht mehr weiter.

Am einfachsten geht es, wenn Sie die Leistungsdaten nicht erst erfassen, wenn die Probleme bereits vorhanden sind. Erfassen Sie die Daten frühzeitig! Am besten sofort nach Inbetriebnahme des neuen Servers oder sofort nach einer Hauptspeichererweiterung. So stehen Ihnen später genügend Informationen zur Verfügung und Sie können schnell anhand entsprechender Vergleiche erkennen, welche Applikationen jetzt mehr Ressourcen benötigen oder ob sich die Antwortzeiten Ihrer Platten verändert haben.

Schauen wir uns jetzt erst einmal die Möglichkeiten an, die der Job Watcher uns für die Analyse anbietet.

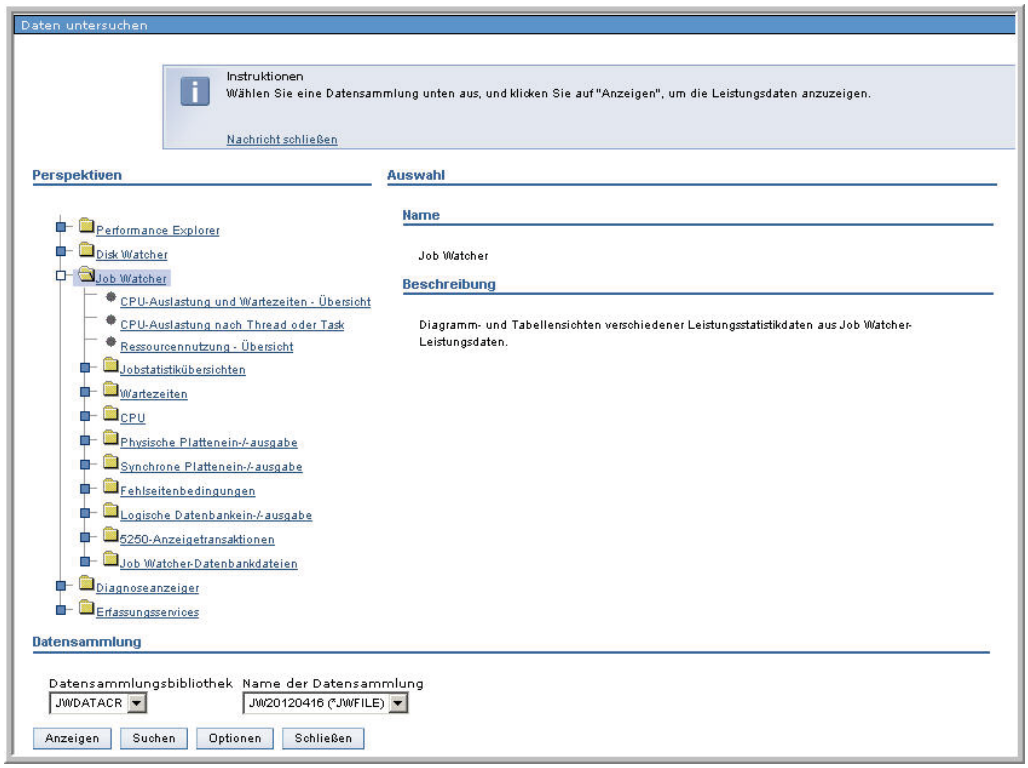
Als Erstes benötigen wir wieder eine Verbindung zum IBM Systems Director Navigator for i5/OS. Da wir bereits in den vorangegangenen Kapiteln die Systemvoraussetzungen und den Anmeldevorgang besprochen haben, sollten Sie problemlos das Menü „Leistung“ öffnen können:



Jobwatcher – Datensammlungen

2.11.2.2
Seite 2

Sie können in der oberen Abbildung sehen, dass ich wieder den Eintrag „Datensammlungen“ öffne, aber jetzt den Punkt „Daten untersuchen“ anwähle. Daraufhin erscheint ein neuer Dialog:

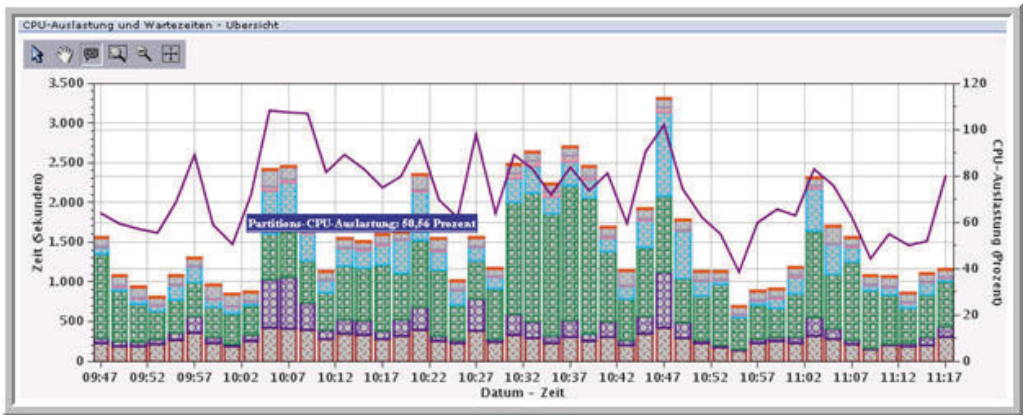


Datensammlung auswählen

Wichtig ist hier die korrekte Auswahl der Datensammlung. Sie können in der oberen Abbildung erkennen, dass ich zunächst die Bibliothek „JWDATA“ auswähle. Anschließend stehen mir die verschiedenen Teildateien zur Analyse zur Verfügung. Wichtig hierbei ist, dass Sie die Teildateien gut unterscheiden können; insbesondere dann, wenn Sie zwei unterschiedliche Zeiträume miteinander vergleichen wollen. Ich wähle eine der vorhandenen Datensammlungen aus.

Nun müssen wir uns noch für eine der Analysen entscheiden. Ein geeigneter Einstieg in die Analyse ist es meist, die Übersicht der CPU-Auslastung mit den aufgetretenen Wartezeiten zu vergleichen. Vergessen Sie nicht auf den Button „Anzeigen“ zu klicken, sonst passiert nämlich gar nichts.

Wenn alles funktioniert hat, erhalten Sie einen Überblick über die CPU-Auslastung und die Wartezeiten des Analysezeitraums.



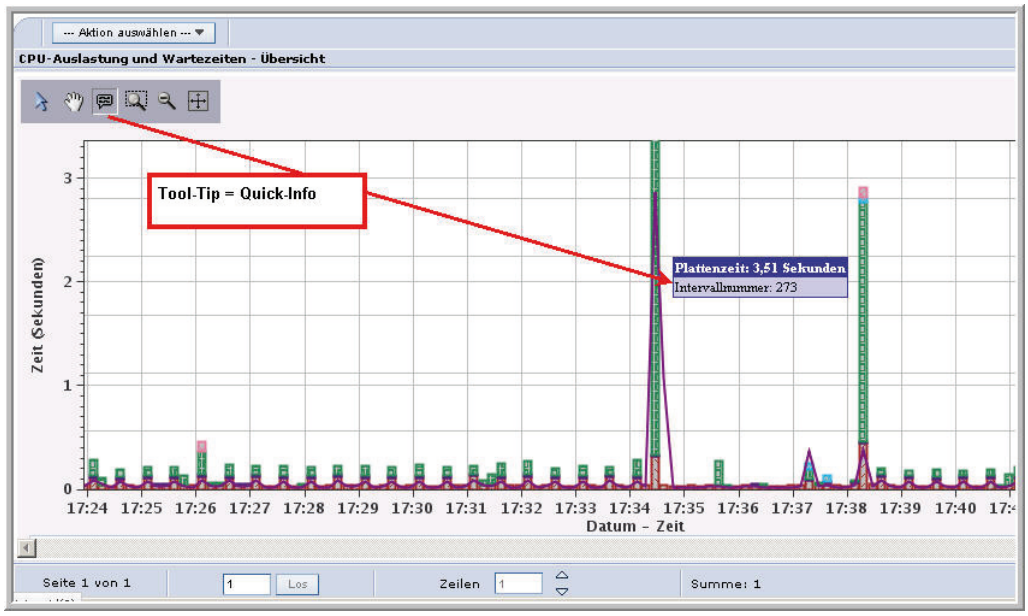
Die Übersicht der Wartezeiten

Falls bei Ihnen keine grafische Darstellung erfolgt und Sie nur Zahlenkolonen sehen, sollten Sie nochmals die Installationsvoraussetzungen prüfen!

Dargestellt wird ein Basisdiagramm, das die CPU-Auslastung einerseits und die Wartezeiten andererseits in 2-Minuten-Intervallen anzeigt. Die Zeit in Sekunden für die verschiedenen Wartezeiten kann somit auch nur ein Mittelwert aller erfassten Intervalle dieses Zeitraums sein. Sie erinnern sich, dass wir alle 10 Sekunden die Daten erfasst habenim vorangegangenen Kapitel erfasst haben.

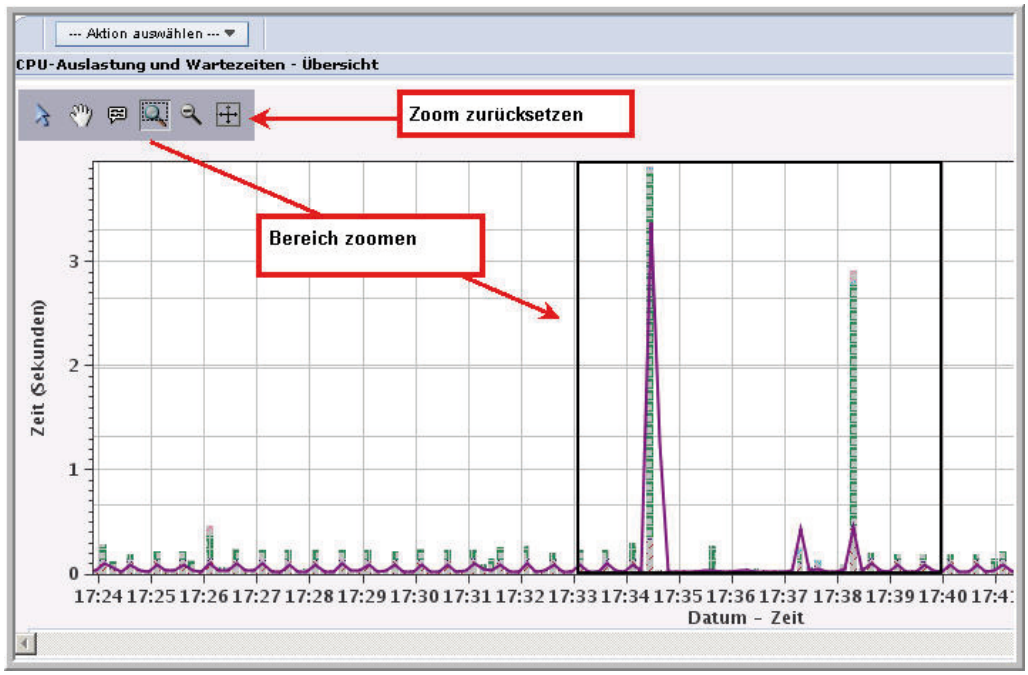
Oben links am Bildschirm finden Sie eine Toolbar. Klicken Sie auf das mittlere Icon (Tool-Tipp). Jetzt können Sie den Mauszeiger auf einen beliebigen Punkt der Auswertung ziehen und erhalten immer einen Hinweis mit Informationen über die genutzte Zeit in Sekunden und die Bedeutung des entsprechenden Eintrags.

2.11.2.2
Seite 4



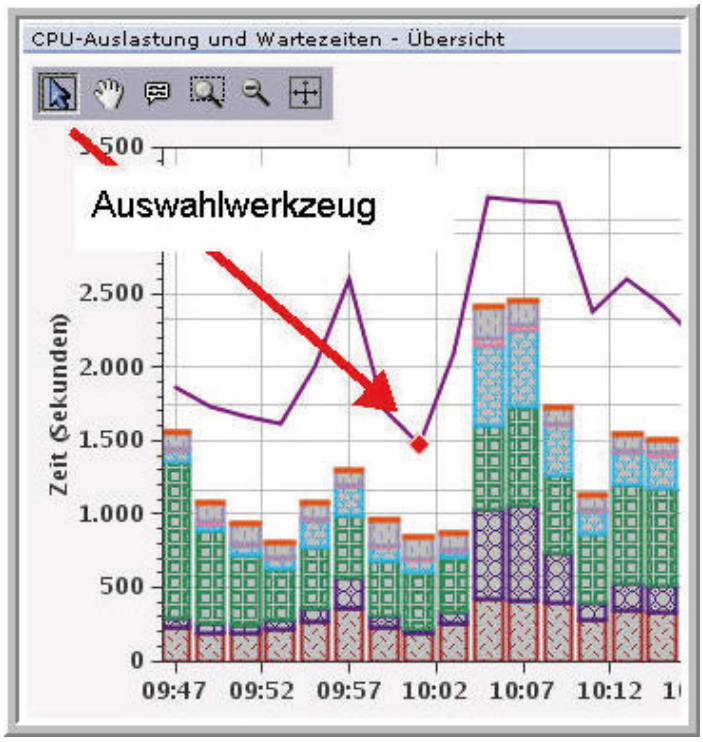
Tool-Tipp

Anhand der oberen Abbildung können Sie gut erkennen, dass die CPU-Auslastung auf diesem Server stark schwankt. Wenn wir jetzt einen ausgewählten Zeitraum betrachten wollen – beispielsweise, die Zeit zwischen 17:34 und 17:36, so können wir den Zeitraum zoomen.



Zoom-Bereich

Durch das Zoomen haben Sie jedoch nicht diesen Bereich für weitere Analysen ausgewählt. Sie haben lediglich die Ansicht optisch vergrößert. Sollen nur einzelne Zeiträume für weitere Analysen ausgewählt werden, dann müssen Sie die zu verwendenden Datenpunkte mit Hilfe eines anderen Werkzeugs auswählen:

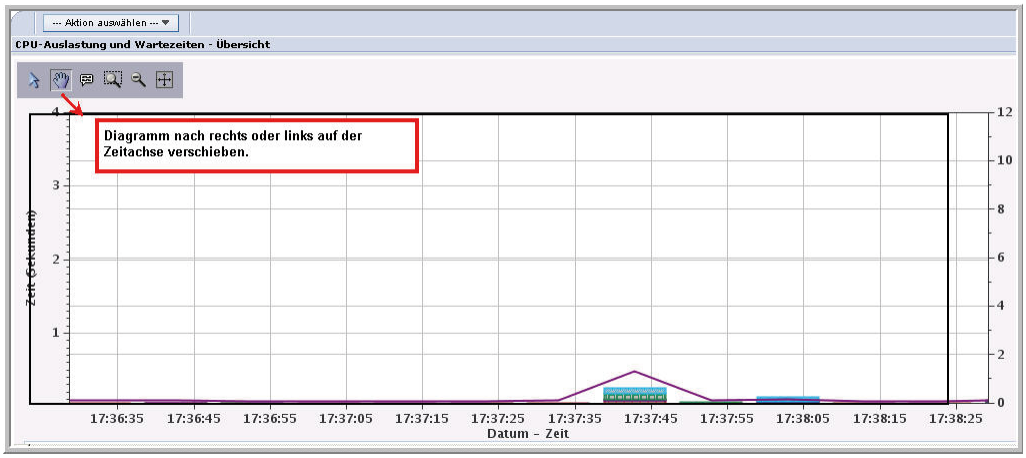


Auswahl einzelner Datenpunkte

Mit Hilfe des Auswahlwerkzeugs können wir ein oder mehrere Intervalle durch einen entsprechenden Punkt auf der CPU-Auslastungskurve markieren. Alle weiteren Auswertungen beziehen sich jetzt nur noch auf das jeweilige Intervall bzw. die jeweiligen Intervalle. Seine/ihre Daten werden als Input-Werte für alle folgenden Detailausgaben verwendet. Falls Sie keine Intervalle kennzeichnen, werden stets alle Daten der Datensammlung herangezogen. Dies gilt auch dann, wenn Sie das Zoom-Werkzeug verwenden, um ein oder mehrere Intervalle „auszuschneiden".

2.11.2.2
Seite 6

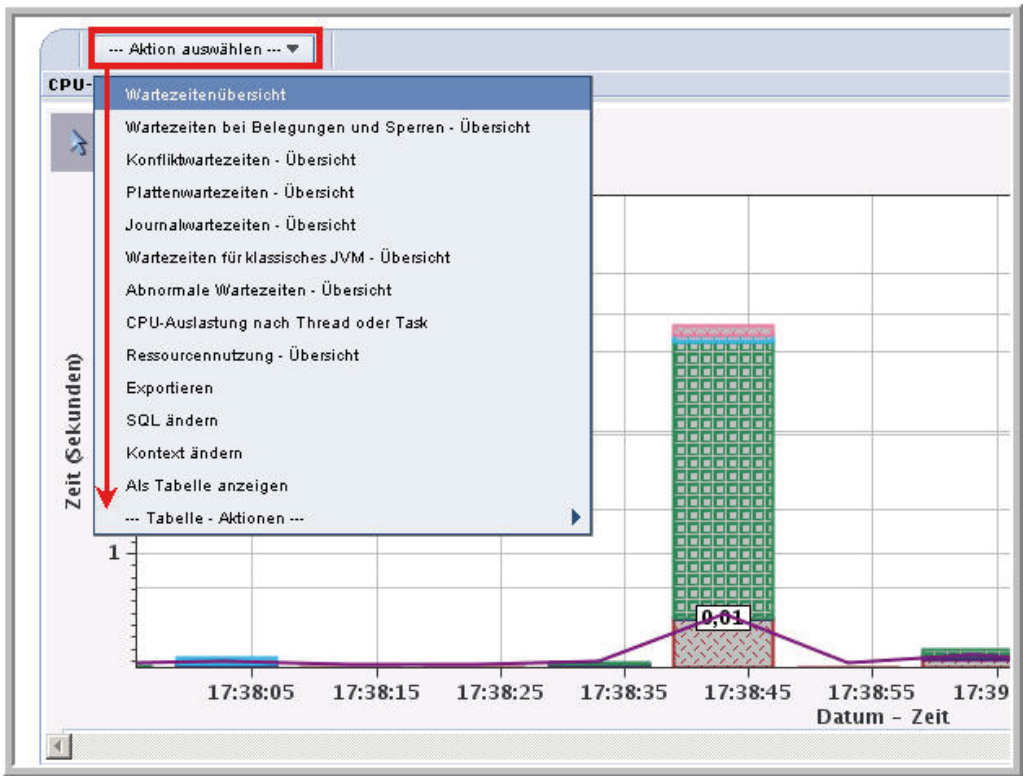
Schauen wir uns nun noch das letzte Icon der Toolbar an.



Zeitachsen verschieben

Mit Hilfe dieses Icons können Sie die Zeitachsen nach rechts bzw. links verschieben und so andere Zeiträume in den Focus ziehen.

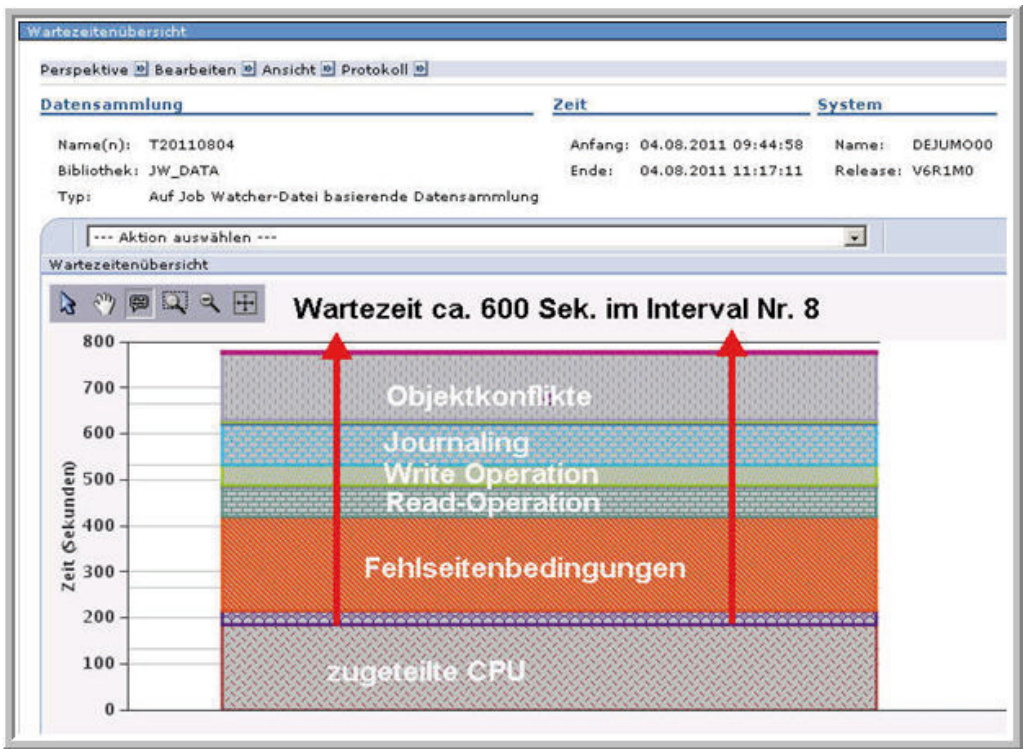
Nachdem wir uns nun mit den Werkzeugen der Toolbar vertraut gemacht und uns ein wenig über die schwankende CPU-Auslastung gewundert haben, versuchen wir diese Schwankungen detaillierter zu betrachten:



Detailanalysen starten

Über eine Aktionsliste am oberen Bildschirmrand stehen weitere Auswertungen zur Verfügung. Die Informationen variieren in Abhängigkeit des jeweiligen Ausgangsbildschirms oder anders gesagt, mit jeder weiteren Analyse über die Aktionsliste bewegen Sie sich ein Stück „tiefer“ in die Auswertungen hinein.

Ich entscheide mich zunächst dafür, die Wartezeiten etwas genauer zu betrachten und lasse mir eine Wartezeitenübersicht anzeigen:



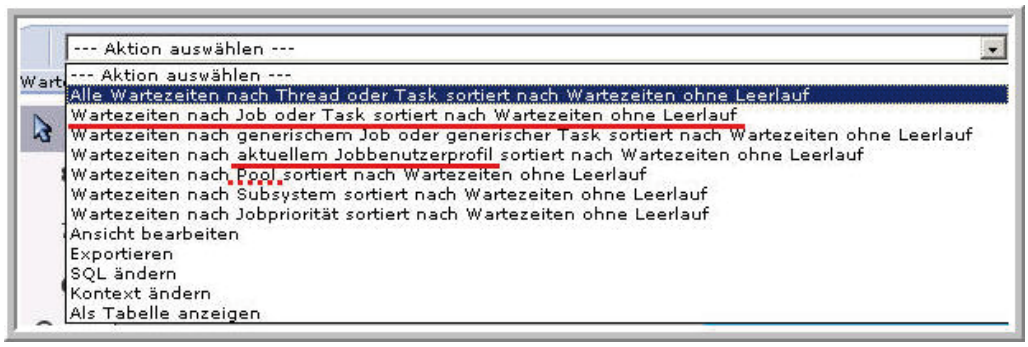
Wartezeitenübersicht

Zur besseren Übersicht habe ich zuvor ein einzelnes Intervall ausgewählt. Dieses Intervall zeigt in der Übersicht ca. 600 Sekunden Wartezeiten an. Lassen Sie mich hier kurz ein wenig ausholen: Das ausgewählte Intervall zeigt, dass alle Jobs, die zu diesem Zeitpunkt liefen, insgesamt 789 Sekunden Zeit beansprucht haben. Diese Laufzeit unterteilt sich in 186,28 Sekunden CPU-Verarbeitungszeit zzgl. 29,43 Sekunden CPU-Queuing. An diesen Zeiten können wir nichts ändern. Aber die Jobs mussten insgesamt 573,29 Sekunden warten, weil beispielsweise Daten von der Festplatte in den Arbeitsspeicher zurückverlagert wurden (= Fehlseitenbedingungen), d. h. der Arbeitsspeicher reichte nicht aus für das Datenvolumen. Auch das Journaling der Dateien nahm eine gewisse Zeit in Anspruch. Zudem gab es Objektkonflikte, weil Sätze oder Objekte gesperrt waren. Hier ergibt sich der Ansatzpunkt unserer Optimierung: Wenn wir in der Lage wären, die Wartezeiten zu eleminieren oder wenigstens zu verkürzen, werden die Jobs schneller ausgeführt werden.

2.11.2.2
Seite 8

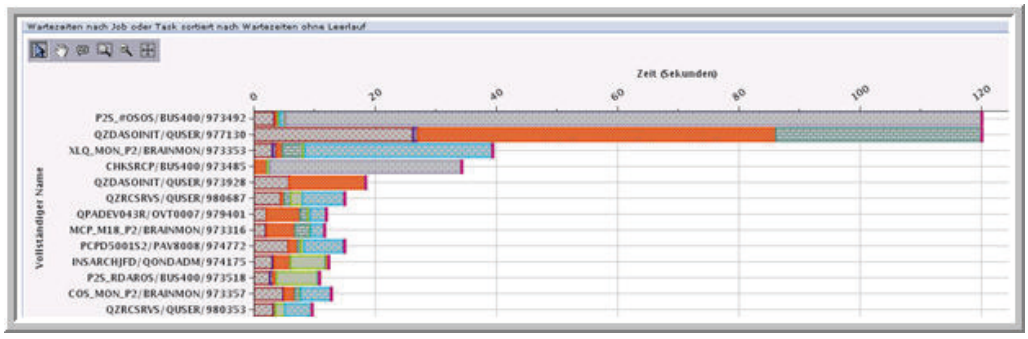
Jetzt müssen weitere Analyseschritte folgen, die in der Regel auf unserer Erfahrung und dem Wissen beruhen, welche Applikationen auf dem Server laufen.

Wir könnten z. B. analysieren, welche Jobs genau die Wartezeiten verursacht haben, indem wir eine Übersicht der Wartezeiten nach Jobs aufrufen:



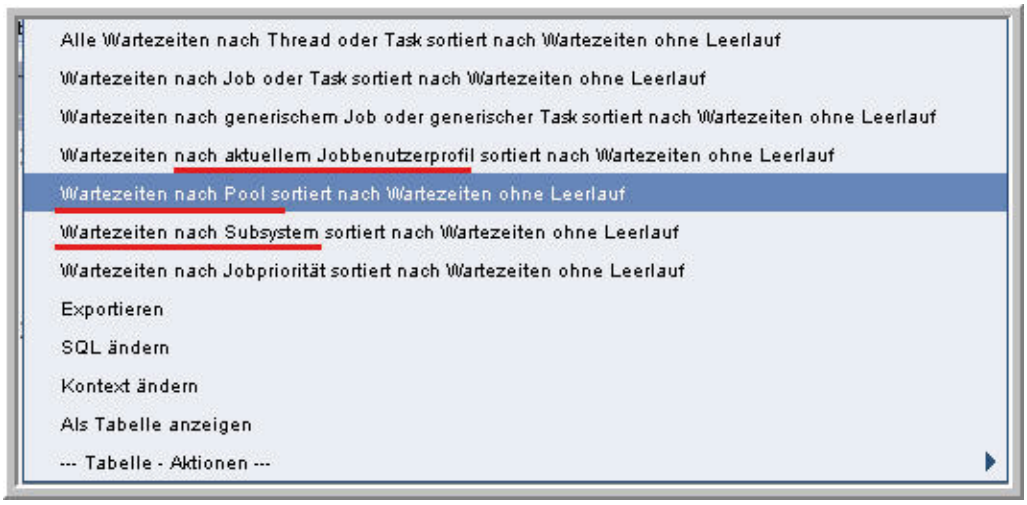
Detaillierte Wartezeitenübersicht

Ich habe mich für eine Übersicht der einzelnen Jobs oder Tasks sortiert nach Wartezeiten entschieden. Die Anzeige ist bereits nach der Wartezeit sortiert, d. h. Jobs mit den höchsten Wartezeiten stehen oben in der Liste.



Wartezeiten nach Job oder Task

Soll die Anzeige nach bestimmten Wartezeiten sortiert werden, können Sie aus der Aktionsliste einen anderen Eintrag wählen.



Weitere Detailübersichten der Wartezeiten

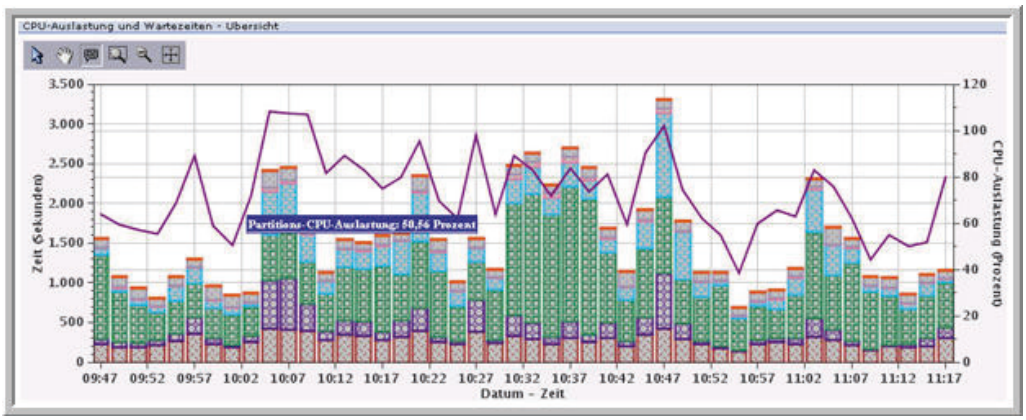
Kommen wir noch einmal auf die Fehlseitenbedingungen zurück. Fehlseitenbedingungen treten immer dann auf, wenn der Server Daten vom Arbeitsspeicher auf die Festplatte auslagert, weil nicht genügend Arbeitsspeicher vorhanden ist. Normalerweise ist das Paging auf unserem Server kein Problem, werden jedoch die Fehlseitenbedingungen zu hoch oder arbeiten wir mit externen Festplatten, könnten wir hier Performanceprobleme bekommen. In der Abbildung „Wartezeiten nach Job oder Task“ konnte man erkennen, dass insbesondere zwei Serverjobs, nämlich QZDASOINIT/QUSER/977130 und QZDASOINIT/QUSER/973928, lange Wartezeiten aufgrund von Fehlseitenbedingungen aufwiesen. Bei beiden Jobs handelte es sich um SQL-Anwender, die SQL-Statements absetzten und dadurch entsprechend große Datenmengen im Arbeitsspeicher hielten. Grundsätzlich aber gibt IBM angesichts der immer komplexer werdenden Applikationen keine technische Empfehlung mehr bzgl. der tolerierbaren Höhe von Fehlseitenbedingungen im Benutzerpool. Vielmehr müssen Sie anhand der Jobs und der Antwortzeiten entscheiden, ob eine Hauptspeichererweiterung notwendig ist. Im gezeigten Beispiel konnte das Problem übrigens sehr schnell durch eine geeignete Indexierung der Datenbank behoben werden!

2.11.2.2
Seite 10

Konzentrieren wir uns nun auf die Objektzugriffskonflikte. Die Konfliktzeit betrug 147 Sekunden. Auch hier bietet es sich an, erst einmal zu analysieren, welche Jobs diese Objektkonflikte verursacht haben:

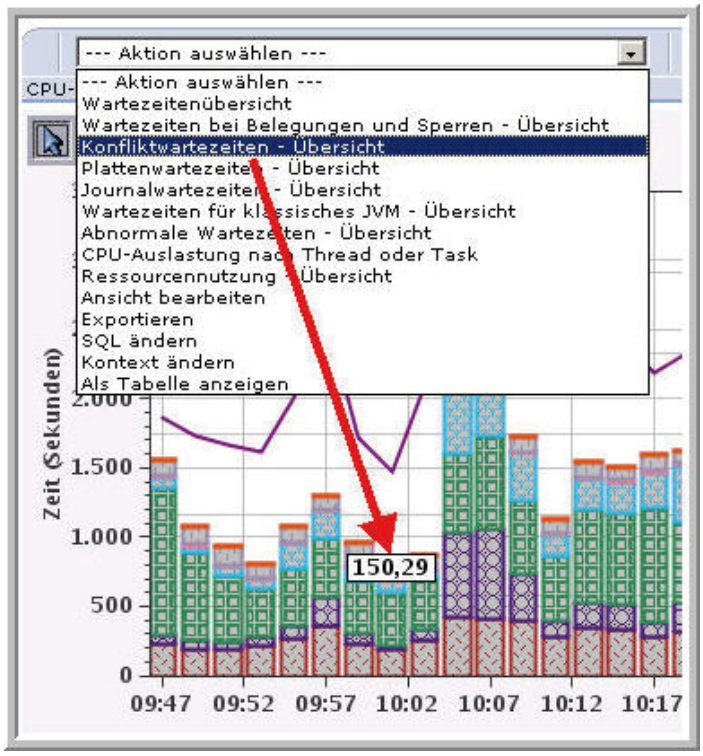
Detailansicht Konfliktwartezeiten

Die Konfliktwartezeiten betreffen vor allem zwei Jobs. Einer der Jobs ist ein BUS/400-Job. Wir können hier noch weiter analysieren und herausfinden, auf welche Objekte solange gewartet werden muss. Dazu kehren wir aber erst einmal zur Ausgangsbasis zurück. Es gibt allerdings keinen Button „Abbrechen". Die aktuelle Anzeige beenden Sie stets, indem Sie auf den Button „Fertig" klicken. Auf diesem Weg und mit ein wenig Geduld gelangen Sie zurück zum Ausgangsbildschirm:



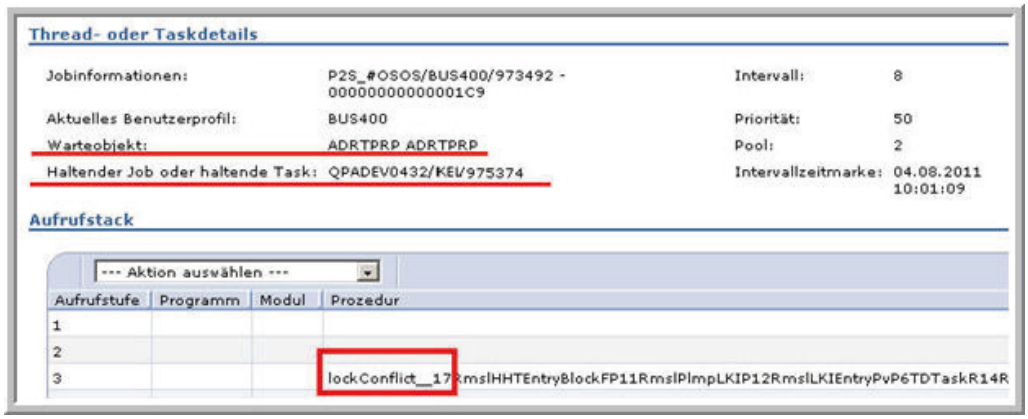
Die Ausgangsbasis

Wieder markieren wir das relevante Intervall Nummer 8. Dieses Mal wählen wir jedoch die Auswertung „Konfliktwartezeiten – Übersicht“:



Konfliktwartezeiten – Übersicht

Anschließend öffnen wir die Auswertung „Alle Wartezeiten für einen Thread oder eine Task“, die uns daraufhin zur Verfügung steht:



Konfliktwartezeiten – Thread-Details

Immer dann, wenn Intervalldetails für einen Thread/Task oder für einen Job/Task als Auswahl zur Verfügung stehen, sehen Sie den Aufrufstack des entsprechenden Jobs und können sehr genau identifizieren, welcher Job und welche Objekte die entscheidenden Objektkonflikte auslösen. Im gezeigten Beispiel wird beispielsweise die Datei ADRTPRP vom User KEI gesperrt, so dass die Daten nicht gespiegelt werden können. Hier wäre zu prüfen, warum die Datei so lange gesperrt wird und ob es sich dabei um einen Anwendungs- oder einen Anwenderfehler handelt. Mehr Informationen können wir aus dieser Auswertung zurzeit nicht mitnehmen. Also kehren wir mit dem Button „Fertig“ wieder zum Ausgangsbildschirm zurück.

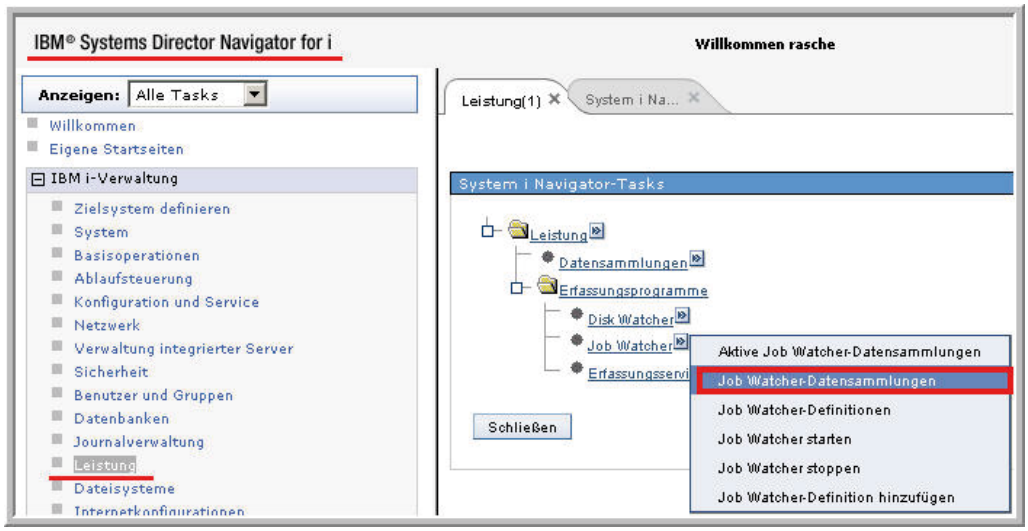
Fazit: An die Bedienung der Webtools muss man sich sicherlich gewöhnen. Doch es steht außer Frage, dass diese Tools viele unterschiedliche Analysemöglichkeiten bieten. Die Zeiten langer Zahlenkolonnen sind damit endgültig vorbei. Aber auch die visualisierten Daten müssen interpretiert werden und hier liegt vielleicht das eigentliche Problem. Woher wissen wir, dass die Fehlseitenbedingungen zu hoch sind oder die Plattenantwortzeiten zu niedrig? Diese Fragen können Sie nur anhand entsprechender Vergleichszeiträume beantworten sowie mit dem entsprechenden Kenntnisstand bzgl. Ihres Servers!

Job Watcher-Datensammlungen importieren

Es kann vorkommen, dass Sie die erstellten Datensammlungen nicht auf dem Produktionssystem analysieren können. Sei es um den Server nicht unnötig zu belasten oder weil Sie vielleicht einen fremden Server analysieren – so wie es bei mir häufig der Fall ist. In diesen Fällen sichern Sie sich die Leistungsdaten und spielen sie anschließend auf einem anderen Server wieder ein.

Bei der Datensicherung müssen Sie lediglich auf den Releasestand des Zielservers achten. Auf dem Zielsystem können Sie jedoch die Daten nicht einfach mit dem Befehl RSTLIB (Restore Library) wiederherstellen; Job Watcher würde die Datensammlungen anschließend nicht finden.

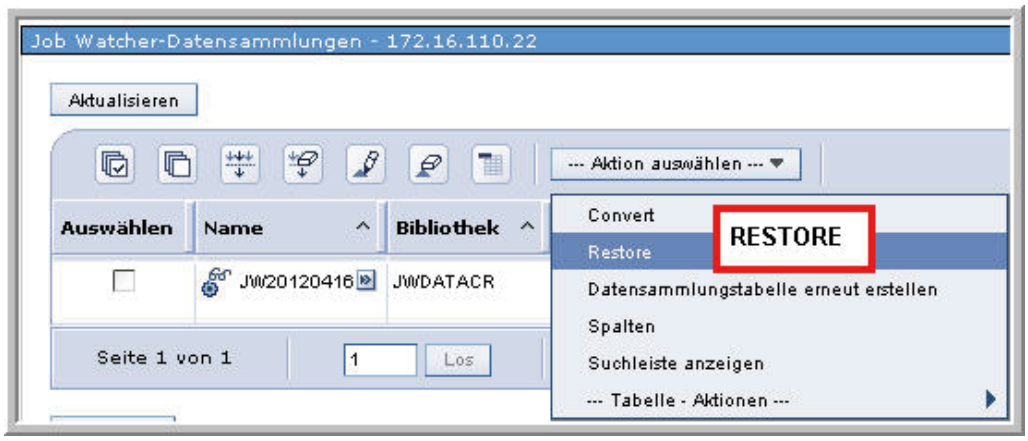
Schauen wir uns an, wie Sie Job Watcher-Datensammlungen importieren können. Im Beispiel habe ich eine Sicherungsdatei (Savefile) mit der Bibliothek TEST, in der sich die Datensammlungen eines Kunden befinden. Um die Daten zu analysieren, muss ich als Erstes die Savefile per FTP auf unseren Server übertragen. Anschließend melde ich mich beim IBM Systems Director Navigator for i an und öffne die Leistungstasks. Anschließend öffnen Sie die Datensammlungen, wie folgende Abbildung zeigt:



Job Watcher-Datensammlungen öffnen

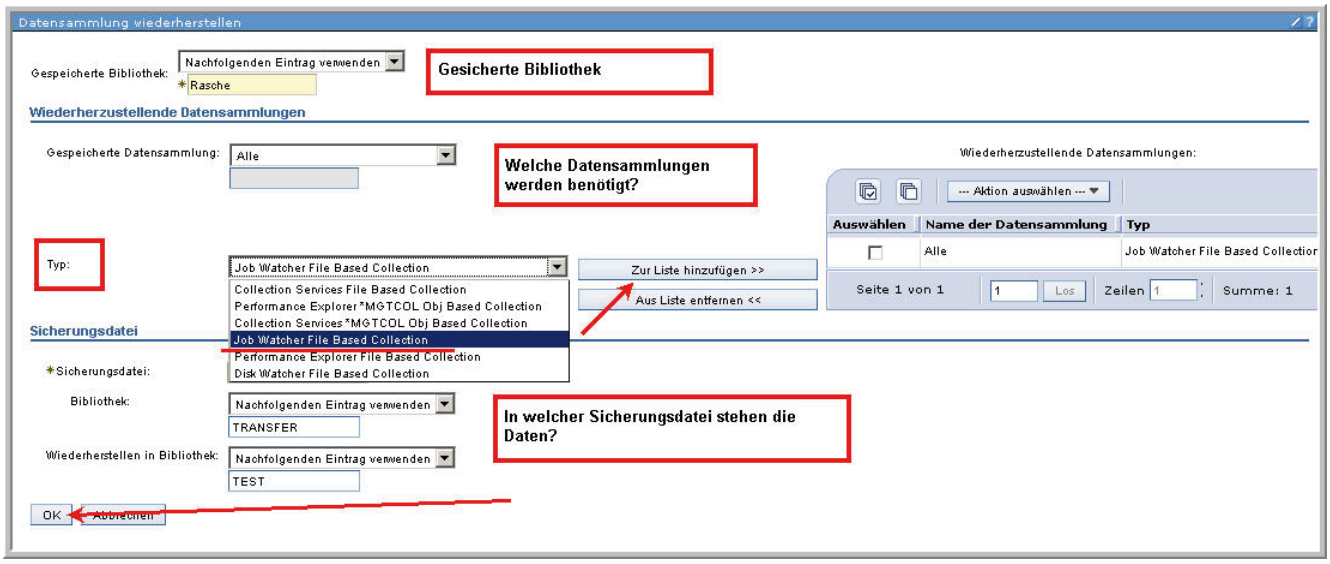
2.11.2.3 Seite 2

Wenn alles korrekt ist, müssten sie jetzt den folgenden Bildschirm sehen:



Übersicht der Job Watcher-Datensammlungen

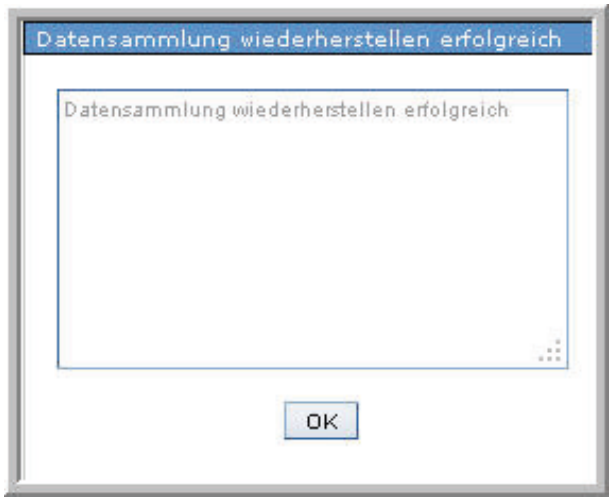
Wie die obere Abbildung zeigt, können Sie jetzt über die Aktionsleiste den Punkt RESTORE auswählen. Es öffnet sich der Dialog zur Rücksicherung der Daten:



Job Watcher-Datensammlungen wiederherstellen

Zunächst müssen Sie den Namen der gesicherten Bibliothek angeben. Anschließend können Sie wählen, ob alle gesicherten Datensammlungen oder spezifische Datensammlungen wiederhergestellt werden sollen. Wichtig ist, dass Sie den Typ der Datensammlung korrekt benennen. In meinem Fall handelt es sich um verschiedene Job Watcher-Datensammlungen, die in der Bibliothek RASCHE gesichert wurden. Vergessen Sie nicht, die Angaben in die rechts stehende Liste zu übernehmen. Abschließend müssen Sie noch bekanntgeben, wo sich die Sicherungsdatei befindet und in welche Bibliothek die Daten zurückgespeichert werden. Erst wenn diese Angaben vollständig gemacht wurden, können Sie „OK“ drücken.

Je nachdem, wie viele Datensammlungen wiederhergestellt werden müssen, kann der Prozess durchaus einige Minuten dauern. Sobald die Rücksicherung beendet ist, erhalten Sie aber ein Fenster mit einer hoffentlich positiven Nachricht:

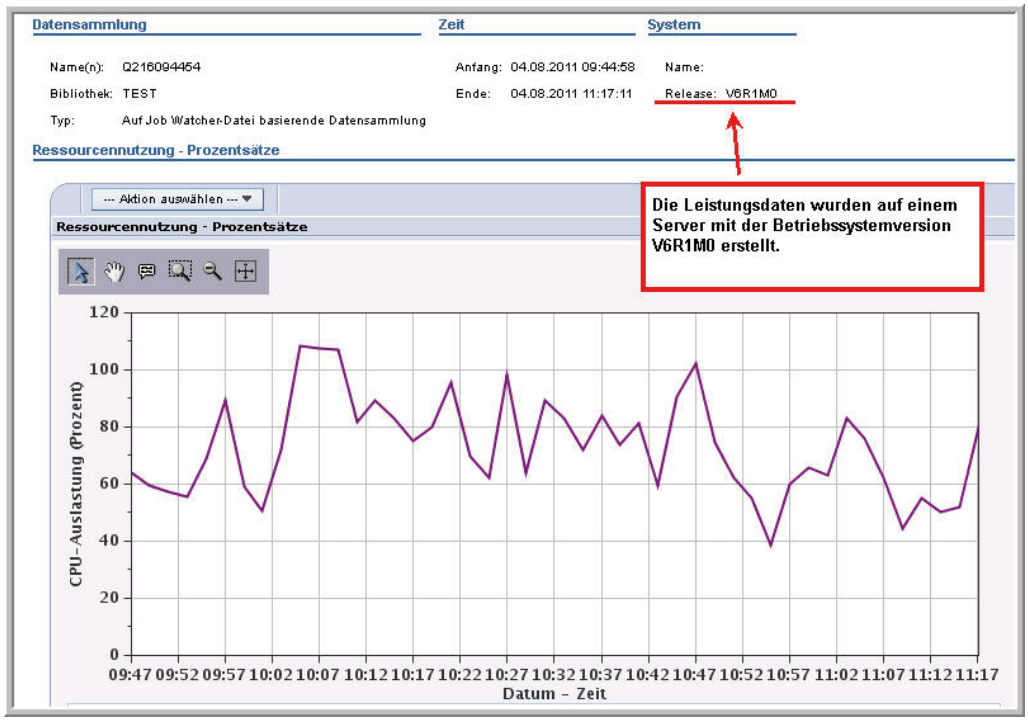


Wiederherstellung der Datensammlung erfolgreich

Bestätigen Sie den Dialog mit „OK“, so dass Sie anschließend wieder in den Ausgangsbildschirm zurückkehren. Hier müssen Sie den Bildschirm noch einmal aktualisieren bevor die importierten Datensammlungen angezeigt werden.

2.11.2.3
Seite 4

Anschließend können Sie die Daten für Untersuchungen sofort verwenden:



Datensammlungen können sofort genutzt werden

In der oberen Abbildung können Sie deutlich erkennen, dass die Leistungsdaten auf einem Server mit dem Release i6.1 erstellt wurden. Der Server, auf dem die Daten untersucht werden, hat die Betriebssystemversion i7.1.

System i Navigator-Monitore

Monitore im System i Navigator geben Ihnen Auskunft über die Performance Ihres Systems. Sie erfahren, was gerade auf Ihrem Server los ist. Grafiken visualisieren die aktuellen Geschehnisse. Dabei können Sie auf die Daten des Erfassungsservice zurückgreifen oder davon unabhängige Monitore definieren. Zusätzlich können Sie diese Monitore aber auch als „Warnsystem" nutzen, wenn eine spezifische Situation auftritt. Im folgenden Kapitel wollen wir uns die Möglichkeiten der verschiedenen Monitore im Überblick anschauen. Dabei können wir nicht alle Möglichkeiten detailliert aufführen, sondern müssen uns mit einigen Beispielen begnügen, so dass Sie wissen, wie Monitore konfiguriert und anschließend visualisiert werden. Welche Monitore in Ihrem Unternehmen nützlich sind, müssen Sie anschließend selbst entscheiden, im System i Navigator stehen uns diverse Monitore zur Verfügung, um verschiedene Situationen zu durchleuchten.

System-Monitor
Mit einem System-Monitor können Sie Ihre System-Performance global überwachen. Dafür stehen eine Reihe vordefinierter Analysen zur Verfügung. So könnten Sie z. B. die durchschnittliche CPU-Auslastung untersuchen. Bei der Analyse können Sie auf jeden Datenpunkt der Grafik klicken, um Detailinformationen über die Jobs zu erhalten, die die meiste CPU-Zeit beanspruchen. Und selbstverständlich können Sie nach einem Rechtsklick mit der Maus sofort mit dem Job weiterarbeiten. Außerdem nutzt der System-Monitor die Daten der Erfassungsservices, oder anders gesagt: Mit Hilfe des System-Monitors können Sie auch diese Analysedaten auswerten.

Job-Monitor
Der Job-Monitor ist hilfreich, wenn Sie einen bestimmten Job, bestimmte Jobtypen oder ein Subsystem überwachen wollen. Auch hier gibt es wieder viele vordefinierte Analysen, um die Performance, aber auch den Status oder die Fehlermeldungen dieser Jobs zu überwachen. Auch der Job-Monitor kann Daten der Erfassungsservices visualisieren und auswerten.

Nachrichten-Monitor
Um herauszufinden, ob eine Applikation erfolgreich ausgeführt wurde, könnten Sie einen Nachrichten-Monitor starten. Aus diesem Monitor heraus können Sie Nachrichtendetails ansehen, aber auch auf eine Nachricht antworten oder sie einfach löschen, wenn sie nicht mehr gebraucht wird.

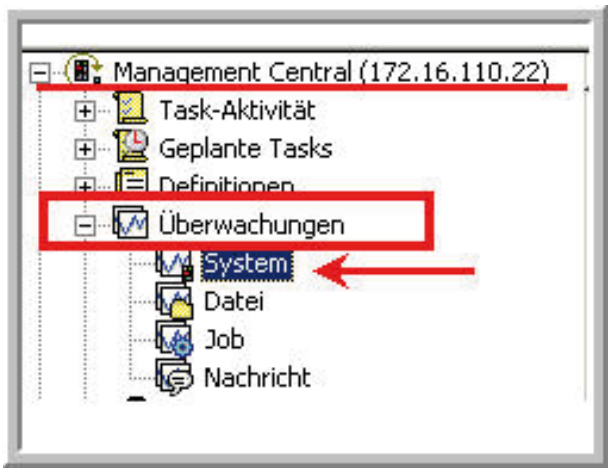
2.11.3
Seite 2

B2B-Monitor
Wenn Sie in Ihrem Unternehmen B2B-Transaktionen durchführen, kann dieser Monitor helfen die B2B-Transaktionen zu überwachen. Es können zudem automatische Befehle ausgelöst werden, wenn definierte Schwellenwerte erreicht werden. Sie können aber auch spezifische Transaktionen suchen, visualisieren und detalliert untersuchen.

Datei-Monitor
Datei-Monitore überwachen beispielsweise Modifikationen an Dateien und informieren Sie über diese Ereignisse.

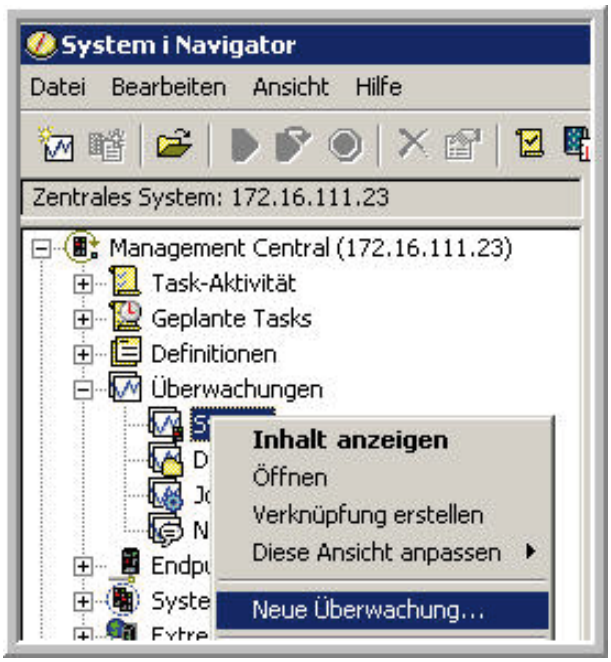
System i Navigator – einen Monitor konfigurieren

Um einen neuen Monitor zu definieren, müssen Sie zunächst einmal eine Verbindung zum zentralen System im System i Navigator herstellen. Anschließend öffnen Sie den Punkt „Überwachungen“.



Verbindung zum Management Central

Um nun einen neuen Monitor zu konfigurieren, klicken wir mit der rechten Maustaste auf den Eintrag „System“ und wählen „Neue Überwachung“, wie in der folgenden Abbildung gezeigt.



Neue Systemüberwachung konfigurieren

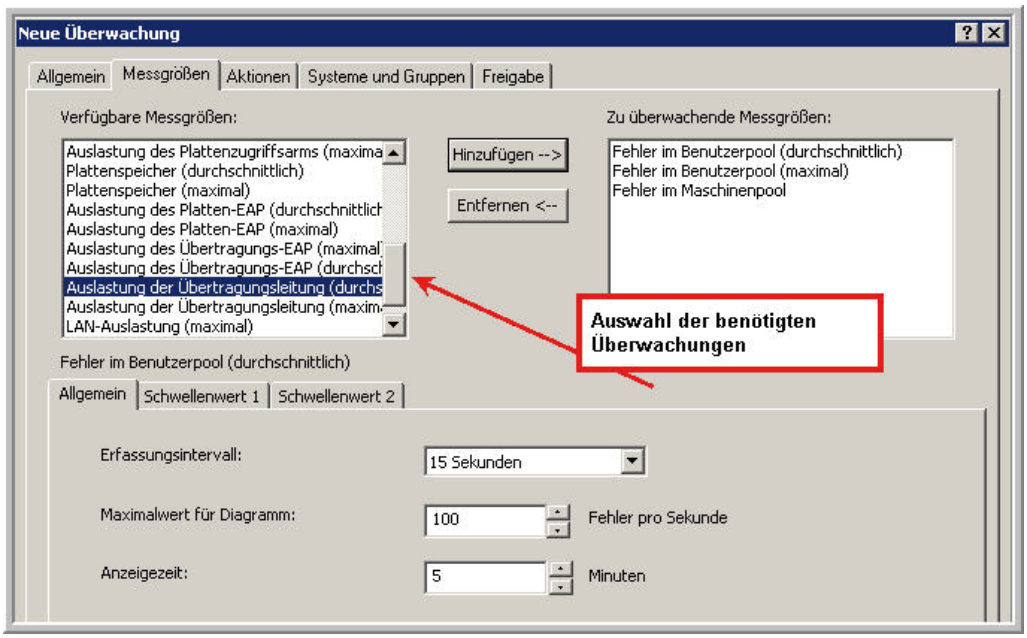
2.11.3.1
Seite 2

Daraufhin erhalten Sie einen neuen Dialog, in dem Sie einen Namen vergeben und natürlich den beschreibenden Text hinzufügen:

Neue Systemüberwachung konfigurieren – Allgemein

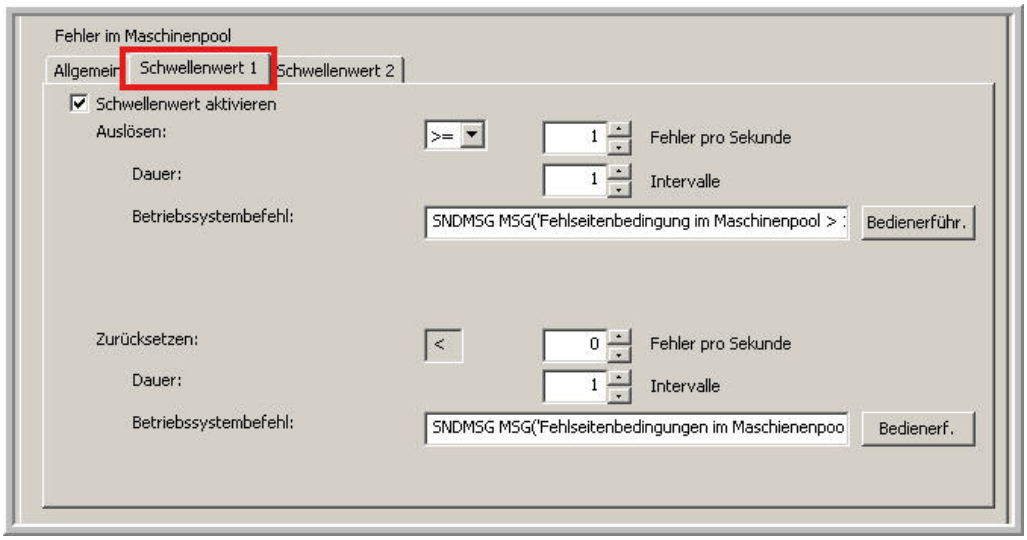
Im Register „Messgrößen" wählen Sie die Analysedaten:

Sie legen fest, welche Informationen benötigt werden, wie häufig die Erfassung der Daten ausgeführt werden soll, welches der Maximalwert im späteren Diagramm sein soll und welche Zeitintervalle im Diagramm dargestellt werden sollen. Ich analysiere Im folgenden Beispiel die Fehlseitenbedingungen alle 15 Sekunden und möchte alle 5 Minuten Datenpunkte erstellt haben, um darüber weitere Informationen zu erhalten. Der Maximalwert meiner Y-Achse im Diagramm soll der Wert 200 sein.



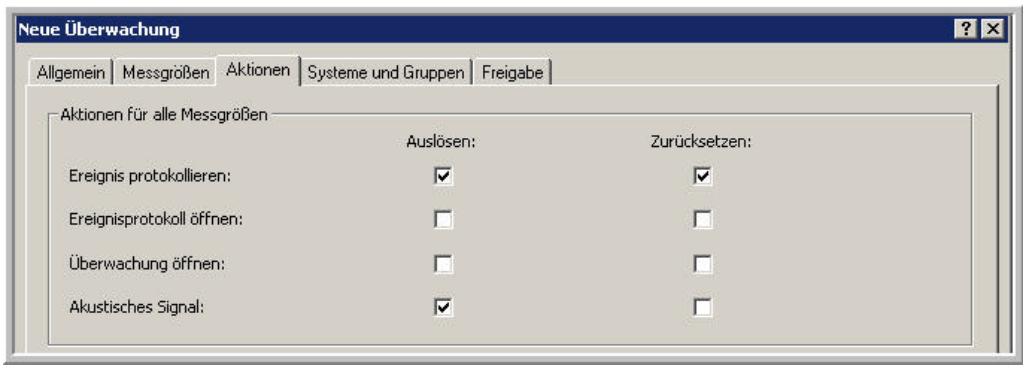
Neue Systemüberwachung konfigurieren – Messgrößen

Für jede Auswertung können sie zudem Schwellenwerte definieren. Sind diese Schwellenwerte erreicht, wird der Befehl ausgelöst, den Sie im Parameter „Betriebssystembefehl“ hinterlegt haben. Grundsätzlich wäre hier natürlich auch ein Programmaufruf denkbar oder der Start einer Job Watcher-Erfassung, um die augenblickliche Situation detaillierter zu analysieren. Sinkt der Schwellenwert wieder, könnten Sie z. B. die Job Watcher-Erfassung automatisch beenden. Ich begnüge mich im unteren Beispiel mit einer einfachen Nachricht an den Operator.



Neue Systemüberwachung konfigurieren – Schwellenwerte

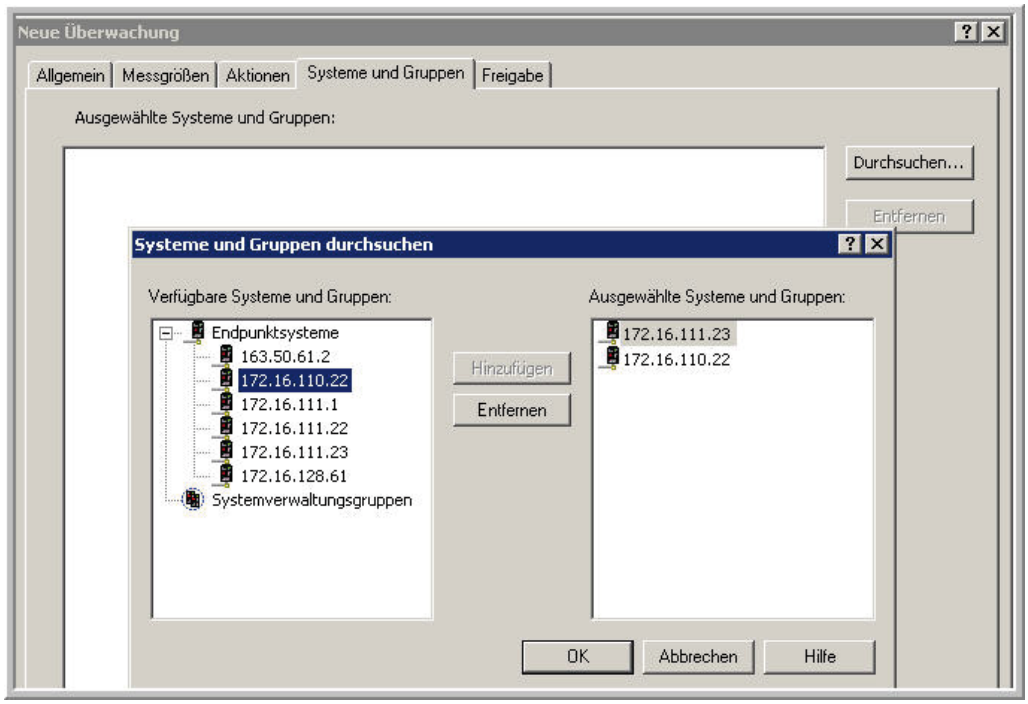
Jetzt können Sie im Register weitere „Aktionen“ planen, wenn die Schwellenwerte erreicht sind.



Neue Systemüberwachung konfigurieren – Aktionen

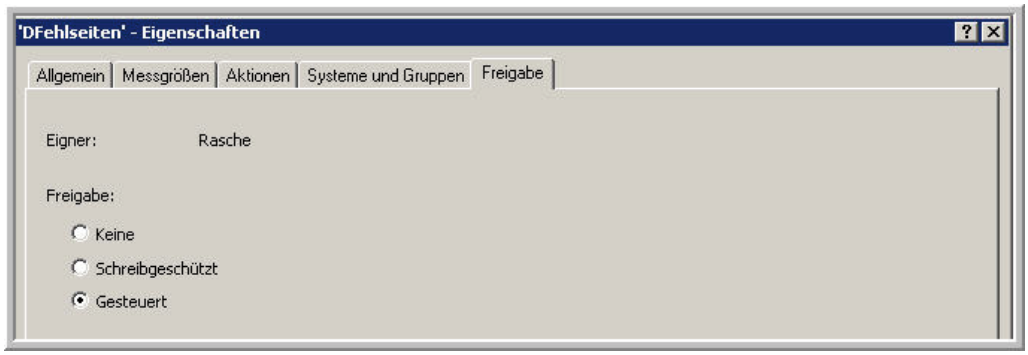
2.11.3.1
Seite 4

Ich kann meistens gut auf die angebotenen Aktionen verzichten und nutze eigentlich nur die Protokollfunktion. Nun müssen wir im letzten Schritt die Server auswählen, auf denen die Überwachung gestartet werden soll.



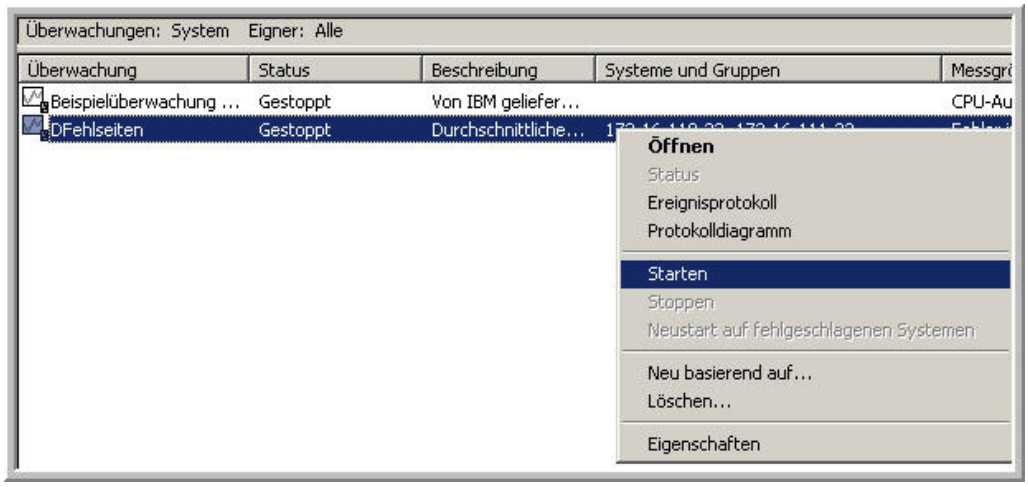
Neue Systemüberwachung konfigurieren – Systeme und Gruppen

Im letzten Register können Sie den Monitor freigeben, damit auch andere Administratoren damit arbeiten können. Die Freigabe ist selbst dann erforderlich, wenn der Anwender über *ALLOBJ-Rechte verfügt.



Neue Systemüberwachung konfigurieren – Freigabe

Sind alle Eingaben vorgenommen, klicken Sie auf „OK“. Dadurch gelangen Sie in den Ausgangsbildschirm zurück und sollten jetzt den neuen Monitor sehen.



Monitor starten

Abschließend muss der neue Monitor nur noch gestartet werden.

2.11.3.1

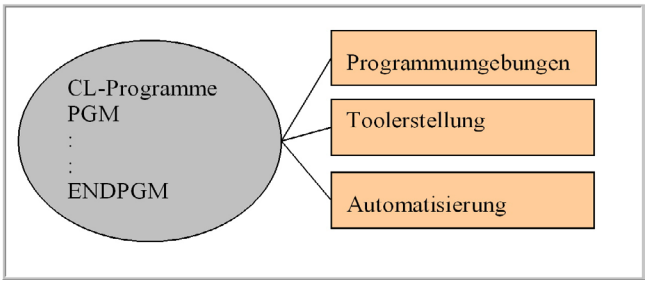
Seite 6

CL-Programmierung

CL steht für Control Language. CL ist eine Programmiersprache, die hauptsächlich für Steuerungsaufgaben eingesetzt wird und in den frühen 80er Jahren mit dem System /38 eingeführt wurde. Im Laufe der Jahre wurde CL ständig erweitert und verbessert. Erst jetzt wurden mit dem Release V5R3 neue Datentypen und Anweisungen eingeführt. Im Gegensatz zu den höheren Programmiersprachen müssen Sie für CL keine Lizenz erwerben. CL ist Bestandteil des Betriebssystems und basiert auf den Systembefehlen. Die meisten Programmanweisungen sind Befehle, die Sie auch manuell zur Steuerung Ihrer iSeries eingeben würden.

Einführung in die CL-Programmierung

Auf jeder iSeries werden mindestens zwei Sprachen eingesetzt – und CL ist eine von beiden. Der Aufbau der höheren Programmiersprachen lässt eine zunehmende Bedeutung des modularen Programmdesigns erkennen, d.h.: Die meisten Verbesserungen der höheren Programmiersprachen hatten zum Ziel, komplexe Prozesse in kleinere, überschaubarere Subprozesse zu unterteilen. In einer derartigen Umgebung sind CL-Programme häufig die Bauteile, die die einzelnen Programmkomponenten zusammenhalten und den Ablauf der Applikation steuern.



Einsatzgebiete von CL-Programmen

CL verfügt über einige Funktionen, die in höheren Programmiersprachen fehlen. Es lassen sich drei grundsätzliche Einsatzgebiete der CL-Programme unterscheiden:

Automatisierungsaufgaben

- Steuern des IPL (Initial Program Load)
- Systembeendigung anfordern
- Einstellung der Systemkonfiguration
- Erstellung von Device-, Controler- und Leitungsbeschreibungen
- Bereinigungsaufgaben

Sicherheitsprüfungen

- Änderung von Benutzer- und Objektberechtigungen
- Kommunikationsaufbau zwischen Benutzern oder anderen Systemen
- Objektverwaltung: CL-Programme können Objekte erstellen, kopieren, ändern, löschen, reorganisieren, bereinigen, umbenennen oder zuordnen.
- Steuerung der Arbeitsabläufe

Zuordnung von Programmumgebungen

- Starten anderer Jobs durch direkten Programmaufruf oder Übergabe von Stapeljobs an das System
- Steuerung des Jobs durch die dynamische Zuordnung von Dateien
- Aufbau der Jobumgebung mit Hilfe von Subsystemen, Jobwarteschlangen, Prioritäten, Speicher-Pools, Zeitscheiben etc.

Tool-Erstellung

- Datenbankpflege automatisieren
- Systempflege automatisieren
- Dokumentationsunterstützung
- Performanceauswertungen

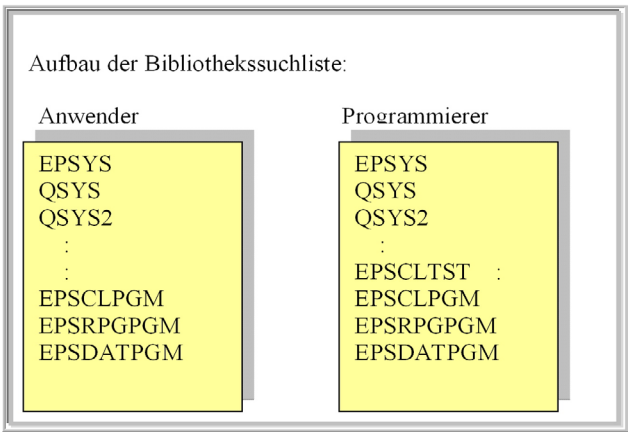
Wenn Sie ein CL-Programm erstellen wollen, müssen Sie den Quelltext (=Sourcecode) mit einem Editor erfassen.

Doch zuvor nehmen Sie sich noch ein paar Minuten Zeit, um einige organisatorische Fragen zu klären:

- In welcher Bibliothek wollen Sie Ihre Programme entwickeln?
- In welcher Bibliothek sollen die frei gegebenen Programme gestellt werden, d.h., welches ist die offizielle Programmbibliothek?
- Vielleicht ist es auch sinnvoll, eine Entwicklungsbibliothek mit den Kollegen gemeinsam zu nutzen?

Egal wie Sie sich entscheiden, diese Fragen sollten Sie klären, bevor Sie anfangen, Programme zu entwickeln. Aber es lohnt sich auch nachträglich, eine einheitliche Struktur aufzubauen. Sie ersparen sich viel Zeit bei der Pflege und Wartung Ihres Systems. Ich gehe in den Beispielen von folgender Situation aus:

Die CL-Programme werden in der Bibliothek EPSCLTST entwickelt und getestet. Nach der Testphase werden die Programme freigegeben, indem sie in die Bibliothek EPSCLPGM gestellt werden. Alle Programmierer sind in der Benutzergruppe EPSPGM zusammengefasst. Eigentümer der Objekte wird das Gruppenprofil.



Aufbau der Bibliothekssuchliste

In der Entwicklungsbibliothek benötigen Sie eine physische Quellendatei, die die CL-Sourcecodes aufnehmen kann. Als Standardwert für CL-Programme wird von der IBM der Name QCLSRC vorgeschlagen. Sie müssen diesen Namen allerdings nicht nutzen, sondern können Ihrer Quellendatei einen beliebigen Namen geben. Der Nachteil, den die Veränderung des Namens mit sich bringt, ist, dass die Umwandlungsbefehle den Standardwert QCLSRC als Namen für die Quellendatei annehmen. Sie könnten natürlich die Default-Einstellungen ändern, aber wozu sollte man sich diese Mühe machen?

Wenn Sie also bislang keine Quellendatei QCLSRC haben, erstellen Sie die Quellendatei mit folgendem Befehl, wobei EPSCLTST der Name unserer Entwicklungsbibliothek ist.

```
CRTSRCPF FILE(EPSCLTST/QCLSRC)      TEXT(,CL-QUELLEN')
```

2.12
Seite 4

CL-Programme erstellen

Ihren Quelltext erfassen Sie jetzt mit einem Editor. Sie können entweder mit der traditionellen Entwicklungsumgebung – dem PDM (Program Development Management) – arbeiten oder die neue Entwicklungsumgebung des IBM WebSphere Development Studios, den CODE400-Editor oder den LPEX-Editor nutzen. Schauen wir uns zunächst einmal den traditionellen Weg an.

Mit Hilfe des PDM kann der Programmierer durch die einheitliche Oberfläche mit Bibliotheken, Objekten und Teildateien arbeiten, ohne die entsprechenden Systembefehle zu kennen. Vom iSeries-Hauptmenü gelangen Sie über Punkt 5 in das Programmierer-Hauptmenü. Über die Auswahl 2 erreichen Sie das Hauptmenü des PDM. Das PDM kann auch direkt über den Befehl STRPDM von jeder Kommandozeile aufgerufen werden.

```
                AS/400 Programmentwicklungsverwaltung (PDM)

Auswahlmöglichkeiten:

     1. Mit Bibliotheken arbeiten          <-- WRKLIBPDM
     2. Mit Objekten arbeiten              <-- WRKOBJPDM
     3. Mit Teildateien arbeiten           <-- WRKMBRPDM

     9. Mit benutzerdefinierten Auswahlmöglichkeiten arbeiten
```

Das Menü Programmentwicklungsverwaltung

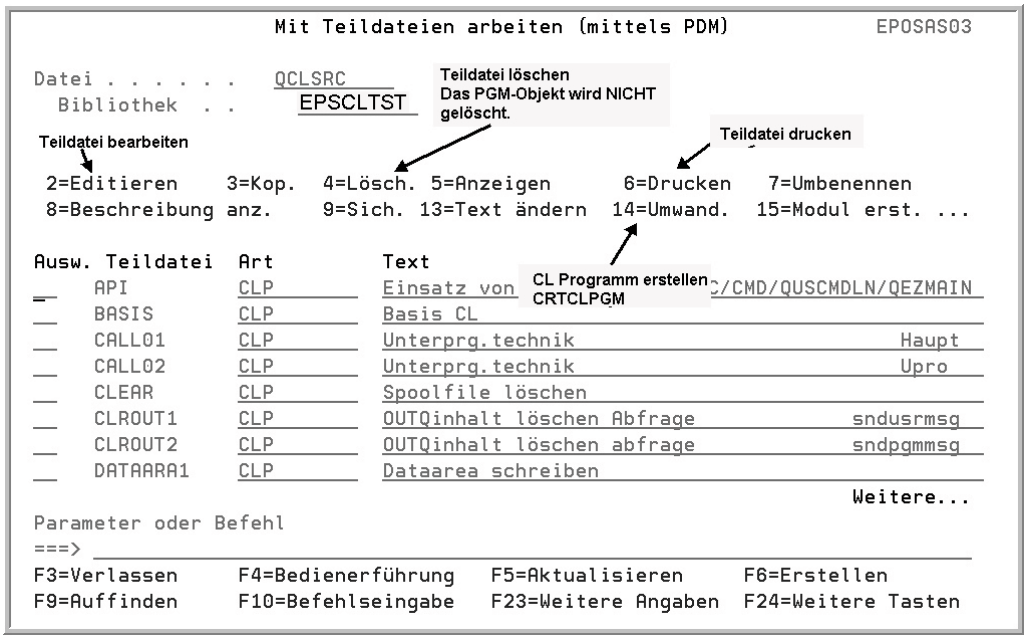
Damit Sie Ihre CL-Programme bearbeiten können, wählen Sie den Menüpunkt 3: „Mit Teildateien arbeiten“.

```
                    Arbeitsteildateien angeben

Auswahl eingeben und Eingabetaste drücken.

  Datei . . . . . .      <-- Name Ihrer Quellendatei

    Bibliothek  . . . . .     <-- Name Ihrer Testbibliothek

  Teildatei:
    Name  . . . . . .    *ALL        *ALL, Name, *generisch*
    Art . . . . . .      *ALL        *ALL, Art, *generisch*, *BLANK
```

Eingabe der Quellendatei und Testbibliothek

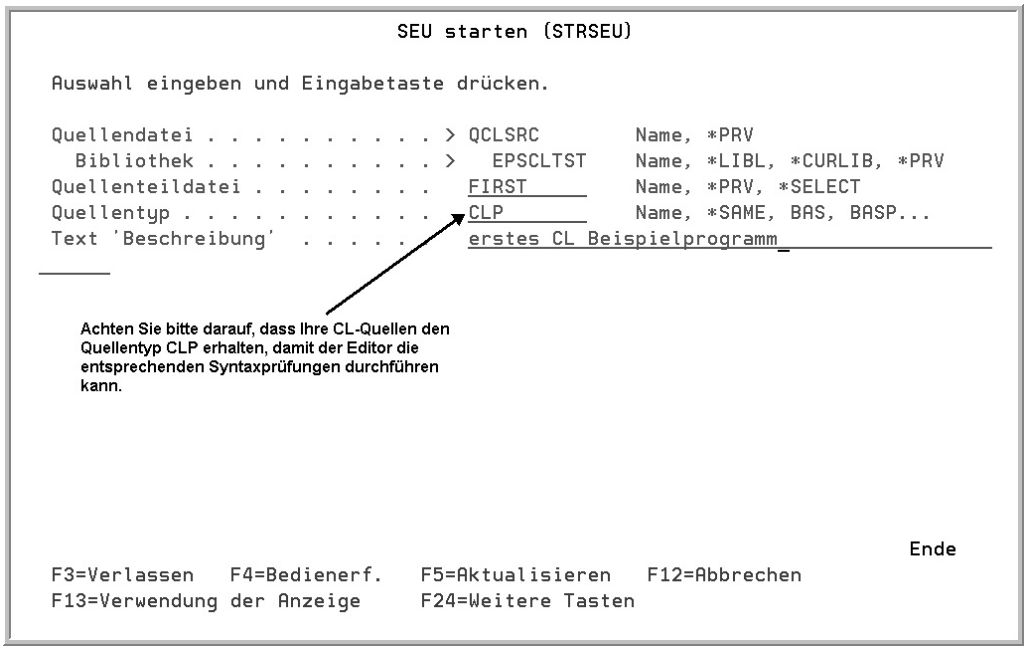
Im Parameter „Datei“ geben Sie den Namen Ihrer Quellendatei ein. Die Parameter „Teildateiname“ und „Teildateiart“ dienen als Filter für Ihre CL-Programmquellen. In der Anfangszeit ist es allerdings besser, auf die Selektion der Teildateien zu verzichten und wie in der Abbildung alle Teildateien anzuzeigen. Nach der Eingabe erhalten Sie eine Übersicht aller CL-Sourcecodes, die Sie nun mit entsprechenden Auswahlen bearbeiten können.

2.12.1
Seite 2



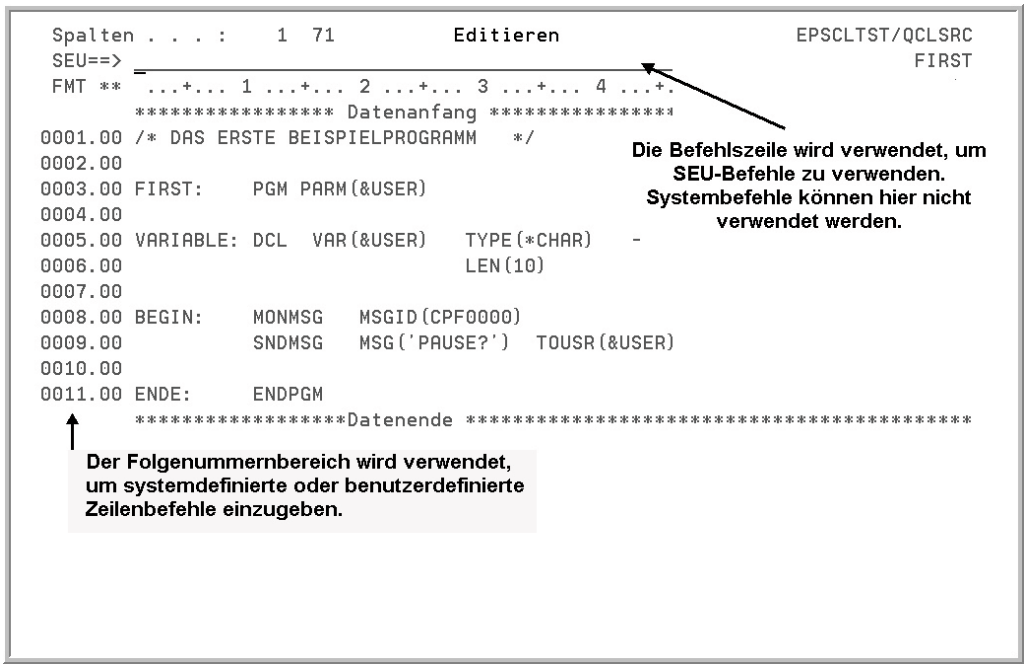
Übersicht über die CL-Sourcecodes

Um SEU vom PDM aus zu starten, nutzen Sie die Funktionstaste F6 oder Sie kopieren mit der Auswahl 3 eine bestehende Teildatei, die Sie anschließend mit der Auswahl 2 modifizieren können.



Das Menü SEU

SEU ist der Standard-Editor der iSeries.



Editieren mit SEU-Befehlen

Sie können innerhalb des Editors SEU-Befehle auf der Befehlszeile, Zeilenbefehle innerhalb des Folgenummernbereichs oder Funktionstasten nutzen. Wir können uns an dieser Stelle nur auf die wichtigsten Funktionen beschränken. Eine ausführliche Beschreibung des Editors entnehmen Sie bitte der Bedienerhilfe.

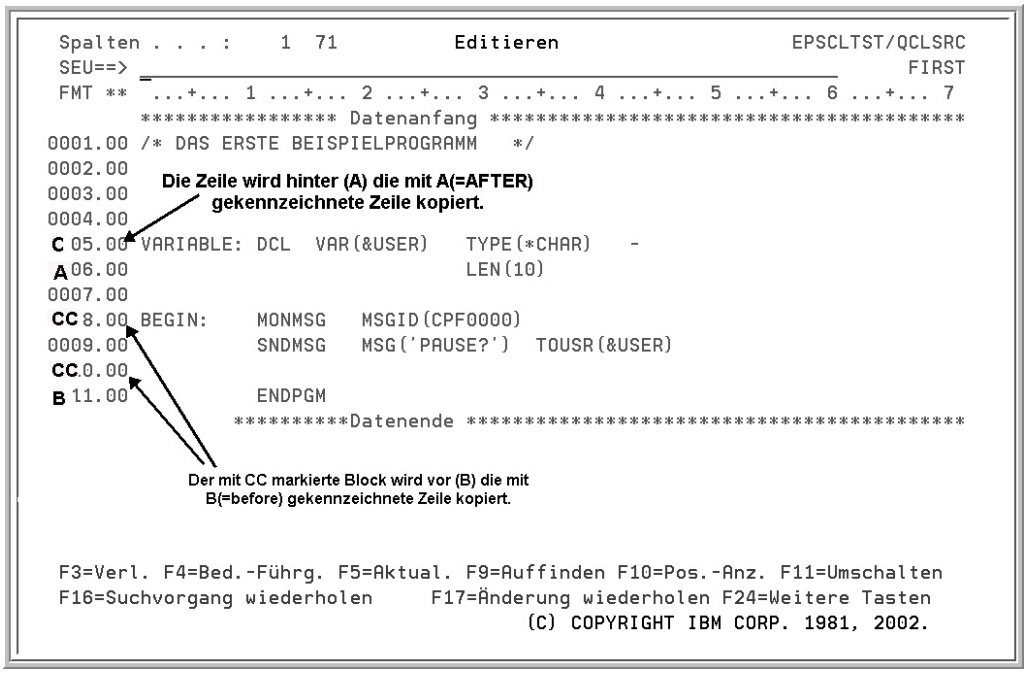
SEU-Befehle sind Befehle im freien Format. Einige der Befehle können abgekürzt werden. Beispielweise kann F oder FIND zur Ausführung des Befehls FIND eingegeben werden. Den Hilfetext der einzelnen Befehle rufen Sie auf, indem der Befehl in die Befehlszeile eingegeben und die Hilfetaste gedrückt wird. Folgende SEU-Befehle stehen Ihnen zur Verfügung:

SEU-Befehle

BOTTOM B	An das Ende der Teildatei positionieren.
CANCEL	Die aktuelle Sitzung wird abgebrochen.
CHANGE C	Zeichenfolge suchen und in eine andere Zeichenfolge ändern.
FILE	Die Änderungen in der Teildatei werden gesichert und die Sitzung wird verlassen.
FIND F	Zeichenfolge in einer Teildatei suchen.
SAVE	Die aktuelle Sitzung wird gesichert.
TOP T	An den Anfang der Teildatei positionieren.

Der Folgenummernbereich dient der Eingabe von Zeilenbefehlen, d.h.: Sie können einzelne Zeilen kopieren, wiederholen oder löschen.

2.12.1
Seite 4



Editieren mit Zeilenbefehlen

Es stehen folgende Auswahlmöglichkeiten zur Verfügung:

Zeilenbefehle

Befehle für die Zielangabe	
A	Versetzen, kopieren oder einfügen der Sätze hinter diesen Satz.
B	Versetzen, kopieren oder einfügen der Sätze vor diesen Satz.
O	Überlagerung dieses Satzes mit dem ersten Satz, für den ein Zeilenbefehl eingegeben wurde.
Befehle zum Kopieren	
C	Die Zeile vor oder hinter einer anderen Zeile kopieren.
Cn	n bestimmt die Anzahl zu kopierender Zeilen.
CC	Block kopieren
Befehle zum Löschen	
D	Zeile löschen.
Dn	Diese Zeile und die nächsten n-Zeilen löschen.
DD	Block löschen.
Befehle zum Einfügen	
I	Eine Zeile einfügen.
In	Hinter der markierten Zeile werden n-Zeilen eingefügt.

Befehle zum Verschieben	
M	Zeile verschieben
Mn	Diese und n Zeilen verschieben
MM	Block verschieben
Befehle zum Wiederholen	
RP	Die Zeile wiederholen
RPn	Die Zeile und n weitere Zeilen wiederholen

Der SEU-Bildschirm verfügt zusätzlich über einige Funktionstasten. Wird der Positionsanzeiger in den Funktionstastenbereich gestellt und die Hilfetaste gedrückt, erhalten Sie detaillierte Informationen über die verfügbaren Funktionen. Nachstehende Übersicht enthält eine kurze Beschreibung der wichtigsten Funktionstasten:

SEU-Funktionstasten

F13	Sitzungsstandardwerte verändern
F14	Auswahlmöglichkeiten zum Suchen/Ändern der Teildatei
F15	Zeilen einer anderen Teildatei in die aktuelle Sitzung kopieren
F16	Suchvorgang wiederholen
F17	Änderungen wiederholen
F21	Systembefehl eingeben

Wenn Sie bspw. die Funktionstaste F13 drücken, können Sie Ihre Editorstandardeinstellungen verändern. Schauen Sie sich hierfür die folgende Abbildung an:

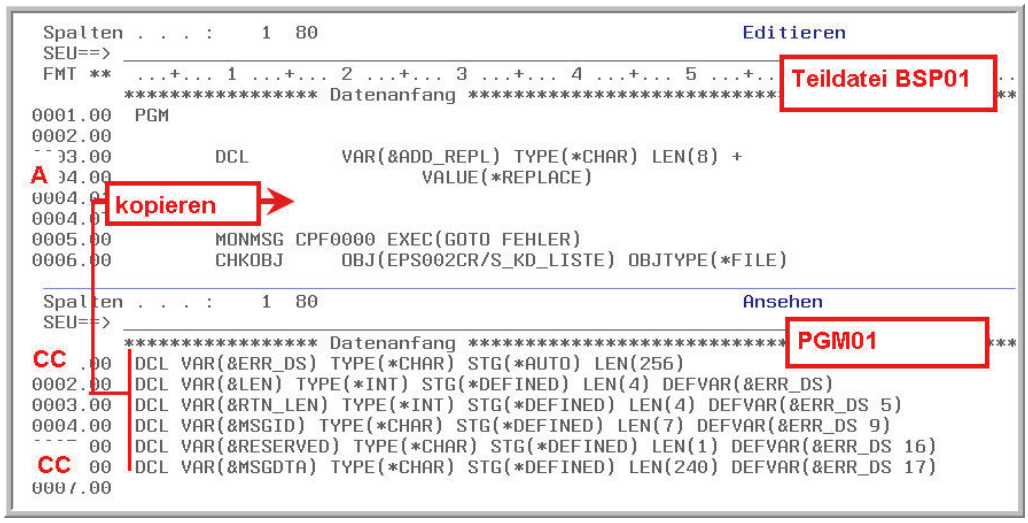
```
                    Standardwerte für Sitzung ändern

uswahl eingeben und Eingabetaste drücken.

 Blättern . . . . . . . . . . . . . . .   H__          H=Halbe Seite, F=Ganze S.
                                                       C=Positionsanzeiger,
                                                       D=Daten, 1-999
 Eingabe nur in Großbuchstaben  . . .   J            J=Ja, N=Nein
 Tabulator ein  . . . . . . . . . . .   N            J=Ja, N=Nein
 Erhöhungswert bei Satzeinfügung  . .   0.01__       0.01-999.99
 Gesamtanzeigemodus . . . . . . . . .   N            J=Ja, N=Nein
 Bildschirmgröße  . . . . . . . . . .   1            1=27x132, 2=24x80
 Quellenart . . . . . . . . . . . . .   CLP_____
 Syntaxprüfung:
   Bei Hinzufügung/Modifizierung  . .   J            J=Ja, N=Nein
   Von Folgenummer  . . . . . . . . .   _______      0000.00-9999.99
   Bis Folgenummer  . . . . . . . . .   _______      0000.00-9999.99
```

F13 Sitzungsstandardwerte ändern

2.12.1
Seite 6

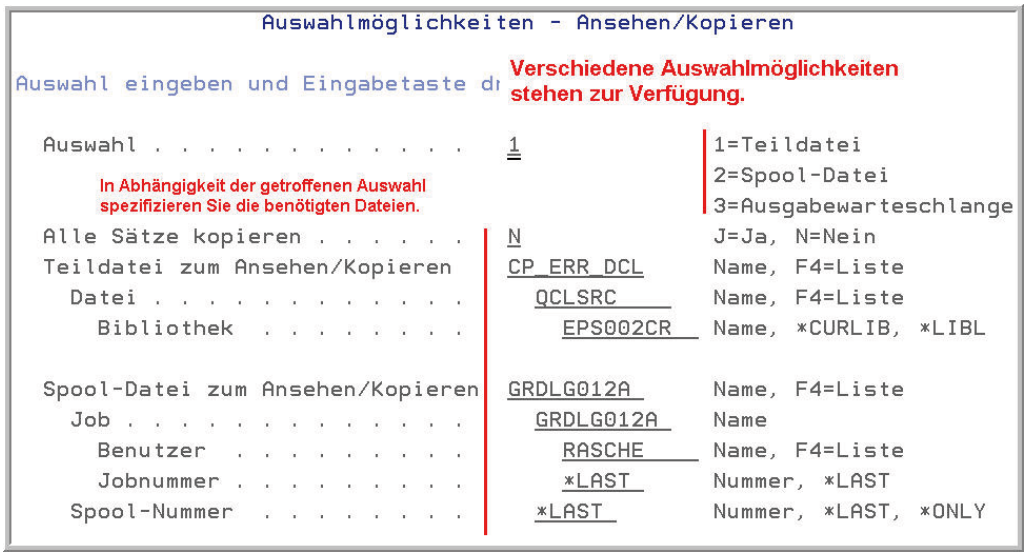
Wenn Sie in Ihrem Sourcecode die Blätter-Taste verwenden, wird normalerweise der Bildschirm um eine halbe Seite nach oben bzw. unten verschoben. Sie können dieses Verhalten jederzeit ändern und ganze Seiten durchblättern oder SEU veranlassen, cursorabhängig zu blättern. Ebenso haben Sie die Möglichkeit, die Eingabe von kleinen Buchstaben zu erlauben. Dies kann nützlich sein, wenn Sie anfangen, auf die Bedienerführung zur Eingabe der CL-Befehle zu verzichten und sowohl konstante Werte aber auch CL-Befehle in Groß- und Kleinbuchstaben schreiben wollen.

Häufiger als die F13-Taste werden Sie in einer 5250-Sitzung die Funktionstaste F15 benötigen, denn Sie erlaubt Ihnen das Kopieren zwischen zwei Teildateien. Sehen Sie, was gemeint ist:



Wirkungsweise der Funktionstaste F15

Im oberen Bildschirm ist die Teildatei BSP01 geöffnet. Nur in dieser Teildatei können Änderungen vorgenommen werden. Allerdings können Zeilen aus dem unteren Bildschirm in die aktive Teildatei hineinkopiert werden; eine Funktion, die Sie sicherlich sehr häufig benötigen. Im unteren Teilbildschirm müssen aber nicht zwingend Teildateien geöffnet werden. Sie können sich auch Ihr Umwandlungsprotokoll anzeigen lassen, um ggf. besser die aufgetretenen Fehler korrigieren zu können. Ob Sie eine Quellenteildatei oder ein Spoolfile anzeigen, entscheiden Sie, wenn Sie mit der Funktionstaste den folgenden Dialog öffnen:



Der Dialog „Auswahlmöglichkeiten – Ansehen/Kopieren“

2.12.1
Seite 8

RDI – Die moderne Entwicklungsumgebung

Grundsätzlich haben Sie bei der Erstellung Ihrer CL-Programme die Wahl zwischen alten Werkzeugen wie PDM und SEU und der modernen RDI-Alternative von Rational. Grundsätzlich lässt sich sagen, dass ILE-RPG die beste RDI-Unterstützung erfährt. Cobol-Programmierer werden etwas schlechter unterstützt und CL-Programmierern stehen praktisch nur die Funktionen zur Verfügung, die auch im SEU vorhanden sind. Dennoch habe ich mich entschlossen, Ihnen auch RDI kurz vorzustellen, denn letztlich muss man die Zielgruppe im Auge behalten. Ein Operator, der lediglich einige CL-Programme schreiben will, muss sich meiner Meinung nach nicht zwingend mit der komplexen RDI-Umgebung vertraut machen. Andererseits gibt es aber auch viele Programmierer, die u. a. CL-Programme entwickeln und denen sollte der Einsatz von PDM und Co. erspart bleiben. Ich bin der Meinung, dass Sie selbst entscheiden müssen, welches das geeignete Entwicklungswerkzeug an Ihrem Arbeitsplatz ist.

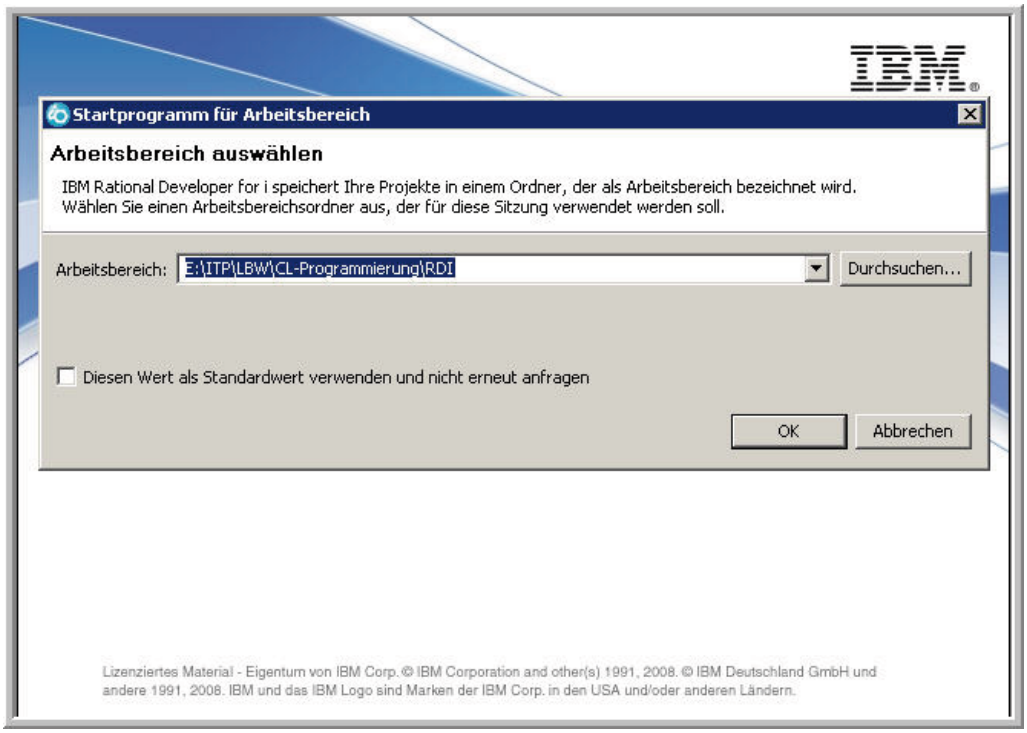
Wollen Sie lieber mit dem IBM Rational Developer for i (RDI) arbeiten, müssen Sie als erstes RDI starten. Einige Leser werden sich jetzt wundern, weil auf Ihrem PC WDSC und nicht RDI installiert ist. WDSC ist der Vorgänger von RDI und wurde lange Zeit kostenlos von IBM ausgeliefert. Die Weiterentwicklung von WDSC wurde allerdings zugunsten des IBM Rational Developer for i bereits vor einiger Zeit aufgegeben. Es ist mir leider nicht möglich, sowohl WDSC als auch RDI zu beleuchten. Ich werde mich also auf den Einsatz von RDI beschränken. Jedoch ähneln sich die beiden Entwicklungsumgebungen in unserem Kontext sehr stark, so dass es kein Problem sein dürfte, auch mit WDSC die Beispiele nachzuvollziehen.

Als erstes müssen wir RDI starten:

Start ➜ Programme ➜ IBM Software Delivery Plattform ➜ IBM Rational Developer for i ➜ IBM Rational Developer for i

Kurz danach erscheint der Dialog „Startprogramm für Arbeitsbereich“.

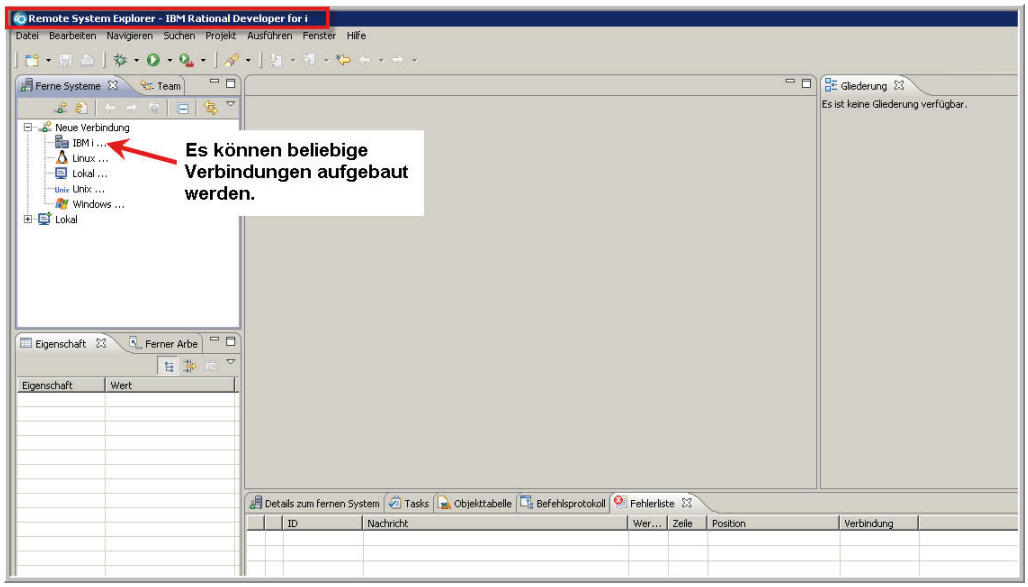
2.12.1.1
Seite 2



Startprogramm für Arbeitsbereich

Der Arbeitsbereich ist eine lokale Verzeichnisstruktur auf Ihrem PC, in dem RDI temporäre und permanente Daten ablegt. Wenn Sie später eine Verbindung zum System i Server aufbauen, werden beispielsweise die Konfigurationsdaten der Verbindung im Arbeitsbereich abgelegt. Im Grunde genommen ist es egal, welches Verzeichnis Sie für die Konfigurationsdaten wählen, solange Sie den Pfad „Dokumente und Einstellungen“ nicht als Speicherort für den Arbeitsbereich verwenden. Muss Windows neu installiert werden, so kann es passieren, dass der Bereich „Dokumente und Einstellungen“ durch die Neuinstallation überschrieben wird und Ihr Arbeitsbereich zerstört wird. Viele Konfigurationen und persönliche Einstellungen müssten dann neu hinterlegt werden.

Sobald Sie Ihren Arbeitsbereich bestimmt haben und den Dialog „Startprogramm für Arbeitsbereich“ bestätigen, erscheint ein Willkommensdialog, wenn Sie RDI zum ersten Mal öffnen, andernfalls öffnet sich sofort die Perspektive des Remote System Explorer (RSE). Die Perspektive des Remote System Explorer (RSE) sollten Sie auch erhalten, wenn Sie den Willkommensdialog schließen.



Die Perspektive Remote System Explorer (RSE)

Die Oberfläche von RDI besteht aus vielen verschiedenen Perspektiven, die unterschiedliche Komponenten und verschiedene Fenster (Sichten) beinhalten. Ich denke, es ist verständlich, dass wir uns auf die Sichten beschränken werden, die erforderlich sind, um CL-Programme zu kodieren, zu testen und zu kompilieren.

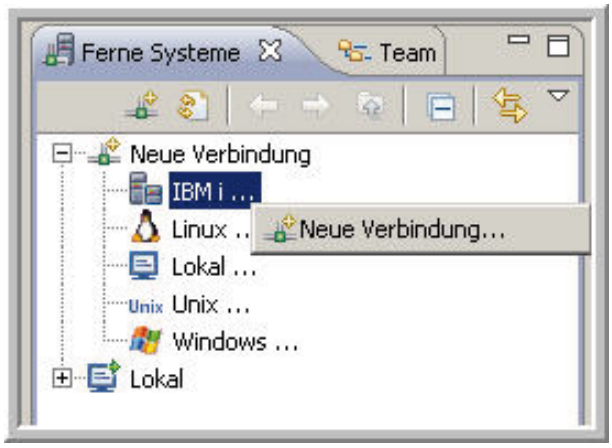
2.12.1.1

Seite 4

Die Arbeitsumgebung konfigurieren

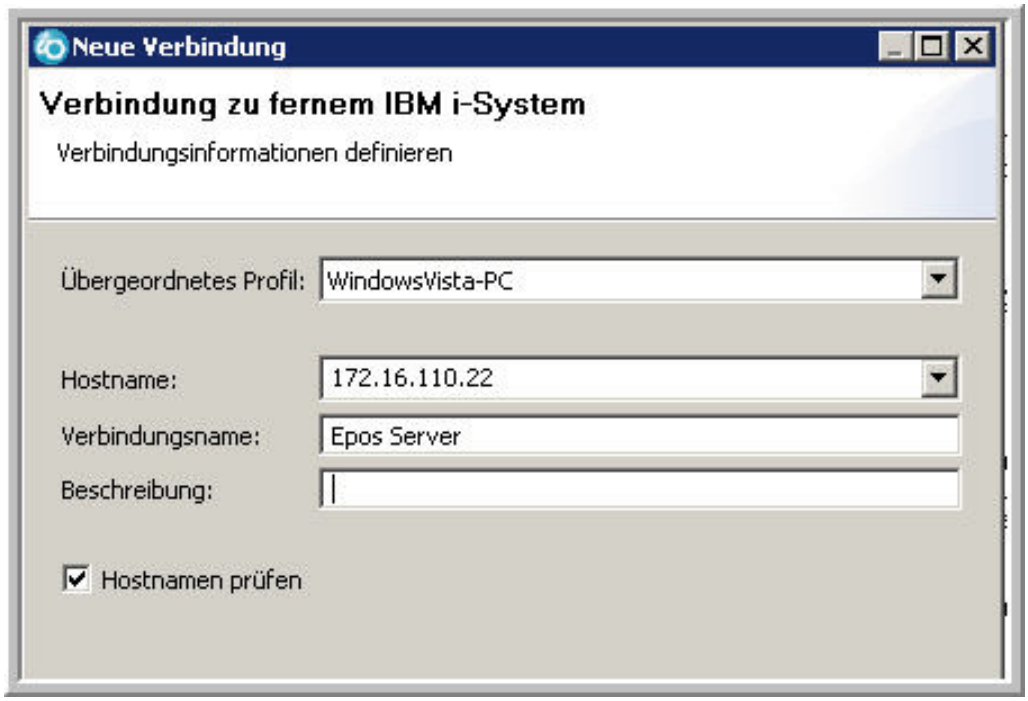
Die Perspektive Remote System Explorer (RSE) ist vergleichbar mit PDM. Innerhalb der Perspektive RSE können wir Quellcode bearbeiten, Programme kompilieren oder Umwandlungslisten prüfen.

Um allerdings mit den Daten auf einem System i Server zu arbeiten, müssen wir zuerst eine Verbindung zu unserem System herstellen. Dazu öffnen Sie den Eintrag „Neue Verbindung“ und anschließend das Kontextmenü des Eintrags „IBM i“, wie es folgende Abbildung zeigt.



Neue System i-Verbindung herstellen

Wenn Sie eine erste Verbindung in einem leeren Arbeitsbereich erstellen, erscheint zunächst einmal ein weiterer Dialog:

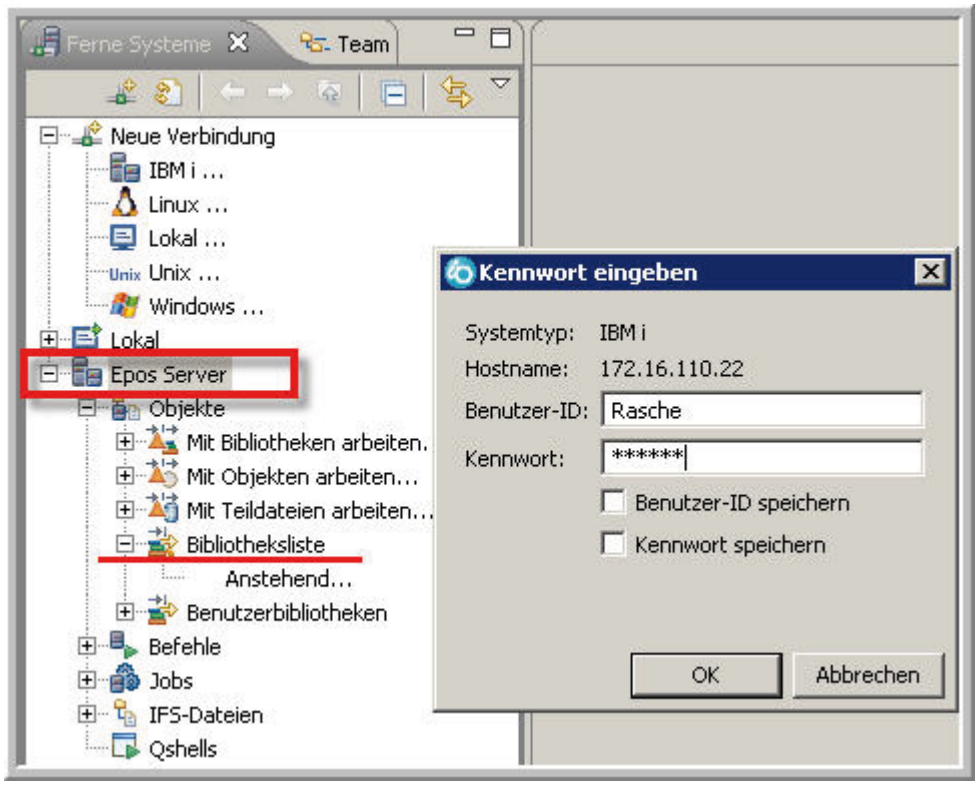


Der Dialog „Verbindung zu fernem IBM i-System“

2.12.1.1.1
Seite 2

Als übergeordnetes Standardprofil wird der PC-Name verwendet, der die Verbindung aufbaut. Profilnamen werden erst dann interessant, wenn Sie in komplexen Umgebungen arbeiten. Für unser Ziel „CL-Programmierung“ können wir es beim Standard belassen, geben nur noch den Hostnamen des System-i-Servers ein und fügen ggf. einen beschreibenden Text hinzu. Wenn alle Informationen hinterlegt sind, bestätigen Sie auch diesen Dialog mit dem Button „Fertig stellen“. Wurde alles korrekt umgesetzt, sollte im Fenster „Ferne Systeme“ jetzt ein Eintrag mit dem gewählten Verbindungsnamen stehen.

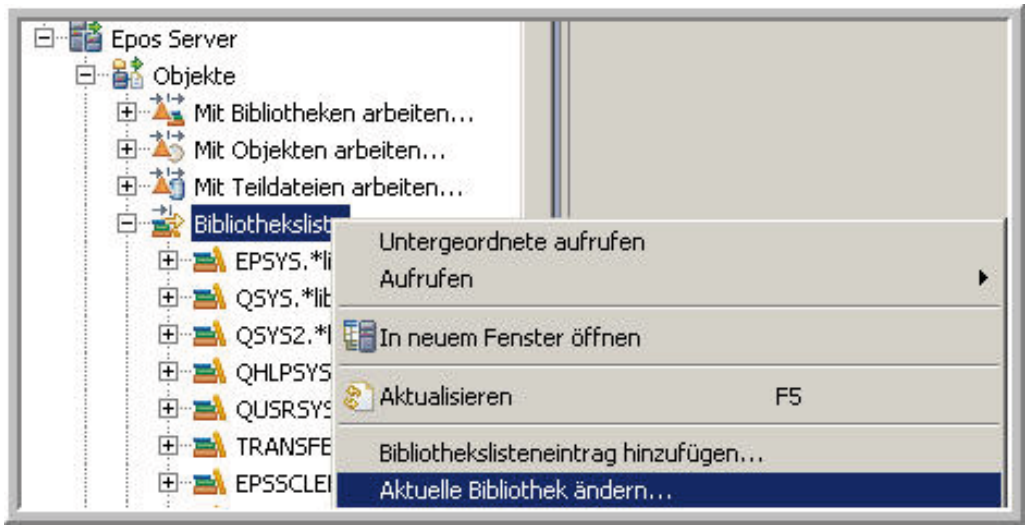
Um die Verbindung zum Server herzustellen, klicken Sie jetzt auf das Pluszeichen vor dem Verbindungsnamen und öffnen bspw. den Eintrag „Bibliotheksliste“. Die folgende Abbildung zeigt, was gemeint ist:



Eine Verbindung zum Server herstellen

Egal, welche Funktion Sie aufrufen, sobald dafür eine Verbindung zum Server erforderlich ist, erscheint ein Anmeldedialog. Sie müssen Ihren Benutzernamen und das Kennwort eingeben und anschließend den Anmeldedialog bestätigen, um mit den verschiedenen Einträgen im Fenster „Ferne Systeme“ arbeiten zu können.

Bevor ich mit meiner Arbeit beginne, muss ich zunächst einmal meine Arbeitsbibliotheken der Bibliotheksliste hinzufügen. Dazu öffne ich das Kontextmenü des Eintrags „Bibliotheksliste“ und wähle den Punkt „Aktuelle Bibliothek ändern“. Sie sehen in der nächsten Abbildung, dass ich alternativ auch mehrere Arbeitsbibliotheken der Suchliste hinzufügen könnte, wenn ich den Eintrag „Bibliothekslisteneintrag hinzufügen“ nutzen würde.



Bibliotheksliste anpassen

Für viele CL-Programme ist eine Bibliothek ausreichend, so dass ich mich für die Bekanntgabe der aktuellen Bibliothek entscheide. Im daraufhin erscheinenden Dialog „Aktuelle Bibliothek ändern“ trage ich meine Arbeitsbibliothek ein und bestätige auch dieses Fenster. Wenn alles korrekt umgesetzt wurde, muss anschließend der gewählte Bibliotheksname in der Liste angezeigt werden.

Die Bibliotheksliste steuert wie im 5250-Umfeld die Adressierung der Objekte. Haben Sie später Probleme, die darauf zurückzuführen sind, dass Objekte nicht gefunden werden, so gibt es nur zwei Möglichkeiten:

- Die nicht gefundenen Objekte existieren tatsächlich nicht
- oder die Bibliotheksliste in der Verbindung ist nicht korrekt.

Wenn ich jetzt RDI beenden würde und später erneut aufrufe, ist meine Arbeitsbibliothek nicht mehr in der Bibliotheksliste vorhanden. Die eben durchgeführten Modifikationen sind nur solange gültig, solange die Verbindung zum Server besteht. Wenn die Verbindung unterbrochen wird, sind die Modifikationen verloren. Ganz ähnlich wie bei einer 5250-Sitzung, in der Sie die Bibliotheksliste mit CHGCURLIB, ADDLIBLE oder anderen Befehlen auf einer 5250-Befehlszeile manipulieren.

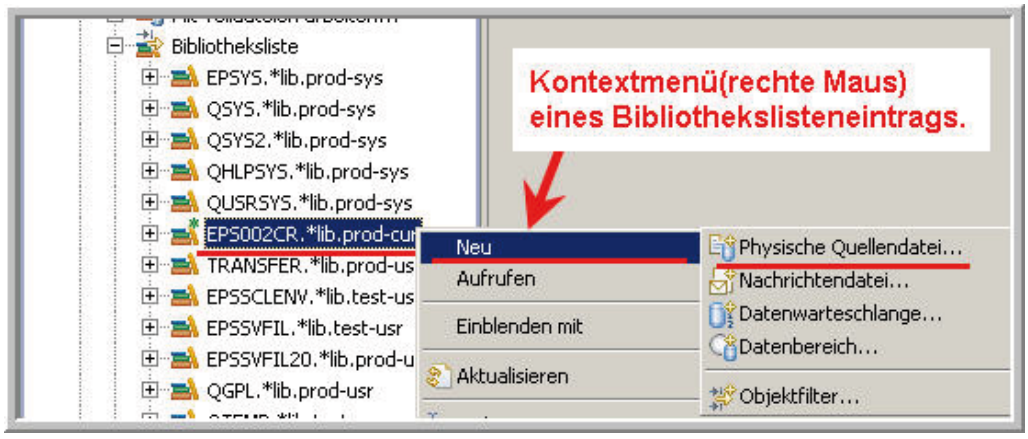
2.12.1.1.1
Seite 4

Sollen also die Bibliotheken dauerhaft hinterlegt werden, müssen Sie die Eigenschaften Ihrer Verbindung ändern. Am schnellsten öffnen Sie den dafür notwendigen Dialog über die Tastenkombination ALT+ENTER:

Eigenschaften der Verbindung definieren

Im Eigenschaftendialog hinterlegen Sie einerseits den Standardbenutzer, andererseits können Sie im Register „Subsysteme“ die Bibliotheksliste manipulieren. Auch diesen Dialog bestätigen Sie mit dem Button „OK“.

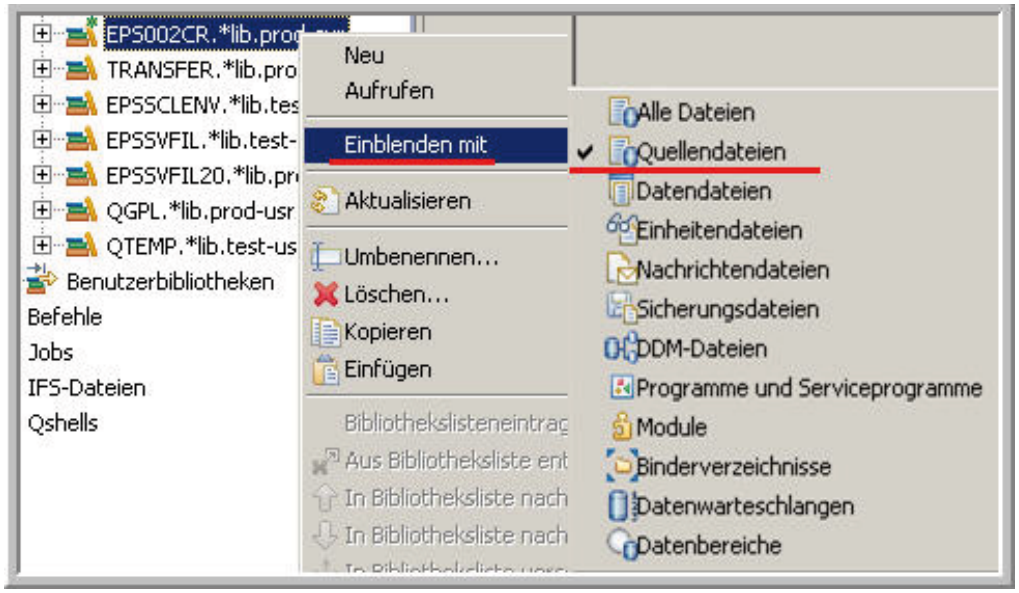
Für unsere weitere Arbeit benötigen wir jetzt noch die Quellendatei QCLSRC oder alternativ QCLLESRC, wenn sie bislang noch nicht erstellt wurde. Grundsätzlich gibt es viele verschiedene Möglichkeiten, neue Objekte zu erstellen. Ich bevorzuge die Methode über das Kontextmenü der jeweiligen Bibliothek, wie in folgender Abbildung gezeigt wird.



Physische Quellendatei QCLSRC erstellen

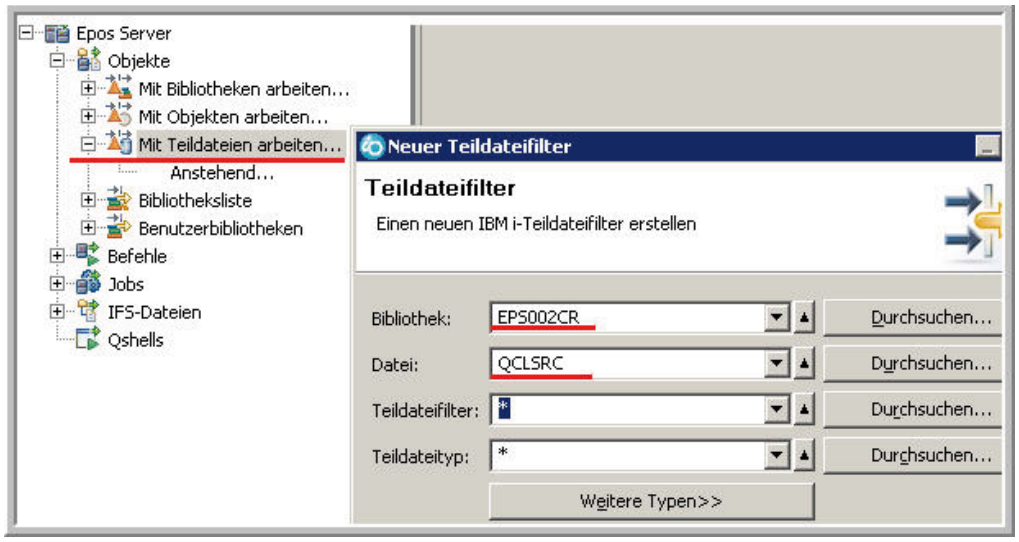
Es öffnet sich daraufhin der Dialog „Neue physische Quellendatei“. Dieser Dialog ist weitestgehend selbsterklärend. Auffällig ist lediglich die gewählte Standardsatzlänge von 112 Byte. Im 5250-Umfeld werden standardmäßig nur 96 Byte vorgeschlagen. Der Unterschied liegt in der Unterscheidung zwischen OPM-CL-Programmen und ILE-CL-Programmen. Wenn Sie OPM-CL-Programme entwickeln, sind 96 Byte als Satzlänge besser geeignet.

Nun wollen wir den Inhalt einer Quellendatei anzeigen. Dazu wählen Sie im Kontextmenü (rechte Maustaste) der entsprechenden Bibliothek den Eintrag „Einblenden mit Quellendateien“.



RDI – Quellendateien anzeigen

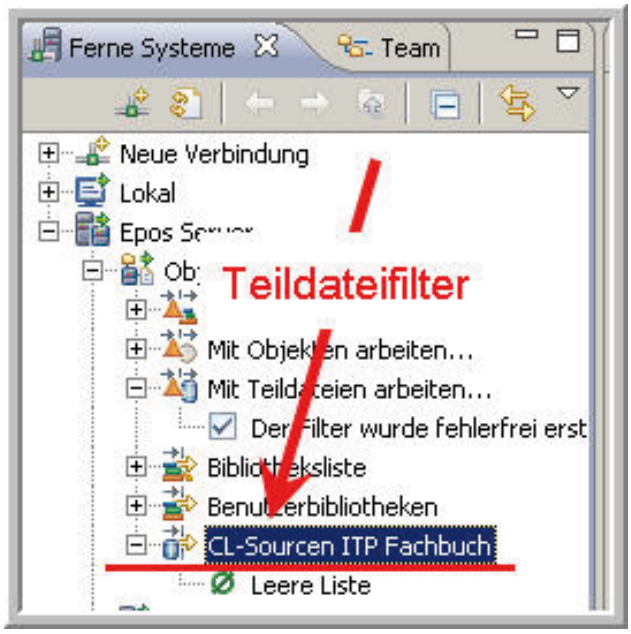
Die Funktion „Einblenden mit“ erlaubt eine gefilterte Anzeige der Bibliotheksobjekte und erleichtert die Arbeit mit den verschiedenen Objekten innerhalb der Bibliothek erheblich. Um den Zugriff auf die relevanten Objekte zu beschleunigen, können aber auch sogenannte Filter erstellt werden. Filter sind nichts anderes als Selektionen auf Bibliotheks-, Objekt- und Teildateiebene, die als eigenständige Verzeichnisse in Ihre Verbindung aufgenommen werden. Dafür muss zunächst ein Teildateifilter erstellt werden. Klicken Sie hierfür auf den Eintrag „Mit Teildateien arbeiten“, um das Dialogfenster zu öffnen. Die folgende Abbildung zeigt, was gemeint ist.



RDI – Teildateifilter erstellen

2.12.1.1.1
Seite 6

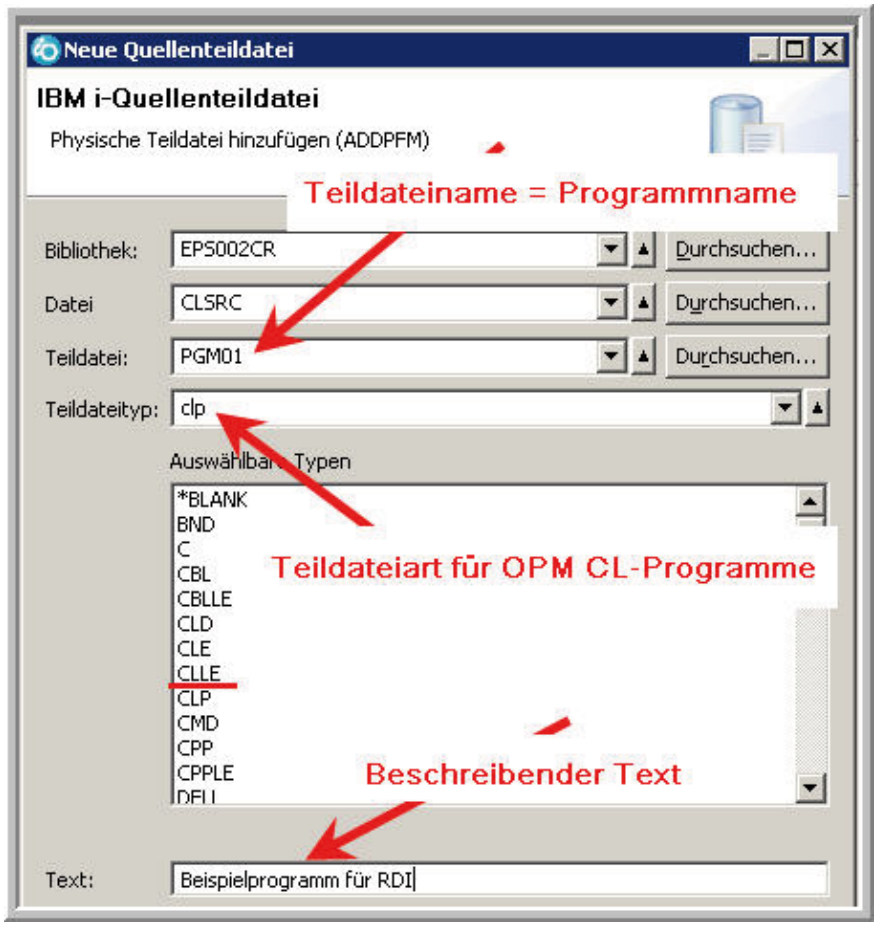
Der Dialog erinnert stark an PDM. Auch dort mussten wir die Bibliothek und den Quellendateinamen nennen, um mit den Teildateien einer Quellendatei arbeiten zu können. Sie können weitere Selektionskriterien hinterlegen, beispielsweise bzgl. des Teildateityps oder der Teildateinamen. Anfangs ist es allerdings ratsam, auf diese Dinge zu verzichten, um nicht versehentlich vorhandene Teildateien auszublenden. Klicken Sie nun auf den Button „Weiter“ und vergeben Sie einen sprechenden Filternamen. Ich verwende für meinen Filter den Namen „CL-Sourcen ITP-Fachbuch“. Wenn der Filter korrekt erstellt wurde, wird er anschließend in Ihrer Verbindung angezeigt:



Der Teildateifilter „CL-Sourcen ITP-Fachbuch“

Teildateien mit dem LPEX-Editor bearbeiten

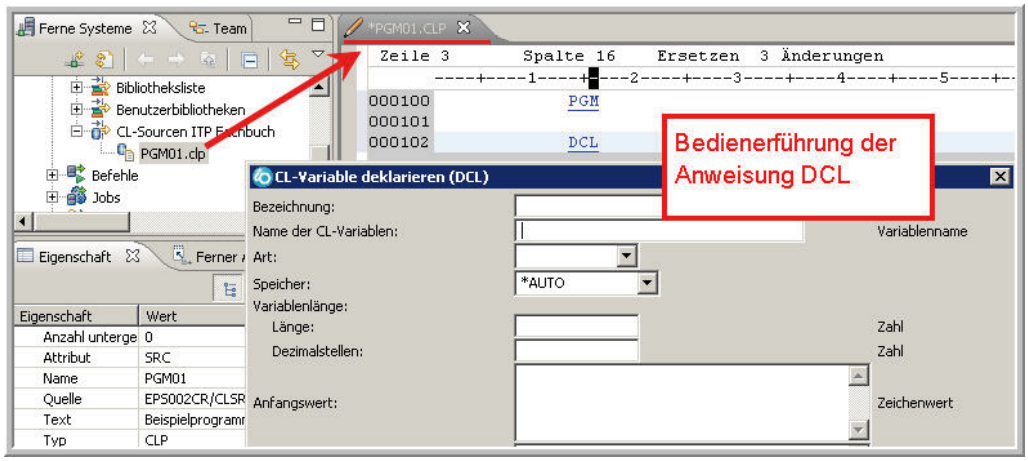
In meiner Quellendatei befinden sich derzeit keine Teildateien. Um eine neue Teildatei zu erstellen, nutze ich einfach das Kontextmenü meines soeben erstellten Teildateifilters „CL-Sourcen ITP-Fachbuch“ und öffne den Dialog „Neue Quellenteildatei“.



Neue Teildatei erstellen

Bezüglich der einzugebenden Informationen unterscheidet sich der Dialog nicht vom SEU-Dialog. Auch hier müssen ein Teildateiname und die Teildateiart gewählt werden. Sobald ich die Teildatei „Fertig stelle“, erscheint die neue Teildatei in meinem Filter und gleichzeitig wird die Teildatei im LPEX-Editor geöffnet. Der Aufruf einer bereits vorhandenen Quelle ist noch einfacher. Ein Doppelklick auf eine vorhandene Teildatei öffnet den Editor.

2.12.1.1.2
Seite 2

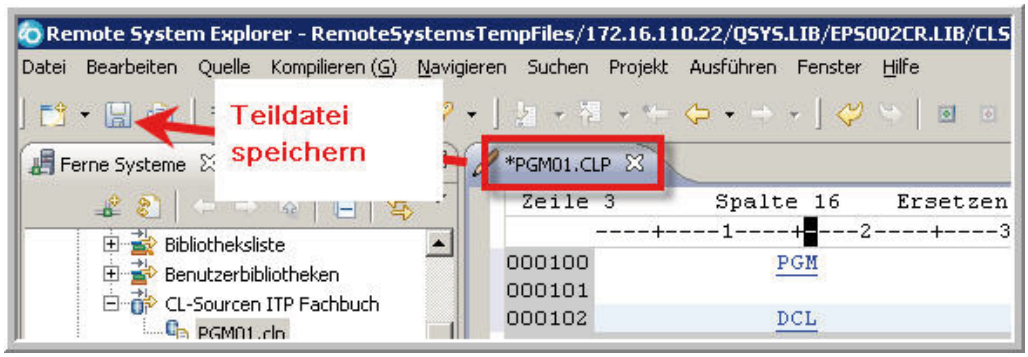


Die Bearbeitung einer Teildatei

Für Programmierer ist der Editor das zentrale Tool. RDI verfügt über einen sehr komfortablen und wandlungsfähigen Editor, der für CL-, RPG- und Cobol-Programme gut geeignet ist:

Jede Teildatei wird in einem eigenen Register geöffnet. Dadurch ist es möglich, mehrere Teildateien parallel zu öffnen. Diese Eigenschaft ist nützlich, wenn wir beispielsweise bereits vorhandene Quellen in eine neue Editor-Sitzung übernehmen wollen. Es kann problemlos über das Windows-Clipboard mit den Tasten STRG+C und STRG+V kopiert und eingefügt werden. Der Editor ist ansonsten dem SEU sehr ähnlich. Im Folgenummernbereich können Sie beispielsweise die aus SEU bekannten Zeilenbefehle bedenkenlos einsetzen. Außerdem steht für jede Anweisung selbstverständlich auch über die Taste F4 die Bedienerführung zur Verfügung.

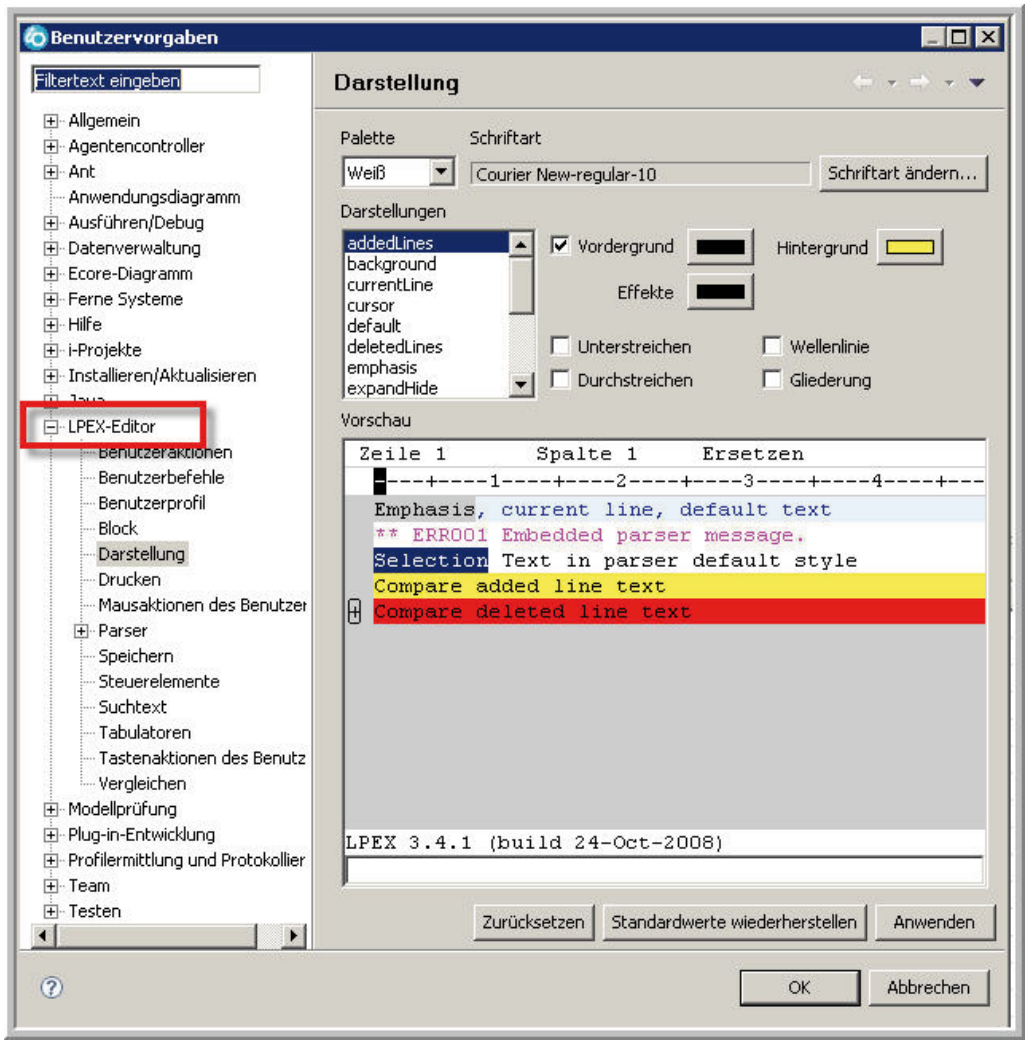
Achten Sie in der folgenden Abbildung auf das kleine Sternchen vor dem Dateinamen.



Teildatei speichern

Dieses Sternchen bedeutet, dass eine Quelle modifiziert, aber bislang noch nicht gespeichert wurde. Sie können in der Toolleiste auf das Disketten-Symbol klicken, um die Änderungen zu speichern. Ansonsten werden Sie spätestens beim Schließen des Registers aufgefordert, die Teildatei zu speichern. Auch wenn das Programmobjekt erstellt wird, wird der Quellcode selbstverständlich gespeichert.

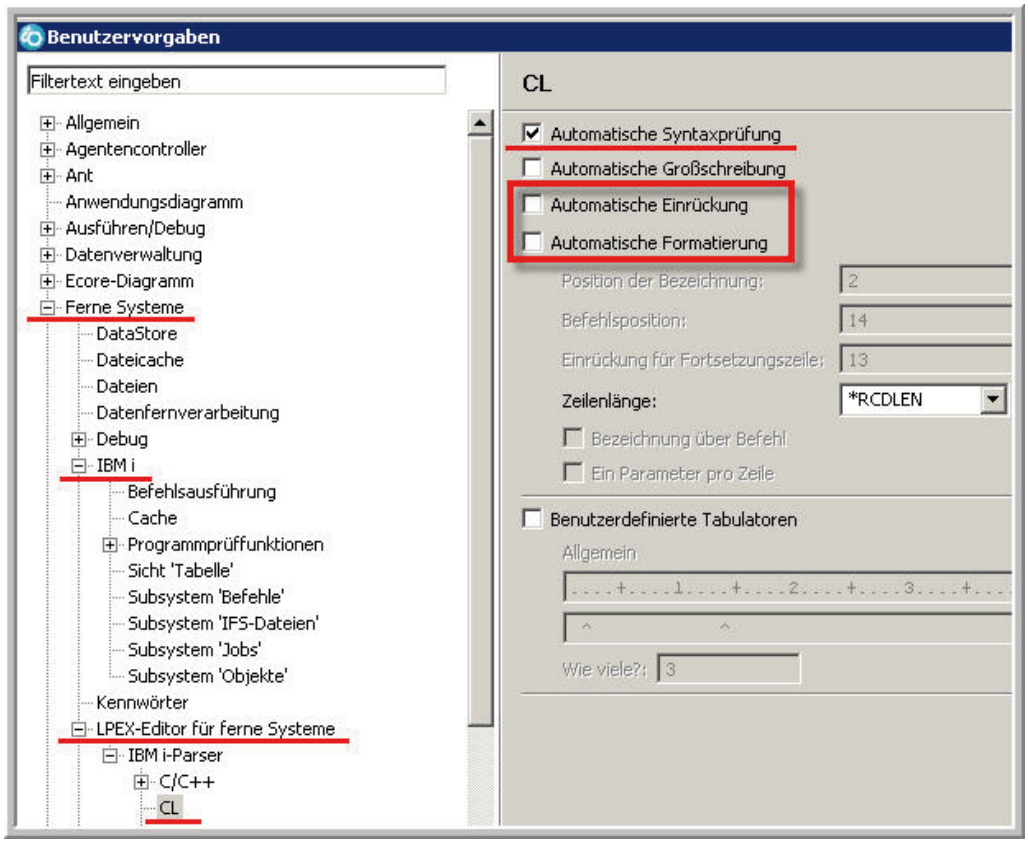
Wenn Ihnen die Grundkonfiguration des Editors nicht gefällt, können Sie viele individuelle Einstellungen vornehmen. Über das Menü „Fenster“ können Sie die „Benutzervorgaben“ öffnen.



Fenster ➜ *Benutzervorgaben* ➜ *LPEX-Editor*

2.12.1.1.2
Seite 4

Sie sehen in der Abbildung, dass Sie viele spezifische Anpassungen vornehmen können. Zusätzlich können Sie auch für die CL-Programmierung einige spezifische Anpassungen vornehmen, so können Sie Angaben bzgl. der Druckoptionen machen, können generelle Vorgaben für Suchoptionen hinterlegen oder die farbliche Darstellung anpassen. Doch Sie können auch CL-spezifische Vorgaben machen. Dafür öffnen Sie wiederum im Menü „Fenster“ die „Benutzervorgaben“. Dieses Mal öffnen Sie aber den Pfad „Ferne Systeme“ ➜ LPEX-Editor für ferne Systeme“ ➜ „IBM i Parser“ ➜ „CL“. Daraufhin sollte der folgende Dialog geöffnet werden:



Benutzervorgaben für CL-Programme

Ich deaktiviere hier beispielsweise gern die automatische Großschreibung und die automatische Formatierung der Bedienerführung, aktiviere aber die automatische Syntaxprüfung.

Wie Sie sicher bereits vermuten, ist der LPEX-Editor ein mächtiges Werkzeug, das je nach verwendeter Programmiersprache unterschiedliche Funktionen anbietet. Leider können wir die vielen Dialoge hier nicht im Detail besprechen, da dies den Umfang des Buches sprengen würde. Jedoch sind viele Optionen selbsterklärend und wenn nicht, stehen umfangreiche Hilfetexte zur Verfügung, die Sie über die F1-Taste aktivieren können.

Bestandteile eines CL-Programms

Schauen wir uns nun den Aufbau eines einfachen Beispielprogramms an:

CL-Anweisungen können frei formatiert werden. Anders als bei DDS oder RPG-Quellen haben die Spalten keinen Einfluss auf die Anweisung. Sie können die Zeilen nach Ihren eigenen Vorstellungen in der Quelle anordnen. Die meisten Programmierer überlassen das Formatieren allerdings der Bedienerführung, die auch innerhalb eines CL-Programms wirksam ist.

```
/*  Das erste Beispielprogramm  ,PAUSE'  */
PAUSE:          pgm   parm(&user)
variable:       dcl var(&user)  type(*char)   len(10)
BEGIN:          monmsg  msgid(CPF0000)
                sndmsg   msg('PAUSE?')  tousr(&user)
ende:           endpgm
```

Der CL-Compiler unterscheidet nicht zwischen Groß- und Kleinschreibung. Sie können selbst entscheiden, welche Schreibweise Ihnen lieber ist. Wenn Sie in der SEU-Eingabezeile den Befehl SET CAPS ON bzw. SET CAPS OFF eingeben, wechseln Sie zwischen Groß- und Kleinschreibung.

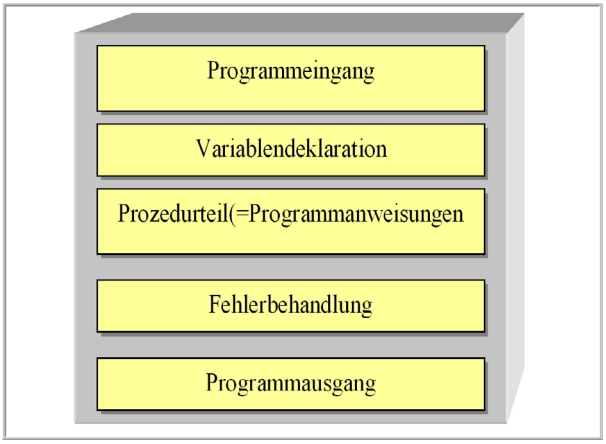
```
/*  Das erste Beispielprogramm  ,Pause'  */
pause:          pgm   parm(&user)
variable:       dcl  var(&user)  type(*char)   -
                len(10)
begin:          monmsg  msgid(CPF0000)
                sndmsg   msg('Pause?')  tousr(&user)
ende:           endpgm
```

Ihre CL-Programme sollten Kommentare enthalten. Die Kommentare erleichtern die spätere Wartung der Programme, denn Sie müssen auch noch nach einem Jahr wissen, wie Sie das Programm aufgebaut haben.

```
/*  Das erste Beispielprogramm  ,Pause'  */
```

Kommentare werden in den Zeichenfolgen /* und */ eingeschlossen. Wenn Sie längere Kommentare schreiben, fassen Sie die Kommentarzeilen mit dem Fortsetzungszeichen (+) zusammen.

2.12.2
Seite 2



Bestandteile eines CL-Programms

```
PAUSE:      pgm   PARM(&user)
```

Jedes CL-Programm hat einen Programmeingang. Der Programmeingang wird mit der Anweisung PGM gekennzeichnet. Die Klausel PARM listet alle Parameter auf, die das Programm anfordert bzw. wieder zurückgibt. Unser Beispielprogramm fordert den Parameter &USER an. PAUSE: ist eine Marke (=Label). Marken sind nicht zwingend erforderlich. Sie können aber gut zur Dokumentation und zur Strukturierung eingesetzt werden.

Die Anweisung DCL (Declare) definiert eine Programmvariable.

```
variable:     dcl   var(&user)          type(*char)   -
                                        len(10)
```

Alle Variablen müssen am Programmanfang deklariert werden. Viele CL-Anweisungen enthalten mehr Parameter, als in eine Zeile passen, so dass mehrere Zeilen für eine Anweisung benötigt werden. Die fortgesetzten Zeilen werden mit einem Pluszeichen (+) oder einem Minuszeichen (–) gekennzeichnet. Wenn Sie mit der Bedienerführung arbeiten, werden die Fortsetzungszeichen automatisch eingefügt.

```
variable:     dcl    &user     *char    10
```

Befehlsparameter können auf zwei Arten eingegeben werden: als positionsgebundene Werte oder mit Schlüsselwörtern. Positionsgebundene Werte sind zwar schneller geschrieben, aber Sie müssen die Lesbarkeit Ihrer Anweisungen beachten. Ihre Programme müssen gepflegt werden und häufig hat der mit der Pflege beauftragte Programmierer das Programm nicht selbst geschrieben. Damit auch andere Kollegen den Quellcode lesen und verstehen können, sollten Sie Ihr Programm so klar wie möglich gestalten.

```
BEGIN:          monmsg  msgid(CPF0000)
                sndmsg   msg('Hallo')  tousr(&user)
```

Nach den Variablen-Deklarationen folgen die eigentlichen Programmanweisungen. Sie bilden den Kern des Programms. CL-Programme können beliebig viele Leerzeilen enthalten. Leerzeilen können frei verwendet werden, außer zwischen Zeilen, die durch Fortsetzungszeichen miteinander verbunden sind.

```
BEGIN:          monmsg  msgid(CPF0000)
```

Die Anweisung MONMSG (Nachrichten überwachen) wird zwar noch ausführlicher betrachtet, dennoch soll hier die Wirkung kurz vorstellt werden. Falls das System bei der Programmausführung Fehler feststellt, können mit der Anweisung MONMSG Maßnahmen zur Fehlerbehandlung festgelegt werden. Da die Anweisung zwischen dem Deklarationsteil und den Programmanweisungen steht, dient die MONMSG-Anweisung der globalen Nachrichtenüberwachung, d.h.: Sie gilt für den gesamten Programmablauf. An jeder anderen Stelle gilt die MONMSG-Anweisung nur für die unmittelbar vorhergehende Anweisung.

```
ende:           endpgm
```

Die Anweisung ENDPGM markiert das Ende des Programms. Sie ist eigentlich optional, denn das Programm wird beendet, sobald keine ausführbaren Anweisungen mehr vorhanden sind. Die Anweisung ENDPGM macht allerdings die Struktur des Programms klarer und definiert einen eindeutigen Endpunkt.

2.12.2
Seite 4

Umwandeln eines CL-Programms

CL ist eine Compiler-Sprache. Nachdem der Quellcode eingegeben wurde, müssen Sie das Programm in Maschinensprache umwandeln. CL-Programme werden mit dem Befehl CRTCLPGM (Create Control Language Program) erstellt. Der folgende Befehl wandelt das Programm PAUSE um:

```
                    CL - Programm erstellen (CRTCLPGM)

Auswahl eingeben und Eingabetaste drücken.

Programm . . . . . > PAUSE       Name
  Bibliothek . . . . . >   EPSCLTST   Name, *CURLIB
Quellendatei . . . . . .   QCLSRC     Name
  Bibliothek . . . . . .     *LIBL     Name, *LIBL, *CURLIB
Quellenteildatei . . . . .   *PGM       Name, *PGM
Text 'Beschreibung'  . . . . . .   *SRCMBRTXT
_____

                         Zusätzliche Parameter

Quellenlistenauswahl . . . . . .   _________     *SOURCE, *NOSOURCE, *SRC...
            + für weitere Werte    _________
Generierungsangaben  . . . . . .   ________      *NOLIST, *LIST, *NOXREF...
            + für weitere Werte    _________
Benutzerprofil . . . . . .   *USER          *USER, *OWNER
Befehle protokollieren . . . . .   *JOB           *JOB, *YES, *NO
                                                                  Weitere ...
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

CL-Programme mit CRTCLPGM erstellen

Die Anweisung CRTCLPGM hat mehrere Parameter:

Programm (PGM):
Der Parameter PGM spezifiziert die Zielbibliothek und den Namen des fertigen Programms. Wenn kein Bibliotheksname angegeben wird, wird der Standardwert *CURLIB angenommen. Wenn Sie für Ihren Job keine aktuelle Bibliothek definiert haben, wird das Programm in der Bibliothek QGPL erstellt.

Quellendatei (SRCFILE):
Im Parameter SRCFILE geben Sie die Quellendatei an, in der sich die Teildatei mit dem Sourcecode befindet. Als Standardwert wird QCLSRC in *LIBL angenommen.

Quellenteildatei (SRCMBR):
In dem Parameter SRCMBR geben Sie den Namen Ihrer Teildatei an, die den Quellcode für das Programm enthält. Der Standardwert *PGM bedeutet, dass die Teildatei denselben Namen hat wie der angegebene Programmname.

Beschreibungstext (TEXT):
Mit dem Parameter TEXT dokumentieren Sie Ihre CL-Objekte. Der Standardwert *SRCMBRTXT legt fest, dass der beschreibende Text Ihrer Quellenteildatei verwendet wird.

Quellenlistenauswahl (OPTION):
Der Parameter OPTION steuert den Inhalt der produzierten Druckausgabe.

- *SRC oder *NOSRC
  Der Parameterwert *SRC bestimmt, dass der Quellcode in dem Umwandlungsprotokoll gedruckt wird. Wenn der Parameterwert *NOSRC angegeben wird, werden nur die Umwandlungsfehler ausgegeben.
- *XREF oder *NOXREF
  Das Umwandlungsprotokoll enthält eine Querverweisliste aller Variablen und Marken, die Sie in Ihrem Programm verwenden. Bei *NOXREF wird die Querverweisliste nicht gedruckt.
- *GEN oder *NOGEN
  Wenn Ihr Quelltext keine Fehler enthält, wird ein Programmobjekt erstellt. Die Einstellung *NOGEN sollten Sie verwenden, wenn Sie zwischenzeitlich Ihre Arbeit auf Fehler überprüfen wollen, ohne das Objekt zu erstellen.
- *SECLVL oder *NOSECLVL
  Bei dem Standardwert *NOSECLVL wird der Fehlernachrichtentext der ersten Ebene ausgegeben. Wenn Ihnen die Fehlermeldungen des Compilers nicht ausreichen, können Sie mit dem Wert *SECLVL den Nachrichtentext der zweiten Ebene mit ausgeben.

Benutzerprofil (USRPRF):
Beim Programmaufruf ordnet das System dem Programm ein bestimmtes Benutzerprofil zu, um die Berechtigungen zu prüfen. Normalerweise ist dies der Name des Anwenders, der das Programm aufruft. Das Programm wird unter der Berechtigung dieses Anwenders ausgeführt. Wenn Sie den Parameterwert *OWNER angeben, werden die Berechtigungen des Programmeigners verwendet. Sie erzwingen, dass das System sich so verhält als wäre das Programm vom Objekteigner aufgerufen worden. Dieses Verfahren nennt man „Übernehmen der Berechtigung“.

Befehle protokollieren (LOG):
Mit dem Parameter „Befehle protokollieren“ (LOG) können Sie festlegen, ob die im Programm enthaltenen CL-Befehle im Jobprotokoll aufgezeichnet werden sollen. Der Standardwert *JOB übernimmt die aktuellen Einstellungen des interaktiven Jobs. Sie können die aktuellen Einstellungen Ihres Jobs mit dem Befehl DSPJOB feststellen und gegebenenfalls mit dem Befehl

CHGJOB ändern. Der Parameterwert *YES bewirkt, dass alle CL-Befehle protokolliert werden. Der Wert ist sehr nützlich, weil Sie dadurch den Ablauf Ihres CL-Programms genau nachvollziehen können.

RTVCLSRC erlauben (ALWRTVSRC):
Mit dem Befehl RTVCLSRC (CL-Quelle auffinden) ist es möglich, den Quellcode eines bestehenden Programmobjekts wiederherzustellen. Diese Möglichkeit wird unterbunden, wenn Sie den Wert *NO verwenden.

Existierendes Programm ersetzen (REPLACE):
Der Parameter REPLACE bestimmt, dass beim Erstellen ein bereits existierendes Programm mit demselben Namen ersetzt werden soll. Mit REPLACE (*YES) wird das bestehende Programm zunächst in die Bibliothek QRPLOBJ gestellt und das neue Programmobjekt wird in der angegebenen Bibliothek erstellt. Wenn Sie REPLACE (*NO) angeben wird das Programm nicht automatisch ersetzt.

Ziel-Release (TGTRLS):
Soll das Programm auf einem anderen System unter einem früheren Release eingesetzt werden, können Sie mit diesem Parameter das Programm für das Ziel-Release kompilieren. Das umgewandelte Objekt muss anschließend mit dem Befehl SAVOBJ – ebenfalls mit dem Parameter TGTRLS – gesichert werden, so dass es dann auf dem anderen System zurückgespeichert und genutzt werden kann. Welche Sonderwerte akzeptiert werden, entnehmen Sie der Bedienerhilfe, denn die Werte sind von Ihrem eigenen Release-Stand abhängig.

Berechtigung (AUT):
Mit diesem Parameter bestimmen Sie die öffentlichen Berechtigungen an dem Programmobjekt. Als Standardwert wird der Parameterwert *LIBCRTAUT verwendet. Dieser Wert verweist auf den Parameter „Berechtigung für neue Objekte“ (CRTAUT) des Befehls CRTLIB (Create Library). Der Wert dieses Parameters wird standardmäßig im Systemwert QCRTAUT hinterlegt. Wenn Sie die Default-Einstellungen in Ihrem System nicht verändert haben, werden Sie dort den Wert *CHANGE finden. Sie können die öffentlichen Berechtigungen natürlich auch direkt durch die Objektberechtigungsklassen *EXCLUDE, *USE, *CHANGE oder *ALL festlegen. Es ist ebenfalls möglich, den Namen einer Berechtigungsliste anzugeben, die den Zugriff auf das Objekt verwaltet. Eine ausführliche Beschreibung der Objektberechtigungen und deren Wirkung finden Sie in dem Kapitel „Ressourcenschutz“.

Der Befehl CRTCLPGM erstellt eine Umwandlungsliste, die den Quellcode, ein Verzeichnis der Programmvariablen und gegebenenfalls Warnungen und Fehlermeldungen enthält – abhängig davon, welche Parameterwerte Sie im Befehl CRTCLPGM verwenden.

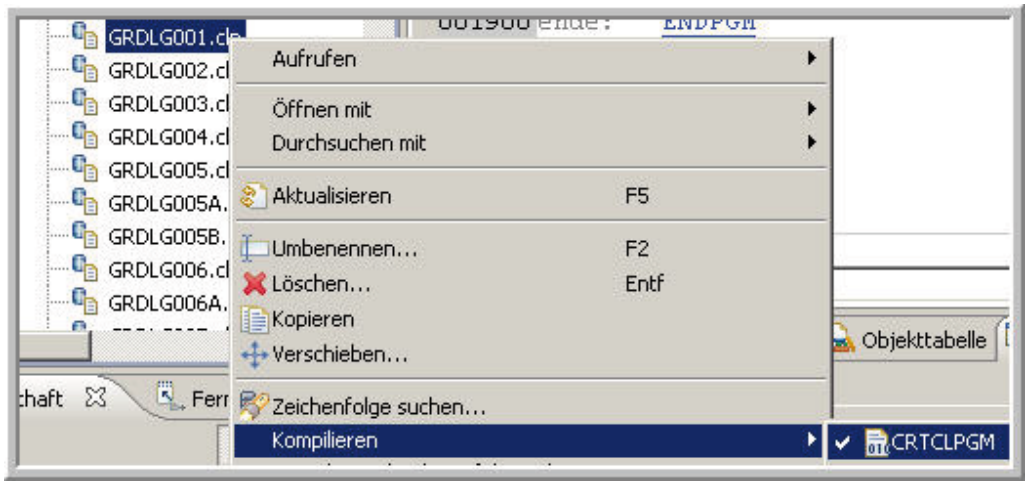
2.12.3
Seite 4

```
                          Spool-Datei anzeigen
Datei . . . . . :   FIRST                          Seite/Zeile 2/16
Steuerung . . . .   _______                        Spalten     1 - 78
Suchen  . . . . .   ______________________________
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
 Definierte Marken
 Marke            Definiert   Referenzen
 BEGIN              800
 ENDE              1100
 FIRST              300
 VARIABLE           500
                    * * * * *  E N D E  D E S  Q U E R V E R W E I S E S
 5722SS1 V5R2M0  020719               Steuersprache (CL)              EPS
                                   Nachrichtenübersicht
              Bewertung
 Gesamt       0-9  10-19  20-29  30-39  40-49  50-59  60-69  70-79  80-89  90
     0          0      0      0      0      0      0      0      0      0
 Programm FIRST erstellt in Bibliothek EPSCLTST am 08.10.04 um 13:33:07.
 Programm FIRST in Bibliothek EPSCLTST erstellt. Höchste Fehlerbewertung ist 0
                           * * * * *  E N D E  D E R  N A C H R I C H T E
                              * * * * *  E N D E  D E R  U M W A N D L U N
                                                                      Ende
F3=Verlassen  F12=Abbrechen   F19=Links   F20=Rechts   F24=Weitere Tasten
```

Das Programm wurde erfolgreich umgewandelt

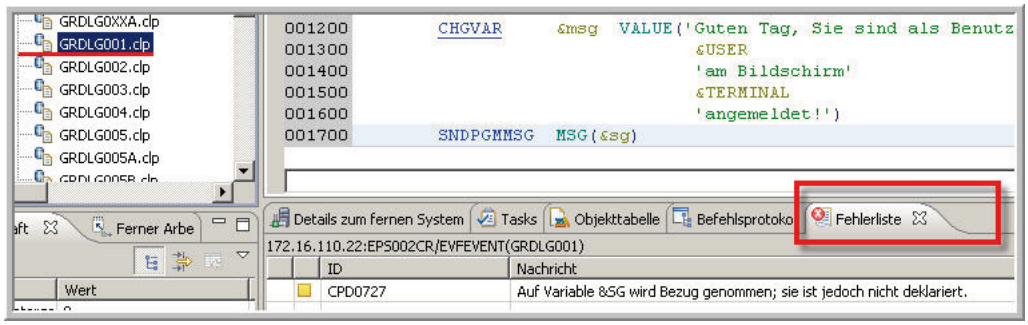
War die Umwandlung fehlerfrei, wird ein Programmobjekt des Typs *PGM mit dem Attribut CLP erstellt. Das Programm ist jetzt ausführbar. Weist die Umwandlung Fehler auf, müssen Sie den Quellcode korrigieren und dann erneut versuchen, das Programm umzuwandeln.

Natürlich können Sie auch aus der RDI-Umgebung heraus Ihre CL-Programme kompilieren. Am einfachsten geht es, wenn Sie die entsprechende Teildatei markieren und im Kontextmenü den Eintrag „Kompilieren“ wählen.



CL-Programme mit RDI kompilieren

Das Ergebnisprotokoll erscheint unterhalb der im Editor geöffneten Teildatei im Register „Fehlerliste“.



Das Register „Fehlerliste“

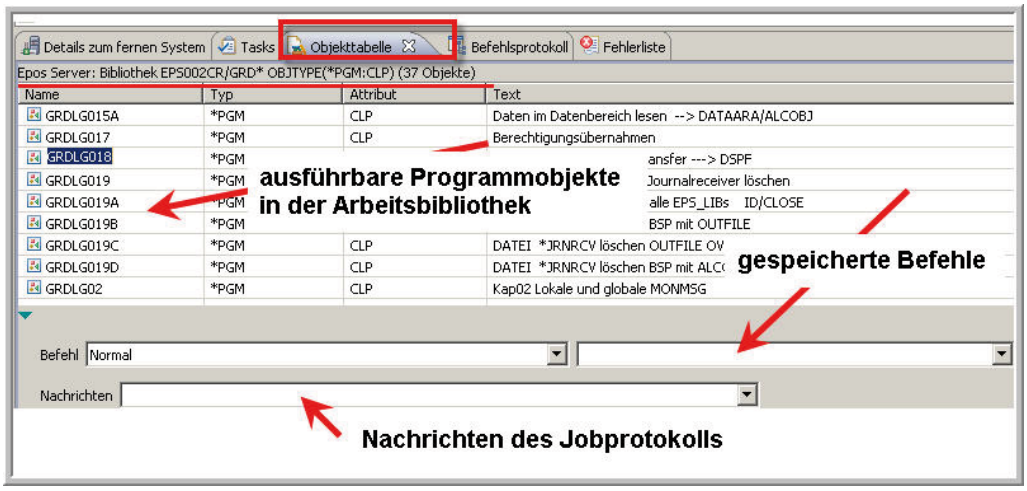
Enthält das Protokoll Fehler wie im gezeigten Beispiel, können Sie auf den Eintrag doppelt klicken und sofort wird die Zeile, die den Fehler enthält, in der geöffneten Teildatei markiert. Die Arbeit mit den Fehlerprotokollen ist dadurch im RDI sehr viel einfacher als in einer 5250-Umgebung.

Benötigen Sie in diesem Umfeld die Bedienerführung des Befehls CRTCLPGM, müssen Sie den Eintrag „Kompilieren (Bedienerführung)“ öffnen. Die Befehlsdefinition des CRTCLPGM-Befehls wird heruntergeladen und kann genauso angepasst werden wie innerhalb einer 5250-Sitzung.

2.12.3
Seite 6

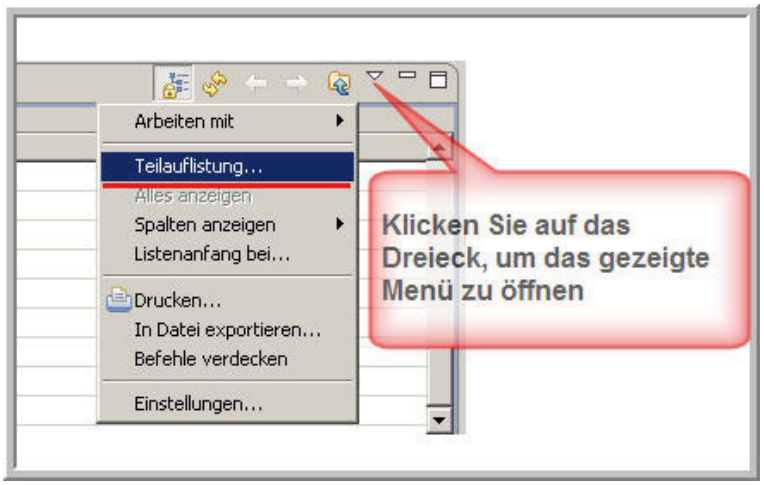
Etwas schwieriger ist es, die erstellten CL-Programme zu testen. Zwar ist RDI so konzipiert, dass Sie ferne native System-i-Programme ausführen können, aber Sie benötigen hierfür entsprechende Startkonfigurationen. Als erstes benötigen wir das ausführbare Programmobjekt. Natürlich könnte ich wieder einen Filter erstellen; dieses Mal auf die Programmobjekte meiner Arbeitsbibliothek.

Wir werden einen anderen Weg gehen und unsere ausführbaren Programmobjekte im Register Objekttabelle am unteren Bildschirmrand anzeigen. Folgende Abbildung zeigt, was gemeint ist:



Das Register Objekttabelle

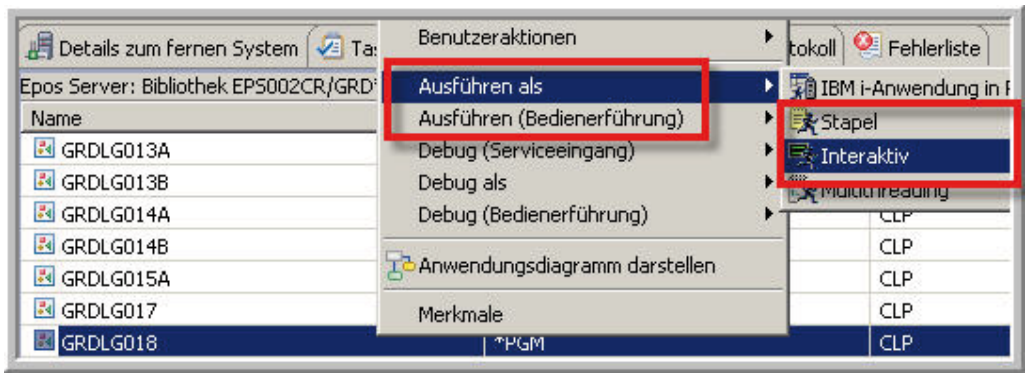
Die „Objekttabelle“ kann beliebige Objekte enthalten, mit denen Sie anschließend arbeiten können. Dafür müssen Sie aber zunächst einmal einen Filter definieren, wie folgende Abbildung zeigt:



Teilauflistung Register Objekttabelle

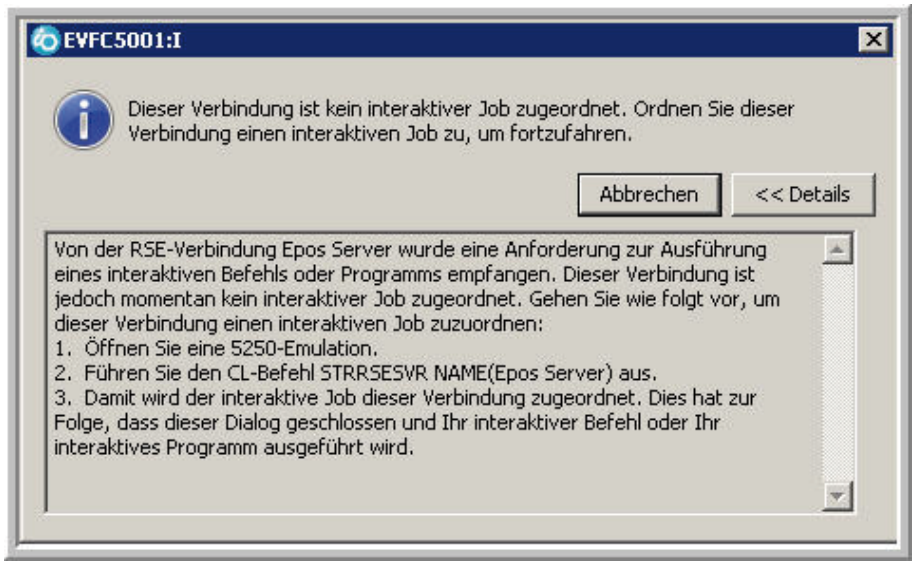
Im darauf folgenden Dialog können Sie anhand des Objektnamens, anhand des Objekttyps und des Objektattributs die anzuzeigenden Daten selektieren. Die Arbeit ist jetzt recht komfortabel, da Sie gleichermaßen Zugriff auf die Programmquelle und das Programmobjekt haben.

Nun wollen wir aber endlich unser CL-Programm aufrufen. Dabei müssen wir unterscheiden, ob das Programm Parameter erwartet oder ob es als Stapeljob (SBMJOB) oder interaktiv ausgeführt werden soll.



Das CL-Programm starten

Die obere Abbildung zeigt, dass es zwei verschiedene Methoden gibt das Programm aufzurufen: mit und ohne Parameter. Wird die Option „Ausführen als“ ausgewählt, wird das ausgewählte Objekt mit Standardwerten ausgeführt. Erfordert Ihr Programm hingegen Parameter, müssen Sie die Version „Ausführen (Bedienerführung)“ verwenden, um eine Startkonfiguration zu erstellen. Startkonfigurationen sind gesicherte Konfigurationen, die Sie aufrufen können, um Programme und Einstellungen mehrmals auszuführen. Das in der Abbildung markierte Programm erfordert keine Parameter, so dass ich die Methode „Ausführen als“ verwenden kann. Als nächstes muss ich entscheiden, ob das Programm als IBM i-Stapelanwendung oder als interaktive IBM i-Anwendung ausgeführt wird. Die Option „Stapel“ verwendet den Befehl SBMJOB, um die Anwendung auszuführen. Bei der Option „Interaktiv“ wird die Anwendung in dem interaktiven Job ausgeführt, der der ausgewählten Verbindung zum fernen System zugeordnet ist. Bei meinem Programm wird ein Bildschirm gesendet, der diverse Eingaben des Anwenders erfordert. Ich muss das Programm daher zwingend interaktiv aufrufen. Doch bevor das gelingt, muss ein interaktiver Job der RDI-Verbindung zugeordnet werden, andernfalls erhalten Sie die folgende Meldung auf dem Bildschirm:

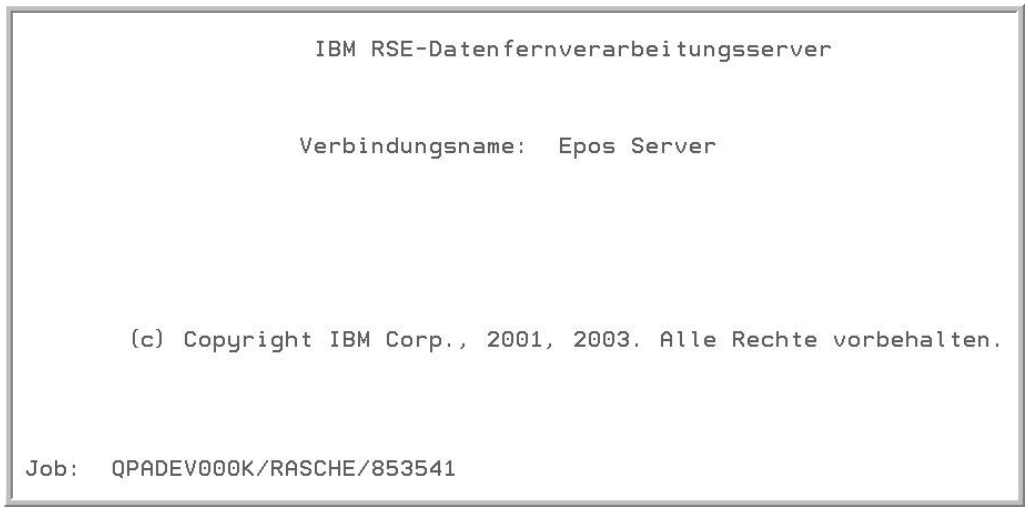


Interaktiven Job zuordnen

Wir werden aufgefordert, mit dem Befehl STRRSESVR der Verbindung einen interaktiven Job zuzuordnen. Die einzelnen Arbeitsschritte werden auch gleich mitgeliefert. Ich starte daher jetzt eine 5250-Emulation und gebe auf der Befehlszeile dieser Sitzung den folgenden Befehl ein:

```
STRRSESVR NAME(Epos Server)
```

Wenn Ihr Verbindungsname Sonderzeichen wie bspw. Leerzeichen enthält, sollten Sie für die Befehlseingabe die Bedienerführung verwenden. Ich habe die Erfahrung gemacht, dass andernfalls Fehlermeldungen in der 5250-Sitzung ausgelöst werden. Wenn alles funktioniert hat, müssten Sie anschließend folgende Anzeige erhalten:



Die RSE-Verbindung wurde aufgebaut

Jetzt kann ich mein Programm aufrufen, der gesendete Bildschirm wird natürlich in der 5250-Sitzung angezeigt:

Die interaktive Programmausführung

Für einen Batch-Job ist im Übrigen keine RSE-Verbindung notwendig. Die Stapelverarbeitung können Sie direkt aufrufen.

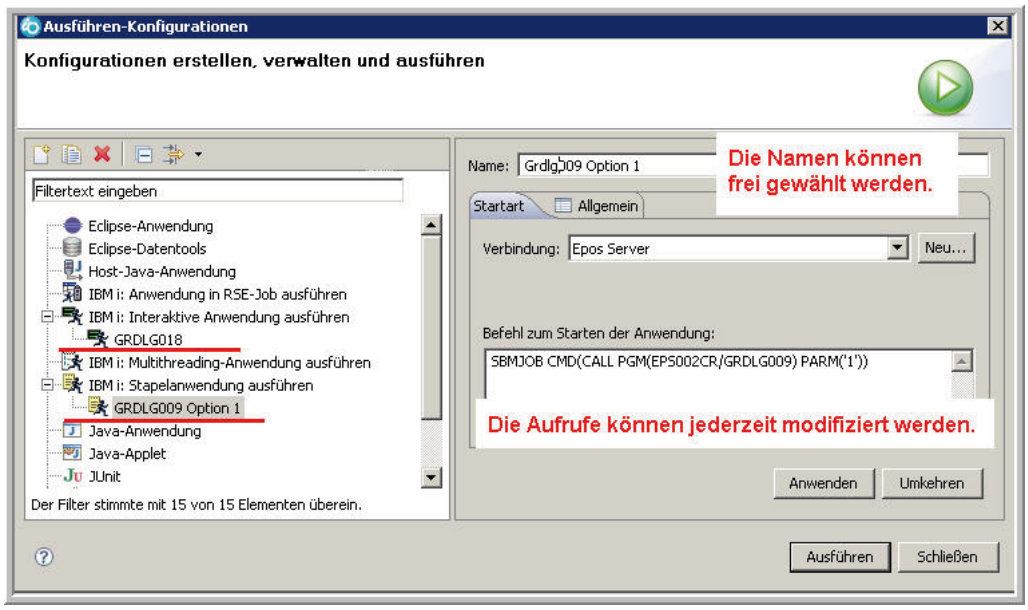
Wird die Option „Ausführen" ohne Bedienerführung für den Aufruf der Programme ausgewählt, wird immer eine Startkonfiguration mit Standardeinstellungen erstellt und gesichert. Startkonfigurationen sind gesicherte Konfigurationen, die Sie aufrufen können, um Programme mehrfach auszuführen. Sie können Programme natürlich auch – wie ich es getan habe – unmittelbar ausführen, Startkonfigurationen ermöglichen jedoch eine höhere Produktivität, da sie die Informationen, die zur Ausführung Ihrer Anwendung erforderlich sind, einschließlich Parameter und Umgebungsvariablen, bereits enthalten. Sie können auch mehrere Startkonfigurationen für dieselbe Anwendung erstellen. Ihre Startkonfigurationen werden dem Menü hinzugefügt und sind auch in der Toolbar verfügbar, wie folgende Abbildung zeigt:

Mit Startkonfigurationen arbeiten

2.12.3
Seite 10

Sie können einfach eine der Startkonfigurationen in der Liste auswählen, um das Programm aufzurufen. Sie können die Startkonfigurationen auch jederzeit manipulieren oder löschen, wenn sie nicht mehr benötigt werden. Dazu müssen Sie lediglich die Option „Ausführen-Konfigurationen“ öffnen.

Es sollte sich daraufhin der folgende Dialog öffnen:



Startkonfigurationen anpassen

Hier kann der Aufruf verändert werden, die Konfigurationsnamen können umbenannt werden und selbstverständlich kann die Konfiguration auch gelöscht werden.

CL-Quellen auffinden (RTVCLSRC)

Eine Besonderheit von CL-Programmen ist es, dass sie einfach „dekompiliert“ werden können, um den Quellcode von Programmen wiederherzustellen, die nur noch als Objekt vorhanden sind. Diese Möglichkeit kann sehr nützlich sein, wenn der Quellcode verloren gegangen ist, oder wenn Sie ein fremdes Programm nur als Objekt zur Verfügung haben, aber eine Änderung daran vornehmen wollen.

Um den Quellcode abzurufen, muss das Programm allerdings mit dem Parameter ALWRTVSRC(*YES) erstellt worden sein. Hierbei handelt es sich jedoch um den Standardwert:

```
                    CL - Programm erstellen (CRTCLPGM)

Auswahl eingeben und Eingabetaste drücken.

RTVCLSRC erlauben  . . . . . . . ALWRTVSRC       *YES
Programm ersetzen  . . . . . . . REPLACE     >   *NO
Ziel-Release . . . . . . . . . . TGTRLS          *CURRENT
```

Der Parameter ALWRTVSRC

Falls Sie nicht wollen, dass Ihre Kunden die Quellen rekonstruieren können, müssen sie diesen Parameter auf *NO ändern.

Um die CL-Quellen abzurufen, nutzen wir den Befehl RTVCLSRC (retrieve cl source).

```
                    CL-Quelle auffinden (RTVCLSRC)

Auswahl eingeben und Eingabetaste drücken.

Programm . . . . . . . . . . . .   qstrup      Name
  Bibliothek . . . . . . . . . .     qsys      Name, *LIBL, *CURLIB
Quellendatei . . . . . . . . . .   QCLSRC      Name
  Bibliothek . . . . . . . . . .     EPS002cr  Name, *LIBL, *CURLIB
Quellenteildatei . . . . . . . .   *PGM        Name, *PGM
Eingeschlossene Quelle abrufen     *NO         *NO, *YES
```

Der Befehl RTVCLSRC

2.12.3.1
Seite 2

Der Befehl hat vier Parameter:
Unter Programm (Parameter PGM) wird der qualifizierte Name des Programms angegeben, das „dekompiliert“ werden soll. Der Parameter „Quellendatei(SRCFILE)“ enthält den qualifizierten Namen der Quellendatei, in der die wiederhergestellte Quelle gespeichert werden soll. Im Parameter „Quellenteildatei(SRCMBR)“ steht schließlich der Teildateiname. Als Standardwert für SRCMBR wird *PGM angenommen, was bedeutet, dass die Teildatei denselben Namen erhält, den auch das Programm hat. Die Teildatei muss bei Ausführung des Befehls noch nicht bestehen. Der vierte Parameter „Eingeschlossene Quelle abrufen(RTVINCSRC)“ gibt an, ob CL-Befehle, die mit INCLUDE-Befehlen in das Original-CL-Quellenprogramm eingeschlossen wurden, abgerufen werden sollen oder nicht. Wenn Sie den Wert *NO wählen, und das Original-CL-Quellenprogramm INCLUDE-Befehle enthielt, wird die abgerufene Quelle dieselben INCLUDE-Befehle enthalten wie das Original, jedoch keine CL-Befehle, die beim Kompilieren der CL-Quelle eingeschlossen wurden. Verwenden Sie hingegen *YES, wird die abgerufene Quelle die eingeschlossenen CL-Quellenbefehle statt der Original INCLUDE-Befehle enthalten. So kann eine CL-Quelle erzeugt werden, die weitestgehend mit der vollständigen CL-Quelle übereinstimmt, die zur Erstellung des CL-Programms benötigt wird.

RTVCLSRC erstellt zwar Quellcode, aber in einem etwas unansehnlichen Format:

```
0001.00 /*************************************************************************/
0002.00 /*                                                                       */
0003.00 /* 5761SS1 V6R1M0 080215 RTVCLSRC-Ausgabe        26.10.09 19:51:59  */
0004.00 /*                                                                       */
0005.00 /* Programmname . . . . . . . . . . . . . :   QSTRUP             PN*/
0006.00 /* Bibliotheksname  . . . . . . . . . . . :   QSYS               PL*/
0007.00 /* Ursprüngliche Quellendatei . . . . . . :   S000028739         SN*/
0008.00 /* Bibliotheksname  . . . . . . . . . . . :   $BLDSS1           SL*/
0009.00 /* Ursprüngliche Quellenteildatei . . . . :   S000028739         SM*/
0010.00 /* Änderung der Quellendatei                                       */
0011.00 /*   Datum/Uhrzeit  . . . . . . . . . . . :   30.09.07 16:27:39 SC*/
0012.00 /* Korrekturangabe  . . . . . . . . . . . :   *NOPATCH           PO*/
0013.00 /* Benutzerprofil . . . . . . . . . . . . :   *USER             UP*/
0014.00 /* Text . . . :                                                   TX*/
0015.00 /* Eigner . . . . . . . . . . . . . . . . :   QSYS               OW*/
0016.00 /* Benutzermodus  . . . . . . . . . . . . :   *NO                UM*/
0017.00 /* Retrieve include source  . . . . . . . :   *YES               RI*/
0018.00 /*                                                               ED*/
0019.00 /*************************************************************************/
```

Das Ergebnis

Am Anfang stehen 20 Kommentarzeilen mit statistischen Informationen über das Programm, aus dem die Quelle wiederhergestellt wurde. Unmittelbar nach den Kommentarzeilen folgt der Quellcode. Alle Anweisungen beginnen einheitlich in der Spalte 6. Leerzeichen, Einrückungen zur besseren Gliederungen oder Kommentare zum besseren Verständnis fehlen leider.

```
/* Eigner . . . . . :   QSYS              OW*/
/* Benutzermodus  . . . . . :   *NO               UM*/
/* Retrieve include source  . . . . . :   *YES            RI*/
/*                                                      ED*/
/*********************************************************************/
     PGM
     DCL VAR(&STRWTRS) TYPE(*CHAR) LEN(1)
     DCL VAR(&CTLSBSD) TYPE(*CHAR) LEN(20)
     DCL VAR(&CPYR) TYPE(*CHAR) LEN(90) VALUE('5761-SS1 (C) COPYRIGHT-
 IBM CORP 1980, 2007. LICENSED MATERIAL - PROGRAM PROPERTY OF IBM')
     QSYS/STRSBS SBSD(QSERVER)
     MONMSG MSGID(CPF0000)
     QSYS/STRSBS SBSD(QUSRWRK)
     MONMSG MSGID(CPF0000)
     QSYS/RLSJOBQ JOBQ(QGPL/QS36MRT)
     MONMSG MSGID(CPF0000)
```

Der Quellcode mit den CL-Anweisungen

Nachdem Sie den Befehl RTVCLSRC ausgeführt haben, sollten Sie den Quellcode mit einem Editor überarbeiten, um ihn lesbarer zu machen:

Löschen Sie als erstes die Kommentarzeilen zu Beginn des Quellcodes

Rufen Sie sich für jede Zeile die Bedienerführung auf, indem Sie den Cursor auf die Anweisung platzieren und anschließend F4 drücken. Wenn Sie anschließend die Eingabetaste drücken, wird der Befehl vom Editor entsprechend formatiert.

Fügen Sie auch entsprechende Leerzeilen ein, um Blöcke zu gliedern und Befehlsgruppen logisch zu kennzeichnen, die zusammen gehören.

Und Sie sollten auch unbedingt versuchen Kommentare einzufügen, um das Programm zu dokumentieren

Selbstverständlich können Sie das Programm beliebig modifizieren und/oder ergänzen. Nur eines sollten Sie unbedingt beachten:

Erstellen Sie das neue, angepasste Programmobjekt in einer separaten Bibliothek! Diese Bibliothek stellen Sie in der Suchliste vor die Originalbibliothek. So bleibt das Original erhalten und Sie profitieren dennoch von den Modifikationen. Und falls doch mal etwas schief laufen sollte, steht das Original schnell wieder zur Verfügung!

2.12.3.1

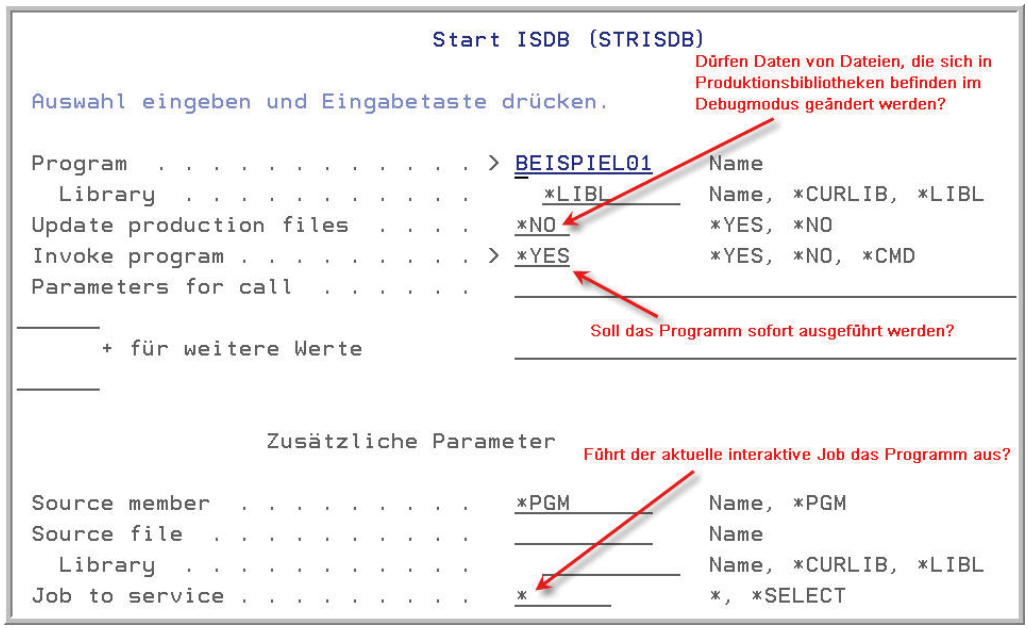
Seite 4

CL-Programme mit dem ISDB-Debugger testen

Was tun, wenn das Programm abstürzt oder sich andere Logikfehler zeigen? Auch erfahrene Programmierer machen Fehler. Und je länger und komplizierter ein CL-Programm ist, desto mehr Gelegenheiten gibt es, Fehler zu machen. Als Hilfsmittel zum Testen der Programme stehen Debugger zur Verfügung. Ein Debugger ermöglicht beispielsweise die Anzeige von Variablen und lässt auch die Veränderung der Variablenwerte zu. Hierzu kann das Programm an einem beliebigen Punkt unterbrochen werden. Außerdem können Sie das Programm „step by step" ausführen. Diese Funktion ist besonders sinnvoll, wenn ein Programm viele IF-Anweisungen und Schleifen enthält, die die Verfolgung des Programms erschweren.

Im 5250-Umfeld müssen Sie sich entscheiden, ob Sie den Befehl STRDBG (Start Debugger) oder STRISDB (Start Interactive Source Debugger) nutzen wollen bzw. müssen. Der Befehl STRISDB ist nur zulässig, wenn ein OPM-Programm getestet wird – also ein CL-Programm der Art CLP. Für ILE-CL-Programme steht hingegen nur der Befehl STRDBG (Start Debugger) im 5250-Umfeld zur Verfügung oder der Debugger des RDIs.

Da das Beispielprogramm ein CLP-Programm ist, kann problemlos der interaktive Source-Debugger genutzt werden:



Bedienerführung STRISDB

Der Befehl STRISDB hat sechs Parameter:

Program (PGM) gibt den qualifizierten Namen des Programms an, das getestet werden soll. In der späteren Debug-Sitzung können weitere Programme mit dem Befehl AddIsdbPgm hinzugefügt werden. Der Debug-Befehl RmvIsdbPgm entfernt überflüssige Programme aus der Debug-Sicht.

2.12.3.2
Seite 2

Update production files(**UPDPRD**) legt fest, ob die Aktualisierung von Dateien in Produktionsbibliotheken erlaubt ist (*YES) oder nicht (*NO). Wenn der Standardwert *NO beibehalten wird und das zu testende Programm einen Satz in der Datei zu aktualisieren versucht, wird eine Fehlermeldung ausgegeben. Wenn das zu testende Programm Daten einfügen oder modifizieren muss, dann sollten Sie entweder den Wert *YES wählen oder die entsprechenden Dateien in Testbibliotheken kopieren.

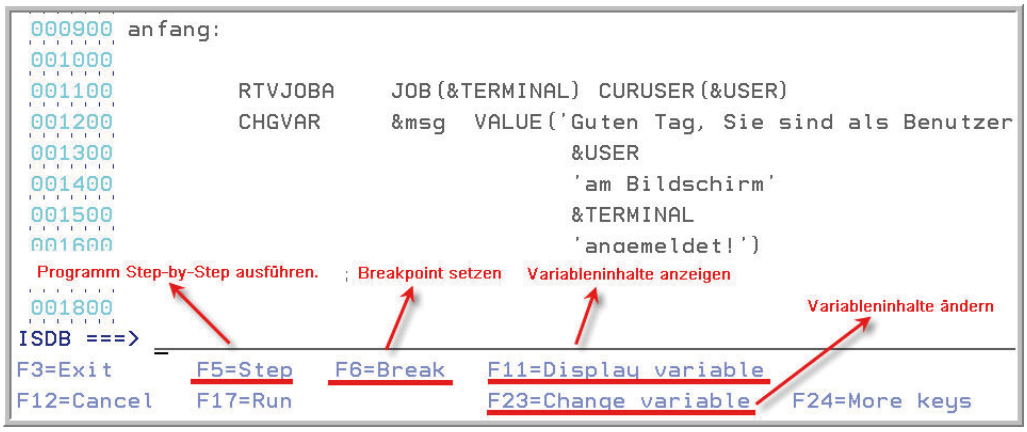
Invoke program (**INVPGM**) ermöglicht den direkten Programmaufruf. Sie müssen in diesem Fall das zu testende Programm nicht mehr mit einer eigenen CALL-Anweisung aufrufen.

Parameters for call (**PARM**) ist nur im Zusammenhang mit dem Programmaufruf zu verstehen. Falls das zu testende Programm sofort ausgeführt wird und Parameter erwartet, können Sie die Parameterwerte hier benennen.

Die Parameter **Sourcemember** (**SRCMBR**) und Sourcefile (SRCF) sind optional und werden eher selten benötigt. Nur wenn die Teildatei, die die Programm-Quellen enthält, einen anderen Namen hat als das zu testende Programmobjekt, müssen Sie hier die korrekte Teildatei spezifizieren.

Der Parameter **Job to service** (**SRVJOB**) ist nur dann erforderlich, wenn ein Programm getestet werden soll, das nicht in Ihrem eigenen interaktiven Job ausgeführt wird. Es könnte beispielsweise vorkommen, dass Sie ein Programm testen wollen, das Sie zur Stapelverarbeitung übergeben haben oder das von einem anderen Benutzer ausgeführt wird. In beiden Fällen ändern Sie den Parameter „Job to Service“ auf *SELECT. Daraufhin erscheint ein zusätzlicher Dialog, der es Ihnen gestattet, den entsprechenden Job zu selektieren.

Wenn alle Parameter gesetzt wurden und der in Abbildung 35 gezeigte Dialog bestätigt wurde, öffnet sich die interaktive Debug-Sitzung:



Arbeitsbildschirm STRISDB

Mit der F5-Taste können Sie Ihr Programm „step by step" ausführen oder mit der F6-Taste Unterbrechungspunkte (Break-Points) hinterlegen, um die Ausführung des Programms bei bestimmten Anweisungen zu unterbrechen. Dazu positionieren Sie den Cursor auf die gewünschte Zeile und drücken die Taste F6. Soll der Unterbrechungspunkt aufgehoben werden, setzen Sie den Cursor einfach wieder in die Unterbrechungszeile und drücken wiederum die F6-Taste. Alternativ können Sie auch den Zeilenbefehl BREAK *zeilennummer* auf der Befehlszeile verwenden. Ein Breakpoint muss aber nicht zwingend zur Programmunterbrechung führen, sondern kann auch an Bedingungen geknüpft werden. Dies kann mit zwei unterschiedlichen Befehlen realisiert werden:

Zum einen kann der Befehl SKIP verwendet werden:

```
BREAK 1200 SKIP 10
```

Dabei wird das Programm erst unterbrochen, nachdem der Breakpoint zehnmal durchlaufen wurde. Diese Methode ist v. a. dann sinnvoll, wenn Programmvariablen nach einer bestimmten Anzahl von Schleifendurchläufen angezeigt werden sollen.

Zum anderen kann aber auch eine Bedingung angegeben werden, unter der der Breakpoint wirksam werden soll:

```
BREAK 1200 WHEN &USER *EQ 'RASCHE'
```

Das Programm wird bei der Anweisung 1200 nur unterbrochen, wenn die Variable &USER den Wert 'RASCHE' enthält.

Sicherlich kann es auch Situationen geben, in denen die Unterbrechungspunkte wieder gelöscht werden sollen. Der Befehl CLEAR *zeilennummer* entfernt den in der Zeile gesetzten Breakpoint. Sollen alle Unterbrechungspunkte entfernt werden, geht das am schnellsten, wenn Sie den Befehl BREAK * verwenden.

2.12.3.2
Seite 4

Nachdem die Breakpoints gesetzt wurden, können Sie das Programm mit der F17-Taste starten. Sobald ein Breakpoint erreicht ist, wird die Programmausführung unterbrochen und die bisherigen Ergebnisse können analysiert werden oder das Programm wird „step by step“ (F5) fortgesetzt.

Zusätzlich können Sie jederzeit Variableninhalte anzeigen. Dazu positionieren Sie den Cursor einfach auf die entsprechende Variable und drücken die F11-Taste – wie folgende Abbildung zeigt:

Funktionen der Testhilfe

Alternativ steht der Befehl WATCH *variablenname* zur Verfügung, um sich dauerhaft die Inhalte wichtiger Variablen anzeigen zu lassen.

Haben Sie Ihr Programm ausreichend getestet, dann können Sie den Debugger einfach über die F3-Taste verlassen. Der Debug-Modus für das Programm wird dadurch automatisch beendet.

Der interaktive Debugger ist zum Testen der OPM-Programme gut zu gebrauchen. Er ist übersichtlich und bietet alle Funktionen, die notwendig sind. Schade, dass hiermit keine ILE-Programme getestet werden können, und der Debugger reagiert leider auch auf COPY-Strecken empfindlich.

CL-Programme aufrufen

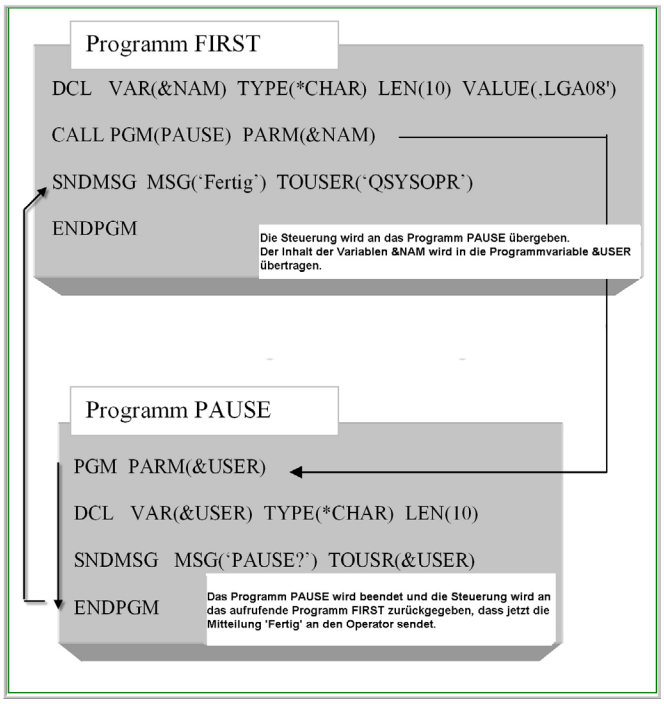
Das CL-Programm können Sie anschließend mit dem Befehl CALL ausführen.

```
                          Programm aufrufen (CALL)

 Auswahl eingeben und Eingabetaste drücken.

 Programm . . . . . . . . . . . . > PAUSE      Name
   Bibliothek . . . . . . . . . .    *LIBL     Name, *LIBL, *CURLIB
 Parameter  . . . . . . . . . . . > 'WERNER'
 ______
             + für weitere Werte   ____________________________________
 ______

                                                                  Ende
 F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
 F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Aufruf des Programms mit CALL

Der Befehl CALL ruft das Programm PAUSE auf und übergibt ihm den Benutzernamen 'WERNER'.

Die Anweisung CALL kann auch innerhalb eines Programms verwendet werden, um ein weiteres Programm aufzurufen. Nach Beendigung des aufgerufenen Programms wird die Steuerung wieder an den Anwender oder an das aufrufende Programm zurückgegeben.



Aufruf eines anderen Programms mit CALL

Wenn Sie – wie bei diesem Beispiel –Variablen als Parameter übergeben, müssen Sie sicherstellen, dass die Parameter zueinander passen. Die Variablennamen müssen nicht gleich sein, da die Parameter sequentiell und nicht nach Namen übergeben werden. In der Regel müssen die Variablen aber denselben Datentyp und die identische Länge haben. Eine Ausnahme bilden alphanumerische Variablen: Sie können in der Länge variieren. Allerdings sind die Ergebnisse in diesem Fall nicht immer vorhersehbar und daher sollten Sie diese Situationen vermeiden. Eine Besonderheit sind auch Gleitkommazahlen. Dieser Datentyp ist einem CL-Programm nicht bekannt, daher können CL-Programme Gleitkommazahlen auch nicht verarbeiten. Sie können Gleitkommazahlen aber als Parameter übergeben, indem Sie die Gleitkommazahl in einer 8-stelligen alphanumerischen Variablen empfangen und so auch an ein anderes Programm weitergeben. Bei der Übergabe konstanter Werte müssen Sie mehr Regeln beachten:

Alphanumerische Konstanten müssen dieselbe Länge wie die Empfangsvariable haben. Es sei denn, das aufgerufene Programm erwartet eine Variable mit 32 Byte oder weniger. In diesem Fall wird die übertragende Zeichenkette mit Leerzeichen aufgefüllt oder abgeschnitten.

Dezimale Konstanten werden grundsätzlich mit der Länge 15,5 übergeben. Ihre Empfangsvariable muss somit mit dem TYPE (*DEC) LEN(15 5) deklariert sein. Erwartet das aufgerufene Programm eine andere Variablenlänge, müssen Sie hexadezimale Konstanten verwenden.

Hexadezimale Konstanten müssen die korrekte Länge haben. Wenn das aufgerufene Programm z.B. einen 6-stelligen numerischen Wert mit zwei Nachkommastellen erwartet, müssen Sie eine 8-stellige hexadezimale Konstante übergeben.

Die Anweisung ENDPGM kennzeichnet das Programm-Ende. Wird die Anweisung erreicht, erhält das aufrufende Programm die Steuerung zurück und führt die Anweisung aus, die dem Aufruf folgt. Wie bereits erwähnt, kennzeichnet die Anweisung ENDPGM nur das physische Ende des CL-Programms, führt aber nicht unmittelbar zur Beendigung des Programms. Soll die Steuerung an das aufrufende Programm unmittelbar zurückgeben werden, können Sie die Anweisung RETURN nutzen. Die Anweisung RETURN beendet an jeder beliebigen Stelle die Ausführung des Programms.

Grundwerk 10/2004

2.12.4

Seite 4

Konstanten, Variablen

Konstanten und Literale sind Werte, die sich im Programm nicht verändern. Konstanten können alphanumerisch, dezimal, hexadezimal und logisch sein.

```
/* alphanumerische Konstante: , System wird abgeschaltet'  */
SNDMSG          MSG(' Das System wird gleich   +
                abgeschaltet')     TOUSR('QSYSOPR')
```

Alphanumerische Konstanten enthalten eine beliebige Anzahl von Zeichen – einschließlich des Leerzeichens. Alphanumerische Konstanten, die Leerzeichen beinhalten oder nur aus Ziffern bestehen, müssen in einfache Anführungszeichen eingeschlossen werden.

```
/* dezimale Konstante: 100,50           */
CHGVAR  VAR(&ZAHL)  VALUE(&ZAHL   +   100.50)
```

Dezimale Konstanten dürfen nur die Ziffern 0 bis 9, ein Vorzeichen und ein Dezimaltrennzeichen enthalten.

```
/* Hexadezimale Konstante: X'00'         */
IF COND(&NAME = X'00')
```

Hexadezimale Konstanten werden für Werte verwendet, die nicht über die Tastatur erreichbar sind. Sie beginnen grundsätzlich mit einem X, gefolgt von paarweise angegebenen Elementen, die aus den Werten 0 bis 9 und A bis F bestehen.

```
/* Logische Konstante: '1'            */
DCL   VAR(&LG)   TYPE(*LGL)
CHGVAR  VAR(&LG)     VALUE('1')
```

Logische Konstanten können nur die Werte '0' oder '1' annehmen.

Variable haben im Gegensatz zu Konstanten veränderbare Inhalte. Sie werden genutzt, um Daten für die Dauer der Programmausführung im Arbeitsspeicher zu halten oder zu verändern. Innerhalb eines Programms erhalten Variablen einen eindeutigen Namen. Bevor Sie eine Variable verwenden können, muss die Variable deklariert werden. Die CL-Variablen müssen unmittelbar nach der PGM-Anweisung und vor den eigentlichen Programmanweisungen deklariert werden. Sie deklarieren die Variablen mit der Anweisung DCL (Declare). Die Anweisung DCL hat vier Parameter:

Grundwerk 10/2004

2.12.5
Seite 2

Der Parameter VAR enthält den Namen der Variablen. Der Variablentyp wird mit dem Parameter TYPE deklariert. Die Variablenlänge steht im Parameter LEN und ein möglicher Anfangswert kann im Parameter VALUE hinterlegt werden.

VAR	Name der Variablen	Alle Variablennamen beginnen mit einem vorangestellten &-Zeichen. Das zweite Zeichen muss alphabetisch oder eines der Zeichen @, # oder $ sein. Der weitere Variablenname kann die Zeichen A-Z, 0-9, @, #, $, _ beinhalten.	Zwingend erforderlich
TYPE	Datentyp	*DEC Dezimal gepackter Wert	Zwingend erforderlich
		*CHAR Zeichenwert	
		*LGL Boolesche Variable, logischer Wert '0' bzw. '1'	
		*INT Binärwert mit Vorzeichen	
		*UINT Binärwert ohne Vorzeichen	
LEN	Länge der Variablen	*DEC max. 15 Byte, davon max. 9 Nachkommastellen	optional
		*CHAR max. 9999 Byte	
		*LGL 1 Byte	
		*INT 2 oder 4 Byte	
		*UINT 2 oder 4 Byte	
VALUE	Initialisierungswert	Maximal 3000 Zeichen	optional

```
DCL     VAR(&OUTQ) TYPE(*CHAR)  LEN(10)
        +     VALUE(,QSYSOPR')
/* Deklariert die 10-stellige alphanumerische Variable OUTQ,  +
die den Wert QSYSOPR enthält */
DCL     VAR(&ZAHL)    TYPE(*DEC)     LEN(3  1)    +   VALUE(16,0)
/* Deklariert die 3-stellige numerische Variable ZAHL mit     +
    einer Nachkommastelle. Sie können maximal den Wert 99,9   +
    speichern     */
DCL      VAR(&LG)    TYPE(*LGL) VALUE(,1')
/* Deklariert die logische Variable LG mit dem Wert wahr(=1)   */
```

Sie können bei der Variablendeklaration auf Initialisierungswerte verzichten.

Zeichenvariablen werden dann mit Leerzeichen, Dezimal- bzw. Integervariablen mit 0 und logische Variablen mit dem Wert FALSCH (=0) initialisiert.

```
DCL   VAR(&NME)     TYPE( *CHAR)
/* Deklariert die Zeichenvariable & NME der Länge          +
      32 Byte. Die Variable ist mit Leerzeichen initialisiert.   */
```

Auch die Längenangabe ist nicht zwingend erforderlich. Integervariablen erhalten standardmäßig die Länge 4 und logische Variablen werden mit der Länge 1 angelegt. Wird der Parameter LEN nicht angegeben, nimmt das System für alphanumerische Variablen den Standardwert 32 Byte und für numerische Variablen den Wert 15,5 an, es sei denn, Sie verwenden einen Initialisierungswert, der dann die Längenangabe bestimmt.

```
DCL  VAR(&NME)     TYPE( *CHAR)      VALUE(,Pause?')
/* Deklariert die Variable & NME der Länge 6 Byte.          +
    Die Länge entspricht dem Initialisierungswert, da die    +
    explizite Längenangabe fehlt. Die Variable ist mit       +
    dem Wert ,Pause?' initialisiert.   */
```

Ein CL-Programm ist schwer zu verstehen, wenn die Längenangaben nicht explizit angegeben werden. Denn dieses Vorgehen setzt voraus, dass Sie die vom System vergebenen Standardwerte kennen. Um die Programme wartungsfreundlicher zu gestalten, sollte der Parameter LEN also grundsätzlich angegeben werden.

Variablen können innerhalb eines CL-Programms jeden Parameterwert ersetzen. Wird ein Befehl manuell eingegeben, übergeben Sie stets konstante Werte. Stellen Sie sich unseren SNDMSG-Befehl vor:

```
SNDMSG  MSG('Pause?')  TOUSR('WERNER')
```

Der Parameter enthält den 6-stelligen alphanumerischen Wert 'PAUSE?' und für TOUSR wird der Wert 'WERNER' übergeben. Wenn Sie den Befehl in einem CL-Programm verwenden, haben Sie die Möglichkeit, anstelle der Konstanten Variablen zu verwenden. Variablen ermöglichen wesentlich flexiblere Programme. Betrachten wir zwei Beispiele:

```
PAUSE:       PGM
BEGIN:       SNDMSG  MSG('PAUSE?')  TOUSR(WERNER)
ENDE:        ENDPGM
```

Das erste Programm sendet an den Benutzer WERNER die Nachricht 'PAUSE?'. Wenn Sie eine andere Nachricht senden wollen oder wenn die Nachricht an einen anderen Benutzer gesendet werden soll, brauchen Sie ein

neues CL-Programm. Werden die konstanten Werte aber durch Variablen ersetzt, ist der Einsatz des Programms flexibler:

```
PAUSE:          PGM  PARM(&USER &TEXT)
VARIABLE:       DCL   VAR(&USER)    TYPE(*CHAR)  LEN(10)
                DCL   VAR(&TEXT) TYPE(*CHAR) LEN(100)
BEGIN:          SNDMSG   MSG(&TEXT)  TOUSR(&USER)
ENDE:           ENDPGM
```

Das Programm ersetzt beide Parameterwerte durch Variable. Sie bestimmen jetzt im Aufruf des Programms, wer die Nachricht erhält und welcher Text gesendet wird.

Grundlegende Funktionen und Operationen

Sie wissen jetzt, wie Sie Variablen und Konstanten verwenden können. Es fehlen uns allerdings die grundlegenden Funktionen und Operationen, um ein erstes CL-Programm zu schreiben.

Ich bin der Meinung, dass man durch die praktische Anwendung schneller lernt, als wenn man nur Syntaxdiagramme und die dazugehörigen Erläuterungen liest. Daher werde ich alle Anweisungen in ein CL-Programm integrieren, so dass wir am Ende jeweils ein in sich geschlossenes Programm erhalten, das alle vorgestellten Funktionen und Operationen im praktischen Einsatz zeigt. Bitte bedenken Sie, dass Sie das Programm nicht unbedingt eins zu eins in Ihrer Umgebung verwenden können, da der Einsatz der Tools natürlich sehr stark von Ihrer eigenen Organisation abhängig ist.

2.12.6
Seite 2

Systemwerte nutzen

Im Herbst stellen wir wieder von Sommer- auf Winterzeit um. Sie müssen dann den Systemwert QHOUR manuell ändern. Natürlich gibt es mittlerweile auch viele Tools, die die Zeitumstellung automatisieren. Wir nutzen dieses Problem, um gemeinsam ein CL-Programm zu entwickeln.

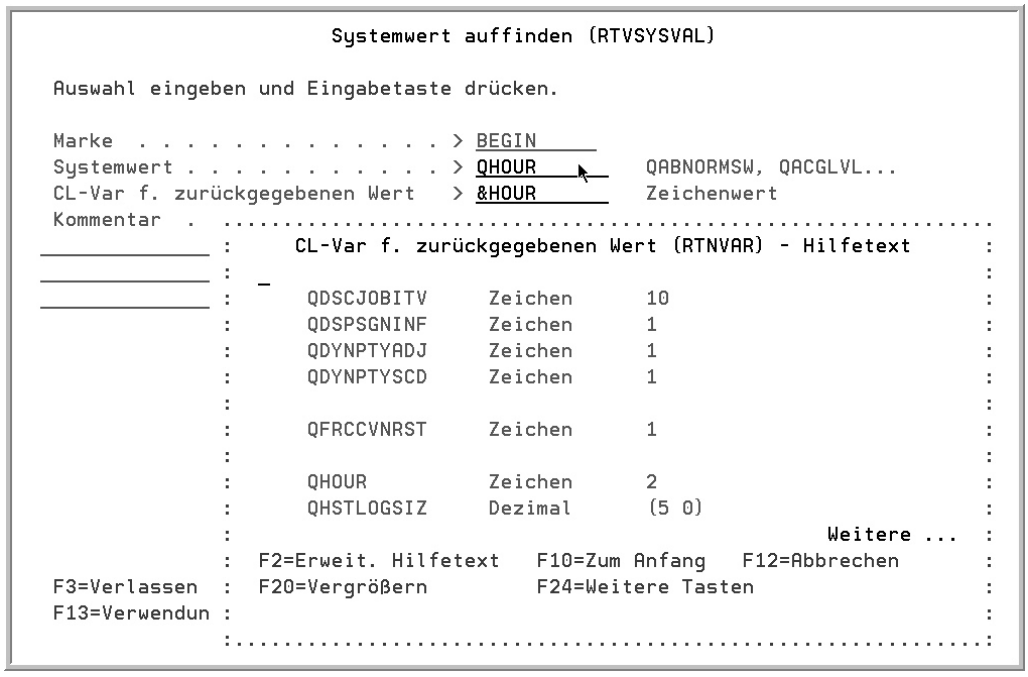
Systemwerte haben generell viel Einfluss auf die Arbeitsweise der iSeries. Und es könnte notwendig sein, bestimmte Systemwerte in einem CL-Programm festzustellen. Mit der Anweisung RTVSYSVAL (Retrieve System Value) können Sie beliebige Systemwerte in eine CL-Variable kopieren und anschließend behandeln wie jede andere Variable.

```
                    Systemwert auffinden (RTVSYSVAL)

Auswahl eingeben und Eingabetaste drücken.

Marke  . . . . . . . . . . . . . > BEGIN
Systemwert . . . . . . . . . . .   qhour       QABNORMSW, QACGLVL...
CL-Var f. zurückgegebenen Wert     &hour_      Zeichenwert
Kommentar  . . . . . . . . . . .   _____________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________________


                                                                        Ende
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Systemwerte mit RTVSYSVAL auffinden

Die Variable, in die der Systemwert gespeichert wird, muss dieselbe Datenart haben wie der abgefragte Systemwert. Der Systemwert QHOUR ist ein 2-stelliger alphanumerischer Wert. Welcher Datentyp für die verschiedenen Systemwerte notwendig ist, können Sie der Bedienerhilfe entnehmen.

2.12.6.1
Seite 2



Bedienerhilfe zum zurückgegebenen Wert

Sie können jetzt den Systemwert QHOUR in einer Variablen speichern, die Variable manipulieren und anschließend kann das CL-Programm den Systemwert mit Hilfe der Anweisung CHGSYSVAL verändern. Die Anweisung CHGSYSVAL erwartet als Parameter den Namen des zu ändernden Systemwertes und den neuen Wert, der als Konstante oder Variable übergeben wird.

```
PGM
DCL VAR(&HOUR) TYPE(*CHAR) LEN(2)
RTVSYSVAL  SYSVAL(QHOUR) RTNVAR(&HOUR)
/* Der Variablen &HOUR wird der Inhalt des Systemwertes      +
   QHOUR zugewiesen.                    */
                        :
                        :
CHGSYSVAL  SYSVAL(QHOUR) VALUE(&HOUR)
/* Der Systemwert QHOUR wird verändert, indem der Inhalt     +
   der Variablen &HOUR übertragen wird.       */
```

Auf diese Weise ist es möglich, jeden Systemwert in einem CL-Programm abzufragen und gegebenenfalls zu verändern. Es gibt weitere RTV-Befehle, die ausschließlich in CL-Programmen verwendet werden können und dem Programm nützliche Werte zur Verfügung stellen.

RTVAUTLE	Berechtigungslisteneintrag auffinden
RTVBCKUP	Sicherungsattribute auffinden
RTVCFGSRC	Konfigurationsquelle auffinden
RTVCLSRC	CL-Quelle auffinden
RTVCURDIR	Aktuelles Verzeichnis auffinden
RTVDSKINF	Platteninformationen auffinden
RTVDTAARA	Datenbereich auffinden
RTVGRPA	Gruppenattribute auffinden
RTVJOBA	Jobattribute auffinden
RTVJRNE	Journaleintrag auffinden
RTVLIBD	Bibliotheksbeschreibung abrufen
RTVMBRD	Teildateibeschreibung abrufen
RTVMSG	Nachricht auffinden
RTVNETA	Netzwerkattribute auffinden
RTVOBJD	Objektbeschreibung auffinden
RTVPWRSCDE	Ein-/Ausschaltzeit des Systems abrufen
RTVQMFORM	Query Formular auffinden
RTVQMQRY	Query Abfrage auffinden
RTVSYSINF	Systeminformationen auffinden
RTVSYSVAL	Systemwerte auffinden
RTVUSRPRF	Benutzerprofilparameter abrufen

2.12.6.1

Seite 4

Die Anweisung CHGVAR

Mit der Anweisung CHGVAR (Change Variable) weisen Sie einer Variablen einen neuen Wert zu.

```
CHGVAR VAR(&HOUR) VALUE('12')
/* &HOUR enthält jetzt den alphanumerischen Wert 12        */
```

Die Anweisung hat zwei verbindliche Parameter: Parameter VAR beinhaltet die Zielvariable und Parameter VALUE enthält den zu übertragenden Wert. Der Parameter VALUE kann entweder eine Konstante, eine Variable oder einen Ausdruck enthalten.

- Bei alphanumerischen Variablen, darf der VALUE-Parameter alphanumerische oder numerische Werte enthalten. Die Zielvariable wird zunächst mit Leerzeichen initialisiert und der zu übertragende Wert anschließend in das Zielformat konvertiert.
- Bei numerischen Variablen kann der VALUE-Parameter ebenfalls alphanumerische oder numerische Werte enthalten. Die numerische Variable wird mit Null initialisiert, numerische Werte werden direkt übertragen, alphanumerische Werte werden zunächst konvertiert (=umgeformt) und dann übertragen.

```
CHGVAR VAR(&HOUR) VALUE('12')
/* &HOUR enthält jetzt den alphanumerischen Wert 12.         */
CHGVAR VAR(&HOUR_D) VALUE(12,75)
/* Fehler, denn CL verwendet den Punkt als Dezimalzeichen.   */
CHGVAR VAR(&HOUR_D) VALUE(&HOUR)
/* Der Inhalt der alphanumerischen Variable &HOUR wird in       +
    numerisches Format konvertiert.  */
CHGVAR VAR(&HOUR_D) VALUE('1M1')
/* Fehler, denn es dürfen bei der Konvertierung nur die Ziffern  +
     0 bis 9, ein Vorzeichen und ein Dezimalzeichen verwendet    +
     werden.            */
```

2.12.6.2

Seite 2

Arithmetische Operatoren

In der Anweisung CHGVAR können arithmetische Ausdrücke verwendet werden. Wenn es sich um einen Ausdruck handelt, wird er zunächst ausgewertet, bevor das Ergebnis der Variablen zugewiesen wird. CL kennt nur die vier Grundrechenarten:

+	Addition
-	Subtraktion
/	Division
*	Multiplikation

Im Allgemeinen ist es egal, ob Sie vor oder nach den Operatoren Leerzeichen verwenden. Die einzige Ausnahme ist der Divisionsoperator, weil der Schrägstrich auch als Trennzeichen für qualifizierte Namen verwendet werden kann. Um Probleme zu vermeiden, sollten Sie allerdings grundsätzlich vor und nach den Operatoren Leerzeichen verwenden.

```
CHGVAR VAR(&ERG) VALUE(2 * 3 * 4)
/*   Die Variable &ERG erhält den Wert 24                    */
CHGVAR VAR(&HOUR_D) VALUE(12)
CHGVAR VAR(&HOUR_D) VALUE(&HOUR_D - 1)
/* &HOUR_D erhält den numerischen Wert 11                    */
```

Wenn wir unseren Systemwert QHOUR um eine Stunde zurücksetzen wollen, müssen wir zunächst die alphanumerische Variable in einen numerischen Datentyp konvertieren. Anschließend kann der Wert 1 subtrahiert werden und der neue Wert wird wieder in die alphanumerische Variable übertragen, um dann den Systemwert QHOUR zu verändern.

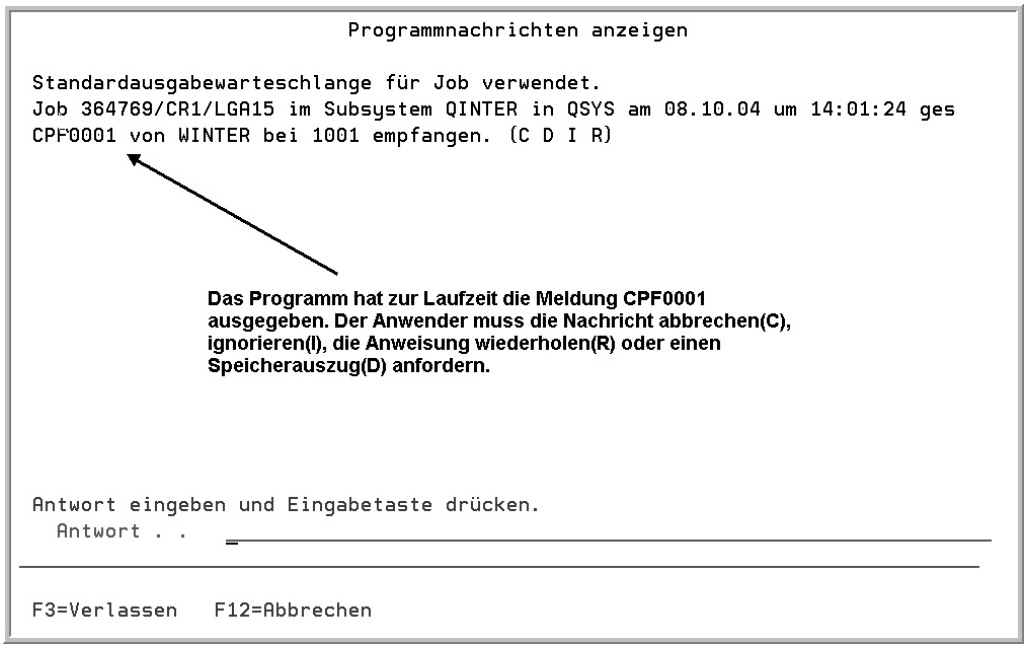
```
PGM
DCL VAR(&HOUR) TYPE(*CHAR) LEN(2)
DCL VAR(&HOUR_D) TYPE(*DEC) LEN(2 0)

RTVSYSVAL  SYSVAL(QHOUR) RTNVAR(&HOUR)
CHGVAR VAR(&HOUR_D) VALUE(&HOUR)
CHGVAR VAR(&HOUR_D) VALUE(&HOUR_D - 1)
CHGVAR VAR(&HOUR) VALUE(&HOUR_D)
CHGSYSVAL  SYSVAL(QHOUR) VALUE(&HOUR)
ENDPGM
```

2.12.6.3

Seite 2

Fehlerbehandlung – Die Anweisung MONMSG

Was passiert aber jetzt in unserem Programm, wenn eine der Anweisungen zur Laufzeit einen Fehler verursacht? Schlägt einer der CL-Befehle fehl, wird das Programm abgebrochen. Ein interaktives Programm sendet dem Anwender die Anzeige für Programmnachrichten. Der Anwender muss die Nachricht beantworten, damit das Programm fortgesetzt wird.



Nachricht nach einem Programmabbruch

Ist das Programm ein Batchjob, dann erhält der Operator die Nachricht. Viele Fehlersituationen können aber bereits im Programm selbst abgefangen werden. Stellen Sie sich vor, der Anwender, der unser Programm aufruft, ist nicht berechtigt, Systemwerte zu verändern. In dieser Situation ist die Anweisung MONMSG (Monitor Message) notwendig. Die Nachrichten, die Sie überwachen, sind größtenteils Abbruchnachrichten. Die entsprechenden Nachrichten-IDs der einzelnen Anweisungen können Sie der Bedienerhilfe oder den CL-Referenzhandbüchern entnehmen.

2.12.6.4
Seite 2

```
                         Systemwert ändern (CHGSYSVAL)
.........................................................................
:                       Systemwert ändern - Hilfetext                   :
:  _                                                                    :
:      CPF18A4     Benutzer zum Ändern des Systemwertes &1 nicht        :
:                  berechtigt.                                          :
:      CPF18C0     Systemwert &1 kann nicht geändert werden.            :
:      CPF1830     Für Systemwert &1 angegebene Werte sind ungültig.    :
:      CPF1831     Benutzer zum Ändern des Systemwertes &1 nicht        :
:                  berechtigt.                                          :
:      CPF1832     Während einleitendem Programmladen kann Systemwert &1 :
:                  nicht geändert werden.                               :
:      CPF1842     Zugriff auf Systemwert &1 nicht möglich.             :
:      CPF1852     Systemwert &1 nicht geändert.                        :
:      CPF1856     Filterart &4 für Systemwert &1 nicht korrekt.        :
:      CPF1857     Für &1 angegebener Wert ist keine codierte Schriftart. :
:      CPF1864     Benutzer zum Ändern des Systemwertes &1 nicht        :
:                  berechtigt.                                          :
:      CPF210C     Bibliothek &1 nicht geändert.                        :
:                                                          Weitere ...  :
:  F3=Hilfetext verlassen    F10=Zum Anfang   F12=Abbrechen             :
:  F13=Unterstützende Info   F14=Hilfetext drucken                      :
:                                                                       :
:.......................................................................:
```

Die Nachrichten-IDs in der Bedienerhilfe

Sie sehen, dass der CHGSYSVAL-Befehl die ESCAPE-Nachricht CPF18A4 senden wird, wenn der Anwender nicht über die erforderlichen Berechtigungen verfügt. Ich habe allerdings die Erfahrung gemacht, dass die genannten IDs häufig nicht die gewünschte Wirkung erzielen. Aus diesem Grund simuliere ich die Fehlersituation am liebsten im System und entnehme die entsprechende Nachrichten-ID der Anzeige für Programmnachrichten. In unserem Beispiel erhalten wir die Nachricht CPF0001, falls der Anwender nicht über die nötigen Berechtigungen verfügt. Diese spezifische Fehlermeldung kann jetzt unmittelbar im Programm bearbeitet werden, indem die MONMSG-Anweisung mit der entsprechenden Nachrichten-ID eingesetzt wird.

```
CHGSYSVAL  SYSVAL(QHOUR) VALUE(&HOUR)
MONMSG    MSGID( CPF0001)
```

Da die Fehlermeldung jetzt im Programm abgefangen ist, erhält der Anwender keine Anzeige mehr, sondern das Programm wird einfach fortgesetzt. Der Parameter MSGID kann 50 IDs aufnehmen. Alle Nachrichten-IDs werden dann identisch behandelt, d.h.: In unserem Beispiel wird das Programm sequentiell fortgesetzt. Da die MONMSG-Anweisung in unserem Beispiel unmittelbar der Anweisung CHGSYSVAL folgt, bezieht sich die Fehlerprüfung auch nur auf diese Anweisung. Alle anderen Laufzeitfehler führen weiterhin zum Programmabbruch. Sie können Fehlermeldungen aber nicht nur auf Anweisungsebene, sondern auch auf Programmebene überwachen.

```
PGM
DCL VAR(&HOUR) TYPE(*CHAR) LEN(2)
DCL VAR(&HOUR_D) TYPE(*DEC) LEN(2 0)

MONMSG  MSGID(CPF0000)   EXEC(                          +
                     GOTO CMDLBL(FEHLER))
RTVSYSVAL  SYSVAL(QHOUR) RTNVAR(&HOUR)
CHGVAR VAR(&HOUR_D) VALUE(&HOUR)
CHGVAR VAR(&HOUR_D) VALUE(&HOUR_D - 1)
CHGVAR VAR(&HOUR) VALUE(&HOUR_D)
CHGSYSVAL  SYSVAL(QHOUR) VALUE(&HOUR)
RETURN
FEHLER:
ENDPGM
```

Jetzt steht die MONMSG-Anweisung am Programmanfang. Dadurch werden **alle** ausführbaren Anweisungen überwacht. Wenn irgendeiner der Befehle eine Fehlernachricht sendet, verzweigt das Programm zur Sprungmarke FEHLER.

Sie können spezifische Fehlermeldungen oder allgemeine Fehlermeldungen überwachen. Bei der spezifischen Fehlerüberwachung werden konkrete Nachrichten-IDs verwendet. Um alle Fehlermeldungen – unabhängig von der konkreten Nachrichten-ID – zu unterdrücken, verwendet man die allgemeine ID CPF0000. So stellen Sie sicher, dass das Programm auf keinen Fall abgebrochen wird. Erinnern Sie sich noch an das modifizierte Programm QSTRUP in der eigenen Systembibliothek, das ausgeführt wird, wenn Sie die iSeries starten? Dieses Programm sollte auf keinen Fall eine Abbruchnachricht senden. Daher verwendet man in diesen Fällen am besten die allgemeine Nachrichten-ID CPF0000 auf Anweisungs- oder Programmebene.

Das Programm QSTRUP.
Achtung überschreiben Sie nicht das Originalprogramm!

```
          ****************** Datenanfang ************
0021.00       PGM
0022.00       CALL PGM(QSYS/QSTRUP)
0023.00       QSYS/STRSBS SBSD(QPGMR)
0023.01       MONMSG MSGID(CPF0000)
0023.02       QSYS/STRSBS SBSD(KDSBS01)
0024.00       MONMSG MSGID(CPF0000)
0025.00       QSYS/STRTCP
0026.00       MONMSG MSGID(CPF0000)
0043.00       ENDPGM
          ******************Datenende **************
```

Das modifizierte Programm QSTRUP

Bisher wird die Fehlermeldung nur unterdrückt und das Programm sequentiell fortgesetzt. Die MONMSG-Anweisung hat aber zwei weitere Parameter: CMPDTA und EXEC.

Der Parameter CMPDTA

wird nur selten genutzt. Er gestattet, Fehlerbehandlungen feiner abzustimmen. Der Nachrichtentext für die Fehlermeldung CPF1842 lautet „Zugriff auf Systemwert &1 nicht möglich“. &1 kennzeichnet eine eingebettete Ersatzvariable. Wenn der Fehler z.B. nur behandelt werden soll, wenn der Zugriff auf den Systemwert QHOUR nicht möglich ist, können Sie den Parameter CMPDTA einsetzen:

```
MONMSG   MSGID(CPF1842)  CMPDTA('QHOUR')
```

Der Parameter EXEC

legt die Aktion fest, die im Fehlerfall ausgeführt werden soll. Wenn die MONMSG-Anweisung keine EXEC-Klausel enthält, wird das Programm im Fehlerfall sequentiell fortgesetzt. Wird die MONMSG-Anweisung auf Programmebene eingesetzt, darf im EXEC-Parameter nur der Befehl GOTO verwendet werden; auf Anweisungsebene können Sie einen beliebigen Befehl verwenden.

```
MONMSG   MSGID(CPF1842)  CMPDTA('QHOUR') EXEC(+ goto
cmdlbl(Fehler))
```

Die Anweisung GOTO

Ein Programmierer wird heute normalerweise die Anweisung GOTO meiden, denn die meisten modernen Programmiersprachen verfügen über Steueranweisungen wie DO WHILE, DO UNTIL, Unterroutinen oder Abbruchbedingungen für Schleifen. CL enthält bis zum Release V5R3 keine dieser Anweisungen. Deshalb mussten viele Strukturen mit GOTO-Befehlen erzeugt werden. Die GOTO-Anweisung ist einfach anzuwenden. Die Anweisung hat nur einen Parameter: CMDLBL. Der Parameter CMDLBL muss den Namen einer Sprungmarke (Label) enthalten, bei der die Ausführung des Programms fortgesetzt werden soll.

```
MONMSG   MSGID(CPF0000)   EXEC(                                    +
                              GOTO CMDLBL(FEHLER))
RTVSYSVAL  SYSVAL(QHOUR) RTNVAR(&HOUR)
        :
/* Da unser CL-Programm sequentiell abgearbeitet wird,            +
     müssen wir verhindern, dass die Fehlermeldungen gesendet     +
     werden, obwohl kein Fehler aufgetreten ist. Darum wird das   +
     Programm mit der Anweisung  RETURN beendet.                  +
     Alternativ könnte auch der Befehl:                           +
     GOTO CMDLBL(ENDE)                                            +
     verwendet werden         */
RETURN
/* Programm-Ende bei fehlerfreier Ausführung  */
FEHLER:
ENDPGM
/* Programm-Ende bei fehlerhafter Ausführung  */
```

In dem Beispiel steht das Label FEHLER hinter dem GOTO-Befehl. Es ist allerdings auch möglich, zu einem vorher ausgeführten Abschnitt des Programms zurückzukehren. Sie müssen nur darauf achten, dass die Namen der Sprungmarken innerhalb des Programms eindeutig sind.

2.12.6.5

Seite 2

Die Anweisung SNDMSG

Mit dem Befehl SNDMSG können frei formulierte Informations- und Abfragenachrichten gesendet werden. Der Nachrichtentext kann max. 512 Zeichen lang sein. Als Empfänger der Nachricht können Sie entweder den Parameter TOUSR oder den Parameter TOMSGQ verwenden. Der Parameter TOUSR kann den Namen eines Benutzerprofils oder den Sonderwert *SYSOPR enthalten, wenn die Nachricht an den Operator geschickt werden soll. Mit dem Parameterwert *REQUESTER erhält der Anwender, der das Programm aufgerufen hat, die Nachricht. Wenn Sie den Parameterwert *ALLACT verwenden, erhalten alle angemeldeten Benutzer die Nachricht. Sollen mehrere Benutzer gleichzeitig die Nachricht erhalten, müssen Sie den Parameter TOMSGQ verwenden, weil er bis zu 50 Einträge erlaubt.

Da das Systemprotokoll QHST eine Nachrichtenwarteschlange ist, können Sie den SNDMSG-Befehl auch dazu verwenden, Einträge im Systemprotokoll vorzunehmen.

Mit dem Parameter MSGTYPE legen Sie die Nachrichtenart fest: *INFO (Informationsnachricht) oder *INQ (Abfragenachricht). Eine Abfragenachricht kann nicht an mehrere Anwender gleichzeitig gesendet werden. Wenn Sie eine Abfragenachricht schicken, müssen Sie mit dem Parameter RPYMSGQ bestimmen, welche Nachrichtenwarteschlange die Antwort erhalten soll. Der Standardwert *WRKSTN legt fest, dass die Antwort in die Nachrichtenwarteschlange des Anwenders gestellt wird, der die Abfragenachricht gesendet hat. Wahlweise kann jede Nachrichtenwarteschlange im System verwendet werden.

```
MONMSG   MSGID(CPF0000)   EXEC(                                  +
                          GOTO CMDLBL(FEHLER))
RTVSYSVAL  SYSVAL(QHOUR) RTNVAR(&HOUR)
      :
FEHLER:
SNDMSG MSG('Sie können keine Systemwerte                         +
                          verändern.') TOUSR(*REQUESTER)
SNDMSG MSG('Systemwerte sollten unberechtigterweise              +
                          verändert werden.')  TOMSGQ(*HSTLOG)
ENDPGM
```

2.12.6.6
Seite 2

Zeichenkettenoperatoren

Für die Ausgabe differenzierterer Nachrichten, werden Zeichenkettenoperatoren benötigt. CL verfügt über die Verknüpfungsoperatoren *CAT, *BCAT und *TCAT für alphanumerische Variablen und Konstanten.

***CAT**

verknüpft zwei Zeichenketten einschließlich aller enthaltenen Leerzeichen. Enthält die erste Zeichenkette vier Leerzeichen am Ende, werden die Leerzeichen in das Ergebnis übernommen.

```
DCL     VAR(&T1)    TYPE(*CHAR)   LEN(10)  VALUE(‚Ilse’)
DCL     VAR(&ERG)   TYPE(*CHAR)   LEN(25)
CHGVAR  VAR(&ERG)   VALUE(&T1  *CAT         +
                    ’, wann machst du Pause?’)
```

I	l	s	e							,	w	a	n	n		.	.	.

***TCAT**

verknüpft zwei Zeichenketten ohne nachfolgende Leerzeichen.

```
DCL     VAR(&T1)    TYPE(*CHAR)   LEN(10)  VALUE(‚Ilse’)
DCL     VAR(&ERG)   TYPE(*CHAR)   LEN(25)
CHGVAR  VAR(&ERG)   VALUE(&T1  *TCAT         +
                    ’, wann machst du Pause?’)
```

I	l	s	e	,	w	a	n	n		.	.	.	.	.	.

***BCAT**

verknüpft zwei Zeichenketten so, dass nur ein Leerzeichen zwischen den Zeichenketten steht.

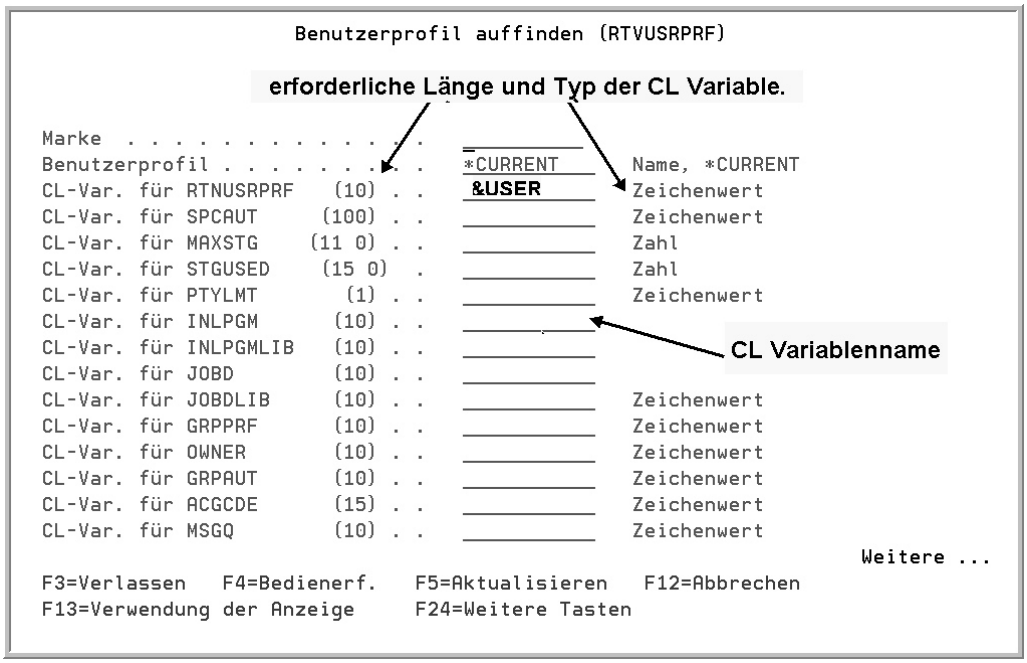
```
DCL     VAR(&T1)    TYPE(*CHAR)   LEN(10)  VALUE(‚Ilse’)
DCL     VAR(&ERG)   TYPE(*CHAR)   LEN(25)
CHGVAR  VAR(&ERG)   VALUE(&T1  *BCAT         +
                    ’, wann machst du Pause?’)
```

I	l	s	e	,		w	a	n	n	.	.	.	.	.	.

Grundwerk 10/2004

2.12.6.7
Seite 2

Wir können jetzt mit Hilfe der Zeichenkettenoperatoren eine qualifizierte Nachricht in das Systemprotokoll schreiben. Zunächst ermitteln wir den Namen des Anwenders, der das Programm aufgerufen hat. Anschließend schreiben wir die Nachricht in das Systemprotokoll. Mit der Anweisung RTVUSRPRF können in einem CL-Programm die Einstellungen von Benutzerprofilen abgefragt und an CL-Variablen übergeben werden – analog zum Befehl RVTSYSVAL. Die Bedienerführungsanzeige des Befehls RTVUSRPRF informiert über Datenart und Länge der einzelnen Attribute des Benutzerprofils.



Die Bedienerführungsanzeige des Befehls RTVUSRPRF

Der erste Parameter ist der Name des abzufragenden Benutzerprofils. Als Standardwert ist *CURRENT vorgegeben. *CURRENT bedeutet, dass das Benutzerprofil verwendet wird, das das Programm aufgerufen hat.

```
PGM
DCL VAR(&HOUR) TYPE(*CHAR) LEN(2)
DCL VAR(&HOUR_D) TYPE(*DEC) LEN(2 0)
DCL VAR(&USR) TYPE(*CHAR) LEN(10)
        :
        :
FEHLER:
/* Nachricht an den Anwender:                                    */
SNDMSG   MSG('Sie können keine Systemwerte verändern.')
TOUSR(*REQUESTER)
/* Qualifizierte Nachricht für das Systemprotokoll              */
RTVUSRPRF USRPRF(*CURRENT)  RTNUSRPRF(&USR)
/* die Variable &USR enthält jetzt den Namen des Anwenders,     +
    der das Programm gestartet hat.                             */
SNDMSG MSG('Anwender' *BCAT  &USR    *BCAT    +  'wollte  den Sys-
temwert QHOUR auf den Wert'   *BCAT          + &HOUR  *BCAT   'än-
dern!')    TOMSGQ(*HSTLOG)
ENDPGM
```

2.12.6.7

Seite 4

Zeichenkettenfunktionen

Die Zeichenkettenfunktion %SST ist eine Built-in-Funktion, die Sie nutzen können, um einen Teil einer alphanumerischen Zeichenkette herauszutrennen. Die Zeichenkette kann in einer alphanumerischen Variablen oder in einem Datenbereich gespeichert sein. Es ist egal, ob Sie die Built-in-Funktion als %SST oder %SUBSTRING kodieren. Sie benötigen in beiden Fällen drei Argumente:

- der Variablenname, aus dem ein Teil extrahiert werden soll,
- die Angabe für die Position, an der die Unterzeichenfolge beginnt,
- die Zahlenangabe für die Länge der Unterzeichenfolge.

```
%SST(&FELD   2    4)
```

In diesem Beispiel ist die Variable &FELD das Referenzfeld. Die Unterzeichenfolge ist 4 Byte lang und beginnt an der zweiten Position. Enthält die Variable den Wert ABCDEFG, dann wird die %SST-Funktion das Ergebnis BCDE liefern.

Sie können für die Start- und Längenangabe der Funktion numerische Variablen einsetzen:

```
%SST(&FELD   &START    &LEN)
```

Da die %SST-Funktion genau einen Zeichenwert als Ergebnis liefert, kann die Built-in-Funktion in allen Anweisungen verwendet werden, die als Parameterwert eine Zeichenfolge erwarten.

```
DCL VAR(&TT)       TYPE(*CHAR)   LEN(2)
DCL VAR(&MM)       TYPE(*CHAR)   LEN(2)
DCL VAR(&JJ)       TYPE(*CHAR)   LEN(2)
DCL VAR(&DATUM)    TYPE(*CHAR)   LEN(6)
DCL VAR(&DATDRK)   TYPE(*CHAR)   LEN(8)
RTVSYSVAL  SYSVAL(QDATE)  RTNVAR(&DATUM)
/* Der Systemwert QDATE enthält das aktuelle Datum im       +
   Format: TTMMJJ      */
CHGVAR   VAR(&TT)      VALUE(%SST(&DATUM   1  2))
CHGVAR   VAR(&MM)      VALUE(%SST(&DATUM   3  2))
CHGVAR   VAR(&JJ)      VALUE(%SST(&DATUM   5  2))
/* Anschließend können Sie das Datum z.B. mit Trennzeichen    +
    aufbereiten   */
CHGVAR   VAR(&DATDRK)  VALUE(&TT *CAT  '.'                    +
*CAT   &MM   *CAT  '.'   *CAT  &JJ)
```

2.12.6.8
Seite 2

Wenn Sie eine zeitlang mit Funktionen gearbeitet haben, werden Sie eventuell anfangen, Ihre Programme zu optimieren, indem Sie z.B. die %SST-Funktion direkt einsetzen.

```
DCL VAR(&DATUM)      TYPE(*CHAR)   LEN(6)
DCL VAR(&DATDRK)     TYPE(*CHAR)   LEN(8)
RTVSYSVAL  SYSVAL(QDATE)  RTNVAR(&DATUM)
CHGVAR   VAR(&DATDRK)  VALUE-
(%SST(&DATUM   1  2)       *CAT   '.'    *CAT                    +
 %SST(&DATUM   3  2)        *CAT  '.'     *CAT                   +
 %SST(&DATUM   5  2))
```

Wir könnten die %SST-Funktion in unserem Programm nutzen, um die Veränderung der Systemzeit zu dokumentieren.

```
PGM
                  :
DCL  VAR(&DATUM)  TYPE(*CHAR)  LEN(6)
                  :
/* Änderung im Systemprotokoll dokumentieren                  */
RTVSYSVAL  SYSVAL(QDATE)  RTNVAR(&DATUM)
SNDMSG MSG('Anwender' *BCAT  &USR  *BCAT        +  'hat den System-
wert QHOUR am'  *BCAT                            +
 %SST(&DATUM   1  2)  *CAT    '.'    *CAT                        +
 %SST(&DATUM   3  2)  *CAT    '.'    *CAT                        +
 %SST(&DATUM   5  2)  *CAT  'auf den Wert'   *BCAT    +     &HOUR
*BCAT   'geändert!')    TOMSGQ(*HSTLOG)
                  :
RETURN
                  :
                  :
ENDPGM
```

Do-Gruppen

Der EXEC-Parameter des MONMSG-Befehls kann dazu verwendet werden, einen bestimmten Befehl auszugeben. Was passiert aber, wenn – wie in unserem Beispiel – mehrere Befehle ausgeführt werden müssen?

```
RTVSYSVAL  SYSVAL(QHOUR)   RTNVAR(&HOUR)
MONMSG MSGID(CPF0001)  EXEC(                                    +
SNDMSG MSG(' Sie können keine Systemwerte  verändern.')         +
TOUSR(*REQUESTER)
SNDMSG MSG('Der Systemwert QHOUR sollte verändert werden.')
TOMSGQ(*HSTLOG)       )
/* So erhalten Sie einen Syntaxfehler, da der EXEC-Parameter    +
nur eine Anweisung erlaubt     */
```

Mit der Anweisung DO können Sie die Funktionsweise des EXEC-Parameters so verändern, dass eine Gruppe von Anweisungen ausgeführt wird. Die auszuführenden Anweisungen müssen zwischen den zusammengehörigen Anweisungen DO und ENDDO eingeschlossen werden.

```
RTVSYSVAL  SYSVAL(QHOUR) RTNVAR(&HOUR)
MONMSG MSGID(CPF0001)  EXEC( DO)
                   SNDMSG MSG('Sie können keine Systemwerte    +
                   verändern.')     TOUSR(*REQUESTER)
                   SNDMSG MSG('Der Systemwert QHOUR sollte     +
                                    verändert  werden.')
TOMSGQ(QHST)
                   ENDDO
/* Alle Anweisungen zwischen der DO und ENDDO                   +
    Anweisung werden ausgeführt, wenn ein Fehler auftritt.     */
```

2.12.6.9
Seite 2

Steueranweisungen

Bisher haben wir nur einfache CL-Programme kennen gelernt, deren Anweisungen sequentiell ausgeführt werden. Die meisten Programme laufen nicht so ab, sondern verzweigen abhängig von Bedingungen. Verzweigungen, die mit IF-Anweisungen programmiert werden, enthalten immer einen logischen Ausdruck, der im ersten Parameter COND der Anweisung IF kodiert wird. Der Parameter COND enthält die Bedingung, die entweder wahr oder falsch sein kann. Ein einfacher logischer Ausdruck besteht aus einem Vergleichsoperator zwischen zwei Ausdrücken, wobei die zu vergleichenden Ausdrücke die gleiche Datenart haben müssen.

Vergleichsoperatoren

*EQ	=	Ist gleich
*GT	>	Größer
*LT	<	Kleiner
*GE	>=	Größer gleich
*LE	<=	Kleiner gleich
*NE	*NOT =	Nicht gleich
*NG	*NOT >	Nicht größer
*NL	*NOT <	Nicht kleiner

IF COND(&ZAHL *EQ 10) THEN(……..

Wenn die numerische Variable &ZAHL den Wert 10 enthält, ist die Bedingung wahr und die Anweisung, die im zweiten Parameter THEN der IF-Anweisung steht, wird ausgeführt. Der Parameter COND kann auch komplexere Ausdrücke enthalten. Bei komplexen logischen Ausdrücken werden zwei oder mehr Bedingungen über Verknüpfungsoperatoren miteinander verbunden.

```
IF COND(*NOT  &NR  *EQ  1)   THEN(…….
```

***NOT**

ändert einen logischen Ausdruck von wahr zu falsch und umgekehrt. Nur wenn die Variable &NR nicht den Wert 1 enthält, wird die Anweisung ausgeführt.

```
IF COND(&NR  *EQ  1  *AND  &FR *EQ  2)   THEN(…….
```

2.12.6.10
Seite 2

***AND**

kombiniert zwei Ausdrücke zu einem Wert. Sind beide Ausdrücke wahr, ist auch der Gesamtausdruck wahr, sonst ist der Gesamtausdruck falsch. In dem Beispiel wird die Anweisung ausgeführt, wenn die Variable &NR den Wert 1 und die Variable &FR den Wert 2 enthält.

```
IF COND(&NR  *EQ  1  *OR  &NR *EQ  2)   THEN(…….
```

***OR**

kombiniert zwei Ausdrücke zu einem Wert. Nur wenn beide Ausdrücke falsch sind, ist auch der Gesamtausdruck falsch. In allen anderen Fällen, wird die Anweisung des Parameters THEN ausgeführt.

Wenn Sie Operatoren – Zeichenkettenoperatoren, arithmetische Operatoren oder logische Operatoren – einsetzen, müssen Sie die Hierarchie/Priorität der Operatoren beachten. Bei den arithmetischen Operatoren haben Multiplikation (*) und Division (/) Vorrang vor Addition (+) und Subtraktion (-), d.h.: Der Ausdruck 10 + 2 * 4 ergibt den Wert 18 und nicht den Wert 48. Die Auflösungshierarchie kann durch die Verwendung von Klammern außer Kraft gesetzt werden. (10 + 2) * 4 ergibt den Wert 48. Die nachfolgende Tabelle enthält die Hierarchie aller Operatoren, die Sie in Ausdrücken verwenden können. Gleichrangige Operatoren werden von links nach rechts aufgelöst.

Priorität	Operator
1	*NOT
2	Multiplikation (*), Division (/)
3	Addition (+), Subtraktion (-)
4	*CAT, *BCAT, *TCAT
5	*GT, *LT, *EQ, *GE, *LE, *NE, *NG, *NL, >, <, =, >=, <=
6	*AND
7	*OR

Wir könnten jetzt eine Bedingung nutzen, um abhängig von dem Monat auf Sommer- bzw. Winterzeit umzustellen.

```
PGM
        :
        :
/* Wenn Monat = Oktober, dann Stunde um 1 reduzieren         */
IF COND(%SST(&DATUM 3  2)  =  '10')  THEN(                   +
CHGVAR VAR(&HOUR_D) VALUE(&HOUR_D - 1))
/* Wenn Monat = März, dann Stunde um 1 erhöhen              */
IF COND(%SST(&DATUM 3  2)  =  '03')  THEN(
CHGVAR   VAR(&HOUR_D)  VALUE(&HOUR_D + 1))
        :
        :
RETURN
        :
ENDPGM
```

Die Anweisung ELSE ist eine optionale Ergänzung der IF-Anweisung. Sie kann genutzt werden, um Anweisungen festzulegen, die ausgeführt werden sollen, wenn die Bedingung der IF-Anweisung falsch ist.

```
/* Wenn Monat = Oktober, dann Stunde um 1 reduzieren          */
IF COND(%SST(&DATUM 3  2)  =  '10')  THEN(
CHGVAR   VAR(&HOUR_D)   VALUE(&HOUR_D - 1))
/* sonst eine Stunde hinzufügen                               */
ELSE  CMD(
                +
CHGVAR   VAR(&HOUR_D)  VALUE(&HOUR_D + 1))
```

Die IF-Anweisung prüft die Bedingung. Trifft sie zu, wird die Anweisung im THEN-Parameter ausgeführt, andernfalls wird der Befehl der ELSE-Anweisung ausgeführt. Beachten Sie bitte, dass beide Parameter – THEN und CMD – nur eine Anweisung verarbeiten können. Wenn Ihre Programmlogik die Ausführung mehrerer Anweisungen erfordert, müssen Sie wieder mit DO-Gruppen arbeiten. Außerdem können Sie Ihre IF-Anweisungen verschachteln, d.h.: Es ist möglich, innerhalb einer IF-Anweisung weitere IF-Anweisungen zu kodieren.

Es bleibt Ihnen überlassen, ob Sie zwei nacheinander folgende IF-Anweisungen kodieren oder die ELSE-Anweisung nutzen. Ich selbst verwende lieber zwei IF-Anweisungen, da dies – meiner Meinung nach – insbesondere bei langen CL-Programmen die Programmlogik deutlich verbessert. Aber wie gesagt, hier spielt Ihre Vorliebe, Erfahrung und natürlich die jeweilige Pro-

2.12.6.10
Seite 4

grammlogik die entscheidende Rolle. Wir sind mit unserem Beispielprogramm jetzt am Ende.

```
PGM
DCL VAR(&HOUR) TYPE(*CHAR) LEN(2)
DCL VAR(&HOUR_D) TYPE(*DEC) LEN(2 0)
DCL VAR(&USR) TYPE(*CHAR) LEN(10)
DCL VAR(&DATUM) TYPE(*CHAR) LEN(6)
MONMSG   MSGID(CPF0000)   EXEC(GOTO CMDLBL(FEHLER))
RTVSYSVAL  SYSVAL(QDATE)  RTNVAR(&DATUM)
RTVSYSVAL  SYSVAL(QHOUR)  RTNVAR(&HOUR)
CHGVAR   VAR(&HOUR_D)     VALUE(&HOUR)
/* Wenn Monat = Oktober, dann Stunde um 1 reduzieren            */
IF COND(%SST(&DATUM 3  2)  =  '10')  THEN(
CHGVAR   VAR(&HOUR_D)     VALUE(&HOUR_D  -  1))
/* Wenn Monat = März, dann Stunde um 1 hinzufügen               */
IF COND(%SST(&DATUM 3  2)  =  '03')  THEN(
CHGVAR   VAR(&HOUR_D)     VALUE(&HOUR_D  +  1))
CHGVAR   VAR(&HOUR)       VALUE(&HOUR_D)
CHGSYSVAL  SYSVAL(QHOUR)  VALUE(&HOUR)
/* Änderung im Systemprotokoll dokumentieren                    */
SNDMSG MSG('Der Anwender' *BCAT  &USER  *BCAT  'hat den          +
Systemwert QHOUR am'  *BCAT   %SST(&DATUM   1  2)   *CAT    '.'  +
*CAT  %SST(&DATUM  3  2)   *CAT    '.'  *CAT    %SST(            +
&DATUM  5  2)   *CAT   'auf den Wert'   *BCAT  &HOUR   *BCAT     +
'geändert!')   TOMSGQ(*HSTLOG)
RETURN
FEHLER:
SNDMSG   MSG('Sie können keine Systemwerte verändern.')
TOUSR(*REQUESTER)
RTVUSRPRF  USRPRF(*CURRENT)   RTNUSRPRF(&USER)
SNDMSG MSG('Der Anwender' *BCAT  &USER  *BCAT      +  'wollte  den
Systemwert QHOUR auf den Wert'   *BCAT      + &HOUR     *BCAT
'ändern!')    TOMSGQ(*HSTLOG)
ENDPGM
```

Das Programm müsste jetzt getestet und anschließend für die Nutzung freigegeben werden, indem das Programm in der Bibliothek EPSCLPGM erstellt wird.

```
                    CL - Programm erstellen (CRTCLPGM)

Auswahl eingeben und Eingabetaste drücken.

Programm . . . . . > WINTER      Name
  Bibliothek . . . . . > EPSCLPGM  Name, *CURLIB
Quellendatei . . . . . QCLSRC     Name
  Bibliothek . . . . . EPSCLTST   Name, *LIBL, *CURLIB
Quellenteildatei . . . . . *PGM   Name, *PGM
Text 'Beschreibung'  . . . . . *SRCMBRTXT

                         Zusätzliche Parameter

Quellenlistenauswahl . . . . . *SOURCE, *NOSOURCE, *SRC...
            + für weitere Werte
Generierungsangaben  . . . . . *NOLIST, *LIST, *NOXREF...
            + für weitere Werte
Benutzerprofil . . . . . *USER   *USER, *OWNER
Befehle protokollieren . . . . . *JOB   *JOB, *YES, *NO
                                                        Weitere ...
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Das CL-Programm in der Bibliothek EPSCLPGM erstellen

Ob Sie nur das Programmobjekt in die öffentliche Bibliothek übertragen oder eventuell auch die Teildatei, ist von Ihrer innerbetrieblichen Organisation abhängig.

Anschließend können Sie das Programm manuell zum jeweiligen Termin mit dem Befehl CALL interaktiv aufrufen oder Sie hinterlegen das Programm als Scheduler-Job – entweder mit dem Befehl SBMJOB oder mit der Anweisung ADDJOBSCDE.

```
                          Job übergeben (SBMJOB)

Auswahl eingeben und Eingabetaste drücken.

Systembibliotheksliste . . . . .  *CURRENT      *CURRENT, *SYSVAL
Aktuelle Bibliothek  . . . . . .  *CURRENT      Name, *CURRENT, *USRPRF...
Anfangsbibliotheksliste  . . . .  *CURRENT      Name, *CURRENT, *JOBD...
            + für weitere Werte
Nachrichtenaufzeichnung:
  Ebene  . . . . . .  *JOBD       0-4, *JOBD
  Bewertung  . . . . . .  *JOBD   0-99, *JOBD
  Text . . . . . .  *JOBD         *JOBD, *MSG, *SECLVL, *NOLIST
CL-Programmbef. protokollieren    *JOBD       *JOBD, *NO, *YES
Antwort auf Anfragenachricht . .  *JOBD       *JOBD, *RQD, *DFT, *SYSRPYL
Jobwarteschlange anhalten  . . .  *JOBD       *JOBD, *NO, *YES
Geplantes Datum  . . . . . .  31.10.04  Datum  *CURRENT, *MONTHSTR...
Geplante Uhrzeit . . . . . .  *CURRENT  Zeit, *CURRENT
Jobdatum . . . . . .  *JOBD     Datum, *JOBD, *SYSVAL
Jobschalter  . . . . . .  *JOBD  Zeichenwert, *JOBD
Anzeige bei WRKSBMJOB erlauben    *YES         *YES, *NO
                                                        Weitere ...
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Das Menü SBMJOB

Wenn Sie Ihr Programm als Batchjob mit einem geplanten Datum und/oder einer geplanten Uhrzeit in das System stellen, finden Sie den Job innerhalb der gewählten Jobwarteschlange bis zum Ausführungszeitpunkt.

```
                        Mit Jobwarteschlange arbeiten

Warteschlange:   QBATCH        Bibl.:   QGPL          Status:   RLS/SBS

Auswahl eingeben und Eingabetaste drücken.
  2=Ändern   3=Anhalten   4=Beenden   5=Arbeiten mit   6=Freigeben

Opt     Job            Benutzer       Nummer     Priorit.     Status
 =      WINTER         LGAD           364778        5          SCD
```

Der Job WINTER in der Warteschlange

Sie können das Programm aber auch mit dem Befehl ADDJOBSCDE dem System übergeben:

```
                  Jobplanungseintrag hinzufügen (ADDJOBSCDE)

Auswahl eingeben und Eingabetaste drücken.

Jobname  . . . . . . . . . . . . > WINTER      Name, *JOBD
Auszuführender Befehl  . . . . . > CALL PGM(WINTER)
_____________________________________________________________________
_____________________________________________________________________
_____________________________________________________________________
_____________________________________________________________________
_____________________________________________________________________
_____________________________________________________________________
Häufigkeit . . . . . . . . . . . > *once       *ONCE, *WEEKLY, *MONTHLY
Geplantes Datum oder . . . . . . > 31.10.04    Datum, *CURRENT, *MONTHSTR...
Geplanter Tag  . . . . . . . . .   *NONE       *NONE, *ALL, *MON, *TUE...
             + für weitere Werte
Geplante Uhrzeit . . . . . . . . > '06:00:00'  Zeit, *CURRENT



                                                                Weitere ...
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Übergabe des Jobs mit dem Befehl ADDJOBSCDE

Der Job kann anschließend jederzeit wieder entfernt oder verändert werden, wenn Sie mit Ihren Planungseinträgen arbeiten (WRKJOBSCDE).

```
                      Mit Jobplanungseinträgen arbeiten              EPOSAS03
                                                          08.10.04  14:50:24


Auswahl eingeben und Eingabetaste drücken.
  2=Ändern   3=Anhalten   4=Entfernen   5=Details anzeigen   6=Freigeben
  8=Mit letzter Übergabe arbeiten       10=Sofort übergeben

                                                          Wieder-   Nächstes
Aus-                        ------Geplant------  Häufig-  herstel-  Übergabe-
wahl  Job         Status    Datum      Uhrzeit   keit       lung      datum
 _    CLNPFRLIBS   SCD      USER DEF   21:00:00  *WEEKLY   *SBMRLS   08.10.04
 _    WINTER       SCD      31.10.04   06:00:00  *ONCE     *SBMRLS   31.10.04
```

Den Job mit WRKJOBSCDE bearbeiten

2.12.6.10

Seite 8

Kontrollanweisungen in Release V5R3

3. Ergänzung 06/2005

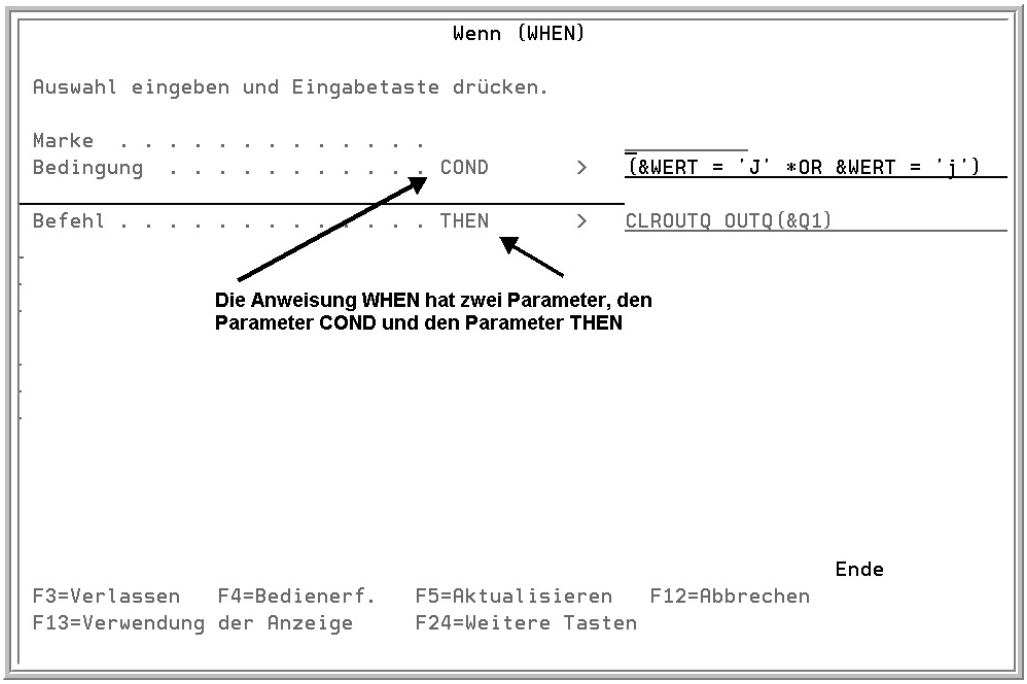
2.12.6.11

Seite 2

Der Befehl SELECT

Mit Release V5R3 wurden einige neue CL-Befehle eingeführt, die die Strukturierung der CL-Programme erheblich vereinfachen. Einer dieser Befehle ist der Befehl SELECT. Mit dem Befehl können Sie auf einfache Art und Weise CASE-Strukturen realisieren. Eine SELECT-Gruppe beginnt immer mit dem Befehl SELECT und endet mit der Anweisung ENDSELECT. Keine dieser Anweisungen hat Parameter. Innerhalb der SELECT-Gruppe wertet der Befehl WHEN einen logischen Ausdruck aus. Ist der logische Ausdruck wahr, wird der im Parameter THEN angegebene Befehl ausgeführt und alle folgenden WHEN-Anweisungen werden weder geprüft noch verarbeitet. Ist der logische Ausdruck dagegen falsch, wird die Steuerung an die nächste WHEN-Anweisung übergeben. Zusätzlich zum Befehl WHEN steht Ihnen der Befehl OTHERWISE zur Verfügung, um die Anweisungen zu hinterlegen, die ausgeführt werden, wenn alle WHEN-Bedingungen nicht zutreffen.

Schauen wir uns zunächst den Befehl WHEN an:



Bedienerführung der Anweisung SELECT WHEN

Der Befehl WHEN

Der Befehl WHEN ist nur innerhalb einer SELECT-ENDSELECT-Befehlsgruppe gültig. Er darf beliebig oft verwendet werden. Mit dem Befehl WHEN kann ein Befehl oder eine Befehlsgruppe angegeben werden, die ausgeführt wird, falls die Bedingung im Befehl „wahr" ist. Sobald eine der Bedingungen zutrifft, werden keine weiteren WHEN-Zweige geprüft. Der Befehl WHEN hat zwei Parameter:

2.12.6.11.1
Seite 2

Der Parameter COND

Der Parameter COND ist ein erforderlicher Parameter. Er beinhaltet den logischen Ausdruck, der ausgewertet wird. Variablen, Konstanten und/oder die integrierten Funktionen %SUBSTR, %SWITCH und %BIN können in dem Ausdruck verwendet werden.

Der Parameter THEN

Im Parameter THEN kann jeder CL-Befehl mit Ausnahme der Befehle ELSE, PGM, ENDPGM, ENDDO, MONMSG, WHEN, OTHERWISE und ENDSELECT kodiert werden.

```
SELECT
WHEN          COND(&OPT *EQ '1')  THEN(DO)
              (Gruppe von CL-Befehlen)
              ENDDO
WHEN          COND(&OPT *EQ '2' *OR &OPT *EQ '3')  THEN(DO)
              (Gruppe von CL-Befehlen)
              ENDDO
ENDSELECT
```

Der Befehl OTHERWISE

Der Befehl OTHERWISE ist nur innerhalb einer SELECT-ENDSELECT-Befehlsgruppe gültig. Er darf nur einmal am Ende der SELECT-Gruppe verwendet werden. Mit dem Befehl OTHERWISE kann ein Befehl oder eine Befehlsgruppe angegeben werden, die ausgeführt wird, falls keine der WHEN-Bedingungen zutrifft.

Der Parameter CMD

Im Parameter CMD kann jeder CL-Befehl mit Ausnahme der Befehle ELSE, PGM, ENDPGM, ENDDO, MONMSG, WHEN, OTHERWISE, ENDSELECT genutzt werden. Ist im Parameter CMD kein Befehl angegeben, wird die Steuerung an den nächsten Befehl nach dem Befehl ENDSELECT übergeben, der diesem Befehl WHEN zugeordnet ist.

Schauen wir uns auch hier wieder ein kleines Beispiel an:


```
CASE:     SELECT
          WHEN  COND(&WERT = 'J' *OR &WERT = 'j') THEN(DO)
                CLROUTQ OUTQ(&Q1)
                SNDMSG  MSG('Inhalt der Nachrichtenwarteschlange +
                wurde gelöscht!') TOUSR(*SYSOPR)
                ENDDO
          WHEN  COND(&WERT = 'N' *or &WERT = 'n') THEN(DO)
                SNDPGMMSG  MSG('Inhalt der Nachrichtenwarteschlange +
                wurde NICHT gelöscht!') TOUSR(*SYSOPR)
                ENDDO
          OTHERWISE  CMD(DO)
                SNDMSG MSG(,Es sind nur die Werte "J" , +
                "j", "N" oder "n" als Antwort gültig. Das Programm +
                wird beendet.') + TOUSR(*REQUESTER)
                ENDDO
          ENDSELECT
RETURN
```

Im ersten WHEN-Zweig wird geprüft, ob die Variable die Werte „J“ oder „j“ enthält. Trifft die Bedingung zu, wird der Inhalt der Ausgabewarteschlange gelöscht, die Nachricht an den Systemoperator gesendet. Anschließend setzt das Programm die Verarbeitung mit dem Befehl RETURN fort. Falls die erste Bedingung falsch ist, d.h. sie trifft nicht zu, wird die nächste WHEN-Bedingung geprüft. Ist die Bedingung „wahr“, werden die dazugehörigen Befehle ausgeführt. In unserem Fall sendet das Programm eine entsprechende Benachrichtigung an den Operator. Auch hier wird das Programm mit der Anweisung RETURN fortgesetzt. Nur wenn keine der Bedingungen zutrifft, kommt der OTHERWISE-Befehl zum Tragen. Die Nachricht wird an den aktuellen Anwender gesendet und anschließend wird der Befehl RETURN ausgeführt.

2.12.6.11.1

Seite 4

Der Befehl DOWHILE

Mit Release V5R3 wurden drei neue Schleifenbefehle eingeführt. Schauen wir uns die unterschiedlichen Syntaxen an einem kleinen Beispielprogramm an:

```
PGM
          DCL &WERT *CHAR 10
GLOBAL:   MONMSG MSGID(CPF0000) EXEC(GOTO CMDLBL(Ende))
START:
          SNDUSRMSG  MSG('Geben Sie den Namen der Ausgabewarte- +
          die bereinigt werden soll oder geben Sie keinen Wert  +
          ein, wennSie das Programm beenden wollen:‘) DFT(' ') +
          MSGRPY(&WERT)
          IF COND(&WERT *EQ ' ') THEN(RETURN)
          CLROUTQ   OUTQ(&WERT)
          GOTO START
ENDE:     ENDPGM
```

Mit Ausnahme des Befehls SNDUSRMSG sind uns alle Anweisungen in dem Programm bekannt. Der Befehl SNDUSRMSG wird zu einem späteren Zeitpunkt noch detailliert besprochen. An dieser Stelle reicht es, wenn wir wissen, dass der Befehl eine Abfragenachricht an den Anwender des Programms sendet. Der Anwender kann den Namen einer Ausgabewarteschlange eintragen oder das Feld leer lassen. Wenn die Variable &Wert gefüllt wurde, versucht das Programm den Inhalt der Ausgabewarteschlange zu löschen, anschließend kann der Anwender einen neuen Wert eingeben. Sie haben Recht, wenn Sie vermuten, dass das Programm im Fehlerfall einfach beendet wird. Dafür sorgt die MONMSG-Anweisung in unserem Programm. Dies ist sicherlich keine gute Lösung. Da wir uns aber mit dem Kodieren von Schleifen beschäftigen wollen, verzichten wir zu diesem Zeitpunkt auf eine gelungene Fehlerroutine. Das Beispielprogramm verwendet die alte GOTO-Syntax, die bis zu Release V5R3 die einzige Möglichkeit war, Schleifen abzubilden. Jetzt stehen uns neue Möglichkeiten zur Verfügung:

2.12.6.11.2
Seite 2

```
PGM
          DCL &WERT *CHAR 10  VALUE(' ')
GLOBAL:   MONMSG MSGID(CPF0000) EXEC(GOTO CMDLBL(Ende))
          SNDUSRMSG  MSG('Geben Sie den Namen der +
          Ausgabewarteschlange ein, die bereinigt werden soll +
          oder geben Sie keinen Wert ein, wenn Sie das Programm +
          beenden wollen:‘) +
          DFT(' ') MSGRPY(&WERT)
          DOWHILE    COND(&WERT *NE '  ')
          CLROUTQ   OUTQ(&WERT)
          SNDUSRMSG  MSG(‘Geben Sie den Namen der Ausgabe-     +
          warteschlange Ausgabewarteschlange ein, die bereinigt +
          werden soll oder geben Sie  keinen Wert ein, wenn Sie +
          das Programm beenden wollen:’) DFT(‘ ‘) MSGRPY(&WERT)
          ENDDO
          RETURN
ENDE:     ENDPGM
```

Der Befehl DOWHILE wertet einen logischen Ausdruck aus und verarbeitet die CL-Befehle, abhängig von der Auswertung des Ausdrucks. Solange der logische Ausdruck wahr ist, werden die Befehle in der DOWHILE-Gruppe verarbeitet. Wird der logische Ausdruck als falsch ausgewertet, wird die Steuerung an den nächsten Befehl übergeben, der auf den zugehörigen ENDDO Befehl folgt. In einer DOWHILE-Schleife wird die Schleifenbedingung bereits vor dem ersten Schleifendurchlauf geprüft. Wenn die Schleifenbedingung nicht zutrifft, also falsch ist, werden die innerhalb der Schleife kodierten Anweisungen nicht ausgeführt. Aus diesem Grund muss die Abfragenachricht in unserem Beispiel bereits das erste Mal außerhalb des Schleifenkörpers kodiert werden, andernfalls wird die Schleife nicht durchlaufen, da die Variable &WERT mit *Blank initialisiert ist. Gibt der Anwender keinen Wert ein, wird das Programm nach der ENDDO-Anweisung fortgesetzt. In unserem Beispiel also mit der Anweisung RETURN. Wir können an dem Beispiel sehen, dass die Schleife mit der Anweisung DOWHILE beginnt und mit der Anweisung ENDDO endet. Die Anweisung DOWHILE hat nur den Parameter COND, der die Schleifenbedingung beinhaltet. Für die möglichen Formulierungen der Schleifenbedingungen gelten die gleichen Regeln wie bei der Bedingungsformulierung der IF-Anweisungen.

Schauen wir uns zwei weitere Beispiele an:

Eine DOWHILE-Befehlsgruppe, die nie verarbeitet wird:

```
DCL   VAR(&LGL)  TYPE(*LGL)  VALUE(,0‘)     /* False */
                    :
DOWHILE   COND(&LGL)
(Gruppe von CL-Befehlen)
ENDDO
```

Die Gruppe der Befehle zwischen DOWHILE und ENDDO wird nicht verarbeitet, da der Anfangswert &LGL falsch ist. Die Steuerung wird an den auf ENDDO folgenden Befehl übergeben.

Endlose DOWHILE-Befehlsgruppe

```
DCL   VAR(&LGL)  TYPE(*LGL)  VALUE(‘1’) /* True */
                    :
DOWHILE &LGL
(Gruppe von CL-Befehlen)
ENDDO
```

Die Gruppe der Befehle zwischen DOWHILE und ENDDO wird so lange verarbeitet, bis der Wert &LGL auf „falsch“ gesetzt wird.

2.12.6.11.2

Seite 4

Der Befehl DOUNTIL


```
PGM
            DCL &WERT *CHAR 10  VALUE(' ')
GLOBAL:     MONMSG MSGID(CPF0000) EXEC(GOTO CMDLBL(ENDE))
            DOUNTIL    COND(&WERT *EQ ' ')
            SNDUSRMSG  MSG('Geben Sie den Namen der +
            Ausgabewarteschlange ein, die bereinigt werden soll   +
            oder geben Sie keinen Wert ein, +
            wenn Sie das Programm beenden wollen:') +
            DFT(' ') MSGRPY(&WERT)
            IF COND(&WERT *EQ ' ') THEN(CLROUTQ   OUTQ(&WERT))
            ENDDO
            RETURN
```

Der Befehl DOUNTIL wertet einen logischen Ausdruck aus und verarbeitet die CL-Befehle, abhängig von der Auswertung des Ausdrucks. Die Anweisung ist eine Programmschleife, deren Bedingung erst nach dem ersten Schleifendurchlauf geprüft wird. Die Bedingungsprüfung wird also erst am Ende des Anweisungsblocks durchgeführt, nachdem die Anweisungen im Schleifenkörper einmal ausgeführt wurden. Die Schleife wird nach dem ersten Durchlauf beendet, wenn die Bedingung zutrifft – also wahr ist, d.h. für unser Programmbeispiel, wenn der Anwender keine Eingabe macht und die Variable &WERT somit den Wert *Blank enthält. Die DOUNTIL-Schleife beginnt mit der Anweisung DOUNTIL und endet mit der Anweisung ENDDO. Die Anweisung DOUNTIL hat wie die Anweisung DOWHILE nur den Parameter COND, der die Schleifenbedingung beinhaltet. Für die möglichen Formulierungen der Schleifenbedingungen gelten die gleichen Regeln wie bei der Bedingungsformulierung der IF-Anweisungen.

Schauen wir uns auch hier zwei weitere Beispiele an:

Eine DOUNTIL-Befehlsgruppe, die mindestens ein Mal verarbeitet wird.

```
DCL   VAR(&INT)  TYPE(*INT)  LEN(2)  VALUE(10)
          :
DOUNTIL  COND(&INT *GT 100)
         (Gruppe von CL-Befehlen)
         CHGVAR   VAR(&INT)  VALUE(&INT + &VAL)
ENDDO
```

3. Ergänzung 06/2005

2.12.6.11.3
Seite 2

Die Gruppe der Befehle zwischen DOUNTIL und ENDDO wird so lange verarbeitet, bis der Wert &INT größer 100 ist. Der Inhalt der DOUNTIL-Gruppe wird mindestens einmal verarbeitet, unabhängig vom Wert &INT am Anfang der Gruppe.

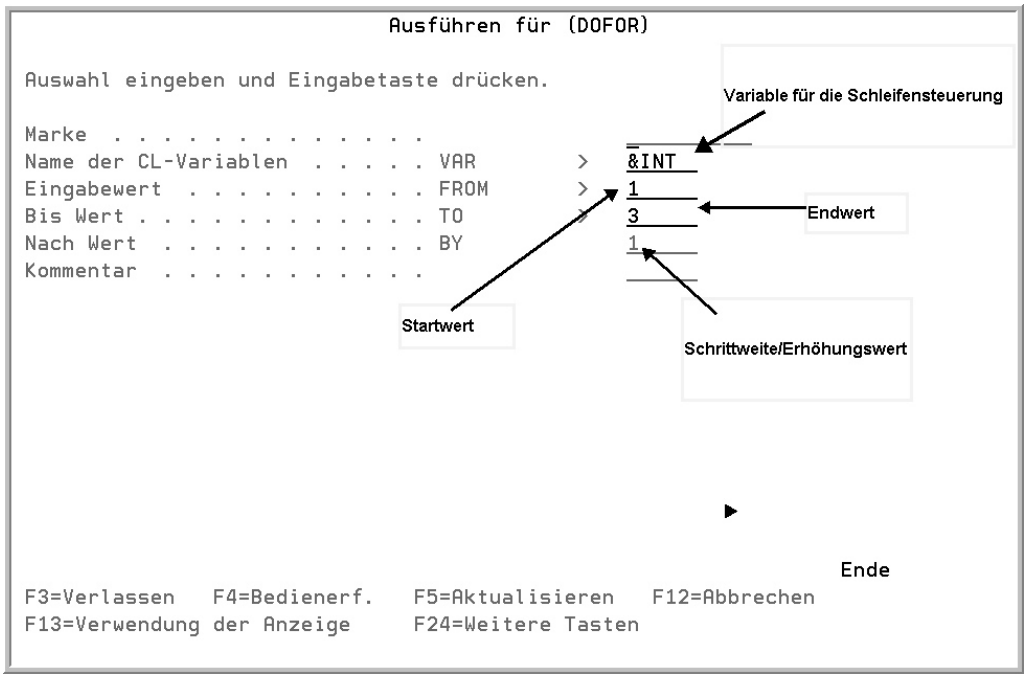
Endlose DOUNTIL-Befehlsgruppe

```
DOUNTIL   COND('0')
          (Gruppe von CL-Befehlen)
ENDDO
```

Die Gruppe der Befehle zwischen DOUNTIL und ENDDO werden verarbeitet.

Der Befehl DOFOR

Die meisten Programmiersprachen kennen die Möglichkeit, Zählschleifen zu programmieren, in denen die Anzahl der Schleifendurchläufe anhand eines Zählers bestimmt wird. Diese Form der Schleife verfügt über eine Kontrollvariable, die einen Startwert enthält, der mit jedem Durchlauf der Schleife solange erhöht oder vermindert wird, bis der angegebene Grenzwert erreicht ist. Diese Zählschleife ist in CL mit dem Befehl DOFOR realisiert.



Bedienerführung des Anweisung DOFOR

DOFOR benötigt drei zwingend erforderliche und einen optionalen Parameter.

Der Parameter VAR

Der Parameter VAR steuert die DOFOR-Schleife. Die Variable muss die Art *INT oder *UINT haben.

Der Parameter FROM

Der Parameter FROM gibt den Startwert der CL-Variablen an, die zur Steuerung der DOFOR-Schleife verwendet wird. Der Wert muss als ganzzahlige Konstante angegeben werden, als CL-Variable der Art *INT oder *UINT oder als ein Ausdruck, aus dem sich ein ganzzahliger Wert ergibt. Der Anfangswert wird der Steuerungsvariablen (Parameter VAR) vor Ausführung des Befehls zugeordnet.

2.12.6.11.4
Seite 2

Der Parameter TO

Der Parameter TO gibt den Endwert der Schleife an. Der Parameter stellt somit den Grenzwert dar, den die Steuerungsvariable im Laufe der Ausführung erreichen darf. Der Wert muss als ganzzahlige Konstante angegeben werden, als CL-Variable der Art *INT oder *UINT oder als ein Ausdruck, aus dem sich ein ganzzahliger Wert ergibt. Der Steuerungsparameter wird mit diesem Endwert vor jedem Schleifendurchlauf verglichen. Wenn der Wert für den Parameter BY negativ ist, endet die Schleife, sobald die Steuerungsvariable kleiner als der TO-Wert ist. Ist der Wert positiv, endet die Schleife, wenn der Steuerungsparameter größer als der TO-Wert ist.

Der Parameter BY

Der Parameter BY legt fest, um welchen Wert sich die Steuerungsvariable bei jedem Schleifendurchlauf verändert. Hier kann sowohl ein positiver als auch ein negativer Wert verwendet werden.

Schauen wir uns auch hier zwei kleine Beispiele an:

```
DCL   VAR(&INT)  TYPE(*INT)  LEN(2)
           :
DOFOR   VAR(&INT)  FROM(1)  TO(10)
           (Gruppe von CL-Befehlen)
ENDDO
```

Die CL-Variable &INT wird zunächst auf den Anfangswert 1 gesetzt und mit Beendigungswert 10 für die Schleife verglichen. Solange der Wert kleiner 10 ist, werden die Anweisungen ausgeführt. Nach jedem Schleifendurchlauf wird die Variable &INT um den Wert 1 erhöht. Der Wert 1 ist der Standardwert für den Parameter BY. Nach der zehnten Schleifenwiederholung nimmt &INT den Wert 11 an, und die Steuerung wird an den Befehl übergeben, der auf die Anweisung ENDDO folgt.

```
DCL   VAR(&INT)  TYPE(*INT)  LEN(2)
DCL   VAR(&START)  TYPE(*INT)  LEN(2)
DCL   VAR(&END)  TYPE(*INT)  LEN(2)
                :
CHGVAR   VAR(&START)  VALUE(100)
CHGVAR   VAR(&END)  VALUE(0)
DOFOR   VAR(&INT)  FROM(&START)  TO(&END)  BY(-5)
        (Gruppe von CL-Befehlen)
ENDDO
```

In diesem Programmbeispiel wird die Schleife 21 Mal verarbeitet. Die CL-Variable &INT wird zunächst auf den Anfangswert 100 gesetzt. Der Endwert ist der Wert 0. Da der Erhöhungswert negativ ist, in diesem Beispiel der Wert -5, wird die Schleife verarbeitet, bis &INT kleiner als 0 ist. Nach jeder Schleifenwiederholung wird &INT um den Wert -5 erhöht und mit dem Wert für TO verglichen. Nach der einundzwanzigsten Schleifenwiederholung nimmt &INT den Wert -5 an, und die Steuerung wird an den Befehl übergeben, der auf die Anweisung ENDDO folgt.

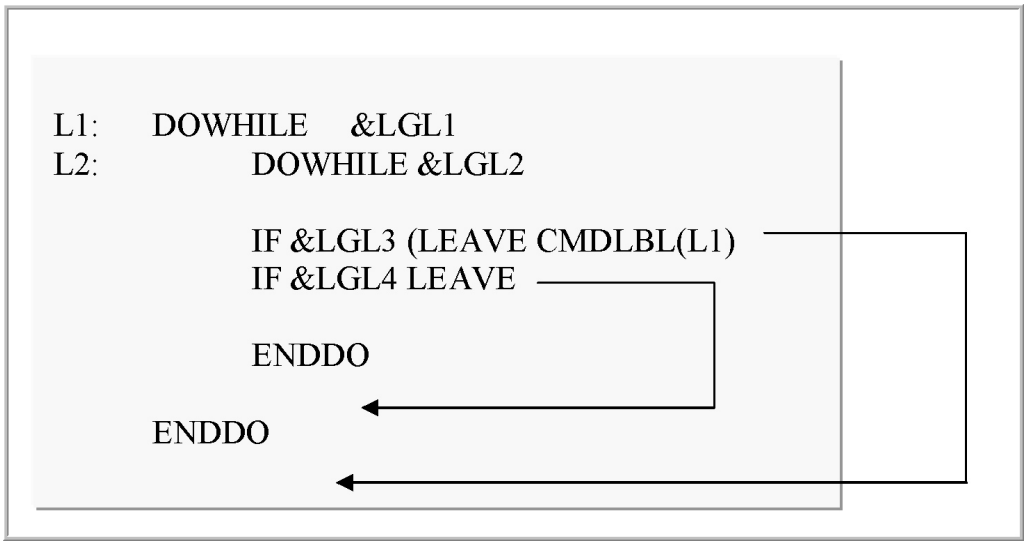
2.12.6.11.4

Seite 4

Die Befehle LEAVE und ITERATE

Die Befehle LEAVE und ITERATE erlauben mit DOUNTIL, DOWHILE oder DOFOR kodierte Schleifen unabhängig von der Schleifenbedingung zu beenden.

Die folgende Befehlsfolge soll die Wirkung des Befehls LEAVE verdeutlichen:



Wirkungsweise der LEAVE-Anweisung

Die Anweisung LEAVE verlässt ohne Kodierung des Parameters CMDLBL stets die aktuelle Schleife. In unserem Beispiel wird die innere Schleife verlassen, wenn &LGL4 wahr ist. Wenn dagegen über den Parameter CMDLBL eine Sprungmarke zugewiesen wird, wird die Schleife beendet, die dieser Sprungmarke zugewiesen ist. Dies ist in unserem Beispiel der Fall, wenn &LGL3 wahr ist.

Im Gegensatz zum Befehl LEAVE unterbricht der Befehl ITERATE lediglich die Verarbeitung und übergibt die Steuerung an die entsprechende Schleifenbedingung zur erneuten Prüfung. Abhängig von der Auswertung wird die Verarbeitung wieder aufgenommen oder beendet.

Schauen wir uns auch hier ein Beispiel an:

2.12.6.11.5
Seite 2

```
DCL   VAR(&INT)  TYPE(*INT)  LEN(2)
DCL   VAR(&NAME)  TYPE(*CHAR)  LEN(10)
              :
DOUNTIL  COND(&INT *GT 100)
              :
              IF   COND(&NAME *EQ *NONE)  THEN(ITERATE)
              CHGVAR   VAR(&INT)  VALUE(&INT + 1)
              :
ENDDO /* ITERATE übergibt die Steuerung bis hierher */
```

Wenn die IF-Bedingung wahr ist, wird der Befehl ITERATE ausgeführt. Die Steuerung wird zunächst an den Befehl ENDDO übergeben. Anschließend wird die im zugeordneten DOUNTIL-Befehl angegebene Bedingung ausgewertet. Wenn der Wert der Variablen &INT kleiner oder gleich 100 ist, wird die Schleife erneut verarbeitet. Ist der Wert der Variablen &INT größer 100 höher, wird die Schleife beendet.

Neue Befehle und Datentypen im Release V5R4

Lange Zeit glaubte man, dass es keine Erweiterungen in der Steuersprache CL mehr geben würde. Doch seit Release V5R3 wurde CL kontinuierlich erweitert und erhielt mächtige Features, die zeigen, dass die Steuersprache CL keineswegs überflüssig wird. Mit Release V5R3 wurden zunächst zahlreiche neue Steueranweisungen implementiert. Lesen Sie hierzu das Kapitel 2.12.6.11. Dies waren sicherlich die weitest reichenden Erweiterungen der Steuersprache seit Einführung der AS/400. Dieser Trend hat sich im Release V5R4 fortgesetzt. So können wir seit Release V5R4 im CL-Programm Subroutinen verwenden, einfache Datenstrukturen aufbauen und Pointer definieren. Schauen wir uns die neuen Features des Release V5R4 einmal an.

2.12.6.12

Seite 2

Subroutinen verwenden V5R4

Erinnern Sie sich noch an unser CL-Programm, das wir gemeinsam entwickelt haben, um Spoolfiles zu löschen? Es gab keine Möglichkeit das Programm zu strukturieren – sieht man einmal von den verschiedenen Sprungmarken ab. In einem komplexen CL-Programm müssen Code-Blöcke entweder mehrfach codiert werden oder separate Module bzw. Programme gebildet werden. Mit Release V5R4M0 wurde dieses Manko durch die Einführung von Subroutinen beseitigt.

Wenn Sie in einer HLL-Sprache programmieren, ist die Verwendung von Subroutinen nichts Neues für Sie. RPG und Co. bieten seit langer Zeit die Möglichkeit, Unterroutinen zu schreiben.

Es gibt Gemeinsamkeiten zwischen Subroutinen, wie sie in HLL-Programmen verwendet werden, und Subroutinen, die in CL-Programmen verwendet werden.

- In den Unterroutinen können nur globale Variablen verarbeitet werden, d. h. es können keine lokalen Variablen definiert oder verwendet werden.
- CL-Subroutinen können keine Eingangsparameter empfangen.
- Innerhalb einer CL-Subroutine können keine weiteren Subroutinen definiert werden.
- Es ist möglich, aus Subroutinen weitere Subroutinen aufzurufen.

Aber es gibt auch einige Unterschiede:

- Aus RPG-Subroutinen kann man mittels einer GOTO- oder CAB-Anweisung die Unterroutine verlassen. Ich weiß, dies ist kein guter Programmierstil, aber es geht! In CL-Subroutinen ist das nicht erlaubt. Eine CL-Subroutine kann nur durch das Erreichen des Endes der Subroutine oder mit dem Befehl RTNSUBR (Return Subroutine) verlassen werden.
- In CL-Subroutinen können Rückgabe-Werte definiert und ausgegeben werden.
- CL-Subroutinen können rekursiv aufgerufen werden, was im Gegensatz dazu nur bei RPG-Prozeduren, nicht jedoch bei Unterroutinen im RPG-Programm zulässig ist.

2.12.6.12.1
Seite 2

Schauen wir uns jetzt ein Beispiel an. Ich verwende dafür wieder das gemeinsam entwickelte Programm. Zunächst möchte ich die vorbereitenden Dateiüberschreibungen in eine eigene Subroutine verlagern:

```
PGM   PARM( &LIB &OUTQ &USRDAT)
:
:

DCLF     FILE(OUT)

         CHKOBJ     OBJ(QTEMP/OUT) OBJTYPE(*FILE)
         MONMSG CPF9801 EXEC(CRTPF FILE(QTEMP/OUT) RCDLEN(132))

         OVRPRTF  FILE(QPRTSPLQ) HOLD(*YES)

         WRKOUTQ    OUTQ(&LIB/&OUTQ) OUTPUT(*PRINT)

         CPYSPLF    FILE(QPRTSPLQ) TOFILE(QTEMP/OUT) SPLNBR(*LAST)+
        CRTDATE(*LAST) TOMBR(*FIRST) MBROPT(*REPLACE)

          DLTSPLF    FILE(QPRTSPLQ) SPLNBR(*LAST) CRTDATE(*LAST)

         OVRDBF     FILE(OUT) TOFILE(QTEMP/OUT) POSITION(*RRN 3)

 LOOP:   RCVF
         MONMSG MSGID(CPF0864) EXEC(DO)
                                    DLTOVR (QPRTSPLQ  OUT)
                                    RETURN
                                    ENDDO
              CHGVAR VAR(&FILDAT) VALUE(    %SST(&OUT 108 2) *CAT +
                                            %SST(&OUT 105 2) *CAT +
                                            %SST(&OUT 102 2))
              IF COND(&FILDAT < &USRDAT) THEN(DO)
                     CHGVAR VAR(&USER) VALUE(%SST(&OUT 13 10))
                     CHGVAR VAR(&JOBNR) VALUE(%SST(&OUT 95 6))
                     CHGVAR VAR(&JOB) VALUE(%SST(&OUT 84 10))
                     CHGVAR VAR(&SPLDAT) VALUE(%SST(&OUT 2 10))
                     CHGVAR VAR(&NR) VALUE(%SST(&OUT 75 4))
                     MONMSG CPF0000
                     DLTSPLF  FILE(&SPLDAT) JOB(&JOBNR/&USER/&JOB)
SPLNBR(&NR)
                     MONMSG CPF0000
                ENDDO

         GOTO LOOP
ENDE: ENDPGM
```

Zusätzlich liefere ich einen entsprechenden Returncode (=Variable &ERROR), der im Hauptprogramm eine Fehlerauswertung zulässt.

```
/***********************SUBROUTINEN*******************************/
        SUBR  CHECK_SUB
           CHKOBJ   OBJ(QTEMP/OUT) OBJTYPE(*FILE)
           MONMSG   CPF9801 EXEC(CRTPF FILE(QTEMP/OUT) RCDLEN(132))
           OVRPRTF  FILE(QPRTSPLQ) HOLD(*YES)
        ENSSUBR

SUBR    COPY_SUB
           WRKOUTQ  OUTQ(&LIB/&OUTQ) OUTPUT(*PRINT)
           CPYSPLF  FILE(QPRTSPLQ) TOFILE(QTEMP/OUT) SPLNBR(*LAST) +
                    CRTDATE(*LAST) TOMBR(*FIRST) MBROPT(*REPLACE)
           DLTSPLF  FILE(QPRTSPLQ) SPLNBR(*LAST) CRTDATE(*LAST)
           OVRDBF   FILE(OUT) TOFILE(QTEMP/OUT) POSITION(*RRN 3)
        ENSSUBR
ENDE: ENDPGM
```

Eine Subroutine beginnt immer mit dem Befehl SUBR (Subroutine) und endet mit dem Befehl ENDSUBR (Ende Subroutine). Zwischen diesen beiden Befehlen wird der auszuführende Quell-Code der Subroutine eingebettet. Der Befehl SUBR hat nur einen Parameter, den Namen der Subroutine. Der Name einer Subroutine muss den bekannten Namenskonventionen entsprechen und darf daher zehn Zeichen nicht überschreiten. Zu Beginn der Subroutine könnte eine Sprungmarke (Label) gesetzt werden. Diese Sprungmarke kann allerdings nur innerhalb der Subroutine verwendet werden. D. h. Sie können das Label nicht außerhalb der Subroutine verwenden. Wer jetzt glaubt, durch die Sprungmarke eine rekursive Ausführung der Subroutine zu codieren, irrt sich! Ein entsprechender GOTO-Befehl bewirkt keinen rekursiven Aufruf, sondern führt die Subroutine ab dem ersten Statement in einer Schleife aus. Der Befehl ENDSUBR hat einen optionalen Parameter (RTNVAL), über den ein Rückgabe-Wert ausgegeben werden kann. Wird der Rückgabe-Wert nicht angegeben, wird 0 (=Zero) zurückgegeben. Als Rückgabe-Werte sind nur ganze Zahlen mit positivem oder negativem Vorzeichen sowie Variablen mit dem Datentypen *INT und einer Länge von 4 Byte zulässig.

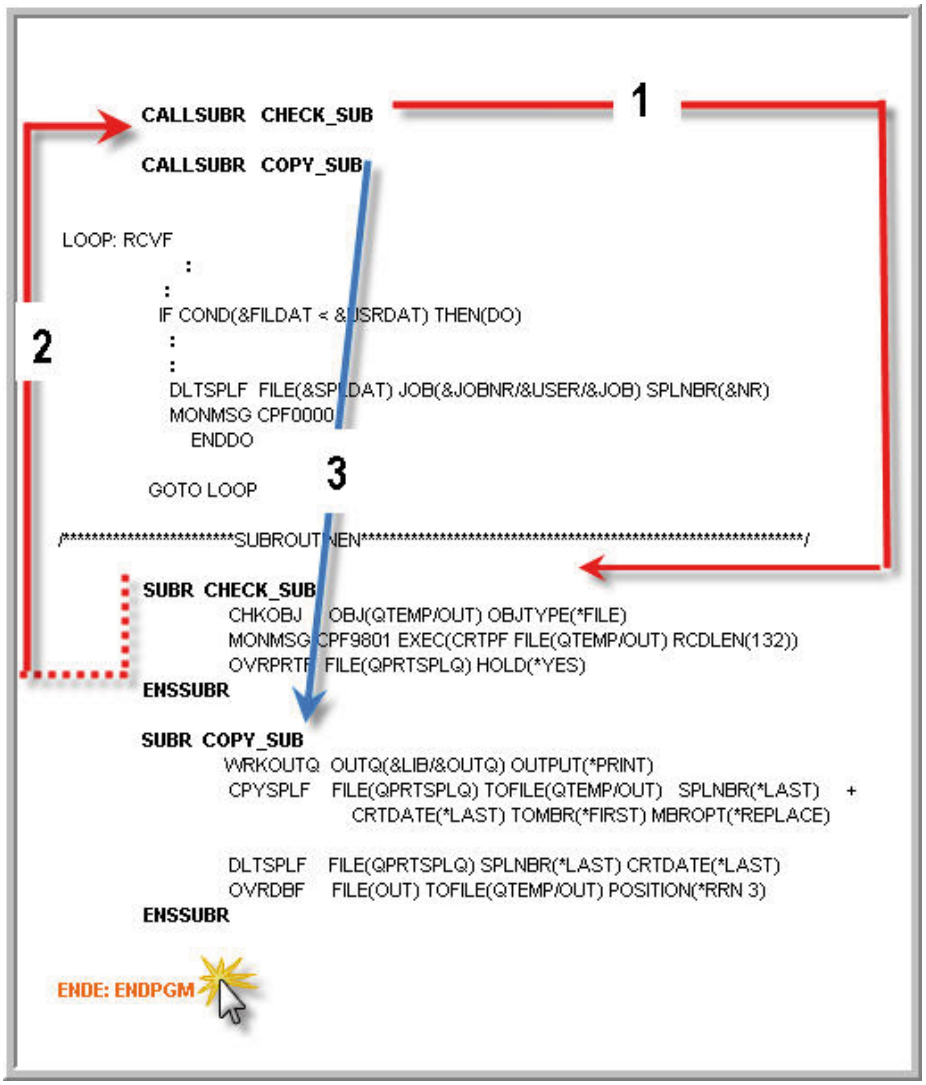
Wie in meinem Beispiel gezeigt, können weitere Subroutinen kodiert werden, indem einfach eine weitere SUBR-Anweisung verwendet wird. Beachten müssen Sie lediglich, dass die ENDPGM-Anweisung – trotz der Subroutinen – die letzte Anweisung im Programm bleibt.

2.12.6.12.1
Seite 4

Im folgenden Quellcode können Sie sehen, wie die Subroutinen aufgerufen werden. Ich verwende zu Beginn meines CL-Programms jetzt den Befehl CALLSUBR, um zuerst die Unterroutine CHECK_SUB und anschließend die Routine COPY_SUB aufzurufen.

```
DCLF FILE(OUT)

/************************Hauptprogramm*************************/
CALLSUBR CHECK_SUB

CALLSUBR COPY_SUB

LOOP: RCVF
```

Die folgende Grafik zeigt, dass die Unterroutine automatisch verlassen wird, wenn die Anweisung ENDSUBR erreicht ist. Das Programm kehrt zum Aufrufpunkt zurück, um den nächsten CL-Befehl auszuführen.



Die Programmlogik

Sie sehen in der Abbildung, dass Subroutinen automatisch verlassen werden, sobald das Ende der Unterroutine erreicht ist. In komplexen Subroutinen könnte es aber sinnvoll sein, die Routine vorzeitig zu verlassen und nicht bis zum Ende auszuführen. Dies ist über den CL-Befehl RTNSUBR möglich. Nützlich in diesen Situationen kann es sein, einen Returnwert zurückzuliefern. Sowohl die Anweisung ENDSUBR als auch die Anweisung RTNSUBR besitzen diese Option. Der CL-Befehl RTNSUBR sowie der CL-Befehl ENDSUBR besitzen hierfür den optionalen Parameter (RTNVAL).

Um zu verstehen, was gemeint ist, habe ich mein Programm um eine Prüfung ergänzt. Zu Beginn der CHECK-Routine überprüfe ich die Benutzereingaben. Ist die Bibliothek oder die Ausgabewarteschlange nicht vorhanden, verlasse ich die Unterroutine umgehend über den Befehl RTNSUBR, da eine weitere Bearbeitung in diesem Fall sinnlos wäre.

```
SUBR CHECK_SUB

CHKOBJ OBJ(&LIB/&OUTQ) OBJTYPE(*OUTQ) AUT(*CHANGE)
       MONMSG MSGID(CPF9801) EXEC(DO)
       CHGVAR &ERROR VALUE('1')
       RTNSUBR RTNVAL(&ERROR)
ENDDO
MONMSG MSGID(CPF9810) EXEC(DO)
       CHGVAR &ERROR VALUE('2')
       RTNSUBR RTNVAL(&ERROR)
ENDDO

CHKOBJ OBJ(QTEMP/OUT) OBJTYPE(*FILE)
MONMSG CPF9801 EXEC(CRTPF FILE(QTEMP/OUT) RCDLEN(132))
OVRPRTF FILE(QPRTSPLQ) HOLD(*YES)

ENDSUBR RTNVAL(&ERROR)
```

2.12.6.12.1
Seite 6

Zusätzlich liefere ich einen entsprechenden Returncode (=Variable &ERROR), der im Hauptprogramm eine Fehlerauswertung zulässt.

```
CALLSUBR CHECK_SUB

IF COND(&ERROR > '0') THEN(DO)

        SELECT
        WHEN   COND(&ERROR = '2') THEN(SNDPGMMSG +
               MSG('Bibliothek nicht gefunden!'))
        WHEN   COND(&ERROR = '1') THEN(SNDPGMMSG +
               MSG('OUTQ nicht gefunden!'))
        ENDSELECT

        GOTO ENDE
        ENDDO

CALLSUBR COPY_SUB
```

Man kann dieses Problem sicherlich auch anders lösen, aber es sollte die Funktionsweise des Befehls RTNSUBR und die Möglichkeiten eines Returnwertes ausreichend erläutern. Und natürlich haben Sie Recht, wenn Sie behaupten, dass man die Variable &Error nicht zurückgeben muss, da ihr Inhalt als globale Variable ohnehin überall im Programm bekannt ist.

Aber man kann diese Routinen auch noch anders kodieren. Dieses Mal rufe ich die Unterroutine mit der Option RTNVAL auf und verwende die Variable &ERG als Returnvariable. Als Rückgabe-Wert dürfen Sie im Übrigen in der Anweisung CALLSUBR lediglich eine Integer-Variable mit einer Länge von 4 Byte verwenden.

```
CALLSUBR CHECK_SUB RTNVAL(&ERG)

IF COND(&ERG > 0) THEN(DO)
```

In meiner Unterroutine verwende ich aber weiterhin die Variable &ERROR.

```
SUBR CHECK_SUB

CHKOBJ  OBJ(&LIB/&OUTQ) OBJTYPE(*OUTQ) AUT(*CHANGE)
        MONMSG MSGID(CPF9801) EXEC(DO)
        CHGVAR &ERROR VALUE(,1‘)
        RTNSUBR RTNVAL(&ERROR)
ENDDO
```

Wird die Unterroutine jetzt beendet, wird der Inhalt der Variablen &ERROR in die Variable &ERG übertragen.

Jeder Aufruf einer Unterroutine erzeugt einen Eintrag im Stack. Der Eintrag wird wieder entfernt, sobald die Subroutine entweder über den CL-Befehl RTNSUBR oder den CL-Befehl ENDSUBR beendet wird. Nun kann man aber CL-Unterroutinen auch rekursiv aufrufen. In diesen Fällen erfolgt bei jedem Aufruf ein neuer Eintrag im Stack, ohne dass der aktuelle beendet wird. Wie viele Rekursivaufrufe zulässig sind, kann über den Befehl DCLPRCOPT festgelegt werden. Er bestimmt somit die Rekursionstiefe im jeweiligen Programm. Der Unterlassungswert für die Verschachtelungstiefe liegt bei 99. Er kann jedoch auf einen beliebigen Wert zwischen 20 und 9.999 gesetzt werden. Da es sich bei dem Befehl um eine Deklaration handelt, darf der Befehl nur ein einziges Mal in der Quelle angegeben werden und muss wie alle anderen DECLARE-Befehle zu Beginn des Programms kodiert werden.

```
PGM
DCL VAR(&Z) TYPE(*INT) LEN(4)
DCLPRCOPT SUBRSTACK(20)
CALLSUBR SUBR(S_REKURSIV)     /*                        1. Aufruf */
        :
        :
/* SUBROUTINE: MIT REKURSIVEM AUFRUF                              */
SUBR    SUBR(S_REKURSIV)
        CHGVAR VAR(&Z) VALUE(&Z + 1)
        IF COND(&Z *GE 100) THEN(RTNSUBR)

        CALLSUBR SUBR(S_REKURSIV) /*            rekursiver Aufruf */
ENDSUBR
```

2.12.6.12.1
Seite 8

In dem kleinen Beispielprogramm wird die Rekursionstiefe zunächst auf 20 festgelegt. Das Programm kann die Unterroutine demnach 20 Mal rekursiv aufrufen, bevor das Programm mit folgender Meldung abstürzt:

```
(C D I R) CPF0822 von REKURSION bei 1800 empfangen.
```

Die Fehlermeldung CPF0822

Wird die zulässige Verschachtelungstiefe überschritten, wird die Fehlermeldung CPF0822 ausgegeben. Sie können den Absturz mit einer MONMSG-Anweisung im Übrigen nicht direkt abfangen:

```
1800-              CALLSUBR   SUBR(S_REKURSIV)
1900-              MONMSG     MSGID(CPF0822) EXEC(RTNSUBR)
CPD0896 30  Nachrichten können bei diesem Befehl nicht überwacht werden.
```

Fehlerüberwachung der Anweisung CALLSUBR

Die Umwandlung des CL-Programms schlägt fehl, sobald Sie die MONMSG Anweisung lokal auf Befehlsebene der Anweisung CALLSUBR kodieren. Alternativ müssen Sie aber eine globale Fehlerroutine verwenden und anschließend ggf. die Fehlermeldung der Programmnachrichtenwarteschlange entnehmen. Sie wissen nicht was gemeint ist?

Zunächst einmal benötigen Sie für diese Lösung vier Variablen:

für den Namen der Nachrichtendatei, der zugehörigen Bibliothek, der Nachrichten-ID und den Nachrichtenersatzwerten. Diese Informationen sendet das Betriebssystem an die Programmnachrichtenwarteschlange des Programms, sobald ein CL-Befehl fehlschlägt und Sie im CL-Programm die Ausnahmesituationen behandeln.

```
PGM
DCL &MSGDTA    *CHAR 150
DCL &ID        *CHAR 7
DCL &MSGF      *CHAR 10
DCL &MSGLIB    *CHAR 10
```

Damit Ihr Programm im Fehlerfall nicht abstürzt, kodieren Sie im Programm zusätzlich eine globale MONMSG-Anweisung:

```
MONMSG MSGID(CPF0000) EXEC(GOTO FEHLER)
```

Wichtig hierbei ist, dass die MONMSG-Anweisung die erste Anweisung im Programm ist; also unmittelbar hinter den Deklarationen kodiert wird.

Die eigentliche Routine, um die Nachrichten des Betriebssystems zu lesen, sieht folgendermaßen aus:

```
FEHLER: RCVMSG MSGQ(*PGMQ) MSGTYPE(*LAST) +
          MSGDTA(&MSGDTA) MSGID(&ID) MSGF(&MSGF) MSGFLIB(&MSGLIB)

          SNDPGMMSG MSGID(&ID) MSGF(&MSGLIB/&MSGF) MSGDTA(&MSGDTA)
```

Die RCVMSG-Anweisung liest hierbei die letzte Meldung des Betriebssystems aus der Programmnachrichtenwarteschlange und die dann folgende SNDPGMMSG-Anweisung sendet diese Nachricht zu einer beliebigen Nachrichtenwarteschlange. Grundsätzlich könnten Sie die Sprungmarke mit den entsprechenden Befehlen überall in Ihrem Programm kodieren. In der Regel finden Sie diese Anweisungen allerdings unmittelbar vor der ENDPGM-Anweisung, da Sie das Programm meistens nach dem Auftreten einer Ausnahmesituation ohnehin verlassen müssen. Wenn Sie allerdings – so wie ich es hier habe – Unterroutinen in Ihrem CL-Programm nutzen, muss die Fehlerroutine VOR den Subroutinen programmiert werden, da nach der letzten ENDSUBR-Anweisung nur noch der ENDPGM-Befehl folgen darf.

Sobald das Programm mit der neuen Fehlerroutine kompiliert ist, kann ich es erneut ausführen. Das Ergebnis sieht bei meinem Programmbeispiel anschließend folgendermaßen aus:

```
Parameter oder Befehl
===>
F3=Verlassen     F4=Bedienerführung   F5=Aktualisieren     F6
F9=Auffinden     F10=Befehlseingabe   F23=Weitere Angaben  F2
Subroutinenstapelüberlauf bei Anweisung 003200.
```

Globale Fehlerroutinen nutzen

Weiterführende Informationen zum Thema „Nachrichten“ finden Sie im Kapitel 2.12.7.1.

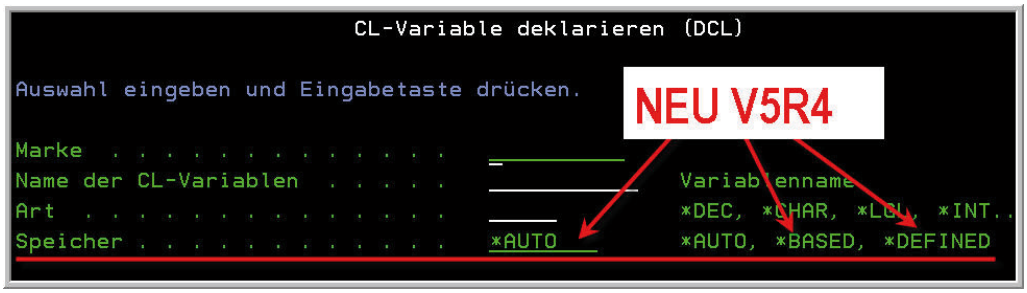
2.12.6.12.1

Seite 10

Datenstrukturen im CL-Programm V5R4

Vor Release V5R3M0 konnten in CL nur alphanumerische und numerische Variablen genutzt werden. Release V5R3 wurde dann um ganzzahlige, numerische Werte mit dem Datentyp Integer (=*INT) und Unsigned Integer (*UINT) mit 2 und 4 Byte Länge ergänzt. In Release V5R4 folgen nun weitere Ergänzungen. So ist es jetzt möglich, einfache Datenstrukturen zu definieren. Datenstrukturen erleichtern den Einsatz von System-APIs (=Application Programming Interface) erheblich, da die meisten Parameter der APIs über Datenstrukturen definiert werden. Bisher konnte man nur die CL-Parameter als alphanumerische Variablen übergeben und anschließend über Substring-Anweisungen in ihre Einzelteile zerlegen.

Um Datenstrukturen zu definieren, wurde die DCL-Anweisung im Release V5R4 erweitert:



Bedienerführung der Anweisung DCL (=Declare)

Haben Sie die neue Option „Speicher“ bereits entdeckt, als Sie Ihr letztes CL-Programm geschrieben haben? Über diese Option wird gesteuert, ob ein einzelnes Feld (*AUTO), ein Datenstruktur-Unterfeld (*DEFINED) oder eine Zeigervariable (*BASED) definiert werden soll. Der Unterlassungs-Wert für die Option STG ist *AUTO.

Zusätzlich zur Option STG wurden die maximale Länge von alphanumerischen Feldern sowie die Länge der Valuewerte geändert. Während in Release V5R3M0 ein alphanumerisches Feld nur bis zu einer Länge von 9.999 Byte definiert werden konnte, können in Release V5R4M0 alphanumerische Felder bis zu einer Länge von 32.767 Byte erstellt werden. Diese Grenze gilt im Übrigen auch für Datenstrukturen. Aber Vorsicht, in einem RPG-Programm dürfen alphanumerische Felder oder benannte Datenstrukturen 65.535 Byte lang sein. Dies muss beim Aufruf von RPG-Programmen oder -Prozeduren aus CL berücksichtigt werden. Auch die Länge der Anfangswerte (Parameter VALUE) wurde von 3.000 auf 5.000 Byte erhöht. Somit kann aber nach wie vor eine mit Maximal-Länge definierte alphanumerische Variable nicht vollständig mit einem String initialisiert werden.

2.12.6.12.2
Seite 2

Kommen wir zurück zur Definition der neuen Datenstrukturen: Bei der Definition von Datenstrukturen wird zunächst eine Variable mit einer vorgegebenen Länge definiert. Die Datenstruktur-Unterfelder überlagern anschließend diese Variable. Wofür benötigen wir überhaupt Datenstrukturen? Nun, es geht auch ohne. Viele Jahre existierten im CL schließlich keine Strukturen, aber es gibt Situationen in denen Datenstrukturen das Leben eines Programmierers erleichtern können. Denken Sie einmal an unsere qualifizierten Objektnamen: In CL-Befehlen werden Bibliothek und Objektname als einzelne Felder angegeben. Im Gegensatz dazu muss der qualifizierte Objekt-Name beim Aufruf von System-APIs in der Regel als 20-stelliger String übergeben werden und auch eigene Commands können qualifizierte Objektnamen als Listen übergeben. Bisher benötigten wir daher zusätzliche Verkettungsbefehle oder entsprechende %SST-Funktionen, um das passende Format bereitzustellen – jetzt können wir dieses Problem mit Datenstrukturen lösen. Schauen Sie sich die folgende Grafik an:

Datenstrukturen definieren

Das Beispiel zeigt eine Datenstruktur für einen qualifizierten Objekt-Namen. In den ersten zehn Stellen wird der Objektname hinterlegt; es folgt der Bibliotheksname.

Definiert wird zunächst eine Variable für den qualifizierten Objektnamen. Dies ist die Variable &QUAL_NAME. Sie dient anschließend als Referenzvariable für die Unterfelder. Die Datenstrukturunterfelder – im Beispiel die Felder &OBJ und &LIB – werden mit der Option STG *DEFINED angegeben. Die zu überlagernde Variable, also die Variable &QUAL_NAME, muss in der Option DEFVAR als Referenzvariable hinterlegt werden. Optional kann bei der Option DEFVAR eine Startposition mitgegeben werden, ab welcher die Referenz-Variable überlagert werden soll. Fehlt die Startposition, wird ab Stelle eins überlagert. Für Datenstruktur-Unterfelder können alle bekannten Datentypen verwendet werden, also CHAR, DEC, LGL, INT, UINT und PTR.

Ein anderes Beispiel sind Fehlercodestrukturen, die viele APIs im Aufruf benötigen. Ein entsprechendes Format solcher Error-Strukturen sehen Sie in der folgenden Tabelle:

Datenstruktur Format ERRC0100				
Offset				
DEC	HEX	Verwendung	Typ	Beschreibung
0	0	INPUT	BINARY(4)	Länge der Datenstruktur (&LEN)
4	4	OUTPUT	BINARY(4)	Länge der Rückgabe-Informationen (&RTN_LEN)
8	8	OUTPUT	CHAR(7)	Nachrichten-ID (&MSGID)
15	F	OUTPUT	CHAR(1)	Reserviert (&RESERVED)
16	10	OUTPUT	CHAR(240)	Nachrichtenersatzwerte (&MSGDTA)

Die Definition dieser Struktur würde im CL-Programm jetzt folgendermaßen aussehen:

```
/*FEHLERSTRUKTUR */
DCL VAR(&ERRC0100) STG(*AUTO) TYPE(*CHAR) LEN(256)

DCL VAR(&LEN) STG(*DEFINED) TYPE(*INT) LEN(4) +
DEFVAR(&ERR_DS)

DCL VAR(&RTN_LEN) STG(*DEFINED) TYPE(*INT) LEN(4) +
DEFVAR(&ERR_DS 5)

DCL VAR(&MSGID) STG(*DEFINED) TYPE(*CHAR) LEN(7) +
DEFVAR(&ERR_DS 9)

DCL VAR(&RESERVED) STG(*DEFINED) TYPE(*CHAR) LEN(1) +
DEFVAR(&ERR_DS 16)

DCL VAR(&MSGDTA) STG(*DEFINED) TYPE(*CHAR) LEN(240) +
DEFVAR(&ERR_DS 17)
```

Anschließend kann die Struktur als Ganzes unter dem Namen &ERRC0100 angesprochen werden, aber auch jedes Strukturfeld kann problemlos genutzt werden.

2.12.6.12.2
Seite 4

Wie in anderen Programmiersprachen können Sie auch Datenstrukturfelder redefinieren, um z. B. Datenstrukturfelder mit unterschiedlichen Formaten zu definieren. So ist es z. B. möglich, ein Datenstruktur-Unterfeld numerisch und gleichzeitig alphanumerisch zu definieren. Je nach Bedarf kann das eine oder andere Datenstrukturfeld verwendet werden. Schauen wir uns auch hierzu ein kleines Beispiel an:

Fehlersituation aufgrund unterschiedlicher Datentypen

Im Beispielprogramm benötige ich die Variable &NR einerseits als numerische Variable, andererseits kann ich die Variable im SNDPGMMSG-Befehl aber nur als alphanumerische Variable einsetzen. Bisher mussten wir in solchen Fällen stets ein Duplikat definieren und dann mittels des Befehls CHGVAR den Inhalt der Variablen &NR in das Duplikat &NR_A übertragen. Jetzt können wir dieses Problem auch direkt mit Hilfe von Datenstrukturen lösen:

```
DCL    VAR(&INFO) STG(*AUTO) TYPE(*CHAR) LEN(40)
DCL    &JOB     STG(*DEFINED) TYPE(*CHAR) LEN(10)   DEFVAR(&INFO 1)
DCL    &USER    STG(*DEFINED) TYPE(*CHAR) LEN(10)   DEFVAR(&INFO 11)
DCL    &SPLDAT  STG(*DEFINED) TYPE(*CHAR) LEN(10)   DEFVAR(&INFO 21)
DCL    &JOBNR   STG(*DEFINED) TYPE(*CHAR) LEN(6)    DEFVAR(&INFO 31)
DCL    &NR      STG(*DEFINED) TYPE(*DEC)  LEN(4 0)  DEFVAR(&INFO 37)
DCL    &NR_A    STG(*DEFINED) TYPE(*CHAR) LEN(4)    DEFVAR(&INFO 37)
```

Redefinierte Datenstrukturen

Leider können wir bislang keine Tabellen (=Arrays) in CL-Programmen definieren, die zum Aufruf mancher APIs sicherlich nützlich wären. Aber auch bei diesem Problem können Datenstrukturen helfen, wenn wir die Tabellenelemente einfach als Datenstrukturunterfelder definieren. Deutlich wird dies an einem weiteren Beispiel. Stellen Sie sich vor, dass Sie mehrere Systemwerteinstellungen im CL-Programm ermitteln wollen. Der CL-Befehl RTVSYSVAL (=retrieve systemvalue) kann nur einen einzigen Systemwert zur Zeit ermitteln. Wenn wir aber mehrere Systemwerte abrufen wollen, müssen wir entweder den Befehl mehrfach kodieren oder das API QWCRSVAL (Retrieve System Values) nutzen. Für den Aufruf dieses APIs sind insgesamt fünf Parameter erforderlich:

- Eine Empfangsvariable, die das Ergebnis enthält.
- Eine Integer-Variable, die die Länge der Empfangsvariablen bestimmt.
- Eine weitere Integer-Variable, die die Anzahl der zu ermittelnden System-Werte enthält.
- Ein Parameter, der als Array beschrieben werden sollte und die Namen der Systemwerte enthält, deren Einstellungen ermittelt werden sollen.
- Zum Schluss wird der Standarderrorcode übergeben.

Stellen Sie sich jetzt vor, dass wir verschiedene Systemwerteinstellungen ermitteln wollen. Um die Namen der Systemwerte zu übergeben, kann ich kein Array definieren, da CL diesen Datentypen (noch) nicht kennt. Also nutze ich einfach eine Datenstruktur namens &Array mit entsprechender Anzahl an Strukturunterfeldern. Die Datenstrukturunterfelder enthalten zum späteren Zeitpunkt die Namen der Systemwerte, die ich benötige.

```
DCL VAR(&ARRAY) TYPE(*CHAR) LEN(40)

DCL VAR(&VALUE1) STG(*DEFINED) TYPE(*CHAR) LEN(10) DEFVAR(&ARRAY  1)
DCL VAR(&VALUE2) STG(*DEFINED) TYPE(*CHAR) LEN(10) DEFVAR(&ARRAY 11)
DCL VAR(&VALUE3) STG(*DEFINED) TYPE(*CHAR) LEN(10) DEFVAR(&ARRAY 21)
DCL VAR(&VALUE4) STG(*DEFINED) TYPE(*CHAR) LEN(10) DEFVAR(&ARRAY 31)
```

Arrays simulieren

Anschließend kann ich das API problemlos aufrufen:

```
CALL PGM(QWCRSVAL) PARM(&RCVVAR &LENRCVVAR &ANZSYSVAL +
&ARRAY &ERR_DS)
```

2.12.6.12.2

Seite 6

CL – Pointer Release V5R4

Seit Release V5R4M0 ist es möglich, Variablen mit dem Datentyp Pointer (=Zeiger) zu definieren. Um Pointer-Variablen sinnvoll einsetzen zu können, ist außerdem eine entsprechende Pointer-Arithmetik notwendig, d.h. die Adressen von Variablen müssen ermittelt und gesetzt und dem Pointer zugeordnet werden. Solange man sich nur mit der klassischen RPG-, Cobol- und CL-Programmierung beschäftigt, spielen Pointer eine untergeordnete oder sogar überhaupt keine Rolle. Zwar werden intern die Adressen der einzelnen Felder sowie die zugehörige Anzahl an Bytes vom System auch über Pointer verarbeitet, aber der Programmierer braucht sich darum nicht zu kümmern. Er muss lediglich den gewünschten Datentypen und die Länge seiner Variablen angeben. In einer ILE-Umgebung werden jedoch immer mehr Funktionen eingebunden, die nicht in RPG, sondern in C geschrieben wurden. Und diese C-Funktionen sowie viele System-APIs setzen die Verwendung von Pointern voraus.

Versuchen wir einfach mal einen Pointer im CL-Programm zu definieren:

```
/* Pointer ohne Initialisierung */
DCL VAR(&P1) TYPE(*PTR) STG(*AUTO)
```

Die Definition ist einfach! Sie wählen lediglich den Datentypen *PTR und schon wird intern eine Pointer-Variable mit einer Länge von 16 Bytes angelegt. Eine eigene Längendefinition ist nicht erlaubt, da das System automatisch immer 16 Byte als Größe reserviert.

Solange dem Pointer keine Adresse zugeordnet ist, ist sein Wert NULL. NULL bedeutet in diesem Fall nicht 0 sondern ist ein feststehender Begriff für einen nicht definierten Speicherinhalt. Sie können allerdings für einen Pointer auch keinen VALUE-Wert angeben. Allerdings können Sie im Parameter ADRESS eine Variable benennen, auf die der Pointer zeigen soll. Zusätzlich kann durch die Angabe des Offsets (=Startposition) der Pointer auf eine beliebige Stelle innerhalb der Variablen positioniert werden. Unter Offset versteht man somit den Abstand zwischen dem Beginn der Variablen und der Position, auf die der Pointer zeigen soll. Soll also der Pointer auf den Beginn der Variablen zeigen, muss der Offsetwert 0 angegeben werden. Für die Verarbeitung ab dem 10. Byte ist der Offset 9 erforderlich. Der zulässige Wert für den Offset liegt zwischen 0 und 32.766 und sollte die Länge der Variablen nicht überschreiten, auf die der Pointer zeigt.

2.12.6.12.3
Seite 2

Schauen Sie sich das folgende Beispiel an, um zu verstehen, wie Pointer initialisiert werden können und was unter einem Offset zu verstehen ist:

```
/* Pointer mit Initialisierung */
DCL     VAR(&Liste) TYPE(*CHAR) STG(*AUTO) LEN(256)
        VALUE('ABCDEFGHIJKLMNOPQRST')
DCL     VAR(&P2) TYPE(*PTR) STG(*AUTO) ADDRESS(&Liste)
DCL     VAR(&P3) TYPE(*PTR) STG(*AUTO) ADDRESS(&Liste 7)
```

Der erste Pointer &P2 wird durch die Angabe des Parameters ADDRESS initialisiert und zeigt somit auf die Variable &LISTE. Achten Sie bei Ihrer Adresszuweisung auf die Reihenfolge Ihrer Variablendefinitionen. Die Variable &LISTE muss vor den Pointern definiert sein, damit die Adresse den Pointern zugewiesen werden kann. Der Pointer &P2 adressiert den Anfang der Variablen &LISTE, da kein Offset verwendet wird. Die Defaulteinstellung für den Offset-Wert ist in diesem Fall 0. Auch hier müssen einige Programmierer umdenken! Wir sind es eigentlich gewohnt, mit dem Wert 1 die Anfangsposition innerhalb eines Strings zu benennen. In der Pointer-Arithmetik wird aber mit dem Wert 0 begonnen. So erklärt es sich auch, dass der zweite Pointer &P3 ebenfalls auf die Variable &Liste zeigt, aber durch die Angabe des Offset-Wertes 7 das 8. Zeichen im String – also den Buchstaben H adressiert.

In dem Beispiel wurden beide Pointer zur Umwandlungszeit initialisiert, d. h. die Adresse ist beim Programmstart fest zugeordnet. Würde der Pointer während des gesamten Programm-Ablaufs auf die gleiche Adresse zeigen, wäre seine Verwendung eher überflüssig. Erst dann, wenn der gleiche Pointer während des Programmablaufs unterschiedliche Variablen oder unterschiedliche Offsets einer Variablen erhält, macht die Verwendung von Pointern Sinn. Es muss also eine Möglichkeit geben, Pointeradressen zur Laufzeit zu ermitteln und/oder zu verändern. Beides wird über die neue Built-in-Function %ADDRESS() realisiert, die einfach im VALUE-Parameter des CHGVAR-Befehls verwendet werden kann. Alternativ können Sie auch die Kurzform %ADDR() nutzen. Die Built-in-Function %ADDR () erfordert einen einzigen Parameter, nämlich den Namen der Variablen, deren Adresse dem Pointer zugewiesen werden soll. Um die vorhandene Adresse eines Pointers zu prüfen, kann die Built-in-Function %ADDRESS() auch einfach in einer IF- oder SELECT-Anweisung eingesetzt werden.

Im folgenden Beispiel wird dem Pointer P1 entweder die Adresse der Variablen VAR1 oder die Adresse der Variablen VAR2 zugeordnet, abhängig von der aktuellen Adresse, auf die der Pointer zeigt.

```
/* Variablen Definition */
       DCL   VAR(&VAR1) TYPE(*CHAR)    STG(*AUTO) LEN(20)
       DCL   VAR(&VAR2) TYPE(*CHAR)    STG(*AUTO) LEN(20)
       DCL   VAR(&P1) TYPE(*PTR)       STG(*AUTO) ADDRESS(&VAR1)

/* zeigt Pointer auf die Variable &VAR1?                          */
/* Dann neue Adresse von &VAR2 zuordnen                           */

       IF COND( &P1 = %ADDR(&VAR1)) +
             THEN(CHGVAR VAR(&P1) VALUE(%ADDRESS(&VAR2) ) )

/* falls der Pointer auf Variable &VAR2 zeigt                     */
/* soll der Pointer auf die Variable &VAR1 zeigen                 */

Else CMD(DO)
       IF COND(&P1 = %ADDR(&VAR2) ) +
             THEN(CHGVAR VAR(&P1) VALUE(%ADDRESS(&VAR1) ) )
       ENDDO
```

Es kann aber auch erforderlich sein, den Pointer innerhalb einer Variablen zu verschieben, d. h. den Offset-Wert zu verändern. Stellen Sie sich vor, dass in einer alphanumerischen Variablen mit einer Gesamtlänge von 50 Byte fünf unterschiedliche Werte mit einer Länge von jeweils 10 Byte hinterlegt sind. In einer Schleife müssen nun alle fünf Werte verarbeitet werden. Dabei wird dem Pointer zunächst die Adresse der ersten Position zugeordnet und bei jedem Durchlauf wird der Pointer um 10 Byte verschoben. Der Offsetwert müsste somit um den Wert 10 mit jedem Schleifendurchlauf erhöht werden. Dies lässt sich relativ einfach mit der neuen Built-in-Function %OFFSET() realisieren. Auch hier gibt es wieder ein Pendant: %OFS() ist die Kurzform.

Diese Built-in-Function erfordert als Argument die Pointer-Variable, für die der Offset-Wert ermittelt werden soll. Das Ergebnis aus der Built-in-Function %OFFSET() wird als 4 Byte unsigned Integer-Wert (*UINT) dargestellt. Zu diesem numerischen Wert kann nun jede beliebige ganze Zahl addiert oder subtrahiert werden, so dass der Pointer um die entsprechende Anzahl Byte verschoben wird.

2.12.6.12.3
Seite 4

Wird die Funktion %OFFSET() im VALUE-Parameter des CL-Befehls CHGVAR verwendet, wird der aktuelle Offset ermittelt, in einen vorzeichenlosen Integer-Wert konvertiert und in der Empfangsvariablen übertragen:

```
DCL VAR(&P3) TYPE(*PTR) STG(*AUTO) ADDRESS(&Liste 7)


CHGVAR VAR(&Z) VALUE(%OFFSET(&P3) )
```

Nach Ausführung der Anweisung CHGVAR würde die Variable &Z den Wert 7 enthalten.

Die Funktion %OFFSET() kann aber auch im Parameter VAR verwendet werden, um den Pointer neu zu positionieren.

```
DCL VAR(&P3) TYPE(*PTR) STG(*AUTO) ADDRESS(&Liste 7)


CHGVAR VAR(%OFFSET(&P3)) VALUE(%OFFSET(&P3) + 7)
```

Der Pointer zeigt jetzt auf das 14. Zeichen der Variablen &Liste. In unserem Beispiel also auf den Buchstaben O.

Nun wissen wir zwar, wo der Pointer steht und können die Pointeradressen auch manipulieren. Doch was ist mit den Dateninhalten? Es muss Möglichkeiten geben, auch auf die Daten zuzugreifen, auf die der Pointer verweist. Es muss also möglich sein, die Daten, auf die der Pointer zeigt, wieder einer Variablen zuzuordnen. Zu diesem Zweck werden *BASED-Variablen benutzt. Im Gegensatz zu anderen Variablen wird für *BASED-Variablen kein Speicherplatz zur Umwandlungszeit reserviert. Stattdessen werden sie einem Pointer zugeordnet. Zeigt der Pointer auf eine gültige Adresse, können die auf die Pointer-Position folgenden Byte über die *BASED-Variable genutzt werden. Sowohl einzelne Felder als auch Datenstrukturen können als *BASED-Variablen definiert werden.

Um eine based Variable zu definieren, muss im Befehl DCL der Parameter STG (Storage) auf *BASED gesetzt werden. Zusätzlich muss über die Option BASPTR (Basing Pointer Variable) der zugeordnete Pointer angegeben werden.

```
            DCL         VAR(&VAR1) TYPE(*CHAR) STG(*AUTO) VALUE('ABCDEFG'
            DCL         VAR(&VAR2) TYPE(*CHAR) STG(*AUTO) VALUE('ABCDEFG'
            DCL         VAR(&P1) TYPE(*PTR) STG(*AUTO)  ADDRESS(&VAR1)

/* ZEIGERVARIABLE  */
            DCL         VAR(&PTR_VAR) TYPE(*CHAR) STG(*BASED) +
                        LEN(10) BASPTR(&P1)

            CHGVAR VAR(&P1) VALUE(%ADDRESS(&VAR2) )   Pointer erhält die Adresse der
                                                      Variablen &P1 OFFSET 0

            SNDPGMMSG  MSG('ERSTE MELDUNG' *BCAT &PTR_VAR) + Ausgegeben wird
                         TOUSR(RASCHE)                       der Wert 'ABCDEFG'


                                                      Offset-Wert wird um
            CHGVAR (%OFFSET(&P1)) VALUE(%OFFSET(&P1) + 3)  3 Byte verschoben


            SNDPGMMSG  MSG('ERSTE MELDUNG' *BCAT &PTR_VAR) + Ausgegeben wird
                         TOUSR(RASCHE)                       der Wert 'DEFG'
```

Zeigervariablen nutzen

2.12.6.12.3

Seite 6

Neue Befehle Release V6R1

Das neue Release V6R1 steht seit dem Frühjahr 2008 zur Verfügung. High-end-Visualisierung, Security, Virtualisierung und Logical Partitioning, Hochverfügbarkeit und i5/OS im Blade Center sind gewiss wichtige Neuerungen! Der klassische System i-Programmierer fragt sich jedoch: Was bringt das neue Release V6R1 für mich? Oder haben die Programmierer dieses Mal das Nachsehen oder gar nur jede Menge Arbeit wie bei der Umstellung von CISC auf RISC, denn auch beim Release V6R1 muss eine Programmkonvertierung erfolgen.

Einige können sich sicherlich noch an die letzte Programmkonvertierung erinnern, 1995 beim Umstieg von den CISC- auf die RISC-Prozessoren und der damit verbundenen Veränderung der Adressbreite von 48 Bit auf 64 Bit. Auch damals war eine Konvertierung aller Programme erforderlich. IBM hat versprochen, dass die jetzige Konvertierung einfacher und schneller ist.

Mit dem neuen Release V6R1 werden neue, verbesserte Möglichkeiten des Machine Interface (MI) ausgenutzt, um signifikante Programmverbesserungen zu erzielen. Neben einem verbesserten Durchsatz stehen auch eine Reihe neuer Betriebssystem- und Prozessorfunktionen zur Verfügung. Um diese Verbesserungen nutzen zu können, ist es erforderlich alle Programme, die mit einem älteren Release erstellt wurden, zu konvertieren. Eine grundlegende Voraussetzung für die Konvertierbarkeit der Programme ist, dass die Erstellungsdaten verfügbar sind. Die Konvertierung ersetzt bestehende Programm- und Dateiobjekte, behält allerdings Eigenschaften wie Name, Bibliothek und Objekt-Eigner bei. Ärgerlich wäre es, wenn Sie während des Upgrades feststellen müssten, dass einige Ihrer Programmobjekte nicht konvertierbar wären. Aus diesem Grund hat IBM daher einen neuen CL-Befehl zur Verfügung gestellt:

```
ANZOBJCVN (Analyze Object Conversion).
```

Der Befehl soll Ihnen bei Ihren Vorbereitungen helfen und kann als PTF für Release V5R3 und V5R4 bezogen werden. Mit ANZOBJCVN ist es möglich festzustellen, ob es Programmobjekte gibt, die nicht konvertiert werden können. Das Tool liefert auch einen Schätzwert für den Zeitaufwand der Konvertierung. Außerdem liefert das Analyse-Tool auch Informationen zur Konvertierung von IFS-Namen und Spooldateien.

2.12.6.13
Seite 2

Es ist nicht unbedingt erforderlich, ANZOBJCVN vor dem Upgrade auf V6R1 durchzuführen. Aber es ist empfehlenswert, um Überraschungen zu vermeiden. Ich möchte an dieser Stelle darauf verzichten, alle Parametereinstellungen des Befehls ANZOBJCVN zu erläutern, da die Bedienerführung weitestgehend selbsterklärend ist. Grundsätzlich bezieht sich der CL-Befehl ANZOBJCVN entweder auf Bibliotheken (Schlüsselwort LIB) oder auf IFS-Verzeichnisse (Schlüsselwort OBJ). Eine Kombination beider Schlüsselwörter ist allerdings möglich. Das Schlüsselwort LIB erlaubt die Auswahl einer oder mehrerer Bibliotheken. Für die Analyse von Programmobjekten im IFS wird das Schlüsselwort OBJ verwendet. Die Analyse erfolgt anschließend in zwei Schritten. Zunächst müssen die Daten gesammelt werden (Auswahl *COLLECT) und in einem zweiten Schritt können Sie entsprechende Berichte (Auswahl *REPORT) erstellen. Für die Berichte sollten Sie ggf. die Druckdatei QPIZARPT ändern. Druckdateien erlauben häufig nur eine begrenzte Anzahl von Einträgen. Überschreitet der Bericht die zulässige Zeilenanzahl, kommt es zur Fehlersituation. Besser, Sie sorgen vor und ändern die Einstellungen des Printerfiles folgendermaßen:

```
CHGPRTF FILE(QSYS/QPIZARPT) MAXRCDS(*NOMAX)
```

Jetzt kann eine beliebige Anzahl von Sätzen in das Printerfile geschrieben werden und Sie erleben keine unangenehmen Überraschungen beim Erstellen Ihres Berichts. Auch für die Berichtserstellung gibt es verschiedene Auswertungsmöglichkeiten. So können Sie beispielsweise eine Auswertung über nicht konvertierbare Objekte erstellen. Hierzu starten Sie Ihren Report mit der **OPTION(*REPORT) RPTTYPE(*CVNPRB).** Falls Sie anschließend im Bericht in der Spalte ‚Erstellung Daten' den Wert ‚NEIN' bei einigen Programmen finden, sind für das Programm keine Erstellungsdaten vorhanden. In diesem Fall muss das Programm neu kompiliert werden. Wurden die Programme von einem Softwarehaus geliefert, so muss der Softwarelieferant informiert werden und eine konvertierbare Version liefern. Objekte, die ihre digitale Signatur verlieren, sind mit *YES in der Spalte ‚Digital Signiert' gekennzeichnet. Die Funktion des Objektes wird dadurch allerdings nicht beeinflusst. Die Spalte „Zustand" liefert Informationen über Objekte, deren Status von *SYSTEM oder *INHERIT in *USER geändert wird. Diese Objekte könnten im Release V6R1 funktional beeinträchtigt werden. Aber es können auch andere Attribute durch die Konvertierung verloren gehen. Welche Objekte hiervon betroffen sind, erfahren Sie, wenn Sie einen Bericht mit der **OPTION (*REPORT) RPTTYPE(*LIBDTL)** erstellen. In der Spalte ‚Einige verlorene Attribute' können Sie anschließend prüfen, welche Objekte betroffen sind.

Sollten die Standardberichte ihren Anforderungen nicht genügen, können Sie auch individuelle Auswertungen mit SQL erstellen. Zu diesem Zweck müssen Sie auch zunächst die Daten mit dem Befehl ANZOBJCVN OPTION(*COLLECT) sammeln. Die Informationen werden in der Bibliothek QUSRSYS in den folgenden Tabellen gespeichert:

- QAIZACVN – Informationen über die analysierten Bibliotheken
- QAIZADIR – Informationen über die analysierten IFS-Verzeichnisse
- QAIZOBJ – Informationen über die Objekte der analysierten IFS-Verzeichnisse
- QAIZASPL – Informationen über die analysierten Spooldateien

Eine detaillierte Beschreibung der Satzformate finden Sie im Anhang B des IBM Redpaper ‚i5 / OS Conversion'.

Sind Ihre Vorbereitungen abgeschlossen, stehen gleich mehrere Möglichkeiten der Konvertierung zur Verfügung:

Der CL-Befehl STROBJCVN Release V6R1
Sie können nach Einspielen des Release V6R1 den CL-Befehl STROBJCVN mit OPTION(*CONVERT) verwenden, um die Programmobjekte, Spooldateien, IFS-Namen und Datenobjekte in den ausgewählten Bibliotheken umzusetzen.

Zurückspeichern der Programme
Wechseln Sie mit dem Release auch Ihre Hardware, könnte es sinnvoll sein, die Konvertierung während der Rückspeicherung der Objekte durchzuführen. Auch wenn Sie Fremdsoftware einspielen, sollten Sie auf die Objektumsetzung achten. Sie können dafür sorgen, dass grundsätzlich alle Objekte, die Sie einspielen, umgesetzt werden sollen. Hierfür müssten Sie allerdings die Defaulteinstellung des Systemwertes QFRCCVNRST ändern. Per Defaulteinstellung werden nur Objekte mit Gültigkeitsfehlern umgesetzt. Ändern Sie hingegen den Systemwert auf einen Wert größer 1, werden auch Objekte umgesetzt, wenn die neue Betriebsversion dies erfordert. Detaillierte Informationen über die verschiedenen Einstellungen entnehmen Sie bitte der Bedienerhilfe.

Natürlich können Sie auch die Umsetzung von Fall zu Fall erzwingen, wenn Sie den Parameter FRCOBJCVN des CL-Befehls RSTLIB entsprechend von *SYSVAL auf *YES ändern.

Parameter FRCOBJCVN des CL-Befehls RSTLIB

In diesem Fall werden alle Objekte, die betroffen sind, automatisch konvertiert. Damit das Objekt umgesetzt wird, müssen natürlich auch jetzt die entsprechenden Erstellungsdaten vorhanden sein. Zwischen dem Systemwert QFRCCVNRST und dem Parameter FRCOBJCVN besteht ein direkter Zusammenhang. Wählen Sie für den Parameter FRCOBJCVN die Werte *YES *RQD, muss der Wert des Systemwerts QFRCCVNRST entweder „0“, „1“ oder „2“ lauten. Bei Angabe von FRCOBJCVN(*YES *ALL) darf der Systemwert dagegen einen beliebigen gültigen Wert annehmen. Wenn Sie die Objekte während des Zurückspeicherns umsetzen, dauert das Zurückspeichern natürlich länger. Die Objekte müssen dann aber bei ihrer ersten Verwendung nicht mehr umgesetzt werden. Auch hier sollten Sie weiterführende Informationen der Bedienerhilfe der jeweiligen CL-Befehle entnehmen.

Während der ersten Programmausführung
Konvertieren Sie Ihre Programme nicht manuell, dann wird das Betriebssystem die Konvertierung automatisch durchführen, wenn das Programm das erste Mal unter Release V6R1 genutzt wird. Der Aufruf der Programme könnte sich dadurch natürlich verzögern. Außerdem wäre es denkbar, dass einige Programme nicht ausgeführt werden können, falls die Konvertierung aufgrund fehlender Informationen nicht durchgeführt werden kann.

Mehrfachverarbeitung von Dateien V6R1

Das Release V5R3 und V5R4 enthielt eine Reihe bedeutender Erweiterungen für die CL-Programmierung. Es wurden Schleifenanweisungen, Subroutinen, Datenstrukturen und Pointer eingeführt. Das Release V6R1 ist aus Sicht eines CL-Programmierers zwar weniger spektakulär, hält aber dennoch einige Neuerungen bereit.

Bereits vor Release V6R1 war es möglich, mehrere Dateien in einem CL-Programm zu verarbeiten, aber es war nicht möglich, eine Datei innerhalb eines CL-Programms zu schließen, wieder zu öffnen und erneut zu verarbeiten. In Release V6R1 wurde diese Beschränkung aufgehoben, d. h. die gleiche Datei kann innerhalb des gleichen Programms mehrfach verarbeitet werden.

Eingeführt wurde zu diesem Zweck der neue CL-Befehl CLOSE. Der Befehl schießt explizit eine Datenbankdatei. Implizit geöffnet wird die Datei wieder über den nächsten RCVF-Befehl. Der Befehl ist nur gültig für Datenbankdateien und kann daher nicht für Bildschirmdateien genutzt werden.

Schauen wir uns auch hierzu ein einfaches Beispiel an. Das Programm liest die Datei Test_close zwei Mal. Bei jedem Schleifendurchlauf werden die gelesenen Sätze ermittelt und als Nachricht ausgegeben.

2.12.6.13.1
Seite 2

```
DCL     &RTNVAR_D *DEC LEN(2 0) VALUE(0)
DCL     &RTNVAR_A *CHAR LEN(2)
DCLF    FILE(TEST_CLOSE)

LOOP:   RCVF
            MONMSG CPF0864 EXEC(DO)
            CHGVAR &RTNVAR_A VALUE(&RTNVAR_D)
            CHGVAR &RTNVAR_D VALUE(0)
            SNDPGMMSG MSG(,Erstes Lesen: , *CAT &rtnvar_A *BCAT
,Sätze‘) +
        TOUSR(RASCHE)
            GOTO NEU
        ENDDO

        CHGVAR &RTNVAR_D VALUE(&RTNVAR_D + 1)
        GOTO LOOP


NEU:    RCVF
            MONMSG CPF0864 EXEC(DO)
            CHGVAR &RTNVAR_A VALUE(&RTNVAR_D)
SNDPGMMSG MSG(,Zweites Lesen: , *CAT &RTNVAR_A *BCAT +
’Sätze’) TOUSR(RASCHE)
        GOTO ENDE
        ENDDO

        CHGVAR &RTNVAR_D VALUE(&RTNVAR_D + 1)
        GOTO NEU

ENDE: ENDPGM
```

Das Ergebnis sieht folgendermaßen aus:

```
Von . . . . :   RASCHE          14.07.09   12:49:31
Erstes Lesen: 02 Sätze
Von . . . . :   RASCHE          14.07.09   12:49:31
Zweites Lesen: 00 Sätze
```

Ergebnis traditionell (ohne CLOSE-Anweisung)

Bislang war es nicht möglich, eine Datei innerhalb eines Programms wiederholt zu lesen. Häufig mussten aus diesem Grund externe Unterprogramme verwendet werden, um ein implizites Schließen zu erzwingen.

Seit Release V6R1 kann die Datei aber im Programm explizit geschlossen werden, so dass derartige Dateiverarbeitungen jetzt auch im CL-Programm keine Probleme mehr bereiten:

```
LOOP:   RCVF
            MONMSG CPF0864 EXEC(DO)
                CHGVAR &RTNVAR_A VALUE(&RTNVAR_D)
                :
                :
/* Datei wird explizit geschlossen                                  */
                CLOSE

                GOTO NEU
                ENDDO

            CHGVAR &RTNVAR_D VALUE(&RTNVAR_D + 1)
            GOTO LOOP

/* Der nächste RCVF-Befehl öffnet die geschlossene Datei wieder   */
NEU:        RCVF
            MONMSG CPF0864 EXEC(DO)
```

Das Ergebnis ist eindeutig:

```
Antwort eingeben (falls erforderlich) und Eingabetaste drücken.
  Von . . . . :    RASCHE         14.07.09   12:50:46
  Erstes Lesen: 02 Sätze
  Von . . . . :    RASCHE         14.07.09   12:50:46
  Zweites Lesen: 02 Sätze
```

Ergebnis V6R1 mit CLOSE-Anweisung

2.12.6.13.1

Seite 4

Copy-Strecken im CL-Programm V6R1

RPG-Programmierer arbeiten bereits seit Langem mit COPY-Strecken und sparen sich dadurch viel Zeit und vor allem die Mehrfachkodierung. Jetzt können auch im CL-Programm Copy-Strecken verwendet werden. Der neue CL-Befehl INCLUDE ermöglicht es, dass der Source Code, der in einer anderen Quelle hinterlegt ist, in die aktuelle Teildatei eingebunden wird. Mit dem Befehl INCLUDE kann ein externer Text in das zu kompilierende Quellenprogramm eingeschlossen werden. Bei dem externen Text muss es sich natürlich um gültige CL-Befehle handeln. Sie können Deklarationsanweisungen, wie z. B. die Anweisung DCL oder DCLF, verwenden, aber auch Steuerungsanweisungen wie IF oder Select und einfache CL-Befehle darf die Copy-Strecke enthalten. Das Einfügen des Textes darf allerdings nicht dazu führen, dass Deklarationsbefehle danach an der falschen Stelle im Programm erscheinen und die kopierten Zeilen die Anweisungen PGM oder ENDPGM enthalten. Auch die INCLUDE-Anweisung darf nicht in der Copy-Strecke enthalten sein.

Schauen wir uns hierzu ein kleines Beispiel an:
Ich verwende häufig globale Fehlerroutinen, um über Fehler informiert zu werden, die nicht direkt im Programm abgefangen werden. Zu diesem Zweck benötige ich in vielen Programmen immer wieder die für die Routine erforderlichen Variablen.

```
****************** Datenanfang *
     DCL &MSGDTA   *CHAR 150
     DCL &ID       *CHAR   7
     DCL &MSGF     *CHAR  10
     DCL &MSGLIB   *CHAR 10
```

Die Copy-Strecke „CPY_D_MSG“

Anstatt die Variablen immer wieder neu zu kodieren, kann ich sie jetzt in einer Teildatei speichern und anschließend in den entsprechenden Programmen einbinden:

```
****************** Datenanfang ********************************
PGM
     INCLUDE    SRCMBR(CPY_D_MSG) SRCFILE(EPS002CR/QITPSRC)
```

Einbinden einer Copy-Strecke

2.12.6.13.2
Seite 2

Der Befehl INCLUDE ist einfach zu kodieren. Doch wenn Sie die Bedienerführung dieses Befehls aufrufen, werden Sie verwundert feststellen, dass Sie nicht sofort die Möglichkeit erhalten den Namen der Quellendatei und die entsprechende Bibliothek zu benennen, wie ich es in meinem Beispiel getan habe.

Bedienerführung des Befehls INCLUDE

Der Grund hierfür ist einfach zu verstehen. Die Quellendatei muss nämlich nicht zwingend benannt werden. Wenn Sie keine Quellendatei im Sourcecode benennen, wird die zu kopierende Teildatei in der Quellendatei gesucht, die Sie im Umwandlungsparameter INCFILE des Befehls CRTCLPGM angeben:

Bedienerführung des Befehls CRTCLPGM

Per Defaulteinstellung würde die Copy-Strecke dadurch in derselben Quellendatei, in der auch der Quellcode des Programms gespeichert ist, gesucht werden. Wollen Sie eine andere Quellendatei nutzen, müssen Sie entweder den Parameter INCFILE des CRTCLPGM-Befehls ändern oder wie in meinem Beispiel den Namen der Quellendatei direkt im Quellcode angeben.

Nachrichtenverwaltung

Die iSeries ist ein nachrichtenorientiertes System. In allen Bereichen werden Nachrichten zur Steuerung des Systems eingesetzt. Deshalb ist es wichtig, das Konzept der Nachrichten zu verstehen und zu wissen, wie diese Nachrichten zu behandeln und zu verarbeiten sind.

2.12.7
Seite 2

Was ist eine Nachricht?

Unter einer Nachricht verstehen wir die Kommunikation zwischen zwei Bereichen. Diese Bereiche können entweder zwei Benutzer, zwei Programme oder auch ein Benutzer und ein Programm sein. Ursachen für die Erstellung oder Verwendung von Nachrichten können in unterschiedlichen Ausgangsbereichen liegen. Zum Einen kennen wir Nachrichten, die im Zuge der Jobausführung automatisch erstellt werden und uns Informationen über den Ablauf des Jobs anzeigen, z. B. die Nachricht „Job wurde normal beendet". In anderen Fällen weisen die erstellten Nachrichten auf gewisse Problemsituationen hin. Nachrichten werden immer gesendet und empfangen. Wird eine Nachricht von dem System erstellt und an den Empfänger gesendet, so wird diese Nachricht vom System als zugestellt definiert. Die empfangende Stelle nimmt die Nachricht automatisch entgegen. Nachrichten werden vom Empfänger aber nicht immer sofort gelesen bzw. verarbeitet. Meistens werden die Nachrichten zunächst gesammelt bis der Empfänger Zeit hat sie zu lesen. Dieses Verfahren kennen wir auch aus anderen Kommunikationsbereichen. Denken Sie an Ihren Anrufbeantworter: Er sammelt die eingehenden Nachrichten und stellt sie bei Bedarf zur Verfügung. Ähnlich verhält es sich bei der iSeries: Sie sammelt die eingehenden Nachrichten in Nachrichtenwarteschlangen. Einträge in den Nachrichtenwarteschlangen werden nicht automatisch entfernt, wenn Sie die Einträge gelesen haben. Ähnlich Ihrem Anrufbeantworter bleiben die Nachrichten als „alte" Nachrichten erhalten, noch nicht gelesene Einträge werden als „neue" Nachrichten definiert.

3. Ergänzung 06/2005

2.12.7.1
Seite 2

Permanente und temporäre Nachrichtenwarteschlangen

Unter einer Nachrichtenwarteschlange verstehen wir somit einen Bereich, in dem Nachrichten gesammelt werden. Nachrichtenwarteschlangen können entweder permanent vorhanden sein oder für bestimmte Verarbeitungen temporär erstellt werden.

Bei einer permanenten Nachrichtenwarteschlange handelt es sich um ein Objekt der iSeries vom Typ *MSGQ, das wie andere Objekte der iSeries in einer Bibliothek gespeichert ist.

Für die Verarbeitung der permanenten Nachrichtenwarteschlangen stehen uns folgende Befehle zur Verfügung:

```
CRTMSGQ        Create Messagequeue
DLTMSGQ        Delete Messagequeue
CLRMSGQ        Clear Messagequeue
WRKMSGQ        mit Nachrichtenwarteschlangen arbeiten
CHGMSGQ        Change Messagequeue
```

Eine permanente Nachrichtenwarteschlange muss zunächst erstellt werden, bevor sie genutzt werden kann. In den meisten Fällen wird die Erstellung manuell mit dem Befehl CRTMSGQ durchgeführt. Permanente Nachrichtenwarteschlangen werden aber auch automatisch beim Anlegen von Benutzerprofilen (CRTUSRPRF) oder beim Erstellen von Einheitenbeschreibungen (CRTDEVD) erstellt. So kommt es, dass jedes Benutzerprofil und jede Workstation über eine eigene permanente Nachrichtenwarteschlange verfügt. Eine der wichtigsten Nachrichtenwarteschlangen ist die Nachrichtenwarteschlange QSYSOPR. Dabei handelt es sich um die Nachrichtenwarteschlange des Systemadministrators. Diese Nachrichtenwarteschlange können wir sowohl mit dem Objektnamen QSYSOPR adressieren als auch über den Sonderwert *SYSOPR. Obwohl permanente Nachrichtenwarteschlangen für alle Bereiche der Verarbeitung auf der iSeries eingesetzt werden können, werden sie in der Regel für die Kommunikation zwischen zwei Benutzern oder für die Sendung einer Nachricht von einem Programm an einen Benutzer verwendet.

Temporäre Nachrichtenwarteschlangen dagegen erstellt das System automatisch. Sie bleiben nur für eine bestimmte Zeit im System. Beim Starten eines Jobs erstellt die iSeries automatisch eine Jobnachrichtenwarteschlange. Diese temporäre Jobnachrichtenwarteschlange besteht für die Dauer des Jobs. Im Falle eines interaktiven Jobs ist der Name identisch mit dem Namen der Arbeitsstation, auf der dieser Job ausgeführt wird. Wird der Job als Batch-Job ausgeführt, wird die Jobwarteschlange dem Systemoperator zugeordnet. Der Name der Jobnachrichtenwarteschlange ist allerdings unwichtig, da diese Nachrichtenwarteschlange als externe Nachrichtenwarteschlange bezeichnet

wird und über den Sonderwert *EXT adressiert werden kann. Wir werden hierzu noch einige Beispiele sehen. Zusätzlich zu der externen Jobnachrichtenwarteschlange erstellt das System temporäre Programmnachrichtenwarteschlangen. Diese Programmnachrichtenwarteschlangen existieren solange, wie sich die Programme oder Prozeduren im Aufruf-Stack befinden. Der Name einer solchen Programm-Nachrichtenwarteschlange ist immer identisch mit dem Namen des Programms oder der Prozedur. Aber auch hier benötigen wir häufig nicht den physikalischen Namen sondern verwenden den Parameterwert *PGMQ, um die Programmnachrichtenwarteschlange zu adressieren. Programmnachrichtenwarteschlangen werden häufig für die Kommunikation zwischen zwei Programmen oder Prozeduren eingesetzt. Die externe Nachrichtenwarteschlange (*EXT) wird dagegen gern für den direkten Nachrichtenaustausch zwischen dem Aufrufer des Programms (Anforderer) und den unterschiedlichen Programmen verwendet, die in dem Job genutzt werden. Wir kennen jetzt den Unterschied zwischen permanenten und temporären Nachrichtenwarteschlangen. Es fehlt allerdings noch die Unterscheidung der unterschiedlichen Nachrichtenarten.

Nachrichtenarten

Nachrichtentyp	Beschreibung
Request (*RQS)	Mit Anforderungsnachrichten wird eine Anforderung zur Ausführung eines Befehls ausgegeben. Wenn Sie einen interaktiven Befehl eingeben und anschließend die Daten-Freigabe-Taste drücken, wird in Wirklichkeit eine Nachricht vom Typ *RQS gesendet. Das Befehls-Anforderungsprogramm, meistens das Programm QSYS/QCMD, empfängt die Nachricht und führt die Anforderung aus. Anforderungs-Nachrichten können nur an eine temporäre Nachrichtenwarteschlange gesendet werden.
Inquiry (*INQ)	Anfragenachrichten sind Nachrichten, die vom Empfänger eine Antwort erwarten. Der Empfänger erhält neben der eigentlichen Nachricht ein Eingabefeld, in dem er seine Antwort auf die Nachricht eingeben kann. Die Antwort wird anschließend an den Sender der ursprünglichen Nachricht als Antwortnachricht (*RPY) zurückgesendet. Abfragenachrichten können nur an permanente Nachrichtenwarteschlangen oder an die externe Jobnachrichtenwarteschlange (*EXT) gesendet werden.
Reply (*RPY)	Antwortnachrichten werden automatisch erstellt, wenn der Empfänger einer Abfragenachricht seine Antwort sendet. Antwortnachrichten können nicht manuell ausgelöst werden. Sie sind immer das Ergebnis einer Abfragenachricht und können nur an permanente Nachrichtenwarteschlangen oder an die externe Nachrichtenwarteschlange *EXT gesendet werden.
Completion (*COMP)	Beendigungsnachrichten informieren den Empfänger darüber, dass ein Befehl fehlerfrei, d.h. normal ausgeführt wurden. Beendigungsnachrichten können an jeden Benutzer oder Prozess gesendet werden.
Diagnostic (*DIAG)	Diagnosenachrichten informieren den Empfänger über Fehler, die während der Ausführung eines Befehls aufgetreten sind. Diagnosenachrichten können an jeden Benutzer oder Prozess gesendet werden.
Escape (*ESCAPE)	Abbruchnachrichten informieren den Empfänger darüber, dass ein zuvor ausgeführter Befehl abnormal beendet wurden. Abbruchmeldungen können nur an Programmnachrichtenwarteschlangen gesendet werden. Das Programm, das die Abbruchnachricht sendet, wird sofort beendet. Das empfangende Programm führt eine Funktionsprüfung aus, die ebenfalls zum Abbruch führt, falls die Ausnahmenachrichten nicht mit dem MONMSG-Befehl abgefangen werden. Abbruchnachrichten müssen in Nachrichtendateien definiert werden.

2.12.7.3
Seite 2

Notify (*NOTIFY)	Eine Hinweisnachricht informiert den Empfänger über eine behebbare Fehlersituation und verlangt, ähnlich einer *INQ Nachricht, eine Antwort vom Empfänger. Benachrichtigungsmeldungen können nur an eine Programmnachrichtenwarteschlange oder an *EXT gesendet werden. Wenn ein Programm eine Benachrichtigungsmeldung an den aufrufenden Prozess sendet, wird die Ausführung des aufrufenden Programms unterbrochen. Die Verarbeitung des sendenden Programms wird nur fortgesetzt, wenn das empfangende Programm die Hinweisnachricht überwacht, oder wenn die Hinweisnachricht an die externe Jobnachrichtenwarteschlange gesendet wurde und der Benutzer eine Antwort eingibt. Hinweisnachrichten müssen in Nachrichtendateien definiert werden.
Information (*INFO)	Mit Informationsnachrichten können Informationen zwischen Programmen und/oder Benutzern ausgetauscht werden. Informationsnachrichten können an permanente und temporäre Nachrichtenwarteschlangen gesendet werden.
Status (*STATUS)	Statusnachrichten werden dazu verwendet, um über den Fortschritt eines laufenden Prozesses zu informieren, insbesondere dann, wenn der Prozess in mehreren Schritten abläuft. Obwohl Statusnachrichten an Programmnachrichtenwarteschlangen und an die externe Jobnachrichtenwarteschlange (*EXT) gesendet werden können, ist deren Verwendung in Verbindung mit Programmnachrichtenwarteschlangen eher selten. Normalerweise wird *EXT verwendet, wodurch die Nachricht in der Statuszeile des Bildschirms erscheint, während das Programm oder der Prozess läuft. Statusnachrichten müssen in Nachrichtendateien definiert werden.

Der Einsatz von Nachrichtendateien

Sie kennen die Vielzahl der Systemnachrichten, die Sie von der iSeries erhalten. Diese Systemnachrichten werden in Nachrichtendateien verwaltet. Die IBM verwaltet alle Nachrichten in der Nachrichtendatei QCPFMSG. Auch Sie können Ihre Nachrichten vordefinieren, um sie dann im Bedarfsfall abzurufen. Der Anwender erhält somit immer identische Mitteilungen – unabhängig davon, welcher Programmierer das Programm geschrieben hat. Sie halten diesen Einsatz für nutzlos? Dann schauen Sie sich einmal an, wie viele Möglichkeiten es gibt, dem Anwender mitzuteilen, dass ein bestimmtes Objekt nicht gefunden wurde. Nachrichtendateien werden spätestens dann unentbehrlich, wenn Sie mehrsprachig arbeiten, oder wollen Sie alle Programme, die Nachrichten versenden, mehrfach kodieren? Was also sind Nachrichtendateien? Nachrichtendateien sind Objekte der Art *MSGF, die mit dem Befehl CRTMSGF erstellt werden.

```
                    Nachrichtendatei erstellen (CRTMSGF)

Auswahl eingeben und Eingabetaste drücken.

Nachrichtendatei . . . . . .   _________     Name
  Bibliothek . . . . . . . .     *CURLIB     Name, *CURLIB
Text 'Beschreibung'  . . . . .   *BLANK
_____

                Zusätzliche Parameter

Dateigröße:
  Anfangsspeichergröße . . . . .   10          Kilobyte
  Erweiterungswert . . . . . .     2           Kilobyte
  Maximale Erweiterungen . . . .   *NOMAX      Zahl, *NOMAX
Berechtigung . . . . . . . . .     *LIBCRTAUT  Name, *LIBCRTAUT, *CHANGE...
Zeichensatz-ID . . . . . . . .     *HEX        *HEX, *MSGD, *JOB...



                                                                      Ende
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Bedienerführung des Befehls CRTMSGF

Nachrichtendateien können in einer beliebigen Bibliothek gespeichert werden, haben zunächst eine Anfangsgröße von 10 Kilobyte und erweitern sich dynamisch. Für die Bearbeitung stehen Ihnen die folgenden Befehle zur Verfügung:

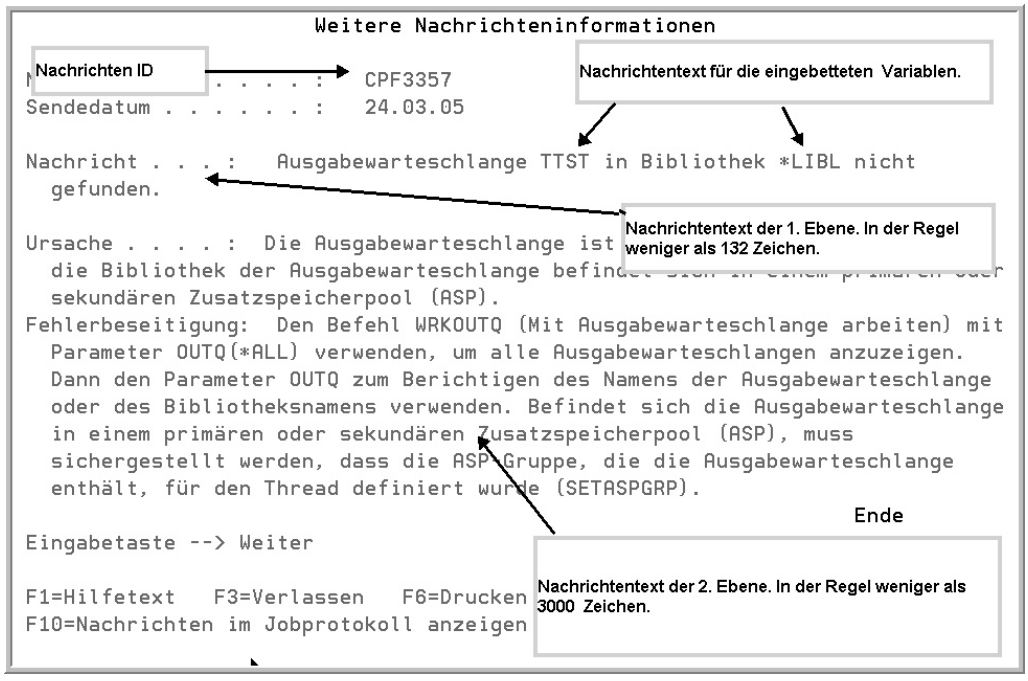
2.12.7.4
Seite 2

CRTMSGF	Create Messagefile	ADDMSGD Nachrichtenbeschreibung hinzufügen
DLTMSGF	Delete Messagefile	RMVMSGD Nachrichtenbeschreibung entfernen
CHGMSGF	Change Messagefile	CHGMSGD Nachrichtenbeschreibung ändern
WRKMSGF	Work Messagefile	DSPMSGD Nachrichtenbeschreibung anzeigen

In den Nachrichtendateien werden die Nachrichtentexte gespeichert. Jede Nachricht erhält einen eindeutigen Nachrichtenschlüssel, die Nachrichten-ID. Die Nachrichten-ID ist ein alphanumerischer 7 Byte langer Schlüssel, wobei IBM folgenden Aufbau empfiehlt:

Zeichen	Zeichen	Zeichen	Ziffer	Ziffer	Ziffer	Ziffer

Zusätzlich zur Nachrichten-ID werden die Nachrichtentexte gespeichert. Jede vordefinierte Nachricht besteht aus einem Nachrichtentext der 1. Ebene und einem Nachrichtentext der 2. Ebene. Schauen wir uns hierzu einfach mal eine vordefinierte Nachricht des Systems an.



Vordefinierte Nachrichtentexte

Sie können eigene Nachrichten in Ihrer Nachrichtendatei erfassen, indem Sie den Befehl ADDMSGD verwenden. Ich bevorzuge allerdings den Befehl WRKMSGD.

```
                    Mit Nachrichtenbeschr.arbeiten (WRKMSGD)

Auswahl eingeben und Eingabetaste drücken.

Nachrichten-ID . . . . . . . . . MSGID          *FIRST
Nachrichtendatei . . . . . . . . MSGF           message
  Bibliothek . . . . . . . . . .                  *LIBL
```

Bedienerführung des Befehls WRKMSGD

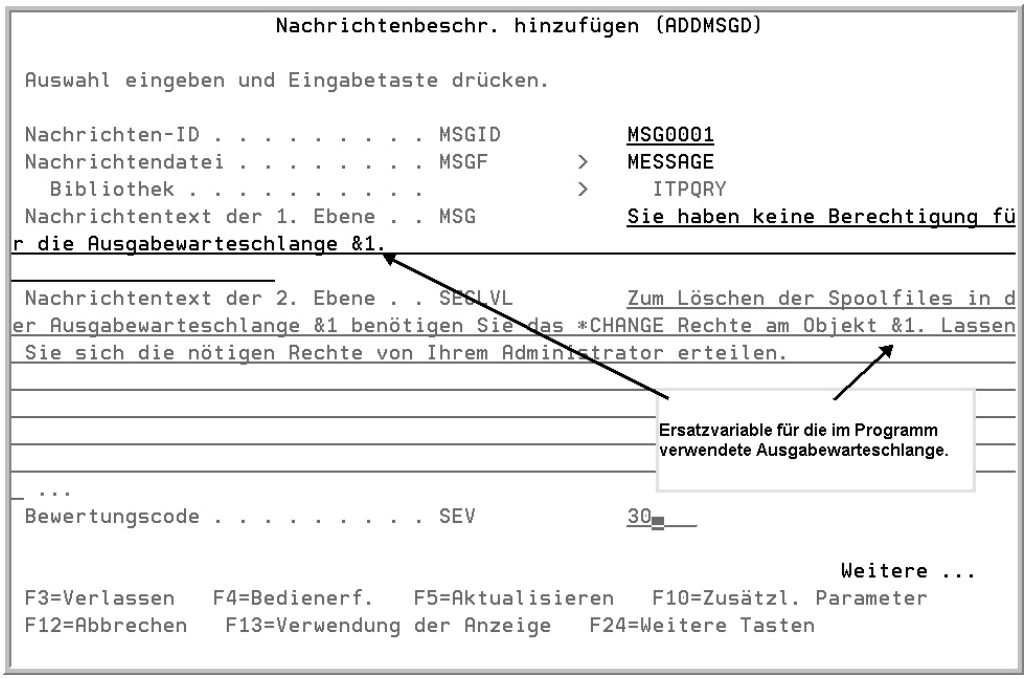
Anschließend erhalten wir alle gespeicherten Nachrichtentexte, die wir jetzt bearbeiten können.

```
                    Mit Nachrichtenbeschreibungen arbeiten
                                                          System:   ITPIS02
Nachrichtendatei:   MESSAGE        Bibliothek:   ITPQRY

CHGMSGD        RMVMSGD        DSPMSGD

  2=Ändern   4=Löschen   5=Details anzeigen   6=Drucken

Aus-  Nachrichten
wahl      ID       Bewertung  Nachrichtentext
 _      MSG0001        30     Sie haben keine Berechtigung für die Ausgabewart



                                ADDMSGD
                                                                Ende
Parameter oder Befehl
===> ____________________________________________________________________
F3=Verlassen   F5=Aktualisieren   F6=Hinzufügen   F12=Abbrechen
F24=Weitere Tasten
```

Die Anzeige WRKMSGD

Um eine eigene vordefinierte Nachricht zu erfassen, nutzen Sie jetzt die Funktionstaste F6. Sie können natürlich auch direkt den Befehl ADDMSGD verwenden. Schauen wir uns den Befehl anhand unseres Beispielprogramms, in dem wir dem Operator und dem Anwender verschiedene Nachrichten senden, einmal an:

2.12.7.4
Seite 4



Bedienerführung des Befehls ADDMSGD Seite 1

Der Parameter MSGID(MSG0001)

Der Parameter MSGID(MSG0001) ist die Nachrichten-ID. Innerhalb einer Nachrichtendatei muss jede Nachricht eine andere Kennung haben. Die Nachrichten-ID sollte aus drei Buchstaben, gefolgt von einer fortlaufenden Zahl bestehen. Vermeiden Sie bitte die Zeichenfolge CPF, CPD, CPI, da IBM diese Systemcodes verwendet.

Der Parameter MSGF(ITPQRY/MESSAGE)

Der Parameter MSGF(ITPQRY/MESSAGE) ist der qualifizierte Name der Nachrichtendatei, zu der die Nachricht hinzugefügt werden soll.

Der Parameter MSG('Sie ..... für die Ausgabewarteschlange &1')

Der Parameter MSG('Sie …… für die Ausgabewarteschlange &1') ist der Nachrichtentext der ersten Ebene. &1 dient als variabler Ersatzwert und wird durch den entsprechenden Objektnamen ersetzt, wenn die Nachricht gesendet wird. Es können bis zu 99 Ersatzvariablen verwendet werden, indem Sie die Werte &2, &3 usw. verwenden.

Der Parameter SECLVL('Zum Löschen .....')

Der Parameter SECLVL('Zum Löschen ……') ist der Nachrichtentext der zweiten Ebene. Diese Nachricht wird angezeigt, wenn der Cursor auf den Nachrichtentext der ersten Ebene positioniert wird und anschließend die Funktionstaste F1 (Hilfe) gedrückt wird. Es können bis zu 3000 in Hoch-

kommata eingeschlossene Zeichen angegeben werden, aber die Größe des Bildschirms sollte in Betracht gezogen werden. Überschreitet die Nachricht 512 Zeichen, können Sie den Text aufgrund des Grenzwerts für OS/400-Bedienerführungen nicht über die Bedienerführung erfassen. In diesem Fall müssen Sie den Befehl in der Befehlseingabezeile oder in einem CL-Programm eingeben. Zusätzlich können Sie die Nachrichtenhilfetexte mit den Formatsteuerzeichen &N, &P und &B formatieren. Die Bedeutung der Steuerzeichen können Sie der Bedienerhilfe des Befehls ADDMSGD entnehmen.

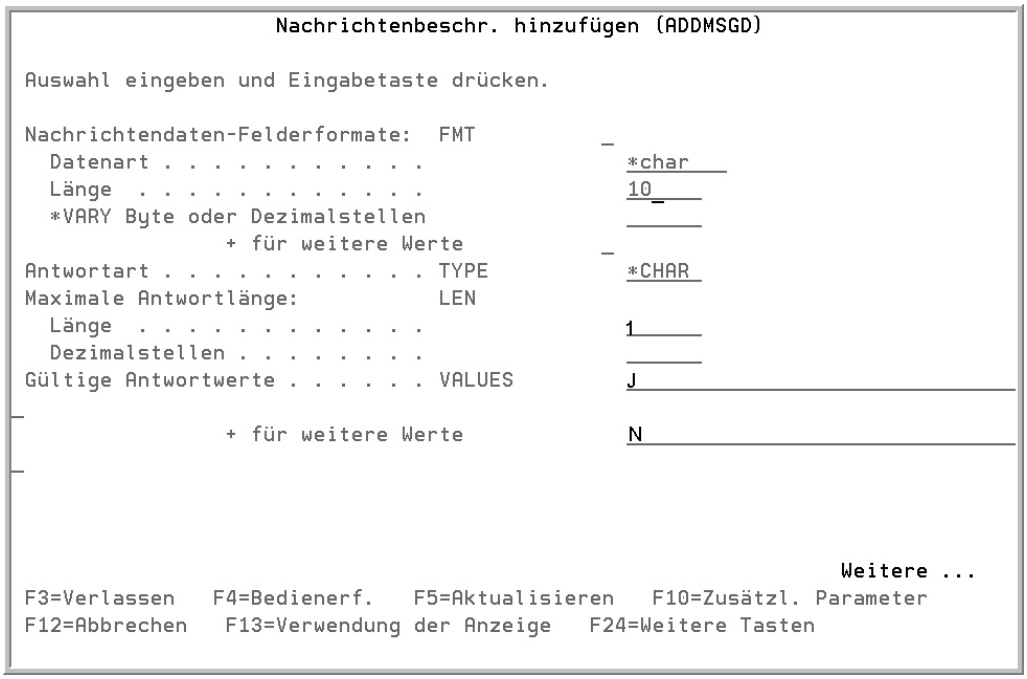
Der Parameter SEV(30)

Der Parameter SEV(30) ist der Bewertungscode dieser Nachricht. IBM nutzt die folgende Zuordnung:

Bewertungscode	Beschreibung
00	Information
10	Warnung
20	Fehler
30	Schwerwiegender Fehler
40	Abnormales Programmende
50	Abnormales Jobende
60	Systemwarnung
70	Einheitenfehler
80	Systemalarm
90	System-/Systemintegrität
99	Bedienereingriff erforderlich

3. Ergänzung 06/2005

2.12.7.4
Seite 6



Bedienerführung des Befehls ADDMSGD Seite 2

Der Parameter FMT(*CHAR 10)

Der Parameter FMT(*CHAR 10) bestimmt das Format der Ersatzwerte. Das erste Element bestimmt die Datenart, in der die Daten formatiert werden, wenn sie im Nachrichtentext ersetzt werden. Der Inhalt des zweiten und dritten Elements ist unterschiedlich je nach der angegebenen Datenart. Sie können die folgenden Datenarten verwenden:

*QTDCHAR	Eine in Hochkommata eingeschlossene Zeichenfolge, z. B. ('Erster Montag')
*CHAR	Eine Zeichenfolge, die nicht in Hochkommata eingeschlossen ist, z. B. ein Objektname. Leerstellen am Ende der Zeichenfolge werden abgeschnitten.
*HEX	Eine hexadezimale Bytefolge.
*DEC	Eine gepackte Dezimalzahl.
*BIN	Ein Binärwert mit Vorzeichen, der entweder 2, 4 oder 8 Byte lang ist.
*UBIN	Ein Binärwert ohne Vorzeichen, der entweder 2, 4 oder 8 Byte lang ist.
*CCHAR	Eine Zeichenfolge, die in die CCSID der Nachrichtenwarteschlange umgesetzt werden kann.

Die Werte *DTS, *SPP, *SYP, *ITV sind nur in von IBM vorgegebenen Nachrichtenbeschreibungen gültig und sollten nicht für andere Nachrichten verwendet werden. Das 2. Element bestimmt die Länge des variablen Ersatzwertes. Wie das zweite Element verwendet wird, hängt von der Datenart ab, die

im ersten Wert verwendet wurde. Falls *QTDCHAR, *CHAR oder *HEX angegeben ist, wird für die Länge standardmäßig der Wert *VARY angenommen. Sie können allerdings auch konstante Längen vereinbaren wie in unserem Beispiel. Falls die Ersatzvariable vom Typ *DEC ist, muss die Gesamtlänge der Variablen einschließlich aller Dezimalstellen vereinbart werden. Für die Datenart *BIN oder *UBIN können die Nachrichtendatenersatzfelder nur 2, 4 oder 8 Byte lang sein. Für die Datenart *CCHAR ist immer ein Feld variabler Länge erforderlich, da die CCSID möglicherweise die Feldlänge ändern kann. Falls Sie eine variable Länge vereinbart haben, bestimmt das 3. Element die tatsächliche Länge der variablen Felder.

Die Parameter TYPE, LEN, VALUES

Die Parameter TYPE, LEN, VALUES können Sie nutzen, um mögliche Antwortwerte einer Abfrage- oder Hinweisnachricht zu spezifizieren. Auch wenn es bei unserer Nachricht eher unwahrscheinlich ist, dass sie als Abfragenachricht versendet wird, wären jetzt in diesem Beispiel nur die Werte „J“ und „N“ als Antworten erlaubt. Für die Spezifizierung der Antworten stehen auf der 3. und 4. Seite der Bedienerführung weitere Parameter zur Verfügung.

3. Ergänzung 06/2005

2.12.7.4

Seite 8

Nachrichten senden und empfangen

Kommen wir zu unseren CL-Programmen zurück. Wir haben verschiedene Befehle, um Nachrichten zu senden. Bisher haben wir meistens den Befehl SNDMSG eingesetzt.

2.12.7.5

Seite 2

Der Befehl SNDMSG

Der Befehl SNDMSG dient zum Versenden von frei formulierten Nachrichten an eine permanente Nachrichtenwarteschlange. Der Befehl SNDMSG eignet sich eher für die interaktive Arbeit, kann aber durchaus auch in CL-Programmen verwendet werden.

Der Parameter MSG

Der Parameter MSG kann einen Nachrichtentext mit einer Länge von bis zu 512 Zeichen aufnehmen.

Der Parameter TOUSR

Der Parameter TOUSR kennzeichnet den Empfänger der Nachricht. In TOUSR kann entweder der Name eines spezifischen Benutzerprofils angegeben werden oder einer der folgenden Sonderwerte:

- *SYSOPR Die Nachricht wird an den Systembediener gesendet.
- *REQUESTER Die Nachricht wird dem aktuellen Benutzer zugestellt.
- *ALLACT Sendet die Nachricht an alle am System angemeldeten Benutzer.

Der Parameter TOMSGQ

Der Parameter TOMSGQ kann anstelle des Parameters TOUSR verwendet werden. Dabei wird die Nachricht an eine permanente Nachrichtenwarteschlange gesendet. Die Werteliste kann 50 Einträge enthalten. So kann ein und dieselbe Nachricht an unterschiedliche Benutzer versendet werden. Jeder dieser Werte kann den qualifizierten Namen einer Nachrichtenwarteschlange oder den Eintrag *SYSOPR enthalten. Da auch das Systemprotokoll QHST eine Nachrichtenwarteschlange ist, kann der Befehl SNDMSG auch dazu verwendet werden, um Nachrichten direkt in dieses Protokoll zu schreiben.

Der Parameter MSGTYPE

Der Parameter MSGTYPE bestimmt den Nachrichtentyp. Der Defaultwert ist *INFO. Die Nachrichtenart *INQ(Abfragenachricht) kann nur verwendet werden, wenn die Nachricht an einen Empfänger gerichtet wird.

Der Parameter RPYMSGQ

Der Parameter RPYMSGQ wird dazu verwendet, um die Nachrichtenwarteschlange zu bestimmen, die die Antwort einer Abfragenachricht aufnimmt. Die Standardeinstellung ist *WORKSTN, was bedeutet, dass die der Arbeitsstation zugeordnete Nachrichtenwarteschlange die Antwort empfängt. Dieser Wert kann durch Angabe des qualifizierten Namens einer beliebigen Nachrichtenwartschlange verändert werden.

2.12.7.5.1
Seite 2

Schauen wir uns hierzu ein kleines Programm an:

```
PGM PARM(&Q1)
    DCL &Q1   *CHAR 10
    DCL &USER *CHAR 10
START:
    CLROUTQ  OUTQ(&Q1)
    MONMSG  MSGID(CPF2207) EXEC(DO)  /*BERECHTIGUNG */
            RTVUSRPRF RTNUSRPRF(&USER)
            SNDMSG  MSG('Keine Berechtigung. Lassen Sie sich die +
               entsprechenden Berechtigungen erteilen')          +
               TOUSR(&USER)
            GOTO ENDE
            ENDDO
    MONMSG  MSGID(CPF3357) EXEC(DO)  /*NICHT GEFUNDEN */
            RTVUSRPRF RTNUSRPRF(&USER)
            SNDMSG  MSG('Objekt nicht gefunden!')  TOUSR(&USER)
            GOTO ENDE
            ENDDO
ENDE: ENDPGM
```

Das Programm empfängt den Parameter &Q1, der den Namen einer Ausgabewarteschlange enthält. Der Inhalt der Ausgabewarteschlange soll gelöscht werden. Wenn es zur Fehlersituation kommt, d.h. wenn die nötigen Berechtigungen fehlen oder das Objekt nicht gefunden wurde, erhält der Anwender eine entsprechende Fehlernachricht und das Programm wird beendet. Es ist allerdings nicht sichergestellt, dass der Anwender, der die Nachricht erhält, diese auch sofort liest. Wie wir bereits wissen, entscheidet der Empfänger, wann die Nachricht gelesen wird und wer weiß schon, wann das sein wird. Dieses Problem können wir umgehen, indem wir einen anderen Befehl für die Nachrichtenübermittlung einsetzen.

Der Befehl SNDBRKMSG

Der Befehl SNDBRKMSG arbeitet im Großen und Ganzen wie der Befehl SNDMSG mit dem Unterschied, dass die Nachricht im Durchbruchmodus gesendet wird, d.h. die Nachricht unterbricht den interaktiven Job und ersetzt die aktuelle Anzeige des Benutzers.

Der Parameter MSG

Der Parameter MSG enthält den Text der Durchbruchnachricht, der bis zu 512 Zeichen lang sein kann.

Der Parameter TOMGSQ

Der Parameter TOMGSQ gibt den Namen einer oder mehrerer Datenstationsnachrichtenwarteschlangen an, welche die Durchbruchnachricht empfangen sollen. In diesem Fall können nur Datenstationsnachrichtenwarteschlangen angegeben werden. Gültige Einträge sind eine oder mehrere Datenstationsnachrichtenwarteschlangen oder der Eintrag *ALLWS. Wenn Sie *ALLWS eingeben, so wird die Durchbruchnachricht an alle Datenstationswarteschlangen gesendet – unabhängig davon, ob diese gerade aktiv sind oder nicht. Die Verwendung von *ALLWS ist allerdings nur für Informationsnachrichten zulässig.

Der Parameter MSGTYPE

Der Parameter MSGTYPE bestimmt die Nachrichtenart. Der Defaultwert ist auch hier wieder *INFO (Informations-Nachrichten). Es ist ebenfalls möglich, Abfragenachrichten mit dem Befehl SNDBRKMSG zu senden.

Der Parameter RPYMSGQ

Der Parameter RPYMSGQ gibt die Nachrichtenwarteschlange an, wenn MSGTYPE(*INQ) definiert wurde. Der Unterlassungswert ist QSYSOPR, es kann aber jede permanente Nachrichtenwarteschlange verwendet werden.

Sicherlich würde der Einsatz des Befehls SNDBRKMSG unserem Programm zunächst nutzen. Wir wissen, dass der Anwender die Fehlernachricht erhält und auch sieht. Dafür müssten wir aber zunächst einmal den Namen der Datenstationsnachrichtenwarteschlange des Aufrufers ermitteln. Das ist mit dem Befehl RTVJOBA zwar einfach, muss aber trotz allem kodiert werden. Es stellt sich allerdings die Frage, ob unsere Informationsnachricht so wichtig ist, dass der Anwender in seiner augenblicklichen Arbeit vollständig unterbrochen wird. Ist es nicht vielleicht ausreichend die Meldung einfach in die Statuszeile des Bildschirms zu schreiben? Hinzu kommt, dass der Befehl SNDBRKMSG und der Befehl SNDMSG nur mit permanenten Warteschlangen arbeiten können, sobald wir temporäre Nachrichtenwarteschlangen oder andere Nachrichtenarten außer den Arten *INFO bzw. *INQ senden wollen,

3. Ergänzung 06/2005

2.12.7.5.2
Seite 2

benötigen wir andere CL-Befehle. Aus diesen Gründen werden in der CL-Programmierung meistens die Befehle SNDPGMMSG oder SNDUSRMSG eingesetzt. Beide Befehle können Sie nicht interaktiv nutzen.

Der Befehl SNDPGMMSG

2.12.7.5.3
Seite 1

Der Befehl SNDPGMMSG (Send Program Message) bietet die vielfältigste, aber auch die komplizierteste Form der Nachrichtenversendung. Mit dem Befehl SNDPGMMSG können Sie Nachrichten an permanente oder temporäre Nachrichtenwarteschlangen versenden. Ein Programm kann die Nachricht auch an die eigene Nachrichtenwarteschlange senden. Dieser Befehl kann sowohl Ausnahmenachrichten als auch jede andere Nachrichtenart senden und er kann frei formulierte Nachrichtentexte oder vordefinierte Nachrichten verwenden. Aufgrund des komplexen Aufbaus möchte ich an dieser Stelle der Vollständigkeit halber alle Parameter kurz erläutern, aber bitte lassen Sie sich nicht abschrecken. Wir schauen uns immer wieder kleine Beispiele an, die Ihnen zeigen werden, dass der Einsatz wesentlich einfacher ist als es zunächst erscheint.

```
                    Programmnachricht senden (SNDPGMMSG)

Auswahl eingeben und Eingabetaste drücken.

Marke  . . . . . . . . . . . . .                 _________
Nachrichtentext oder . . . . . . MSG        >   'Keine Berechtigung. Lassen Sie
sich die entsprechenden Berechtigungen erteilen'
_ ...
Nachrichten-ID . . . . . . . . . MSGID           __________
Nachrichtendatei . . . . . . . . MSGF            __________
  Bibliothek . . . . . . . . . .                   *LIBL

                                                                Weitere ...
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F10=Zusätzl. Parameter
F12=Abbrechen   F13=Verwendung der Anzeige   F24=Weitere Tasten
```

Bedienerführung des Befehls SNDPGMMSG Seite 1

Der Parameter MSG

Der Parameter MSG gibt den zu sendenden Nachrichtentext an. Es können maximal 3000 Zeichen angegeben werden. Wenn der Befehl allerdings über die Bedienerführung eingegeben wird, können maximal 512 Zeichen angegeben werden. Die Zeichenfolge muss in Hochkommata eingeschlossen werden, wenn Sonderzeichen (einschließlich Leerzeichen) verwendet werden. Dieser Parameter kann nicht angegeben wird, wenn die Parameter MSGID und MSGF verwendet werden. Wenn der Parameter MSG genutzt wird, können alle Nachrichtenarten gesendet werden mit Ausnahme der Arten *ESCAPE, *NOTIFY oder *STATUS.

3. Ergänzung 06/2005

2.12.7.5.3
Seite 2

Der Parameter MSGID

Der Parameter MSGID gibt die Nachrichten-ID einer Nachrichtenbeschreibung an. Wenn der Parameter MSGID verwendet wird, kann der Parameter MSG nicht genutzt werden. Vordefinierte Nachrichten können nicht für die Nachrichtenart *RQS verwendet werden.

Der Parameter MSGF

Der Parameter MSGF gibt die Nachrichtendatei an, die die Nachrichten-ID enthält. Dieser Parameter ist zwingend erforderlich, wenn der Parameter MSGID angegeben wurde.

```
                    Programmnachricht senden (SNDPGMMSG)

Auswahl eingeben und Eingabetaste drücken.

Nachrichtendatenfeldwerte  . . . MSGDTA          *NONE
```

Bedienerführung des Befehls SNDPGMMSG Seite 2

Der Parameter MSGDTA

Der Parameter MSGDTA enthält die Ersatzwerte für die variablen Bestandteile innerhalb des Nachrichtentextes, Sie erinnern sich an den Wert &1? Falls mehrere Ersatzwerte erforderlich sind, kann eine CL-Variable verwendet werden, die die Nachrichtendatenfeldwerte enthält.

```
                    Programmnachricht senden (SNDPGMMSG)

Auswahl eingeben und Eingabetaste drücken.

MSGQ des Aufrufstapeleintrags:   TOPGMQ
  Beziehung  . . . . . .                 *PRV
  Aufrufstapeleintrags-ID:
  Aufrufstapeleintrag  . . . . .         *

  Modul  . . . . . .                     *NONE
  Binderprogramm . . . . . .             *NONE
An Nicht-Programm-MSGQ senden  . TOMSGQ
  Bibliothek . . . . . .
              + für weitere Werte

An Benutzerprofil  . . . . . . . TOUSR     >
Nachrichtenart . . . . . . . . . MSGTYPE       *INFO

                                                                  Weitere ...
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F10=Zusätzl. Parameter
F12=Abbrechen   F13=Verwendung der Anzeige   F24=Weitere Tasten
```

Diese Parameter schließen sich gegenseitig aus!

Bedienerführung des Befehls SNDPGMMSG Seite 3

Der Parameter TOPGMQ

Der Parameter TOPGMQ adressiert ausschließlich temporäre Nachrichtenwarteschlangen. Der Parameter besteht aus zwei Elementen. Das erste Element bestimmt zunächst die Beziehung zum 2. Element.

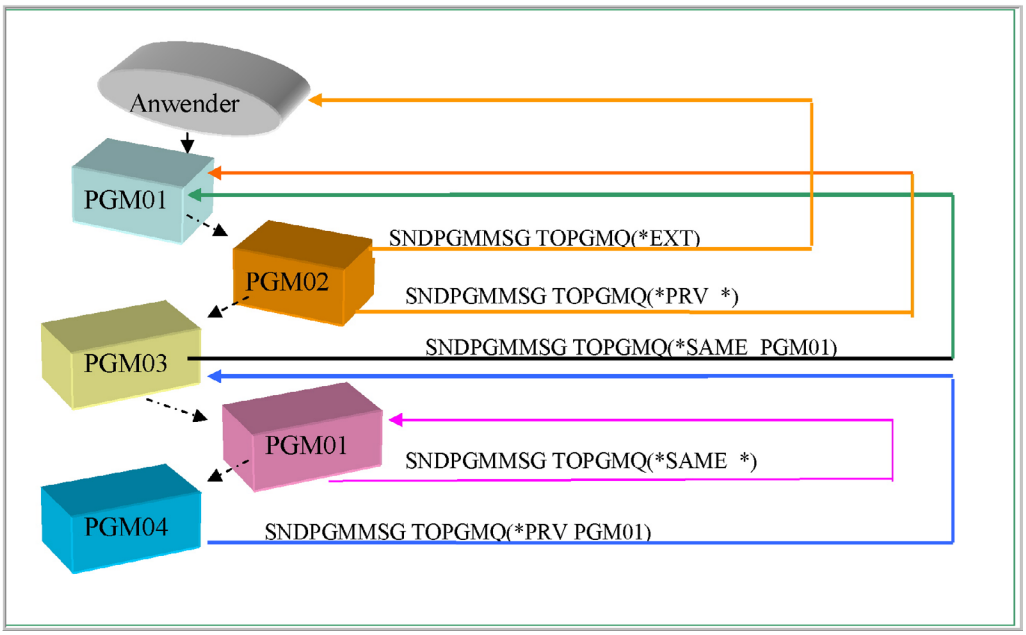
- *PRV bezeichnet den Aufrufer, d.h. den Vorgänger des 2. Elements.
- *SAME dagegen adressiert das im 2. Element benannte Programm.
- *EXT benennt die externe Jobnachrichtenwarteschlange.

Der Sonderwert * im 2. Element kennzeichnet stets das aktuelle Programm, andernfalls muss der Name eines aktiven Programms oder einer Prozedur angegeben werden.

TOPGMQ		
1. Element	2. Element	
*PRV	*	Standardwert; die Nachricht wird an das Programm bzw. die Prozedur gesendet, die das Programm aufgerufen hat, das die Nachricht sendet.
*PRV	Programm-/ Prozedurname	Die Nachricht erhält das Programm, das das im zweiten Element benannte Programm aufgerufen hat.
*SAME	*	Die Nachricht erhält das aktuelle Programm/Prozedur.
*SAME	Programm-/ Prozedurname	Die Nachricht erhält das im 2. Element benannte Programm.
*EXT	*	Die Nachricht wird an die externe Jobnachrichtenwarteschlange gesendet.

2.12.7.5.3
Seite 4

Das folgende Schaubild soll den Parameter TOPGMQ weiter verdeutlichen:

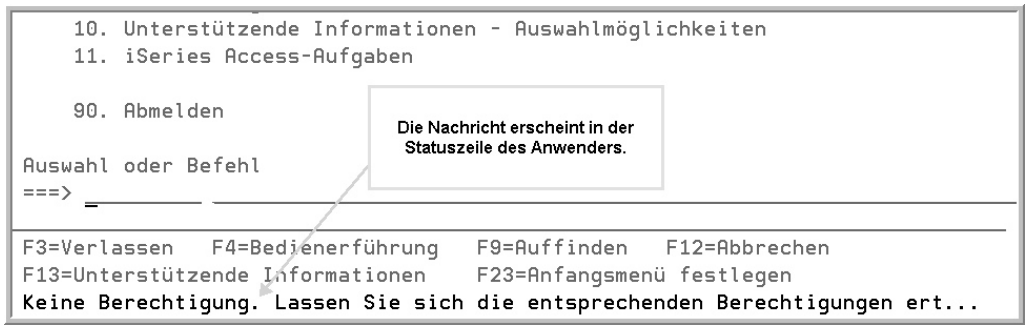


Wirkungsweise des Parameters TOPGMQ

Erinnern Sie sich noch an unser Ausgangsprogramm? Ich habe den ursprünglichen Befehl SNDMSG ausgetauscht gegen den SNDPGMMSG Befehl, der die Informationsnachricht an die Statuszeile des Aufrufers sendet. Für diese frei formulierte Nachricht ist lediglich der Parameter MSG und der Parameter TOPGMQ notwendig, alle weiteren Parameter enthalten Defaultwerte.

```
     CLROUTQ  OUTQ(&Q1)
     MONMSG   MSGID(CPF2207) EXEC(DO)  /*BERECHTIGUNG */
              SNDPGMMSG  MSG('Keine Berechtigung. Lassen Sie sich die +
                   entsprechenden Berechtigungen erteilen!')          +
                   TOPGMQ(*EXT)
              GOTO ENDE
              ENDDO
```

SNDPGMMSG sendet die Informationsnachricht an die Statuszeile des aktuellen Bildschirms. Es ist nicht mehr erforderlich den Namen des Benutzers oder den der Datenstation zu ermitteln. Die Wirkung können Sie in der folgenden Abbildung sehen:



Da der Nachrichtentext mehr als 76 Zeichen enthält, muss der Anwender den Cursor in die Statuszeile positionieren und anschließend die Funktionstaste F1 drücken, um den vollständigen Nachrichtentext zu lesen.

Dieselbe Nachricht können wir aber auch mit Hilfe unserer Nachrichtendatei senden:

```
CLROUTQ  OUTQ(&Q1)
MONMSG   MSGID(CPF2207) EXEC(DO)  /*BERECHTIGUNG */
          SNDPGMMSG  MSGID(MSG0001) MSGF(MESSAGE)  +
          MSGDTA(&Q1)  TOPGMQ(*EXT)
          GOTO ENDE
     ENDDO
```

In diesem Fall benutzen wir die Parameter MSGID, MSGF und MSGDTA. Die Parameter MSGID und MSGF bestimmen, welche Nachricht welcher Nachrichtendatei gesendet wird und der Parameter MSGDTA enthält den Ersatzwert für die im Nachrichtentext enthaltene Variable &1. Die Wirkungsweise des Befehls ist zunächst identisch. Allerdings hat der Anwender jetzt die Möglichkeit sich den erweiterten Nachrichtentext (Nachrichtentext der 2. Ebene) anzeigen zu lassen.

```
                        Weitere Nachrichteninformationen

Nachrichten-ID . . . . :   MSG0001
Sendedatum . . . . . . :   24.03.05      Sendezeit  . . . . . :   13:05:52

Nachricht . . . :   Sie haben keine Berechtigung für die Ausgabewarteschlange
  TEST.

Zum Löschen der Spoolfiles in der Ausgabewarteschlange TEST benötigen Sie das
  *CHANGE Rechte am Objekt TEST. Lassen Sie sich die nötigen Rechte von Ihrem
  Administrator erteilen.









                                                                        Ende
Eingabetaste --> Weiter

F1=Hilfetext   F3=Verlassen   F6=Drucken   F9=Nachrichtendetails anzeigen
F10=Nachrichten im Jobprotokoll anzeigen   F12=Abbrechen   F24=Weitere Tasten
                                _
```

Erweiterter Nachrichtentext

Der Parameter TOMSGQ

Der Parameter TOMSGQ adressiert permanente Nachrichtenwarteschlangen. Es ist möglich die Parameter TOPGMQ und TOMSGQ parallel zu nutzen. Der Parameter kann nicht verwendet werden, wenn der Parameter TOUSR verwendet wird. Es können bis zu 50 permanente Nachrichtenwarteschlangen eingetragen werden. Der Sonderwert *SYSOPR kennzeichnet die Nachrichtenwarteschlange des Systemoperators. Durch die Verwendung des Sonderwertes *HSTLOG können Sie *DIAG oder *INFO-Nachrichten in das Systemprotokoll schreiben. Falls Sie die Sonderwerte *HSTLOG und *SYSOPR gleichzeitig angeben, wird nur eine Nachricht im Systemprotokoll erstellt. Beachten Sie aber bitte, dass in das Systemprotokoll nur wichtige Mitteilungen geschrieben werden sollten.

```
CLROUTQ  OUTQ(&Q1)
MONMSG  MSGID(CPF2207) EXEC(DO)  /*BERECHTIGUNG */
        SNDPGMMSG  MSG('Keine Berechtigung. Lassen Sie sich die +
              entsprechenden Berechtigungen erteilen') TOPGMQ(*EXT)
        GOTO ENDE
        ENDDO
SNDPGMMSG  MSG('Die Ausgabewarteschlange' *BCAT &Q1 *BCAT ,wurde +
          bereinigt!') TOPGMQ(*EXT)  TOMSGQ(*SYSOPR)
```

Hier wurde das Programm dahingehend erweitert, das der Anwender und der Systembediener eine Informationsnachricht über die Ausführung des Befehls CLROUTQ erhält.

Der Parameter TOUSR

Der Parameter TOUSR adressiert die Nachrichtenwarteschlange, die im Benutzerprofil des Benutzers angegeben ist. Der Parameter kann nicht angegeben werden, wenn der Parameter TOMSGQ oder TOPGMQ angegeben wird. Es können die folgenden Sonderwerte verwendet werden:

- *SYSOPR Sendet eine Nachricht an den Systembediener.
- *REQUESTER Sendet die Nachricht an den aktuellen Benutzer bei einem interaktiven Job. Wenn es ein Batch-Job ist, erhält der Systemoperator die Nachricht.
- *ALLACT Die Nachricht wird an die Nachrichtenwarteschlange jedes Benutzerprofils gesendet, für das momentan ein interaktiver Job läuft. *ALLACT kann bei Anfragenachrichten (*INQ) nicht angegeben werden.

Der Parameter MSGTYPE

Der Parameter MSGTYPE bestimmt, welche Nachrichtenart dieser Nachricht zugeordnet werden soll. Die Standardeinstellung ist *INFO (Informationsnachricht). Eine Übersicht aller Nachrichtenarten und deren Bedeutung finden Sie im Abschnitt 2.12.7.3, so dass ich an dieser Stelle auf eine erneute Darstellung verzichten werde. Beachten Sie lediglich, dass Abfragenachrichten (*INQ) nur an die externe Nachrichtenwarteschlange (*EXT) oder an eine im Parameter TOUSR oder TOMSGQ angegebene Nachrichtenwarteschlange gesendet werden können. Falls Sie sich dann für den Parameter TOMSGQ entscheiden, kann als zweite Warteschlange *HSTLOG verwendet werden. Beendigungs- (*COMP), Diagnose- (*DIAG), Abbruch- (*ESCAPE), Hinweis- (*NOTIFY) und Statusnachrichten (*Status) können nur gesendet werden, wenn Sie den Parameter TOPGMQ für die Adressierung verwenden. Dabei ist weiterhin zu berücksichtigen, dass Sie Abbruchnachrichten (*ESCAPE) nicht an die externe Jobnachrichtenwarteschlange schicken können.

Bislang haben wir nur Informationsnachrichten geschickt. Wie wäre es mit einer Status-Nachricht? Kennen Ihre Anwender oder Sie auch das ungute Gefühl, das entsteht, wenn ein Programm interaktiv läuft und läuft und läuft….und man nicht so recht weiß, was gerade passiert? In diesen Fällen bieten sich Statusmeldungen an.

Statusnachrichten erfordern die Angabe MSGID/MSGF/MSGDTA und erlauben daher keine frei formulierten Texte. Das Problem kann umgangen wer-

2.12.7.5.3
Seite 8

den, indem Sie die MSGID CPF9898 der IBM Nachrichtendatei QCPFMSG verwenden. Der Nachrichtentext der Nachricht CPF9898 besteht aus einer einzigen Ersatzvariablen.

```
PGM
                :
    SNDPGMMSG MSGID(CPF9898) MSGF(*LIBL/QCPFMSG) +
    MSGDTA('Die Kundendaten werden kopiert! Einen Moment bitte..')  +
    TOPGMQ(*EXT) MSGTYPE(*STATUS)
                :
    SNDPGMMSG MSGID(CPF9898) MSGF(*LIBL/QCPFMSG) +
    MSGDTA('Die Artikel werden geladen! Einen Moment bitte..')  +
    TOPGMQ(*EXT) MSGTYPE(*STATUS)
                :
```

Die Kombination von TOPGMQ(*EXT) MSGTYPE(*STATUS) sendet eine Echtzeitnachricht an die Arbeitsstation, die Sie lesen können, während das Programm ausgeführt wird. Das System sendet die Nachricht: „Die Kundendaten werden kopiert! Einen Moment bitte…“ in die Statuszeile Ihres Bildschirms.

```
                    Programmnachricht senden (SNDPGMMSG)

Auswahl eingeben und Eingabetaste drücken.

Antwort in Nachr.warteschlange   RPYMSGQ        *PGMQ
  Bibliothek . . . . . . . . . .                  __________
CL-Var. für KEYVAR       (4) . . KEYVAR          __________
Kommentar  . . . . . . . . . . .                 ______________________________
_____________________________________ ______________________________________
_____________________________________________________________________________
_____________________________________________________________________________

                                                                    Ende
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F10=Zusätzl. Parameter
F12=Abbrechen   F13=Verwendung der Anzeige   F24=Weitere Tasten
```

Bedienerführung des Befehls SNDPGMMSG Seite 4

Der Parameter RPYMSGQ

Wenn Sie eine Abfragenachricht (*INQ) senden, soll der Empfänger auf diese Nachricht antworten. Die Antwort muss „irgendwo“ empfangen werden. Der Parameter RPYMSGQ gibt an, wo die Antwort empfangen wird. Der Unterlassungswert ist *PGMQ. Das aktuelle Programm empfängt also die Nachricht direkt in der eigenen Programmnachrichtenwarteschlange. Es ist aber auch möglich den qualifizierten Namen einer Nachrichtenwarteschlange einzutragen. In beiden Fällen benötigen wir anschließend den Befehl RCVMSG, um die Nachricht verarbeiten zu können.

Der Parameter KEYVAR

Jeder Nachricht wird ein Nachrichtenreferenzschlüssel vom System zugeordnet, wenn die Nachricht an eine Programmnachrichtenwarteschlange gesendet wird. Für Nachrichten an permanente Nachrichtenwarteschlangen sind Nachrichtenreferenzschlüssel nur für Anfragenachrichten (*INQ) verfügbar. Wenn eine andere Art von Nachricht an eine permanente Nachrichtenwarteschlange gesendet wird ist kein Nachrichtenschlüssel verfügbar und für KEYVAR wird kein Wert zurückgegeben. Die CL-Variable, die den Nachrichtenreferenzschlüssel aufnehmen soll, muss eine 4 Byte lange Zeichenvariable sein.

Wir haben es geschafft! Alle Parameter des Befehls SNDPGMMSG sind erklärt. Schauen wir uns zum Schluss noch mal unser Beispielprogramm an, um einen sinnvollen Einsatz für die Parameter RPYMSGQ und KEYVAR zu finden.

Wir könnten z.B. in unserem Programm eine Sicherheitsabfrage einbauen, bevor wir den Inhalt der Ausgabewarteschlange löschen. Ja, ich mag auch keine Programme in denen laufend gefragt wird „Sind Sie wirklich sicher, dass Sie löschen wollen?“, aber um die Wirkungsweise des Parameters RPYMSGQ zu verstehen, bietet sich diese Programmlösung an:

```
SNDPGMMSG  MSG('Wollen Sie den Inhalt der Ausgabewarteschlange' +
           *BCAT &Q1 *BCAT  'löschen? [J/N]') TOPGMQ(*EXT)   +
           MSGTYPE(*INQ)   RPYMSGQ(*PGMQ)
```

2.12.7.5.3
Seite 10

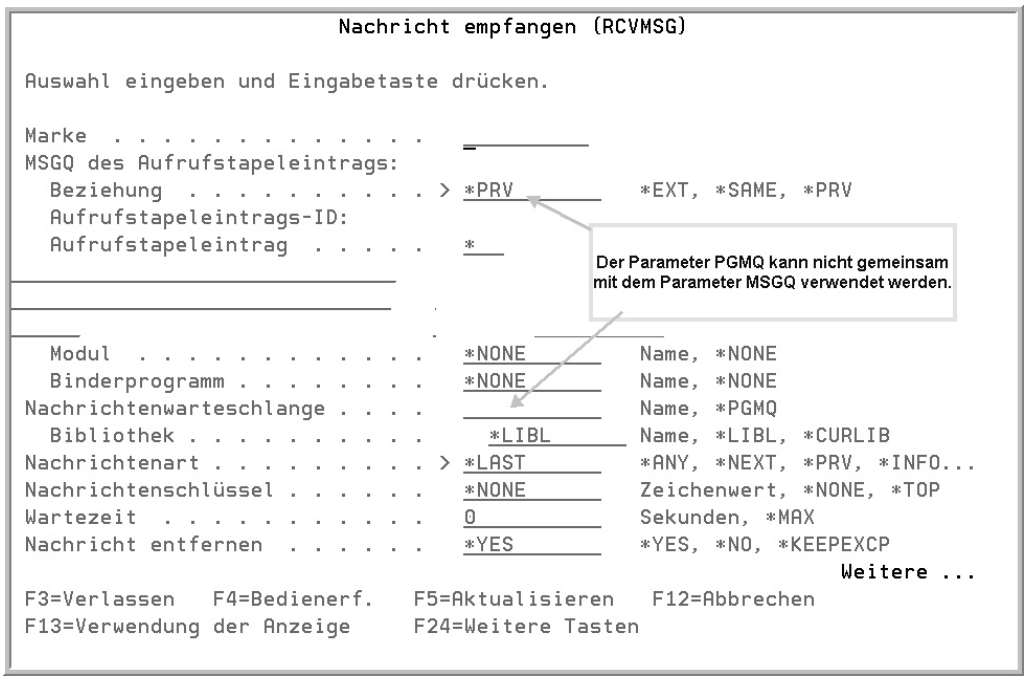
SNDPGMMSG sendet die Anfragenachricht an *EXT, weil eine Kommunikation mit dem Benutzer ausgeführt werden soll. Die Nachrichtenwarteschlange des CL-Programms empfängt die Antwort, die für RPYMSGQ(*PGMQ) angegeben ist. Dies ist die Defaulteinstellung des Parameters RPYMSGQ, so dass Sie auf die Angabe des Parameters auch verzichten könnten. Es wäre aber auch denkbar eine andere Nachrichtenwarteschlange im Befehl SNDPGMMSG anzusprechen.

```
SNDPGMMSG  MSG('Wollen Sie den Inhalt der Ausgabewarteschlange' +
           *BCAT &Q1 *BCAT  'löschen? [J/N]') TOPGMQ(*EXT)   +
           MSGTYPE(*INQ)   RPYMSGQ(*EXT)
```

Der Befehl RCVMSG

2.12.7.5.4
Seite 1

Wie wir in den vorhergehenden Abschnitten gesehen haben, können Nachrichten mit dem Befehl „SNDPGMMSG“ an CL-Programm-Nachrichtenwarteschlangen gesendet werden. Das empfangende Programm muss den Befehl „Nachrichten empfangen“ (RCVMSG) ausführen, um die Antwort zu verarbeiten. Schauen wir uns auch diesen Befehl mit seinen Parametern etwas genauer an:



Bedienerführung des Befehls RCVMSG Seite 1

Der Parameter PGMQ

Zuerst muss festgelegt werden, ob RCVMSG die Nachricht von einer Programmnachrichtenwarteschlange oder einem permanenten Nachrichtenwarteschlangenobjekt empfangen soll. Dies kann durch Angabe des Parameters PGMQ erfolgen. Verwenden Sie die Angabe PGMQ, um eine Programmnachrichtenwarteschlange für den Empfang der Nachrichten zu definieren. Der Parameter PGMQ ist identisch mit dem Parameter TOMSGQ bei der Verwendung des Befehls „SNDPGMMSG“. Hier einige mögliche Kombinationen:

2.12.7.5.4
Seite 2

1.Element	2. Element	Beschreibung
*SAME	*	Defaulteinstellung. Empfängt die Nachricht von der Nachrichtenwarteschlange des aktuellen Programms oder der aktuellen Prozedur.
*SAME	Programm-/ Prozedurname	Empfängt die Nachricht von der Nachrichtenwarteschlange des angegebenen Programms oder der angegebenen Prozedur.
*PRV	*	Empfängt die Nachricht von der Nachrichtenwarteschlange von dem Programm oder der Prozedur, welche das aktuelle Programm aufgerufen hat.
*PRV	Programm-/ Prozedurname	Empfängt die Nachricht von dem Programm oder der Prozedur, die das benannte Programm oder die benannte Prozedur aufgerufen hat.
*EXT	*	Empfängt die Nachricht von der externen Nachrichtenwarteschlange. Der zweite Wert (*) kann weggelassen werden.

Der Parameter MSGQ

Der Parameter MSGQ kann verwendet werden, um eine permanente Nachrichtenwarteschlange qualifiziert anzusprechen. Die Parameter PGMQ und MSGQ können nicht gemeinsam verwendet werden.

Der Parameter MSGTYPE

Der Parameter MSGTYPE spezifiziert den Nachrichtentyp, der empfangen werden soll. Ich verzichte an dieser Stelle auf die Erläuterung aller Sonderwerte und stelle Ihnen lediglich die wichtigsten vor:

*ANY Jede Nachrichtenart, außer einer Nachrichtenkopie wird nach dem Prinzip „First-In-First-Out“ empfangen. Dies ist die Standardeinstellung.

*RPY Eine Antwort auf eine Abfragenachricht soll empfangen werden.

*FIRST Es soll die erste, d.h. die älteste Nachricht in der Nachrichtenwarteschlange empfangen werden.

*EXCP Eine Ausnahmebedingungsnachricht wird empfangen. Diese Art von Nachrichten (Abbruch-, Hinweis- und Statusnachrichten) werden nach dem Prinzip „Last-In-First-Out“ empfangen. Nachrichten, die keine Ausnahmebe dingungsnachrichten sind, werden nach dem Prinzip „First-In-First-Out“ empfangen.

*LAST Es wird die letzte, d.h. die jüngste Nachricht empfangen.

Der Parameter Nachrichtenschlüssel

Der Parameter Nachrichtenschlüssel erlaubt den Zugriff auf eine bestimmte Nachricht ähnlich dem wahlfreien Zugriff auf indexierte Tabellen. Der Standardwert dieses Parameters ist *NONE, d.h. die Nachrichten werden nach dem First-In-First-Out-Prinzip verarbeitet.

Der Parameter WAIT

Der Parameter WAIT gibt die Zeitspanne in Sekunden an, die das Programm auf das Eintreffen der Nachricht wartet. Der Befehl RCVMSG kann vorzeitig abgesetzt werden, auch wenn noch keine Nachricht vorhanden ist. Der CL-Befehl unterbricht dann das Programm für die hier angegebene Zeit oder bis die Nachricht eintrifft. Wenn innerhalb der angegebenen Zeit keine Nachricht eintrifft, werden die CL-Variablen, die die Nachricht empfangen sollten, mit Leerstellen aufgefüllt. Der Unterlassungswert für diesen Parameter ist der Wert 0. Wenn, wie in unserem letzten Programmbeispiel, der für MSGKEY angegebene Wert auf eine Abfragenachricht verweist und MSGTYPE(*RPY) angegeben wurde, ignoriert das Programm den Parameter WAIT.

Der Parameter RMV

Der Parameter RMV gibt an, ob die empfangene Nachricht aus der Nachrichtenwarteschlange entfernt werden soll. Gültige Werte für diesen Parameter sind *YES, *NO und *KEEPEXCP. Der Standardwert ist *YES, so dass alle Nachrichten nach dem Empfang aus der Nachrichtenwarteschlange gelöscht werden. Wenn Sie *NO verwenden, werden die Nachrichten als alte Nachrichten in der Nachrichtenwarteschlange stehen bleiben. Wenn es sich bei der Nachricht um eine Ausnahmenachricht handelt, werden die Nachrichten in beiden Fällen vom System nach dem Empfang der Nachricht behandelt. Der Wert *KEEPEXCP ist nur bei Ausnahmenachrichten wirksam. Er verhindert die automatische Fehlerbehandlung und belässt lediglich unbehandelte Ausnahmenachrichten als neue Nachrichten in der Warteschlange. Behandelte Nachrichten bleiben dagegen als alte Nachricht in der Nachrichtenwarteschlange.

2.12.7.5.4
Seite 4

```
                        Nachricht empfangen (RCVMSG)

Auswahl eingeben und Eingabetaste drücken.

Zeichensatz-ID . . . . . .   *JOB        *JOB, *HEX...
Standardantwort zurückweisen .  *NOALWRJT   *NOALWRJT, *ALWRJT
CL-Var. für KEYVAR       (4) . .           Zeichenwert
CL-Var. für Text der 1. Ebene  . > &WERT    Zeichenwert
CL-Var. für MSGLEN     (5 0) . .           Zahl
CL-Var. für Text der 2. Ebene  .           Zeichenwert
CL-Var. für SECLVLLEN  (5 0) . .           Zahl
CL-Var. für Nachrichtendaten . .           Zeichenwert
CL-Var. für MSGDTALEN  (5 0) . .           Zahl
CL-Var. für MSGID        (7) . .           Zeichenwert
CL-Var. für SEV        (2 0) . .           Zahl
CL-Var. für SENDER      (80) . .           Zeichenwert
Senderformat . . . . . .   *SHORT        *SHORT, *LONG
CL-Var. für RTNTYPE      (2) . .           Zeichenwert
CL-Var. für ALROPT       (9) . .           Zeichenwert
CL-Var. für  MSGF      (10)  . .           Zeichenwert
                                                        Weitere ...
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Bedienerführung des Befehls RCVMSG Seite 2

Wenn eine Nachricht empfangen wird, werden die Informationen dieser Nachricht in CL-Variablen zur Verfügung gestellt. Die Bedeutung der einzelnen Informationen sollen im Weiteren kurz erläutert werden:

KEYVAR

KEYVAR enthält den vom System generierten Nachrichtenschlüssel.

MSG (Text der ersten Ebene)

MSG (Text der ersten Ebene) enthält den Nachrichtentext der ersten Ebene. Die CL-Variable ist mit dem TYPE(*CHAR) zu definieren und kann mit einer beliebigen Länge angegeben werden. Die meisten Nachrichtentexte der ersten Ebene haben eine Länge kleiner 132 Zeichen.

MSGLEN

MSGLEN enthält die tatsächliche Länge des Textes der ersten Nachrichtenebene. Diese Variable muss als TYPE(*DEC) LEN(5 0) definiert werden.

SECLVL (Text der 2. Ebene)

SECLVL (Text der 2. Ebene) enthält den erweiterten Nachrichtentext. Diesen Nachrichtentext erhalten Sie stets, wenn Sie den Cursor auf eine Fehlermeldung positionieren und anschließend die Funktionstaste F1 drücken. Die Variable, die den Text empfangen soll, muss als TYPE(*CHAR) und einer beliebigen Länge definiert werden. In der Regel werden Nachrichtentexte der zweiten Ebene mit einer Länge von 3000 Zeichen oder weniger definiert.

SECLVLLEN

SECLVLLEN gibt die exakte Länge (in Bytes) des Nachrichtentextes der zweiten Ebene an. Diese CL-Variable muss als TYPE(*DEC) LEN(5 0) angegeben werden.

MSGDTA

MSGDTA enthält den Nachrichtentext für die eingebetteten Variablen. Diese muss als TYPE (*CHAR) mit einer beliebigen Länge angegeben werden.

MSGDTALEN

MSGDTALEN enthält die Länge der eingebetteten Ersatzwerte in Bytes. Die Variable muss als TYPE(*DEC) LEN(5 0) definiert werden.

MSGID

MSGID enthält die Nachrichten-ID der zu empfangenden Nachricht. Diese CL-Variable ist mit TYPE(*CHAR) LEN(7) zu definieren. Wenn eine Nachricht empfangen wird, deren Nachrichtentext frei eingegeben wurde, dann ist dieser Wert mit Leerstellen initialisiert.

SEV

SEV enthält die Nachrichtenbewertung. Eine CL-Variable, die diese Information empfangen soll, muss mit TYPE(*DEC) LEN(2 0) definiert werden. Frei eingegebene Nachrichtentexte erhalten die Bewertungsstufe „00“.

SENDER

SENDER muss mit TYPE(*CHAR) deklariert werden. Wenn der Parameter SENDERFMT(*SHORT) angegeben wird, muss die Variable mindestens 80 Bytes lang sein. Ist der Parameter SENDERFMT(*LONG) angegeben, dann muss die Länge der Variablen mindestens 720 Bytes betragen.

2.12.7.5.4
Seite 6

Folgende Informationen enthält der Parameter SENDER (Kurzformat):

Von	Bis	Beschreibung
1	10	Jobname
11	20	Benutzer
21	26	Jobnummer
27	38	Programm-Name. Bei der Verwendung von ILE wird der Name des Binderprogramms verwendet.
39	42	Anweisungs-Nummer. Für ILE-Prozeduren Blank!
43	49	Datum im Format CYYMMDD
50	55	Zeit im HHMMSS Format
56	65	Name des empfangenden Programms. Bei ILE-Prozeduren: Name des Binderprogramms.
66	69	Anweisungs-Nummer. Für ILE-Prozeduren Blank.
70	70	"0": Sender ist ein Programm "1": Sender ist eine ILE-Prozedur.
72	80	FREI

RTNTYPE

RTNTYPE bestimmt die Art der Nachricht.

Wert	Nachrichtenart
01	Beendigungsnachricht
02	Diagnosenachricht
04	Informationsnachricht
05	Anfragenachricht
06	Kopie
08	Anforderungsnachricht
10	Anforderung mit Bedienerführung
14	Hinweisnachricht, die zum Zeitpunkt von RCVMSG noch nicht bearbeitet ist.
15	Abbruchnachricht, die zum Zeitpunkt von RCVMSG bereits bearbeitet wurde.
16	Hinweisnachricht, die zum Zeitpunkt von RCVMSG noch nicht bearbeitet ist.
17	Abbruchnachricht, die zum Zeitpunkt von RCVMSG noch nicht bearbeitet ist.
21	Antwort, die nicht auf Gültigkeit überprüft ist.
22	Antwort, die bereits auf Gültigkeit überprüft ist.
23	Standardantwort

Wert	Nachrichtenart
24	Standardantwort des Systems
25	Antwort aus der Systemantwortliste

Bedienerführung des Befehls RCVMSG 3. Seite

MSGFLIB

MSGFLIB enthält den Bibliotheksnamen der Nachrichtendatei, wenn die Nachricht eine gespeicherte Nachricht ist. Wurde im Sendebefehl die Nachrichtendatei durch den Sonderwert *LIBL adressiert, enthält dieser Parameter auch den Wert *LIBL. Falls die empfangene Nachricht keine vordefinierte Nachricht ist, werden Leerzeichen zurückgegeben. Die hier zu verwendende Variable muss mit dem TYP(*CHAR 10) definiert werden.

```
                         Nachricht empfangen (RCVMSG)

Auswahl eingeben und Eingabetaste drücken.

CL-Var. für MSGFLIB    (10)  . .   ___________   Zeichenwert
CL-Var. für SNDMSGFLIB (10)  . .   ___________   Zeichenwert
CL-Var. für Text-CCSID (5 0) . .   ___________   Zahl
CL-Var. für Daten-CCSID (5 0)  .   ___________   Zahl
Kommentar  . . . . . . . . . . .   ______________________________________________
_______________________________________________________________________________
_______________________________________________________________________________
_______________________________________________________________________________

                                                                        Ende
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

SNDMSGFLIB

SNDMSGFLIB enthält den tatsächlichen Namen der Bibliothek, wenn im Sendebefehl *LIBL angegeben wurde. Die Variable, die die Information aufnehmen soll, muss mit dem Typ(*CHAR 10) definiert werden.

Was fangen wir jetzt mit diesen Informationen an? Schauen wir uns zunächst ein paar einfache Beispiele an.

```
RCVMSG   MSGQ(RASCHE)  MSGKEY(&KEY)  MSG(&WORK)
```

Dieser Befehl empfängt eine Nachricht aus der permanenten Nachrichtenwarteschlange „RASCHE“, wenn die Nachrichtensignatur dem Inhalt der Variablen &KEY entspricht. Dabei ist es egal, ob die Nachricht bereits gelesen

2.12.7.5.4
Seite 8

wurde, also alt ist oder ob sie neu ist. Auf das Eintreffen der Nachricht wird nicht gewartet. Falls die Nachricht vorhanden ist, steht der Nachrichtentext der ersten Ebene dem Programm in der Variablen &WORK zur Verfügung.

```
RCVMSG   MSGQ(RASCHE)  WAIT(120)  MSG(&WORK)
```

Mit diesem Befehl wird eine neue Nachricht aus der Nachrichtenwarteschlange RASCHE in die CL-Variable &WORK kopiert. Das Programm wartet maximal 120 Sekunden auf die Ankunft einer neuen Nachricht, wenn sich keine neuen Nachrichten in der Nachrichtenwarteschlange befinden. Enthält die Warteschlange mehrere neue Nachrichten, empfängt das Programm die erste Nachricht in der Warteschlange.

Schauen wir uns jetzt noch einmal unseren SNDPGMMSG-Befehl des vorherigen Beispiels an:

```
SNDPGMMSG  MSG('Wollen Sie den Inhalt der Ausgabewarteschlange' +
           *BCAT &Q1 *BCAT  'löschen? [J/N]') TOPGMQ(*EXT)      +
           MSGTYPE(*INQ)   RPYMSGQ(*PGMQ)
```

SNDPGMMSG sendet die Anfragenachricht an *EXT, weil eine Kommunikation mit dem Benutzer ausgeführt werden soll. Die Nachrichtenwarteschlange des CL-Programms empfängt die Antwort, da für RPYMSGQ(*PGMQ) angegeben ist. Der dann folgende Befehl RCVMSG muss die Antwort aus der entsprechenden *PGMQ empfangen.

```
RCVMSG     MSGTYPE(*RPY) MSG(&WERT)
CASE:      SELECT
           WHEN  COND(&WERT = 'J' *OR &WERT = 'j') THEN(DO)
           CLROUTQ OUTQ(&Q1)
           SNDPGMMSG  MSG('Inhalt der Nachrichtenwarteschlange    +
           wurde gelöscht!') +
           TOPGMQ(*EXT)
           ENDDO
                      :
```

Der Text der Antwort, in unserem Fall hoffentlich ein „J“ oder ein „N“, wird in die CL-Variable &WERT gestellt und kann ausgewertet werden.

Es wäre aber auch denkbar eine andere Nachrichtenwarteschlange im Befehl SNDPGMMSG anzusprechen. Nur dann muss der nachfolgende RCVMSG-Befehl auch darauf reagieren.

```
SNDPGMMSG  MSG('Wollen Sie den Inhalt der Ausgabewarteschlange' +
           *BCAT &Q1 *BCAT  'löschen? [J/N]') TOPGMQ(*EXT)      +
           MSGTYPE(*INQ)   RPYMSGQ(*EXT)
RCVMSG     PGMQ(*EXT) MSGTYPE(*RPY) MSG(&WERT)
```

Ein letztes Problem bleibt aber noch zu klären: Woher wissen wir, dass der empfangene Antwortwert korrekt ist? Vielleicht hat ein anderer Benutzer die Antwort gesendet? Diese letzte Unsicherheit können wir durch den Parameter KEYVAR lösen.

```
PGM PARM(&Q1)
          DCL &KEY     *CHAR     4
                      :
SNDPGMMSG  MSG('Wollen Sie den Inhalt der Ausgabewarteschlange' +
           *BCAT &Q1 *BCAT 'löschen? [J/N]')                    +
           TOPGMQ(*EXT)  MSGTYPE(*INQ)  RPYMSGQ(*PGMQ)         +
           KEYVAR(&KEY)
RCVMSG MSGTYPE(*RPY)   MSGKEY(&KEY)  MSG(&WERT)
```

Jede Nachricht, die in Programmnachrichtenwarteschlangen gesendet wird, wird vom System mit einem 4 Byte langen Schlüssel signiert. Die Variable &KEY in dem Befehl SNDPGMMSG wird dazu verwendet, die Signatur der soeben gesendeten Nachricht aufzunehmen. Da die iSeries sowohl der Abfragenachricht (*INQ) als auch der Antwort (*RPY) denselben Schlüssel zuweist, können wir mit der nachfolgenden RCVMSG-Anweisung so zu sagen „wahlfrei“ auf die Programmnachrichtenwarteschlange zugreifen und uns genau die Antwort beschaffen, die aufgrund unserer Frage gesendet wurde. Verwechslungen sind somit ausgeschlossen.

Aber nicht immer ist zuerst ein SNDPGMMSG Befehl nötig, um eine Nachricht empfangen zu können. Stellen Sie sich vor, Ihr Programm verursacht einen Ausnahmefehler und stürzt ab. Sie haben jetzt verschiedene Möglichkeiten zu reagieren:

Sie versuchen alle Ausnahmefehler individuell im Programm abzufangen. Aber wer will das schaffen bei der Vielzahl der möglichen Fehler. Also werden Sie für die nicht unmittelbar vorhersehbaren und behebbaren Fehler einen globalen MONMSG-Befehl einsetzen.

2.12.7.5.4
Seite 10

```
GLOBAL:    MONMSG MSGID(CPF0000) EXEC(GOTO CMDLBL(ERRMSG))
                      :
                      :
/* STANDARDFEHLERROUTINE  */
ERRMSG:    SNDPGMMSG MSG('Es ist ein Fehler aufgetreten!')
TOPGMQ(*EXT)
ENDE:  ENDPGM
```

Ihr Programm stürzt jetzt zwar nicht mehr ab, aber Sie kennen auch nicht die Ursache des Fehlers und wenn ein Anwender das Programm aufgerufen hat, erfahren Sie von der Fehlermeldung vielleicht gar nichts, weil er „es einfach morgen noch mal probieren wird".

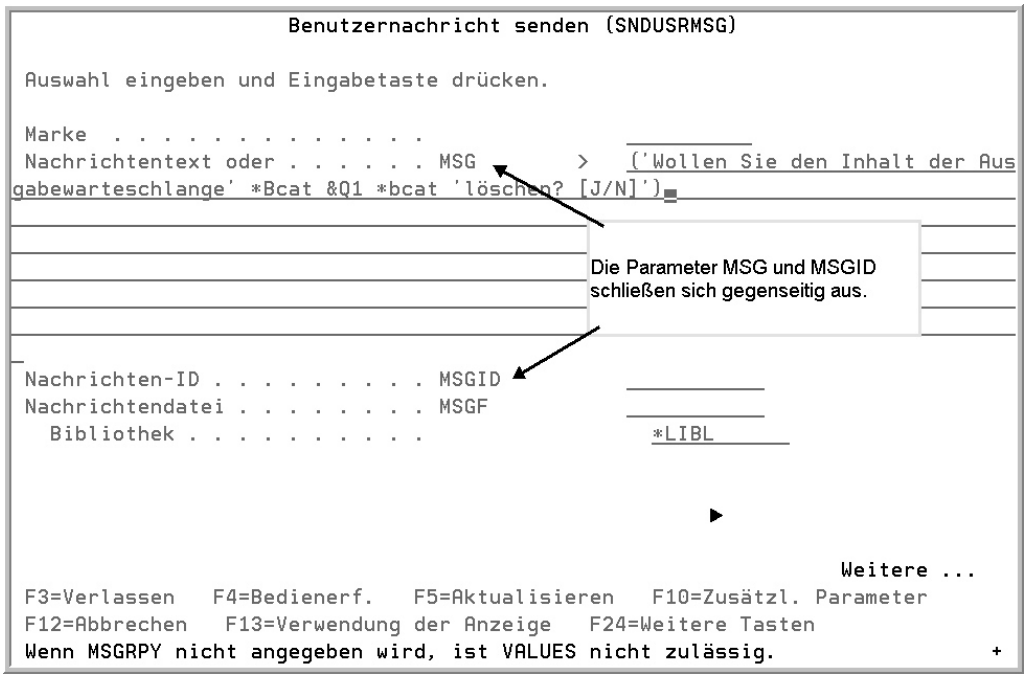
Wir brauchen also eine zuverlässige Standardfehlerroutine, die uns über alle Ausnahmefehler informiert.

```
/*  Standardfehlerroutine   -   Variablen  */
    DCL &MSGF    *CHAR 10
    DCL &LIB     *CHAR 10
    DCL &MSGID   *CHAR 7
    DCL &MSGDTA  *CHAR 3000
GLOBAL:   MONMSG MSGID(CPF0000) EXEC(GOTO CMDLBL(ERRMSG))
                      :
                      :
/* STANDARDFEHLERROUTINE  */
ERRMSG:  RCVMSG MSGTYPE(*LAST) MSGDTA(&MSGDTA)                +
         MSGID(&MSGID) MSGF(&MSGF)  MSGFLIB(&LIB)
         SNDPGMMSG MSGID(&MSGID) MSGF(&LIB/&MSGF)             +
         MSGDTA(&MSGDTA)  TOMSGQ(*SYSOPR)   TOPGMQ(*EXT)
ENDE:  ENDPGM
```

Im Fehlerfall sendet die iSeries an das Programm die Ausnahmenachricht. Der MONMSG Befehl verhindert den unmittelbaren Programmabsturz. Stattdessen empfangen wir in entsprechenden Programmvariablen die gesendete Nachricht und können jetzt aus dem Programm heraus entscheiden, wer die Originalnachricht erhalten soll. In meinem Beispiel sende ich die Mitteilung dem Operator und dem aktuellen Benutzer als Informationsnachricht und beende anschließend das Programm. Eine – wie ich finde – geschickte Art nicht vorhersehbare Ausnahmebedingungen im Programm abzufangen ohne dass entscheidende Informationen verloren gehen.

Der Befehl SNDUSRMSG

Der Befehl SNDUSRMSG (send user message) bietet eine weitere Möglichkeit, Nachrichten von einem CL-Programm aus zu versenden. SNDUSRMSG ist nicht so komplex, aber auch nicht so vielseitig wie der Befehl SNDPGMMSG. Mit dem Befehl SNDUSRMSG können Sie nur Informationsnachrichten und Abfragenachrichten versenden. Außerdem können die Nachrichten nur an Benutzer oder permanente Nachrichtenwarteschlangen gesendet werden. Sie können entweder frei formulierte oder vordefinierte Nachrichten senden. Der entscheidende Vorteil des Befehls SNDUSRMSG ist die einfache Handhabung von Abfragenachrichten, die eine Antwort empfangen sollen. Lassen Sie uns auch hier die Parameter des Befehls im Einzelnen betrachten:



Bedienerführung des Befehls SNDUSRMSG Seite 1

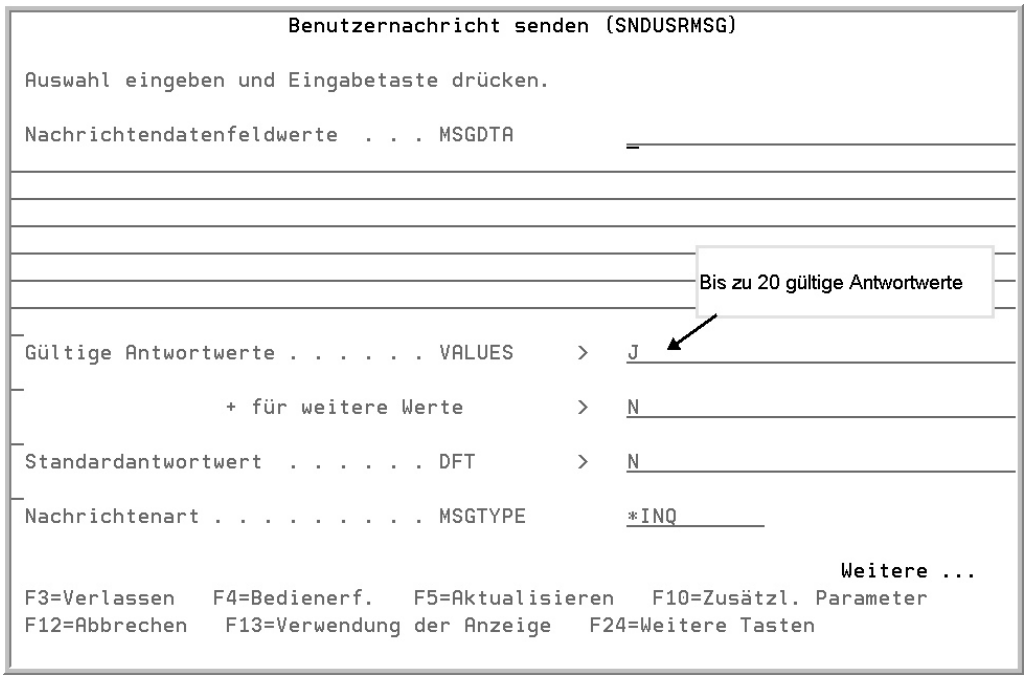
Der Parameter MSG

Der Parameter MSG gibt den Nachrichtentext einer Adhoc-Nachricht an. Maximal können 512 Zeichen angegeben werden. Wenn dieser Parameter angegeben wird, können keine Werte bei den Parametern Nachrichten-ID (MSGID), Nachrichtendatei (MSGF) oder Parameter Nachrichtendatenfeldwerte (MSGDTA) angegeben werden.

Die Parameter MSGID, MSGF und MSGDTA

Die Parameter MSGID, MSGF und MSGDTA bestimmen die Nachrichten-ID, die Nachrichtendatei und die Nachrichtendatenersatzwerte einer vordefinierten Nachricht.

2.12.7.5.4
Seite 12



Bedienerführung SNDUSRMSG Seite 2

Der Parameter VALUES

Der Parameter VALUES bestimmt eine Liste gültiger Antwortwerte. Falls die Antwort auf die Abfragenachricht keinem der angegebenen Werte entspricht, wird an den Absender der Antwort eine Fehlermeldung gesendet und die Anfragenachricht erneut gesendet.

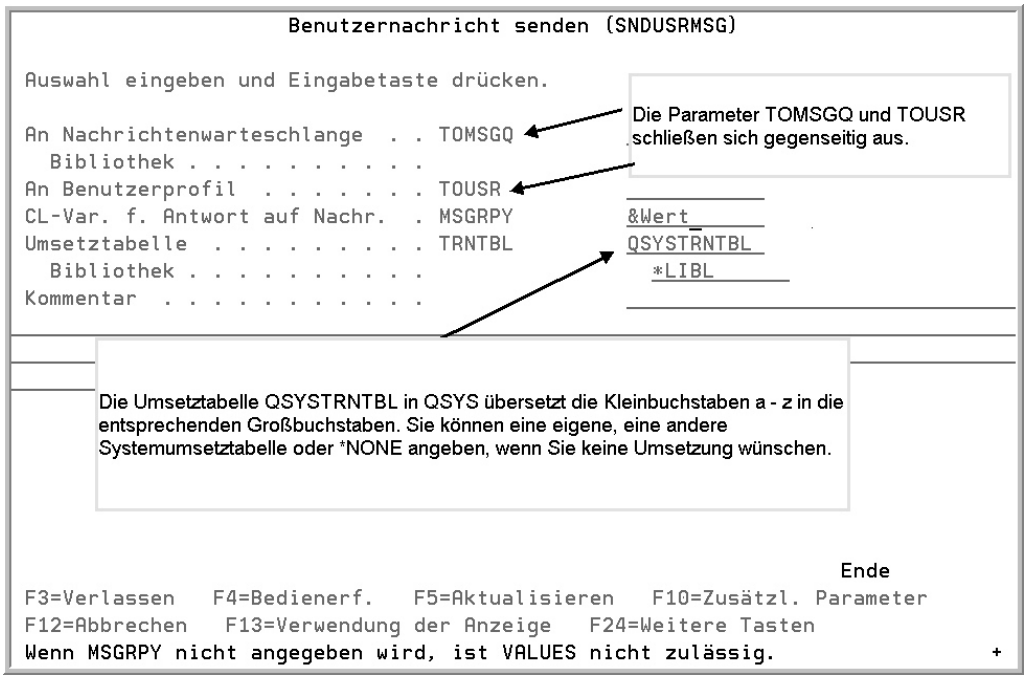
Es können max. 20 Werte angeben werden, die mit der Antwort verglichen werden. Die maximale Länge der einzelnen Werte beträgt 32 Zeichen. Wenn der Parameter VALUES genutzt wird, muss auch der Parameter MSGRPY angegeben werden.

Der Parameter DFT

Der Parameter DFT gibt den Wert an, der als Standardantwort auf eine Anfragenachricht verwendet wird, wenn die Anfragenachricht an eine Warteschlange gesendet wird, die sich im Standardzustellungsmodus befindet oder wenn der Anwender die Anfrage ohne Eingabe bestätigt. Sie können eine Standardantwort als Konstante definieren oder auf den in der Nachrichtenbeschreibung hinterlegten Standardwert verweisen, indem Sie den Sonderwert *MSGDFT verwenden.

Der Parameter MSGTYPE

Der Parameter MSGTYPE gibt die Art der zu sendenden Nachricht an. Der Befehl SNDUSRMSG kann nur Abfragenachrichten (*INQ) oder Informationsnachrichten (*INFO) senden.



Bedienerführung SNDUSRMSG Seite 3

Der Parameter TOUSR und TOMSGQ

Die Parameter bestimmen, wer die Nachricht empfangen soll. Die beiden Parameter schließen sich gegenseitig aus. Die folgende Tabelle zeigt die möglichen Werte:

Parametername	Beschreibung
TOUSR	
*SYSOPR	Systemoperator
*REQUESTER	Aktueller Benutzer eines interaktiven Jobs, Systemoperator eines Batch-Jobs.
Benutzerprofilname	Nachrichtenwarteschlange, die im Benutzerprofil hinterlegt wurde.
TOMSGQ	
*	Externe Jobnachrichtenwarteschlangen bei einem interaktiven Job; Systemoperator bei einem Batch-Job.
*EXT	Externe Jobnachrichtenwarteschlange.
*SYSOPR	Systemoperator
Nachrichten-warteschlange	Angegebene Nachrichtenwarteschlange

2.12.7.5.4
Seite 14

Der Parameter MSGRPY

Der Parameter MSGRPY gibt die CL-Variable an, die die auf eine Anfragenachricht eingegangene Antwort aufnimmt. Die Antwort kann max. 132 Zeichen lang sein. Dieser Parameter ist nur gültig, wenn als Nachrichtenart *INQ gewählt wurde.

Der Parameter TRNTBL

Der Parameter TRNTBL bestimmt die Umsetztabelle für die Antwortwerte. Die Standardtabelle QSYSTRNTBL setzt Kleinbuchstaben in Großbuchstaben um, dadurch wird die Auswertung der Antworten in den CL-Programmen vereinfacht.

Sie sehen, dass der Befehl SNDUSRMSG eine Kombination der Parameter verwendet, die bei den Befehlen SNDPGMMSG und RCVMSG zur Verfügung stehen. Der Befehl erlaubt einem Programm Nachrichten mit Hilfe eines einzigen Befehls zu senden und zu empfangen. Außerdem bietet der Befehl SNDUSRMSG die Gültigkeitsprüfung sowie die Umsetzung der Antwort in Großbuchstaben, wodurch die Handhabung der Antwortwerte in den Programmen vereinfacht wird.

Schauen wir uns zum Schluss, dass Programm, das wir bisher immer nur in Ausschnitten betrachtet haben, im Ganzen an:

```
PGM
    DCL &WERT *CHAR 10  VALUE(' ')
/*  Standardfehlerroutine   -   Variablen  */
    DCL &MSGF   *CHAR 10
    DCL &LIB    *CHAR 10
    DCL &MSGID  *CHAR 7
    DCL &MSGDTA *CHAR 3000
GLOBAL: MONMSG MSGID(CPF0000) EXEC(GOTO CMDLBL(ERRMSG))
LOOP:       DOUNTIL    COND(&WERT *EQ ' ')
            SNDUSRMSG  MSG('Geben Sie den Namen der Ausgabe-     +
               warteschlange ein, die bereinigt werden soll oder +
               geben Sie keinen Wert ein, wenn Sie das           +
               Programm beenden wollen:') DFT(' ') MSGRPY(&WERT)
           IF COND(&WERT *NE ' ') THEN(DO)
               CLROUTQ   OUTQ(&WERT)
               MONMSG MSGID(CPF3357) EXEC(DO)
```

```
                    SNDPGMMSG MSG('Das Objekt' *BCAT &WERT +
                    'existiert nicht!')
                    GOTO LOOP
                    ENDDO
            SNDPGMMSG MSGID(CPF9898) MSGF(*LIBL/ +
            QCPFMSG) MSGDTA('Der Inhalt der Ausgabewarteschlange' +
            *BCAT &WERT *BCAT 'wurde gelöscht!') TOPGMQ(*EXT) +
            MSGTYPE(*STATUS)
            SNDPGMMSG MSGID(MSG002) MSGF(*LIBL/ +
            MESSAGE) MSGDTA(&WERT) TOPGMQ(*SYSOPR) MSGTYPE(*INFO)

           ENDDO
         ENDDO
         SNDPGMMSG MSG(,'Das Programm wurde normal beendet!')
         RETURN
/* STANDARDFEHLERROUTINE  */
ERRMSG:
       RCVMSG MSGDTA(&MSGDTA) MSGID(&MSGID) MSGF(&MSGF)
MSGFLIB(&LIB)
       SNDPGMMSG MSGID(&MSGID) MSGF(&LIB/&MSGF) MSGDTA(&MSGDTA) +
       TOPGMQ(*EXT) TOMSGQ(*SYSOPR)
ENDE:  ENDPGM
```

Das Programm erlaubt dem Anwender beliebig viele Ausgabewarteschlangen anzugeben, deren Inhalt gelöscht werden soll. Für die Eingabe der Ausgabewarteschlangen verwende ich den Befehl SNDUSRMSG, der den Antwortwert in die Variable &WERT stellt. Alle anderen Bestandteile des Programms haben wir bereits im Einzelnen besprochen und müssen daher nicht erneut erläutert werden. Sie werden sicherlich schon vermuten, dass dieses Programm nur die Vorbereitung eines Tools ist. Sie haben Recht! Demnächst werden wir dieses Programm erweitern, um Spoolfiles bedingt zu löschen. Eine Funktion, die im System fehlt.

2.12.7.5.4

Seite 16

Verwendung von Dateien

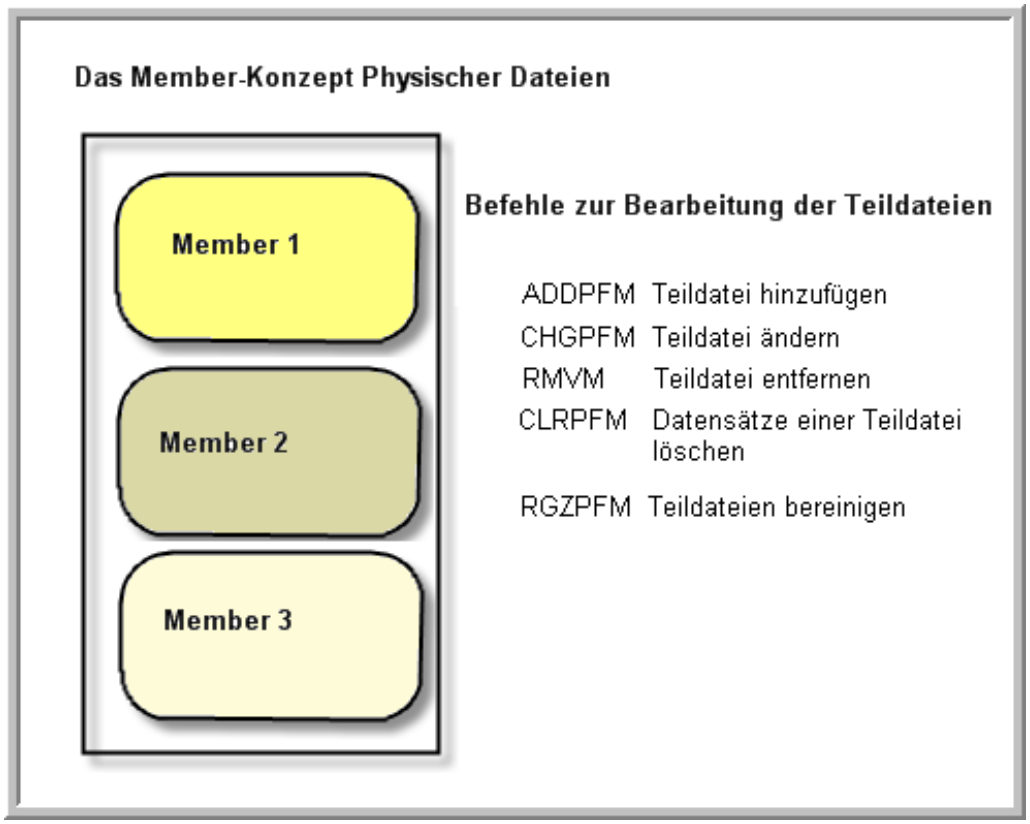
Die Verarbeitung von Dateien ist eigentlich eher eine Aufgabe der höheren Programmiersprachen. Dennoch kann auch CL, obwohl es sich hier um eine Steuersprache handelt, Dateien sowohl als Ganzes als auch Satzweise verarbeiten. Für die Lösung vieler Probleme ist die Dateiverarbeitung in CL-Programmen sogar unumgänglich.

Bildschirmdateien, so genannte Display-Files, können gelesen und gesendet werden und erleichtern zum Beispiel dem Anwender die Parametereingaben. Datenbankdateien können hingegen nur gelesen werden. Das Schreiben von Datensätzen in die Datenbankdatei ist nicht möglich.

2.12.8
Seite 2

Dateiobjekte verarbeiten

Mit Hilfe von CL-Prozeduren können Dateien als Ganzes verarbeitet werden. Datenbankdateien können mit den entsprechenden Befehlen erstellt und auch gelöscht werden. Sie können zum Beispiel eine physische Datei mit dem Befehl CRTPF (create physical file) erstellen und mit dem Befehl DLTPF (delete physical file) wieder löschen. Physische Dateien können also auch ohne Verwendung von SQL oder DDS erstellt werden. In diesem Fall existieren allerdings keine Feldbeschreibungen, d. h. die Datei verfügt nicht über einzelne Felder, so dass alle Informationen in einem einzigen Satz gespeichert werden. Das Erstellen und Löschen von Dateien kann ein sehr Ressourcen-intensiver Prozess sein und sollte daher nach Möglichkeit auf ein Minimum beschränkt werden. Die Datensätze einer physischen Datei werden in den Membern gespeichert, d.h. eine physische Datei besteht aus mindestens einer Teildatei (= Member), die wiederum die Datensätze beinhaltet.



Das Member-Konzept physischer Dateien

Nehmen wir an, Sie verwenden in Ihrem CL-Programm eine Arbeitsdatei „MYFILE“. Anstatt die Datei jedes Mal neu zu erstellen, können Sie den alten Inhalt der Datei mit dem Befehl CLRPFM löschen. Wenn die Datei bereits leer ist, führt der Befehl keine Aktion aus.

2.12.8.1
Seite 2

Durch das Teildateikonzept besteht auch die Möglichkeit, die Member separat einzelnen Jobs zuzuweisen. Eine Teildatei wird z.B. zu Beginn des Jobs erstellt und am Ende wieder gelöscht. Jede der Teildateien erhält einen eindeutigen Namen und steht somit dem Job exklusiv zur Verfügung. Als Teildateiname kann evtl. die Jobnummer verwendet werden.

```
PGM
DCL     &JOBNR *CHAR     6
DCL     &USER  *CHAR     10
DCL     &MBR   *CHAR     10

RTVJOBANBR(&JOBNR)              /* ermittelt aktuelle Jobnummer */
CHGVAR  &MBR   VALUE('JOB' *CAT &JOBNR)
ADDPFM  FILE(MYFILE) MBR(&MBR)     /* Teildatei hinzufügen */

/* Verarbeitung */

RMVM FILE(MYFILE) MBR(&MBR)
ENDPGM
```

Beachten Sie bei dieser Vorgehensweise, dass Ihre physische Datei auch tatsächlich mehr als eine Teildatei enthalten darf.

2.12.8.1
Seite 3

```
                    Physische Datei erstellen (CRTPF)

  Auswahl eingeben und Eingabetaste drücken.

  Datei  . . . . . . . . . . . . . . > MYFILE      Name
    Bibliothek . . . . . . . . . .     *CURLIB    Name, *CURLIB
  Quellendatei . . . . . . . . . .   QDDSSRC     Name
    Bibliothek . . . . . . . . . .     *LIBL      Name, *LIBL, *CURLIB
  Quellenteildatei . . . . . . . .   *FILE       Name, *FILE
  Satzlänge, falls nicht DDS . . .   132         Zahl
  Generierungsbewertungsstufe  . .   20          0-30
  Kennzeichnende Bewertungsstufe     0           0-30
  Dateiart . . . . . . . . . . . .   *DATA       *DATA, *SRC
  Teildatei, falls erforderlich  .   *FILE       Name, *FILE, *NONE
  Text 'Beschreibung'  . . . . . .   *SRCMBRTXT

                Zusätzliche Parameter

Quellenlistenauswahl . . . . . .   _______      *SRC, *NOSRC, *SOURCE
     + für weitere Werte           _______
System . . . . . . . . . . . . .   *LCL
Verfallsdatum für Teildatei  . .   *NONE
Maximale Anzahl Teildateien  . .   5           Zahl, *NOMAX
Größe des Zugriffspfads  . . . .   *MAX1TB      *MAX1TB, *MAX4GB
Zugriffspfadwartung  . . . . . .   *IMMED       *IMMED, *DLY, *REBLD
Zugriffspfad wiederherstellen  .   _______      *NO, *AFTIPL, *IPL
Schlüsselzugriffspfad erzwing.     *NO          *NO, *YES
Größe der Teildatei:
  Anfangsanzahl Sätze  . . . . .   10000        1-2147483646, *NOMAX
```

Hier bestimmen Sie die maximale Anzahl der Member.

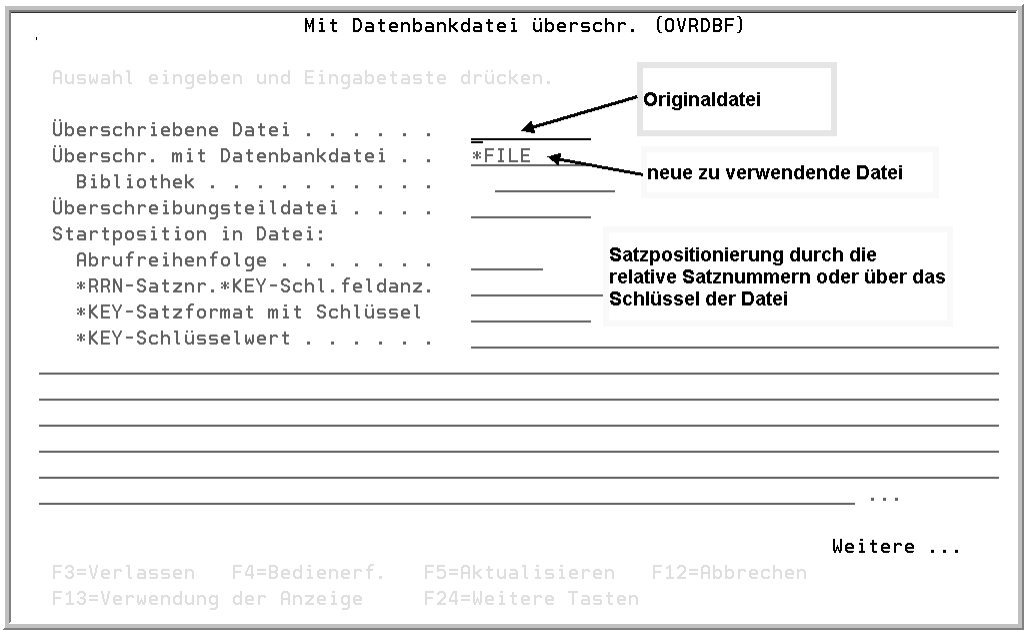
Physische Dateien erstellen

2.12.8.1

Seite 4

Dateiobjekte überschreiben

Es ist möglich, die ursprünglich in einem Programm deklarierten Dateien zu überschreiben. Der Befehl wird verwendet, um entweder eine im Programm benannte Datei zu ersetzen und/oder um bestimmte Dateiattribute der im Programm benannten Datei zu verändern. Der Befehl OVRDBF hat eine Reihe von Parametern:



Der Befehl OVRDBF Seite 1

Der Parameter FILE
Dieser Parameter legt die im Programm benannte Datei fest.

Der Parameter TOFILE
Dieser gibt den Namen der Datenbankdatei an, die anstelle der im Programm deklarierten Datei verwendet wird. Wenn der Sonderwert *FILE verwendet wird, werden nur einzelne Attribute der im Programm verwendeten Datei verändert.

Der Parameter MBR
Der bestimmt die zu verwendende Teildatei. Standardmäßig wird die erste Teildatei überschrieben. Der Sonderwert *LAST verweist auf die letzte Teildatei. Wenn der Sonderwert *ALL genutzt wird, werden alle Teildateien der aktuellen Datei sequentiell verarbeitet.

2.12.8.2
Seite 2

Der Parameter POSITION

Er gibt die Startposition für das Lesen von Sätzen aus der Datenbankdatei an. Der zu lesende Satz kann sich am Anfang (*START) oder am Ende (*END) der Datei befinden. Er kann der n-te Satz in der Datei sein (*RRN) oder es ist der Satz, der durch einen Schlüsselwert (*KEY, *KEYA, *KEYAE, *KEYB, *KEYBE) bestimmt wird. Der Parameter kann nicht verwendet werden, wenn zuvor der Parameter MBR (*ALL) gesetzt wurde.

```
                    Mit Datenbankdatei überschr. (OVRDBF)

  Auswahl eingeben und Eingabetaste drücken.

  Satzformat-Sperre:                =
    Satzformat . . . . . . . . . .   __________     Name
    Sperrstatus  . . . . . . . . .   _______        *SHRRD, *SHRNUP, *SHRUPD...
       + für weitere Werte          _
  Satzanzahl f. zwing. schreiben     ______         Zahl, *NONE
  Satzformat-Auswahlprogramm . . .   __________     Name
    Bibliothek . . . . . . . . . .     *LIBL_____   Name, *LIBL, *CURLIB
  Maximale Dateiwartezeit  . . . .   ______         Sekunden *IMMED, *CLS
  Maximale Satzwartezeit . . . . .   ______         Sekunden *NOMAX, *IMMED
  Gleichzeitig abgerufene Sätze  .   ______         Zahl
  Dateiendewartezeit in Sekunden     ___________    1-99999, *NONE
  Satzformat-Aktualitätsprüfung  .   ___            *NO
  Ablaufdatum prüfen . . . . . . .   ____           *YES, *NO
  Schreiben unterdrücken . . . . .   ____           *YES, *NO
  Vor Überschreibung schützen  . .   *NO            *NO, *YES
  Bereich überschreiben  . . . . .   *ACTGRPDFN     *ACTGRPDFN, *CALLLVL, *JOB
                                                                  Weitere ...
  F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
  F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Der Befehl OVRDBF Seite 2

Der Parameter RCDFMTLCK

Der bestimmt den Sperrstatus der Sätze während der Verarbeitung. Folgende Satzsperren sind zulässig:

Sperrstatus	Andere Programme dürfen
*SHRRD	Lesen und Aktualisieren zulässig
*SHRNUP	Lesen zulässig, Aktualisierungen sind nicht erlaubt
*SHRUP	Lesen und Aktualisieren zulässig
*EXCLRD	Nur Lesen zulässig
*EXCL	Weder Lesen noch Aktualisieren anderer Programme sind zulässig

Der Parameter FRCRATIO
Der Parameter gibt die Anzahl der Einfüge-, Lösch- oder Aktualisierungsoperationen an, die für Sätze durchgeführt werden können, bevor diese Sätze zwingend auf die Festplatte geschrieben werden. Wenn die physische Datei im Journal aufgezeichnet wird, sollte entweder ein hoher Wert für die Anzahl oder *NONE angegeben werden. *NONE kann zu langen Synchronisationszeiten beim Journal und den physischen Dateien führen.

Der Parameter WAITFILE
WAITFILE gibt die Anzahl der Sekunden an, die das Programm wartet, bis die Datei zugeordnet wird. Wenn die Dateiressourcen in der angegebenen Zeit nicht zugeordnet werden können, sendet das System eine Fehlernachricht an das Programm.

Der Parameter WAITRCD
Dieser gibt die Anzahl Sekunden an, die ein Programm auf die Zuordnung eines Satzes für Update-, Delete- oder Read-Operationen wartet.

Der Parameter EOFDLY
Durch diesen Parameter wird das EOF verzögert oder anders gesagt, das End Of File wird nicht erreicht. Dadurch haben andere Programme die Möglichkeit, Sätze der Datei hinzuzufügen, die dann von dem aktuellen Job verarbeitet werden können, ohne dass dieser Job neu gestartet werden muss. D.h. der Job wartet die angegebene Zeit. Nach Ablauf der Verzögerungszeit wird der Job automatisch aktiviert, um festzustellen, ob neue Sätze hinzugefügt wurden. Wenn keine neuen Sätze vorhanden sind, wartet der Job erneut das Verstreichen der Verzögerungszeit ab, ohne dass eine Rückmeldung an das Anwendungsprogramm erfolgt. Die Beendigung des Jobs müssen Sie in diesem Fall durch den Befehl ENDJOB erzwingen.

Der Parameter LVLCHK
Sobald Sie eine Datenbankdatei aktualisieren, verändert sich deren Aktualitäts-ID. Eröffnet ein Programm jetzt eine Datenbankdatei, vergleicht das System die IDs aller vom Programm verwendeten Satzformate mit den zugehörigen Kennzeichen in der Datenbankteildatei. Die Aktualitätsprüfung wird nur durchgeführt, wenn der Parameter LVLCHK den Wert *YES enthält.

Der Parameter INHWRT
Dieser Parameter gibt an, ob die verarbeiteten Sätze in die Datenbankdatei geschrieben, gelöscht oder aktualisiert werden. Der Parameter Schreibsperre ermöglicht das Testen einer Datei, ohne dass die verarbeiteten Datensätze in der Datenbank gespeichert werden.

Der Parameter OVRSCOPE
OVRSCOPE bestimmt den Gültigkeitsbereich der Überschreibung.

*ACTGRPDFN
Die Überschreibung ist innerhalb der Aktivierungsgruppe des Programms gültig, das diesen Befehl aufruft. Handelt es sich bei der Aktivierungsgruppe um die Standardaktivierungsgruppe, entspricht der Gültigkeitsbereich der Aufrufebene des aufrufenden Programms. Ist die Aktivierungsgruppe nicht die Standardaktivierungsgruppe, entspricht der Gültigkeitsbereich der Aktivierungsgruppe des aufrufenden Programms.

*CALLLVL
Der Gültigkeitsbereich der Überschreibung wird durch die aktuelle Aufrufebene bestimmt. Die Überschreibungen betreffen das aktuelle und alle aufgerufenen Programme.

*JOB
Die Überschreibungen gelten für den gesamten Job.

Der Parameter SEQONLY
SEQONLY gibt für Datenbankdateien, deren Sätze nur sequentiell verarbeitet werden, an, dass die Datei nur sequentiell verarbeitet wird. Der Parameter gibt darüber hinaus an, wie viele Sätze als Gruppe verarbeitet werden. Wird kein Wert angegeben, bestimmt das System einen Standardwert. Dieser Parameter wird verwendet, um die Programmleistung zu verbessern. Detaillierte Informationen über die ausschließlich sequentielle Verarbeitung von Datenbankdateien enthält das Handbuch Datenbankinformationen im iSeries Information Center unter http://www.ibm.com/eserver/iseries/infocenter.

Schauen wir uns jetzt einige Beispiele an:
Stellen Sie sich vor, in einem Programm wird die Datei WRKFILE in der Bibliothek PRDLIB verarbeitet. Sie dagegen wollen auf die Dateikopie FILE01 in der Bibliothek MYLIB zugreifen. Anstatt die Bibliotheksliste zu verändern können Sie den Befehl OVRDBF (override with data base file) verwenden.

```
OVRDBF FILE(PRDLIB/WRKFILE) TOFILE(mylib/file01)
```

Die Überschreibung könnte sich auch lohnen, wenn ein RPG-Programm auf eine Datei unter Verwendung eines anderen Namens zugreift. Auch hier hilft die Verwendung des Befehls OVRDBF. Das HLL-Programm verwendet z.B. die Datei PAANS01 in der Bibliothek PRDLIB. Sie wollen aber, dass die Datei PAANS08 verwendet wird, um z. B. die Sortierfolge zu verändern oder einfach weil die Datei zwischenzeitlich umbenannt wurde und Sie das RPG-Programm nicht ändern wollen.

```
OVRDBF FILE(PRDLIB/PAANS01) TOFILE(PRDLIB/PAANS08)
CALL …..
```

Sie möchten in einem Programm eine spezielle Teildatei verwenden.

```
OVRDBF FILE(PAUMS00) MBR(Feb)
CALL ...
```

Das aufgerufene Programm wird jetzt die Teildatei „FEB“ verarbeiten. Wenn keine Teildatei angegeben wird, wird stets das erste Member verarbeitet.

```
OVRDBF FILE(PAANS00) SHARE(*YES)
```

Dieser Befehl überschreibt die Angabe SHARE für die Datei PAANS00. Aufgrund dieser Überschreibung wird bei jedem nachfolgenden Öffnen dieser Datei derselbe ODP für die Datei benutzt.

```
OVRDBF FILE(INPUT) TOFILE(OUT) MBR(O1)    +
     RCDFMTLCK((OUTFMT *EXCL))
```

Der Befehl überschreibt die Datei, die Teildatei und den Sperrstatus des Satzformats OUTFMT. Die Überschreibung bewirkt Folgendes, sobald die Datei INPUT geöffnet wird:

- Die Datei OUT wird an Stelle der Datei INPUT verarbeitet.
- Die Teildatei O1 wird verarbeitet.
- Die Sperre *EXCL wird für das Satzformat OUTFMT angelegt. Durch diesen Sperrstatus kann kein anderes Programm auf das Satzformat zugreifen solange die Überschreibung wirksam ist.

Drucker- und Anzeigedateien können Sie ebenfalls mit den Befehlen OVRPRTF (override with printerfile) und OVRDSPF (override with displayfile) überschreiben. Nehmen wir an, Sie erstellen eine Druckdatei PRTFRM. Diese hat eine Länge von 25 Zeilen und 80 Spalten. Nun müssen Sie die Liste in einem anderen Format drucken. Sie überschreiben einfach die Attribute der Druckdatei und veranlassen so den neuen Ausdruck.

```
OVRPRTF FILE(PRTFRM) FORMTYPE(*STD) PAGESIZE(66 132) OVERFLOW(60)
CALL ….
```

Falls Sie die Überschreibungen wieder aufheben wollen, verwenden Sie einfach den Befehl DLTOVR.

2.12.8.2

Seite 6

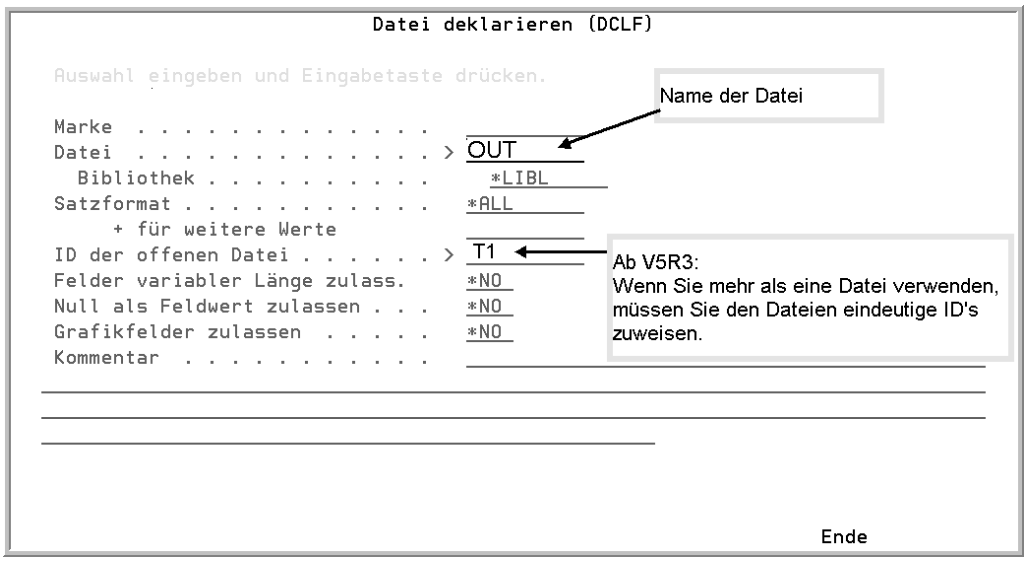
Die Bibliothek QTEMP

Die Bibliothek QTEMP existiert nur für die Dauer eines Jobs. Das System erstellt für jeden Job eine eigene Bibliothek QTEMP, wenn der Job gestartet wird und löscht diese Bibliothek wieder, wenn er beendet wird. Jeder Job hat somit seine eigene Bibliothek QTEMP. Wenn Sie ein Objekt in QTEMP erstellen, kann kein anderer Job in dem System auf dieses Objekt zugreifen. QTEMP ist also der ideale Ort für alle Objekte, die Sie nur für die Zeit der Ausführung des Jobs benötigen. So können Sie zum Beispiel Arbeitsdateien erstellen, die Daten aus unterschiedlichen Quellen aufnehmen und dann im weiteren Verlauf ausgedruckt werden. Nach dem Ausdruck haben die Arbeitsdateien keine weitere Verwendung mehr. Sie können natürlich nicht nur Dateien in QTEMP erstellen, generell lassen sich alle Objekte in der Bibliothek QTEMP erstellen. Da QTEMP gelöscht wird, sobald der Job beendet wird, werden alle Objekte der Bibliothek auch automatisch gelöscht. Sie brauchen sich also um das Aufräumen nicht mehr zu kümmern, und sparen so eventuell wertvollen Plattenplatz. Sollten Sie eine Vielzahl von Objekten in einer interaktiven Sitzung in QTEMP erstellt haben, so kann das Abmelden unter Umständen etwas länger dauern, da das System damit beschäftigt ist alle Objekte der Bibliothek QTEMP zu löschen, um abschließend die Bibliothek selbst auch noch zu löschen.

2.12.8.3
Seite 2

Dateien deklarieren

Wenn Dateien in einem CL-Programm nicht als Ganzes sondern Satz für Satz verarbeitet werden sollen, müssen wir die Dateien zunächst mit dem Befehl DCLF deklarieren. Jeder DCLF-Befehl enthält den Namen einer Bildschirmdatei oder einer Datenbankdatei sowie eine ID für die geöffnete Datei, die die Instanz der geöffneten Datei eindeutig identifiziert. Vor dem Release V5R3 konnte nur eine Datei je CL-Programm verwendet werden, d.h. der DCLF-Befehl konnte nur ein einziges Mal kodiert werden. Seit dem Release V5R3 können bis zu 5 Dateien in einem CL-Programm angesprochen werden. Es ist somit auch möglich, dass mehrere DCLF-Befehle auf dieselbe Datei verweisen, sofern die angegebene ID unterschiedlich ist.



Dateien deklarieren

Der Befehl DCLF (declare file) hat die folgenden Parameter:

Der Parameter FILE
Der Parameter FILE spezifiziert die Datei. Hier müssen Sie den Namen der Datenbank- oder Bildschirmdatei eintragen, die verarbeitet werden soll.

Wenn die angegebene Datei eine Bildschirmdatei ist, gilt Folgendes:

- Die Datei muss zur Umwandlungszeit des Programms vorhanden sein.
- Das Programm kann die folgenden Befehle enthalten: SNDF, RCVF, SNDRCVF, ENDRCV, und WAIT.
- Die Datei wird zur Ein- und Ausgabe geöffnet, d.h. sie kann gelesen und geschrieben werden.

2.12.8.4
Seite 2

Wenn die angegebene Datei eine Datenbankdatei ist, gilt Folgendes:

- Die Datei muss zur Laufzeit des Programms eine Datenbankdatei mit einem Satzformat sein.
- Das Programm kann nur den Befehl RCVF enthalten.
- Die Datei wird zur Eingabe geöffnet, d.h. sie kann nur gelesen werden.
- Die Datei wird geöffnet, wenn der Befehl RCVF verwendet wird und geschlossen, wenn das Programm beendet wird.

Der Parameter RCDFMT
Dieser bestimmt das zu verarbeitende Satzformat. Für jedes Satzformat der Datei, maximal 99, werden die Felder als CL-Variablen deklariert, wenn der Sonderwert *ALL verwendet wird. Falls die Datei mehr als 99 Satzformate hat, werden nur die ersten 99 Satzformate verwendet. Es empfiehlt sich allerdings, ein konkretes Format zu benennen. Dadurch wird das erstellte Programmobjekt kleiner und die Umwandlungsliste ist einfacher zu lesen. Eine CL-Variable kann für die Benennung des Satzformatnamens nicht verwendet werden.

Der Parameter OPNID
OPNID existiert seit dem Release V5R3 und dient zur Unterscheidung mehrerer Dateien. Solange Sie nur eine Datei verarbeiten, können Sie die Default-Einstellung *NONE verwenden.

Der Parameter ALWVARLEN

Der gibt an, ob Felder variabler Längen im Satzformat zulässig sind. Wenn Sie mit variablen Feldlängen arbeiten, werden die zugehörigen CL-Variablen als *CHAR-Felder deklariert. Die CL-Variablen entsprechen der maximal zulässigen Länge des variablen Feldes plus 2 zusätzlichen Byte. Nehmen wir an, in Ihrer Datenbank befindet sich ein VARCHAR-Feld, das maximal 500 Byte lang ist. Die CL-Variable ist in diesem Fall 502 Byte lang. Nach dem Lesen des Satzes enthalten die ersten beiden Bytes die tatsächliche Länge der gelesenen Daten. Die von der Datei empfangenen Daten werden ggf. rechts mit Leerzeichen aufgefüllt.

Der Parameter ALWNULL
Dieser gibt an, ob der Wert NULL zulässig ist. Wenn Sie keine NULL-Werte akzeptieren, wird eine Diagnosenachricht für den gesamten Satz gesendet falls NULL-Werte empfangen werden. Die CL-Variablen enthalten dann die Standardwerte.

Der Parameter ALWGRAPHIC
Der Parameter gibt an, ob Grafikfelder zulässig sind. Die entsprechenden CL-Variablen werden wie Variablen der Art *CHAR behandelt, wobei die Länge gleich der Länge des Grafikdaten-Feldes ist.

Wenn Sie ein CL-Programm kompilieren, das einen DCLF-Befehl enthält, erhalten Sie automatisch für jedes Feld der Datei eine CL-Variable. Für eine Datei mit der OPNID *NONE, wird der Variablenname aus dem Feldnamen der Datei mit einem führenden Ampersand(&) gebildet. Hier ein Beispiel: Stellen Sie sich vor, wir haben eine Datei OUT. Diese Datei enthält zwei Felder: DTName und BiBName. Die Datei OUT wird im CL-Programm mit folgendem Befehl deklariert:

```
DCLF  FILE(OUT) OPNID(*NONE)
```

```
                          CL-Quellenprogramm
FOLGENR *...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+...
         PGM
   200-             DCLF       FILE(OUT)
        QUALIFIZIERTER DATEINAME - EPS002CR/OUT
           SATZFORMATNAME - OUTFMT
             CL-VARIABLE
              &DTNAME
              &BIBNAME
   300-  ENDPGM

* * * *  E N D E  D E S  Q U E L L E N P R O G R A M M S  * * * *
```

Der CL Kompiler hat für die Felder der Datei automatisch die Variablen &DTNAME und &BIBNAME generiert.

Ergebnis der Programmumwandlung Beispiel 1

Sobald der Parameter OPNID einen anderen Wert als *NONE hat, wird der Variablenname wie folgt gebildet: Ampersand(&), der OPNID-Wert, ein Unterstrich und zum Schluss der Feldname. Ich denke, es ist auch hier einfacher sich die Variablenbenennung an einem Beispiel vorzustellen: Die Datei OUT wird im CL-Programm jetzt mit folgendem Befehl deklariert:

```
DCLF  FILE(OUT) OPNID(T01)
```

```
                          CL-Quellenprogramm
FOLGENR *...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+...
   100-  PGM
   200-             DCLF       FILE(OUT) OPNID(T1)
        QUALIFIZIERTER DATEINAME - EPS002CR/OUT
           SATZFORMATNAME - OUTFMT
             CL-VARIABLE
              &T1_DTNAME
              &T1_BIBNAME
   300-  ENDPGM

* * * *  E N D E  D E S  Q U E L L E N P R O G R A M M S  * * * *
```

Der CL Kompiler hat für die Felder der Datei automatisch die Variablen &T1_DTNAME und &T1_BIBNAME generiert.

Die Attribute der CL-Variablen stimmen stets mit den Attributen der Datensatzfelder überein. Felder, die in der Datei numerisch definiert sind, werden

2.12.8.4
Seite 4

Die Attribute der CL-Variablen stimmen stets mit den Attributen der Datensatzfelder überein. Felder, die in der Datei numerisch definiert sind, werden als Dezimalvariablen (*DEC) deklariert. Bezugszahlen werden als logische Variablen deklariert, wobei die Zeichenfolge &IN vorangestellt wird. Variablen, die indirekt durch den DCLF deklariert wurden, können auf die gleiche Weise verarbeitet werden, wie die durch den DCL-Befehl deklarierten Variablen.

Was macht der CL-Kompiler aber mit Dateien, die über keine Feldbeschreibungen verfügen? Stellen wir uns vor, die Datei OUT sei mit folgendem Befehl erstellt worden:

```
CRTPF FILE(OUT) RCDLEN(132)
```

Der CL-Kompiler hat in diesem Fall keine Möglichkeit, Variablen für die Datensatzfelder zu generieren.

```
DCLF  FILE(OUT) OPNID(*NONE)
```

```
                          CL-Quellenprogramm
 FOLGENR
    100-
    200-  In einem solchen Fall erstellt der Kompiler eine einzige Variable, die denselben Namen erhält wie die Datei.
          Diese Variable wird als Zeichenvariable entsprechend der Satzlänge der Datei deklariert.

            SATZFORMATNAME -  OUT
              CL-VARIABLE                ART       LÄNGE      GENAUIGKEIT   TEXT
               &OUT                      *CHAR     132
    300-  ENDPGM
* CPD0729 10  Keine ausführbaren Befehle in Quellendatei festgestellt.
* * * *  E N D E  D E S  Q U E L L E N P R O G R A M M S  * * * *
```

Das Ergebnis der Programmumwandlung Beispiel 3

Datenbankdateien lesen

Nachdem Sie Ihre Datenbankdateien deklariert haben, können Sie die Dateien jetzt mit dem Befehl RCVF (receive file) lesen. Die Sätze der Datei werden sequentiell Satz für Satz gelesen. RCVF liest somit genau einen Datensatz, d.h. wenn mehrere Sätze verarbeitet werden sollen, muss der RCVF-Befehl wiederholt oder in einer Schleife kodiert werden. Sobald das EOF (end of file) erreicht ist, d.h. in der Datei existieren keine weiteren Sätze, wird die Diagnosemeldung CPF0864 ausgegeben.

```
 LOOP:                    RCVF
                                        MONMSG MSGID(CPF0864)
EXEC(RETURN)
                                        :
                                        :
                                        GOTO LOOP
```

Der Befehl RCVF hat vier optionale Parameter:

```
                          Datei empfangen (RCVF)

 Auswahl eingeben und Eingabetaste drücken.

 Marke  . . . . . . . . . . . . . > _________
 Bildschirmeinheit  . . . . . . .   *FILE      Name, *FILE
 Satzformat . . . . . . . . . . .   *FILE      Name, *FILE
 ID der offenen Datei . . . . . .   T1         Name, *NONE
 Warten . . . . . . . . . . . . .   *YES       *YES, *NO
 Kommentar  . . . . . . . . . . .   ____________________________________________
 ____________________________________________________________________________
 ____________________________________________________________________________
 ____________________________________________
```

Der Befehl RCVF

Der Parameter DEV
Dieser bestimmt den Namen der Datei, die verarbeitet wird. Der Standardwert ist *FILE, der bestimmt, dass die Daten von der zuvor deklarierten Datei gelesen werden sollen.

Der Parameter RCDFMT
RCDFMT ist nur erforderlich, wenn die Datei mehr als ein Satzformat enthält. Diese Dateien sind in einem CL-Programm eher unwahrscheinlich, so dass Sie in den meisten Fällen die Default-Einstellung *FILE nutzen können.

hält. Diese Dateien sind in einem CL-Programm eher unwahrscheinlich, so dass Sie in den meisten Fällen die Default-Einstellung *FILE nutzen können.

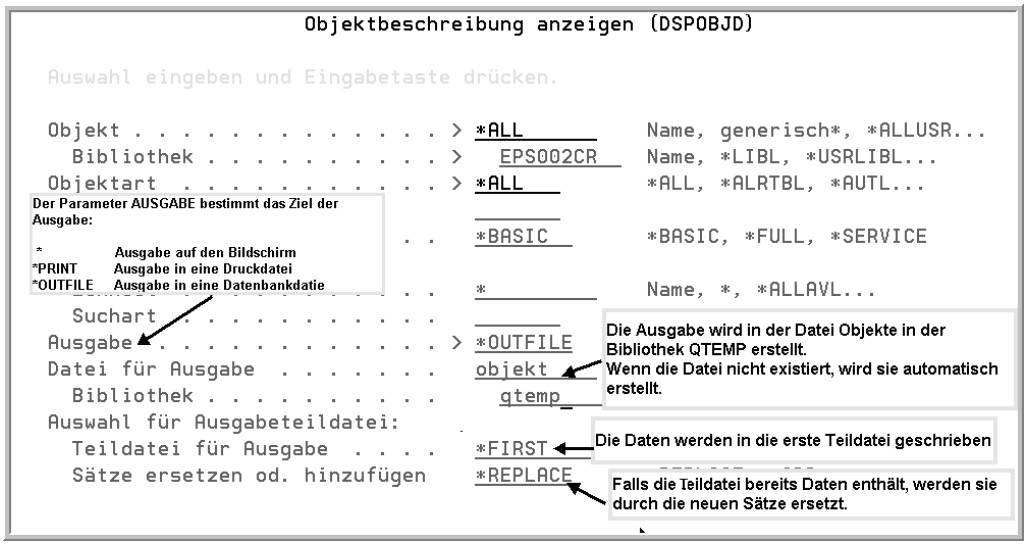
Der Parameter OPNID
OPNID existiert seit dem Release V5R3 und bestimmt, auf welche der deklarierten Dateien zugegriffen werden soll. Der Parameter ist nur erforderlich, wenn Sie mit mehr als einer Datei in Ihrem CL-Programm arbeiten. Der hier verwendete Wert muss dann natürlich mit dem Inhalt des Parameters OPNID der deklarierten Dateien übereinstimmen.

Der Parameter WAIT
WAIT wird nur in Verbindung mit Bildschirmdateien verwendet. WAIT akzeptiert zwei Werte: *NO oder *YES. WAIT(*YES) veranlasst das CL-Programm mit der weiteren Verarbeitung zu warten bis der Anwender seine Bildschirmeingaben beendet hat, d.h. bis der Anwender entweder die Bildschirmausgabe mit der Datenfreigabe- oder einer Funktionstaste abschließt. Der Parameter WAIT(*NO) führt dazu, dass das CL-Programm weiter ausgeführt wird bis ein entsprechender WAIT-Befehl abgesetzt wird. Normalerweise werden Sie auch hier immer mit der Default-Einstellung WAIT(*YES) arbeiten.

Ausgabedateien nutzen

Welche Probleme lassen sich jetzt aber konkret durch den Einsatz von Dateien lösen?

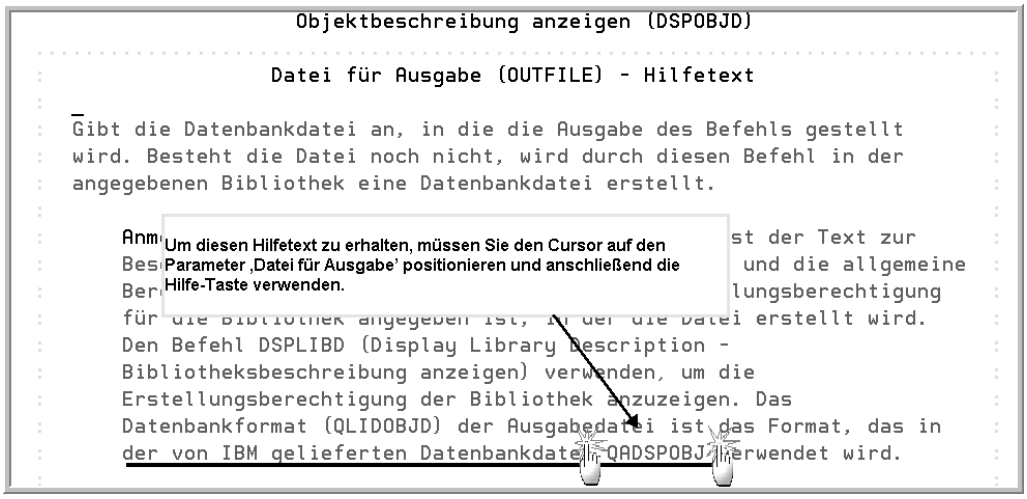
i5/OS-Befehle, die mit dem Verb DSP(Display) und manchmal auch mit dem Verb WRK (work) beginnen, dienen der Ausgabe von Informationen. Die Informationen können angezeigt, gedruckt oder in eine Datei geschrieben werden. Wohin die Informationen gesendet werden ist abhängig von dem Parameter AUSGABE: Der Wert „*“ stellt die Ausgabe bei einem interaktiven Job auf dem Bildschirm und bei einem Batchjob als Druckausgabe dar. Wird der Wert *PRINT verwendet, wird eine Druckausgabe erzeugt. Manche Befehle verfügen über einen dritten Wert, den Wert *OUTFILE. Der Wert *OUTFILE leitet die Druckausgabe direkt in eine Datei. Die Befehle DSPUSRPRF (display userprofile) oder DSPOBJD (display object description) verfügen z.B. über diese Möglichkeit.



*Der Befehl DSPOBJD Parameter *OUTFILE*

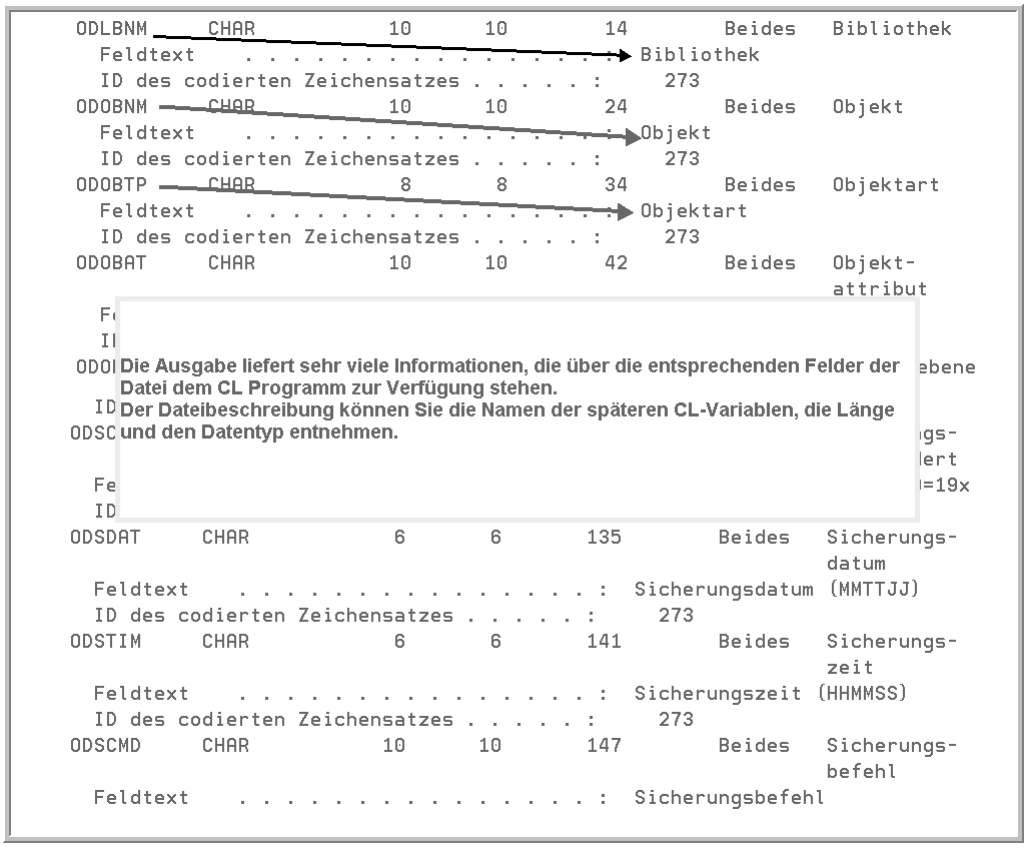
Das System führt hier den Befehl „Objektbeschreibungen anzeigen“ aus. Anstatt das Ergebnis anzuzeigen oder zu drucken, werden die Informationen in die Datenbankdatei „OBJEKT“ in die Bibliothek „QTEMP“ geschrieben. Wenn die Datei nicht existiert, wird sie automatisch erstellt. Aufgrund der Angabe *REPLACE werden bereits vorhandene Einträge überschrieben. Der auszuführende Befehl DSPOBJD enthält für den Parameter OUTFILE nützliche Hilfetexte, die das Dateimodell benennen.

2.12.8.6
Seite 2



Hilfetext des Befehls DSPOBJD

Die Ausgabedatei – in unserem Beispiel die Datei OBJEKT – wird nicht willkürlich erstellt, sondern aufgrund einer Modelldatei generiert, die in der Bibliothek QSYS gespeichert ist. Den Satzaufbau der Modelldatei, in unserem Beispiel die Datei QADSPOBJ, können Sie sich mit dem Befehl DSPFFD (display file field description) anzeigen lassen. Das Layout gibt Ihnen Auskunft darüber, welche Informationen Sie erhalten und in welchen Feldern die Informationen gespeichert werden.



Auszug der verfügbaren Informationen der physischen Datei QADSPOBJ

Die Ausgabedateien können in CL-Programmen problemlos verarbeitet werden. Stellen Sie sich z.B. vor, Sie wollen bei Objekten einer beliebigen Bibliothek den bisherigen Eigner verändern, weil Sie zukünftig mit Gruppenprofilen arbeiten oder weil ein Mitarbeiter aus dem Unternehmen ausgeschieden ist. Wollen Sie diese Tätigkeit manuell ausführen, dann ist dies sowohl ein zeitaufwendiges als auch fehleranfälliges Verfahren. Stattdessen schreiben wir einfach ein CL-Programm, das wir CHGOWNER nennen:

```
PGM PARM(&OldOwner &NewOwner &Lib)
  DCL               &OldOwner                *CHAR LEN(10)
  DCL               &NewOwner              *CHAR LEN(10)
  DCL               &Lib                                *CHAR LEN(10)
```

Das Programm empfängt drei Variablen: den Namen des ursprünglichen Eigners, den neuen Eigner und die Bibliothek deren Objekte untersucht werden sollen.

```
DCLF  FILE(QADSPOBJ)
```

Anschließend wird die Datei deklariert. Wir verwenden zu diesem Zeitpunkt einfach die Modelldatei der IBM, da die Datei zur Umwandlungszeit des Programms vorhanden sein muss.

```
                  DSPOBJD OBJ(&Lib/*ALL) OBJTYPE(*ALL)
OUTPUT(*OUTFILE) +
                  OUTFILE(Qtemp/Objekt) OUTMBR(*FIRST *REPLACE)
```

Jetzt erstellen wir das Duplikat der Modelldatei mit den Informationen in der Bibliothek QTEMP. Nähere Informationen über die Besonderheiten der Bibliothek QTEMP erfahren Sie im Kapitel 2.12.8.3.

```
                   OVRDBF QADSPOBJ TOFILE(QTEMP/Objekt)
```

In der Dateideklaration haben wir die Modelldatei QADSPOBJ der Bibliothek QSYS verwendet. Wenn wir diese Datei jetzt lesen, erhalten wir aber keine Daten, da diese in der Datei OBJEKT in der Bibliothek QTEMP stehen. CL bietet uns aber die Möglichkeit mit Überschreibungen zu arbeiten, d.h. wir überschreiben die ursprüngliche Dateideklaration durch unsere Datei OBJEKT. Das CL-Programm wird ab jetzt bei allen Dateibefehlen nicht mehr die Originaldatei sondern unsere Datei OBJEKT verwenden. Nähere Informationen über den Befehl OVRDBF erhalten Sie im Kapitel 2.12.8.2.

```
  LOOP:          RCVF
                        MONMSG MSGID(CPF0864) EXEC(RETURN)
```

Jetzt können wir den ersten Satz der Datei Objekte lesen. Wenn das Dateiende erreicht ist, wird das Programm beendet.

```
  IF                    COND(&ODOBOW = &OLDOWNER) THEN( +
                                  CHGOBJOWN OBJ(&ODLBNM/
&ODOBNM) OJTYPE(&ODOBTP) +
                                  NEWOWN(&NEWOWNER))
```

Der aktuelle Eigner des Objektes ist in der Variablen &ODOBOW gespeichert. Diese Information müssen wir uns im Vorfeld beschaffen, indem wir uns wie beschrieben die Feldinformationen der Modelldateien anzeigen lassen. Wenn der aktuelle Eigner dem gesuchten Wert entspricht, wird die Eignerschaft durch das Programm auf den neuen Wert geändert. Um den Eigner mit dem Befehl CHGOBJOWN zu verändern, benötigen wir den Namen des Objektes, evtl. die Bibliothek und natürlich den Objekttyp. Auch diese Informationen können wir im Vorfeld der Modelldatei entnehmen. In unserem Fall finden wir die benötigten Daten in den Variablen &ODLBNM, &ODOBNM und &ODOBTP.

```
  GOTO LOOP
```

Nun springen wir an die Marke LOOP, um den nächsten Satz unserer Datei OBJEKT zu lesen und anschließend zu verarbeiten.

Der Befehl DSPOBJD liefert eine Vielzahl von Informationen. Neben der Objekteignerschaft erhalten Sie unter anderem Informationen über die Objekterstellung einschließlich der zugehörigen Objektquellen, die Sicherung des Objektes und die Objektverwendung. Mit diesen Informationen können Sie viele Tools schreiben, die Ihnen Ihre tägliche Arbeit erleichtern. Die Vorgehensweise ist immer identisch wie Sie an einem zweiten Beispiel erkennen werden. Das CL-Programm soll alle Objekte einer beliebigen Bibliothek sichern, die bislang noch nie gesichert wurden:

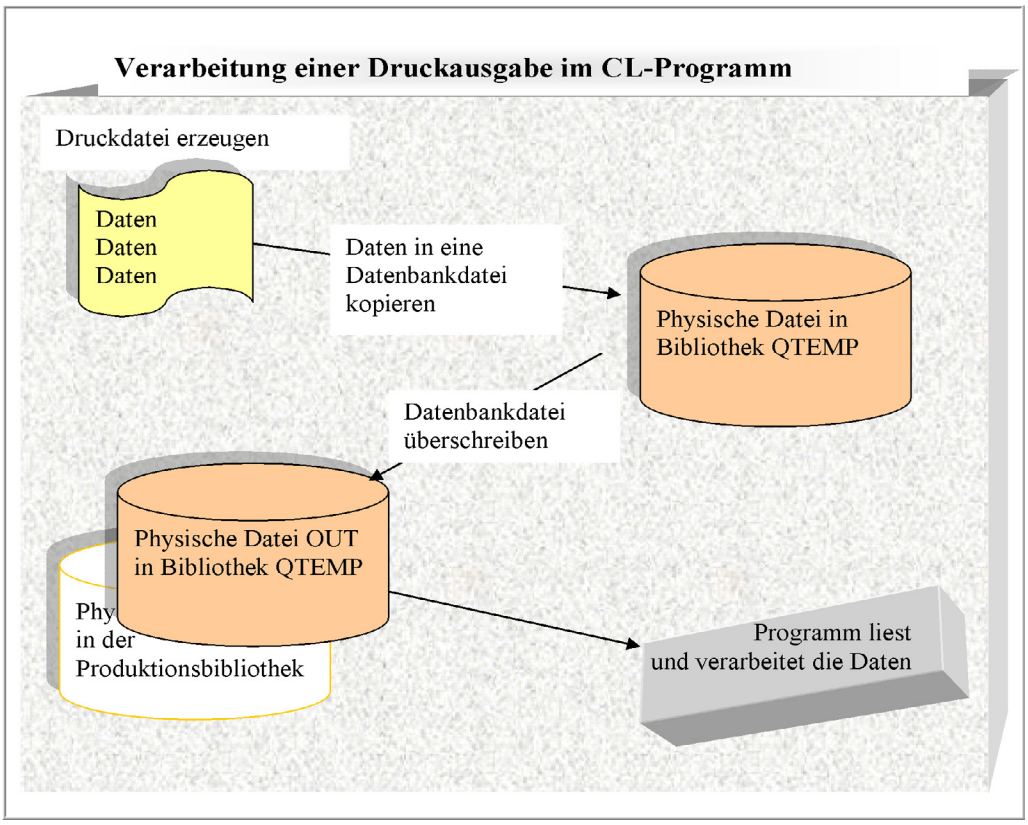
```
PGM PARM(&LIB)          /* Name der Bibliothek, die untersucht
werden soll */

DCL &LIB *CHAR 10

DCLF  FILE(QADSPOBJ)  /* Modelldatei für die Ausgabe DSPOBJD */

/* der Befehl DSPOBJD…OUTPUT(*OUTFILE) liefert wir die
 benötigten Informationen */
DSPOBJD OBJ(&LIB/*ALL) OBJTYPE(*ALL) +
      OUTPUT(*OUTFILE) OUTFILE(QTEMP/OBJEKT)

/* Sicherstellen, dass die soeben erstellten Daten verarbeitet
 werden */
OVRDBF  FILE(QADSPOBJ) TOFILE(QTEMP/OBJEKT)

LOOP:
RCVF  /* Satz lessen */
MONMSG MSGID(CPF0864) EXEC(RETURN)  /* Bei EOF Programm beenden */

/* Verarbeitung, indem zunächst geprüft wird, ob das Objekt bereits
 gesichert wurde, +
 indem geprüft wird, ob ein Sicherungsdatum(&ODSDAT) existiert.*/
IF (&ODSDAT = , ,) THEN(DO)

/* Objekt wird gesichert */
         SAVOBJ  OBJ(&ODOBNM) LIB(&ODLBNM) DEV(TAP01) +
         OBJTYPE(&ODOBTP) ENDOPT(*LEAVE)
  ENDDO

GOTO LOOP    /* der nächste Satz kann verarbeitet warden */

ENDPGM
```

2.12.8.6

Seite 6

Druckausgaben verarbeiten

Leider erlauben nur wenige Betriebssystembefehle die Möglichkeit mit OUTFILES zu arbeiten. Viele DSP-Befehle kennen nur die Ausgabe auf den Bildschirm oder auf den Drucker. Wir können die Druckausgaben in einem CL-Programm nur verarbeiten, wenn wir die Druckausgabe in eine Datenbankdatei kopieren.

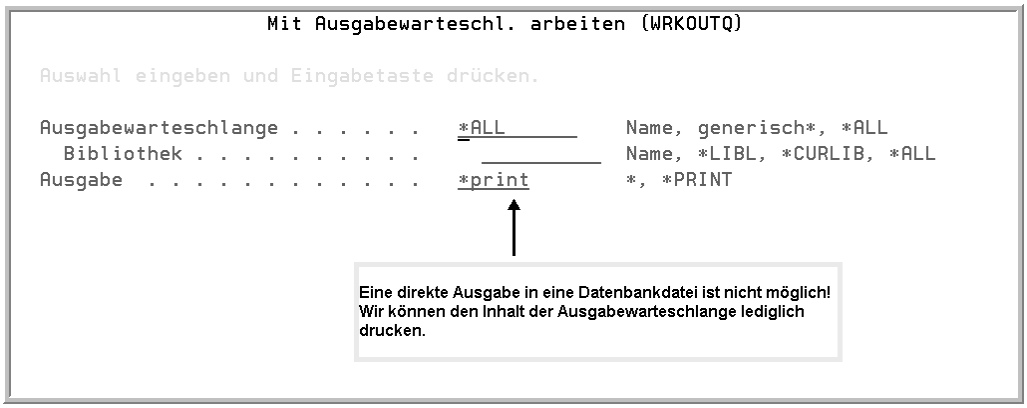


Die Verarbeitung einer Druckausgabe

Erinnern Sie sich noch an unser Beispielprogramm, das den Inhalt einer beliebigen Ausgabewarteschlange löscht? Was ist, wenn wir jetzt nicht mehr alle sondern nur bestimmte Spoolfiles löschen wollen? In unserem nächsten Beispiel sollen die Spoolfiles einer beliebigen Ausgabewarteschlange gelöscht werden, die vor einem bestimmten Datum erstellt wurden. Um auf die einzelnen Spoolfiles einer Ausgabewarteschlange in einem CL-Programm zugreifen zu können, müssen wir den Inhalt der Ausgabewarteschlange in unserem CL-Programm „lesen“. Schauen wir uns die dafür notwendigen Verarbeitungsschritte zunächst im Einzelnen an:

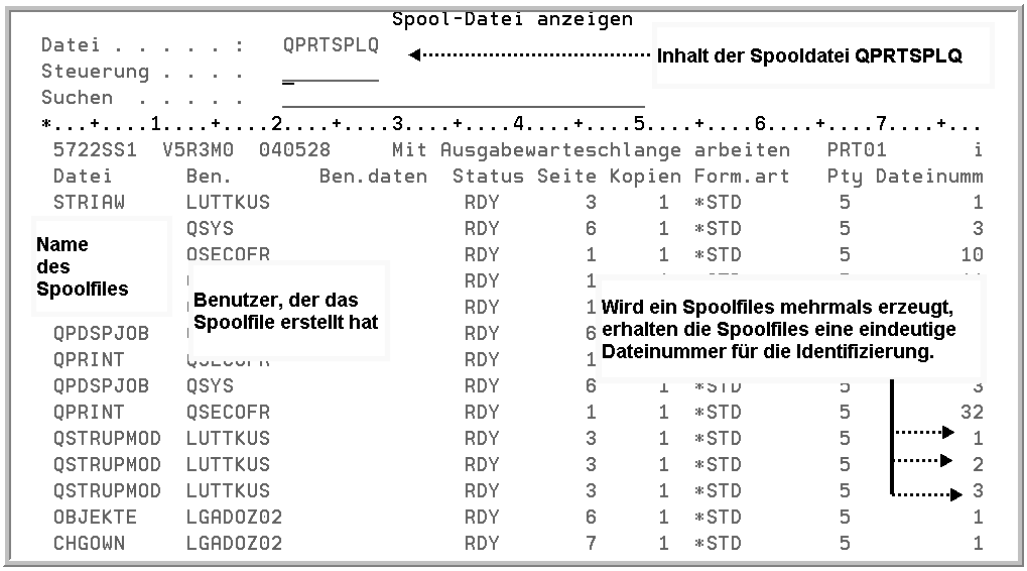
2.12.8.7
Seite 2

Der Befehl WRKOUTQ erlaubt uns nicht die Umleitung der Druckausgabe in eine Datenbankdatei. Dennoch gelingt uns die Umleitung, wenn wir mit *PRINT zunächst eine Druckausgabe erzeugen.



*Der Befehl WRKOUTQ OUTPUT (*PRINT)*

Die erzeugte Druckausgabe enthält alle Spoolfiles des Systems. Diese Informationen stehen anschließend in dem Spoolfile QPRTSPLQ.



Inhalt des Spoolfiles QPRTSPLQ

Leider ist es nicht möglich dieses Spoolfile im CL-Programm zu lesen. Daher kopieren wir die Daten des Spoolfiles in eine physische Datenbankdatei. Die physische Datei muss allerdings bereits existieren, wenn wir die Daten kopieren.

```
                    Physische Datei erstellen (CRTPF)

Auswahl eingeben und Eingabetaste drücken.

Datei  . . . . . . . . . . . . . > OUT          Name
  Bibliothek . . . . . . . . .     *CURLIB      Name, *CURLIB
Quellendatei . . . . . . . . .   QDDSSRC        Name
  Bibliothek . . . . . . . . .     *LIBL        Name, *LIBL, *CURLIB
Quellenteildatei . . . . . . .   *FILE          Name, *FILE
Satzlänge, falls nicht DDS . . > 132           Zahl
Generierungsbewertungsstufe  . .   20          0-30
Kennzeichnende Bewertungsstufe     0           0-30
Dateiart . . . . . . . . . . .   *DATA          *DATA, *SRC
Teildatei, falls erforderlich  .   *FILE        Name, *FILE, *NONE
Text 'Beschreibung'  . . . . . . > 'Ausgabedatei für WRKOUTQ'
_____
```

Die Ausgabedatei erstellen

Die Satzlänge der Datei orientiert sich immer an der Satzlänge der Druckdatei. In unserem Beispiel sind das 132 Byte analog unserer Druckdatei, die auch 132 Byte Satzlänge hat. Erst jetzt können wir die Druckausgabe mit dem Befehl CPYSPLF (copy spoolfile) in die Datenbankdatei kopieren.

```
                    Spool-Datei kopieren (CPYSPLF)

Auswahl eingeben und Eingabetaste drücken.

Spool-Datei  . . . . . . . . . . > QPRTSPLQ
Nach Datenbankdatei  . . . . . . > OUT
  Bibliothek . . . . . . . . . .   EPS002CR
Jobname  . . . . . . . . . . . .   *
  Benutzer . . . . . . . . . . .
  Nummer . . . . . . . . . . . .
Spool-Dateinummer  . . . . . . . > *LAST
Jobsystemname  . . . . . . . . .   *ONLY
Spool-Datei erstellt:
  Erstellungsdatum . . . . . . . > *LAST
  Erstellungszeit  . . . . . . .
Nach Teildatei . . . . . . . . .   *FIRST
Sätze ersetzen od. hinzufügen  .   *REPLACE
```

Name des zu kopierenden Spoolfiles

Wohin soll das Spoolfile kopiert werden?

Welches Spoolfile soll verwendet werden, wenn mehr als ein Spoolfile mit diesem Namen existiert?

Welches Member wird verwendet und was soll mit evtl. bestehenden Sätzen in der Datenbankdatei geschehen?

Der Befehl CPYSPLF

Den Inhalt der Datenbankdatei OUT können Sie sich anschließend mit dem Befehl DSPPFM(display physical file member) ansehen.

2.12.8.7
Seite 4

```
                    Physische Teildatei anzeigen
Datei  . . . . . :   OUT                 Bibliothek . . . :   EPS002CR
Teildatei  . . . :   OUT                 Satz . . . . . . :   1
Steuerung  . . . :   _________           Spalte . . . . . :   1
Suchen . . . . . :   ____________________________________
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
 5722SS1  V5R3M0  040528     Mit Ausgabewarteschlange arbeiten   PRT01       in  QUSRSYS     12.08.05   17:51:13
 Datei      Ben.       Ben.daten  Status Seite Kopien Form.art   Pty Dateinummer   Job        Nummer Datum    Uhrz
 STRIAW     LUTTKUS                RDY       3     1  *STD        5          1     QDFTJOBD   059758 28.07.05 11:3
 QPDSPJOB   QSYS                   RDY       6     1  *STD        5          3     QFILESYS1  059475 29.07.05 16:1
 QPRINT     QSECOFR                RDY       1     1  *STD        5         10     QPRTJOB    060011 29.07.05 17:3
 QPRINT     QSECOFR                RDY       1     1  *STD        5         11     QPRTJOB    060011 29.07.05 18:2
 QPRINT     QSECOFR                RDY       1     1  *STD        5         12     QPRTJOB    060011 29.07.05 19:1
 QPDSPJOB   QSYS                   RDY       6     1  *STD        5          3     QFILESYS1  060553 31.07.05 01:3
 QPRINT     QSECOFR                RDY       1     1  *STD        5         22     QPRTJOB    060011 01.08.05 08:5
 QPDSPJOB   QSYS                   RDY       6     1  *STD        5          3     QFILESYS1  060953 02.08.05 16:0
 QPRINT     QSECOFR                RDY       1     1  *STD        5         32     QPRTJOB    060011 03.08.05 10:2
 QSTRUPMOD  LUTTKUS                RDY       3     1  *STD        5          1     SINTGEF0A1 061414 03.08.05 13:0
 QSTRUPMOD  LUTTKUS                RDY       3     1  *STD        5          2     SINTGEF0A1 061414 03.08.05 13:0
 QSTRUPMOD  LUTTKUS                RDY       3     1  *STD        5          3     SINTGEF0A1 061414 03.08.05 13:0
```

Inhalt der physischen Datenbankdatei OUT

Die Datei OUT enthält den Inhalt unseres Spoolfiles. Es ist natürlich darauf zu achten, dass die physische Datei auch Überschriften und Fußzeilen enthält. Diese Sätze dürfen wir später nicht verarbeiten. Außerdem gibt es keine Datensatzfelder, so dass wir die benötigten Informationen aus dem Satz mit der Funktion %SST ausschneiden müssen. Die jeweiligen Positionen sind im Vorfeld zu ermitteln.

Fangen wir nun also an unser Programm zu kodieren:
Zunächst einmal benötigen wir vom Anwender den Namen der Ausgabewarteschlange und das Grenzdatum, d.h. unser Programm muss zwei Parameter empfangen. Eventuell einen dritten für den Namen der Bibliothek.

```
PGM PARM(&OUTQ &LIB &DATUSR)


DCL              &OUTQ              *CHAR              10
DCL              &LIB                              *CHAR
10
DCL              &DATUSR              *CHAR              6
DCL              &DATFIL              *CHAR              6
```

Wir gehen davon aus, dass der Anwender uns das Datum im Format JJMMTT übergibt, so dass hier eine weitere Aufbereitung nicht erforderlich ist.

Dann benötigen wir auch noch eine Datenbankdatei für die spätere Druckausgabe. Ich habe bereits die Datei OUT mit einer Satzlänge von 132 Byte erstellt.

```
DCLF FILE(*CURLIB/OUT)
```

Bevor wir die Ausgabedaten drucken, sollte sichergestellt sein, dass die Ausgabe nicht versehentlich gedruckt wird. Aus diesem Grund überschreibe ich vorsorglich die Attribute der Druckdatei QPRTSPLQ.

```
OVRPRTF FILE(QPRTSPLQ) HOLD(*YES)
```

Erst jetzt werden die benötigten Daten gedruckt.

```
WRKOUTQ OUTQ(&LIB/&OUTQ) OUTPUT(*PRINT)
```

Der WRKOUTQ Befehl hat die Druckdatei QPRTSPLQ erstellt, die den Status *HLD aufweist. Nun ist es möglich die Ausgabe in die Datenbankdatei zu kopieren. Da wir die Daten allerdings nur während der Ausführung unseres CL-Programms benötigen, erstelle ich eine neue Datei OUT in der Bibliothek QTEMP. Die Daten stehen Ihnen so exklusiv zur Verfügung und werden automatisch gelöscht, wenn Sie sich abmelden. Das System würde allerdings bei erneuter Programmausführung einen Fehler melden, wenn die Datei bereits existiert. Also prüfen wir einfach zunächst die Objektexistenz mit dem Befehl CHKOBJ und erstellen die Ausgabedatei nur, wenn sie bislang in der Bibliothek QTEMP nicht vorhanden ist.

```
CHKOBJ  OBJ(QTEMP/OUT) OBJTYPE(*FILE)
MONMSG CPF9801 EXEC(CRTPF FILE(QTEMP/OUT) RCDLEN(132))
```

Jetzt kopieren wir das Spoolfile in unsere Datenbankdatei. Um Verwechslungen auszuschließen müssen wir das zu kopierende Spoolfile näher spezifizieren, entweder indem wir die Jobdaten oder indem wir das zuletzt erstellte Spoolfile verwenden.

```
CPYSPLF FILE(QPRTSPLQ) TOFILE(QTEMP/OUT) SPLNBR(*LAST)
```

Das Spoolfile QPRTSPLQ ist nach dem Kopiervorgang überflüssig und sollte daher auch gleich wieder gelöscht werden.

```
DLTSPLF FILE(QPRTSPLQ) SPLNBR(*LAST) CRTDATE(*LAST)
```

Da wir im DCLF-Befehl die physische Datei OUT unserer Produktionsbibliothek verwendet haben, müssen wir die benannte Datei noch mit der im Programm erstellten Datei überschreiben. Hier berücksichtigen wir auch gleich die beiden Überschriftenzeilen, indem wir auf den dritten Datensatz positionieren.

```
OVRDBF  FILE(OUT) TOFILE(QTEMP/OUT) POSITION(*RRN 3)
```

Jetzt ist es soweit, wir können beginnen die Datenbankdatei zu lesen und beenden das Programm, wenn der letzte Datensatz verarbeitet wurde. Vergessen Sie aber nicht die Überschreibungen wieder zu löschen.

```
LOOP: RCVF
                  MONMSG MSGID(CPF0864) EXEC(DO)

                                        DLTOVR(QPRTSPLQ OUT)

                                        RETURN

                           ENDDO
```

Der gelesene Satz enthält das Erstellungsdatum der Druckausgaben. Dieses Datum benötigen wir als Vergleichswert. Allerdings müssen wir das Datum in der Datenbankdatei aufbereiten, da es im Format TT MM JJ gespeichert ist und dadurch für den Vergleich unbrauchbar ist.

```
  CHGVAR VAR(&DATFIL)                VALUE(%SST(&OUT 108 2)  *CAT +

                          %SST(&OUT 105 2)  *CAT +

                           %SST(&OUT 102 2))
```

Anschließend können wird die beiden Daten miteinander vergleichen und das Spoolfile löschen, falls die Bedingung wahr ist.

```
  IF COND(&DATFIL < &DATUSR) THEN(DO)
```

Für den DLTSPLF-Befehl benötigen wir weitere Angaben: den Benutzernamen, die Jobnummer, den Jobnamen, den Spooldateinamen und die Dateinummer. Diese Angaben sind erforderlich, um eine Spooldatei **eindeutig** im System zu identifizieren. Die im CL-Programm verwendeten Variablen müs-

sen Sie natürlich deklarieren und die Positionen ermitteln Sie anhand der Ausgabedatei.

```
     CHGVAR VAR(&USER) VALUE(%SST(&OUT 13 10)) /* Benutzer */
     CHGVAR VAR(&JOBNR) VALUE(%SST(&OUT 95 6)) /* Jobnummer */
```

```
     CHGVAR VAR(&JOB) VALUE(%SST(&OUT 84 10)) /* Jobname */
     CHGVAR VAR(&SPLDAT) VALUE(%SST(&OUT 2 10)) /* Spooldateiname */

     CHGVAR VAR(&NR) VALUE(%SST(&OUT 75 4))                /*Datei-
nummer */
     MONMSG CPF0000
```

Sie wundern sich über die MONMSG-Anweisung hinter der letzten CHG-VAR-Anweisung? Die Variable &NR ist eine *DEC Variable und da ich die Fußzeilen der Druckausgabe nicht gesondert berücksichtige, könnten in der letzten Zeile Daten in das Feld &NR übertragen werden, die nicht als Dezimalwerte darstellbar sind. Um in diesem Fall einen Programmabsturz zu verhindern, fange ich mit der MONMSG-Anweisung diese Fehler ab und lasse das Programm einfach weiter arbeiten. Als nächstes wird das Spoolfile gelöscht. Auch hier verwende ich wieder die MONMSG, damit die Fußzeilen keinen Programmabsturz verursachen.

```
                    DLTSPLF FILE(&SPLDAT) JOB(&JOBNR/&USER/&JOB)
SPLNBR(&NR)
                    MONMSG CPF0000
                    ENDDO
```

Das war's! Unser Programm kann jetzt in einer beliebigen Ausgabewarteschlange alle Spoolfiles löschen, die vor einem bestimmten Datum erstellt wurden. Und hier noch einmal das Programm im Zusammenhang, falls Sie vorhaben dieses Tool auf Ihrer Maschine zu nutzen.

2.12.8.7
Seite 8

```
PGM PARM( &LIB &OUTQ &USRDAT)

DCL &FILDAT   *CHAR LEN(6)
DCL &USRDAT *CHAR LEN(6)


DCL &LIB   *CHAR LEN(10)
DCL &OUTQ  *CHAR LEN(10)
DCL &JOB   *CHAR LEN(10)
DCL &USER  *CHAR LEN(10)
DCL &SPLDAT *CHAR LEN(10)
DCL &JOBNR  *CHAR LEN(6)
DCL &NR    *DEC  LEN(4 0)


DCLF  FILE(OUT)

        CHKOBJ  OBJ(QTEMP/OUT) OBJTYPE(*FILE)
        MONMSG CPF9801 EXEC(CRTPF FILE(QTEMP/OUT) RCDLEN(132))



        OVRPRTF FILE(QPRTSPLQ) HOLD(*YES)

        WRKOUTQ OUTQ(&LIB/&OUTQ) OUTPUT(*PRINT)

        CPYSPLF FILE(QPRTSPLQ) TOFILE(QTEMP/OUT) SPLNBR(*LAST)  +
          CRTDATE(*LAST) TOMBR(*FIRST) MBROPT(*REPLACE)

        DLTSPLF FILE(QPRTSPLQ) SPLNBR(*LAST) CRTDATE(*LAST)

        OVRDBF  FILE(OUT) TOFILE(QTEMP/OUT) POSITION(*RRN 3)

LOOP: RCVF
        MONMSG MSGID(CPF0864) EXEC(DO)
                                    DLTOVR (QPRTSPLQ OUT)
                                    RETURN
                                    ENDDO
```

```
     CHGVAR VAR(&FILDAT) VALUE( %SST(&OUT 108 2)    *CAT +
                                %SST(&OUT 105 2)    *CAT +
                                %SST(&OUT 102 2))
        IF COND(&FILDAT < &USRDAT) THEN(DO)
      CHGVAR VAR(&USER) VALUE(%SST(&OUT 13 10))
      CHGVAR VAR(&JOBNR) VALUE(%SST(&OUT 95 6))
      CHGVAR VAR(&JOB) VALUE(%SST(&OUT 84 10))
      CHGVAR VAR(&SPLDAT) VALUE(%SST(&OUT 2 10))
      CHGVAR VAR(&NR) VALUE(%SST(&OUT 75 4))
      MONMSG CPF0000
      DLTSPLF FILE(&SPLDAT) JOB(&JOBNR/&USER/&JOB) SPLNBR(&NR)
      MONMSG CPF0000
          ENDDO

        GOTO LOOP

ENDE:   ENDPGM
```

2.12.8.7

Seite 10

Verwendung von Bildschirmdateien in CL-Programmen

Die Verwendung von Bildschirmdateien ist eigentlich eher eine Aufgabe der höheren Programmiersprachen. Dennoch gibt es auch in CL-Programmen häufig die Notwendigkeit, Bildschirmdateien einzusetzen, um beispielsweise dem Anwender Parametereingaben zu erleichtern. Stellen wir uns vor, dass der Anwender eine SQL-Abfrage aufrufen soll und hierbei die Möglichkeit erhält, nicht nur nach verschiedenen Kriterien zu selektieren, sondern sich darüber hinaus noch verschiedener Sortierkriterien bedienen kann. Natürlich könnten wir für die Lösung ein entsprechendes RPG-Programm schreiben. Wir können diese Aufgabenstellung aber auch mit wenig Aufwand durch ein CL-Programm lösen. Alles, was wir benötigen, sind eine entsprechend parametrisierte Query-Abfrage und ein CL-Programm, das die Abfrage ausführt. Die Eingabe der Selektions- und Sortierkriterien müssten wir dann über einen Bildschirm realisieren. Dieser Bildschirm könnte folgendermaßen aussehen:

```
30.03.10                                                    EPSSRV
14:40:25                                                    RASCHE

                          Kundenliste


Mandantenummer:   1

Postleitzahl ..........: von:      _____        bis:     _____

Sortierung.............: Kundennummer  X        Kundenname  _




F3 = Programmende              F19 Befehlszeile
```

Der Eingabebildschirm

Um die Aufgabenstellung zu realisieren, benötigen wir also zunächst einmal einen Bildschirm oder besser gesagt eine Bildschirmdatei. Um DDS-Quellen von Bildschirmdateien zu bearbeiten bzw. neu zu erstellen, könnten wir natürlich die DDS-Quelle mit einem beliebigen Editor bearbeiten oder wir nutzen ein für die Bildschirmerstellung und -bearbeitung optimiertes Werkzeug, SDA oder – wenn Sie lieber mit RDI arbeiten – den Screen Designer.

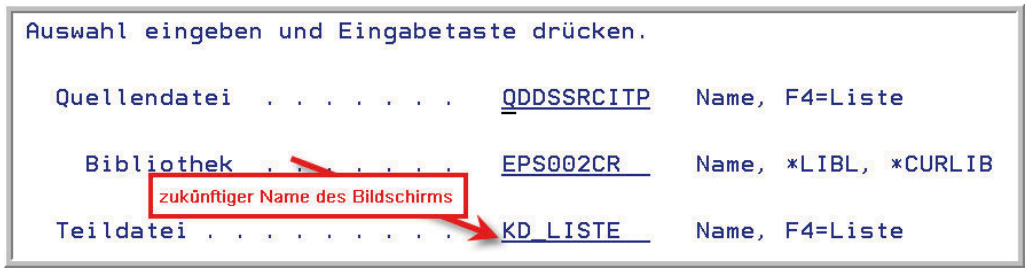
2.12.8.8

Seite 2

SDA – Screen Design Aid

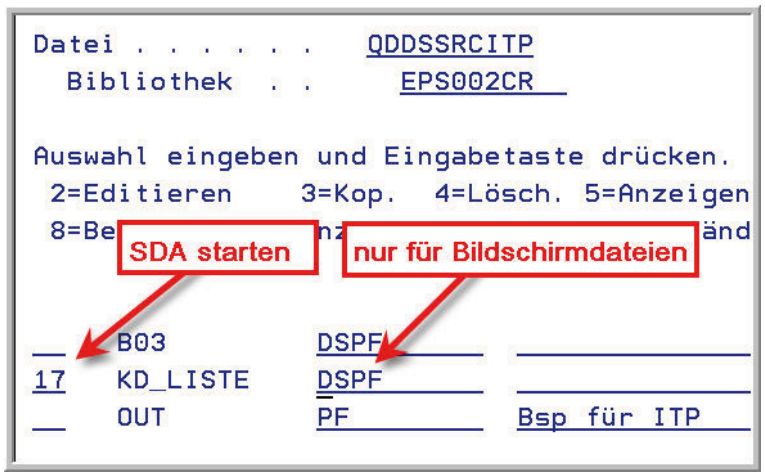
SDA ist ein sehr altes Werkzeug, mit dessen Hilfe wir 5250-Bildschirme gestalten können. Ich habe lange überlegt und viel mit Kollegen darüber diskutiert, ob es wirklich sinnvoll ist, in diesem Buch die alten Werkzeuge vorzustellen. Allerdings werden insbesondere die CL-Programme häufig von Systemadministratoren geschrieben und ich glaube nicht, das jeder Operator sich mit einem Entwicklungstool wie RDI auseinandersetzen wird, nur um mal ein kleines CL-Programm zu schreiben. Aus diesem Grund habe ich mich entschlossen – trotz vorhandener neuer Werkzeuge – auch SDA vorzustellen. Wir haben dieses Tool bereits für die Menüerstellung im Kapitel 2.1.2.1.2 dieses Buches verwendet, so dass Ihnen einige der nachfolgenden Funktionen vielleicht bekannt vorkommen.

Um SDA zu starten, können wir den CL-Befehl STRSDA (start screen design aid) verwenden. Im daraufhin erscheinenden Dialog wählen wir die Auswahl „1 Bildschirme entwerfen“ und müssen anschließend die Quellendatei und den künftigen Bildschirmnamen bekanntgeben:



SDA starten

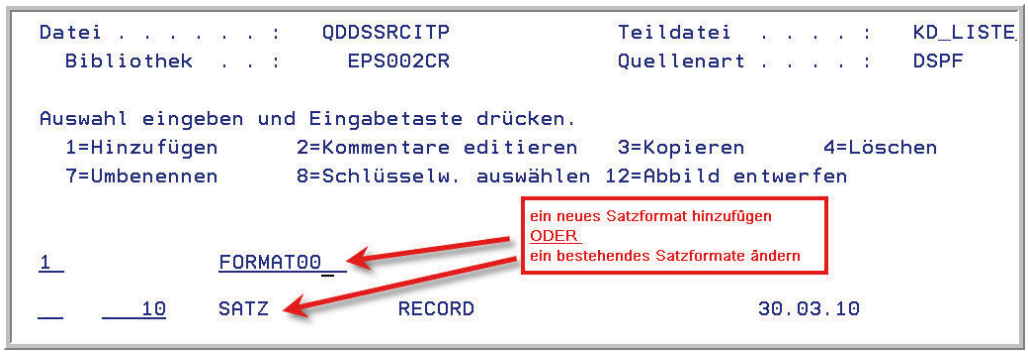
Wenn wir allerdings nur einen bereits bestehenden Bildschirm modifizieren wollen, geht's auch schneller:



SDA direkt aufrufen

2.12.8.8.1
Seite 2

Wir können SDA mit der Auswahl 17 direkt in PDM starten. Diese Option ist allerdings nur auf DSPF-Dateien anwendbar. Egal wie wir SDA starten, anschließend erhalten wir in jedem Fall den folgenden Dialog:



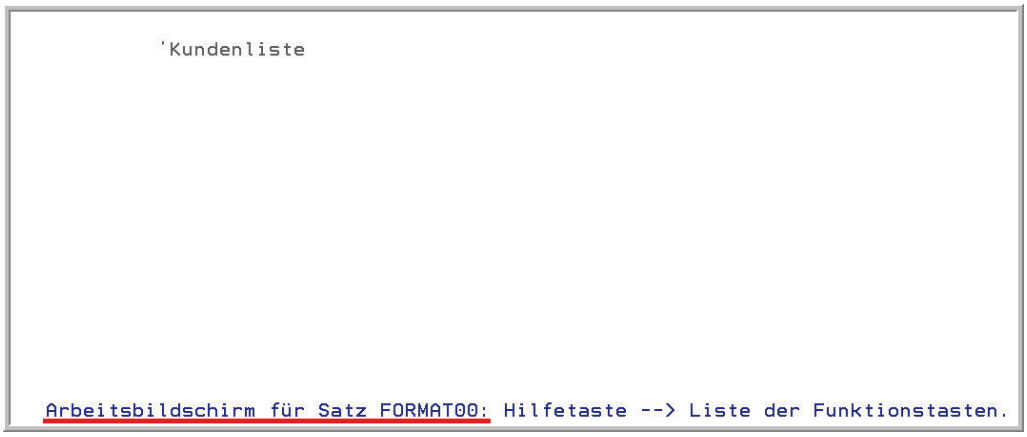
Satzformate hinzufügen oder ändern

Ist die Bildschirmdatei bereits vorhanden, sehen wir den existierenden Satzformatnamen und können diesen mittels Auswahl 12 bearbeiten. Ist die Bildschirmdatei hingegen neu, sehen wir nur den Namen, den Typ und den Speicherort der Datei, die zur Verarbeitung geöffnet wurde. In diesem Fall müssen wir mit der Auswahl „1“ ein neues Satzformat hinzufügen. In meinem Beispiel nenne ich das neu zu erstellende Satzformat „FORMAT00“. Damit teilen wir SDA mit, dass ein neuer Satz innerhalb der Datei anzulegen ist. Natürlich genügen diese Informationen noch nicht.

Satzformatart bestätigen

Mit SDA lassen sich verschiedene Bildschirmtypen erstellen. So erklären sich die vielen verschiedenen Satzformatarten. Für unseren einfachen Bildschirm genügt es jedoch, die Art RECORD beizubehalten; wir bestätigen daher einfach unsere Eingabe.

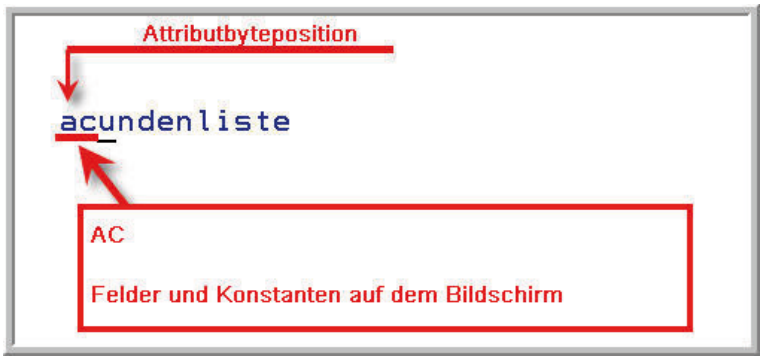
Der daraufhin erscheinende Bildschirm ist nicht sehr ermutigend; die leere Anzeige zeigt unseren Arbeitsbereich, in dem wir nun unseren Bildschirm gestalten können.



Der SDA-Arbeitsbereich

Geben wir hier zunächst unsere Überschrift ein. Positionieren Sie dafür den Cursor in eine beliebige Zeile des Arbeitsbereichs. Nun können Sie Ihre Überschrift einfach eingeben. Stellen Sie dabei lediglich sicher, dass der Text der Überschrift in einfache Hochkommata eingeschlossen ist. Stören Sie sich im Moment nicht daran, dass der Text nicht zentriert ist und bestätigen Sie Ihre Eingabe mit der Enter-Taste.

Wir haben eine Konstante definiert. Konstanten werden in SDA mit Hilfe von einfachen Hochkommazeichen angegeben. Auf die Hochkommata kann dann verzichtet werden, wenn die Überschrift – wie in unserem Beispiel – aus einem einzelnen Wort besteht, das keine Sonderzeichen beinhaltet. Nun wollen wir die Überschrift zentrieren. Wir verwenden dafür den Befehl AC. Geben Sie dazu den Buchstaben „A“ unmittelbar vor der zu zentrierenden Zeichenfolge ein und „überschreiben“ Sie die erste Stelle der Konstanten mit dem Buchstaben „C“, wie es in der folgenden Abbildung gezeigt wird.



Konstanten und Felder zentrieren

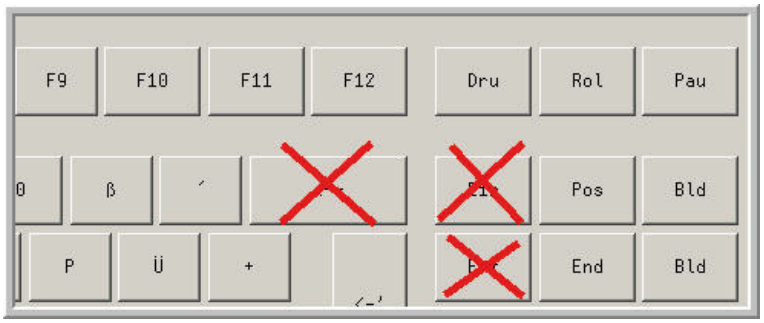
2.12.8.8.1
Seite 4

Jede Konstante und jedes Feld auf dem Bildschirm hat ein sogenanntes Attributbyte, das unmittelbar vor dem ersten Zeichen der Konstante bzw. des Feldes liegt. Sie brauchen sich also keine Gedanken zu machen, dass durch Ihre Eingabe der erste Buchstabe der Konstanten verschwinden würde – SDA erkennt die Eingabe und interpretiert diese entsprechend.

```
                        Kundenliste

Mandant.........:

Postleitzahlengebiet.........von:                bis:

Sortierfolge.........Kundennummer:               Kundenname:
```

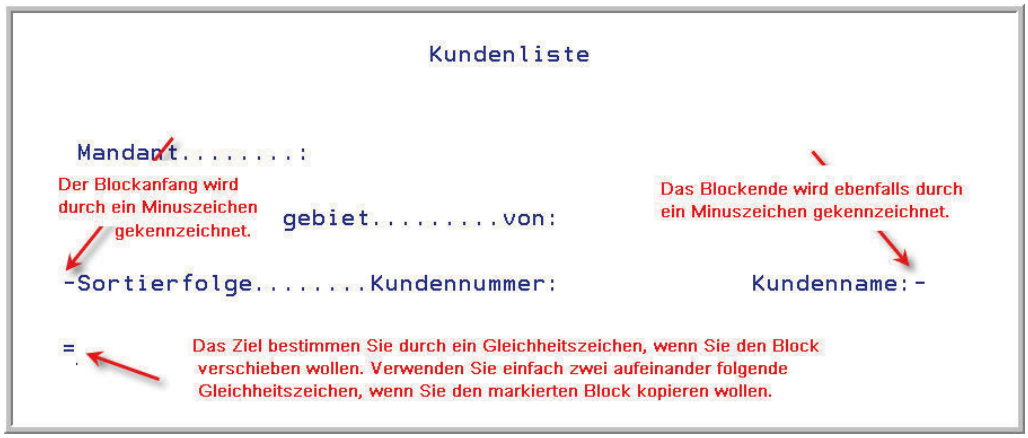
Der Arbeitsbildschirm mit diversen Konstanten

Eigentlich wäre das Arbeiten mit SDA sehr einfach, wenn es da auf der PC-Tastatur nicht die Tasten „Einfg“ und „Entf“ geben würde. SDA versteht den gesamten Bildschirmbereich als einen zusammenhängenden Bereich und verschiebt demzufolge alle Elemente des Bildschirms, sobald Sie die Tasten „EINFG“ oder „ENTF“ verwenden. Außerdem kann es dabei zum Löschen einzelner Attributbytes kommen, so dass SDA daraufhin die Felder und Konstanten nicht mehr zuordnen kann und sie dann einfach vom Bildschirm entfernt. Ärgerlich, wenn Ihnen das bei einem fast fertigen Layout passiert. Sie sollten sich daher von Anfang an daran gewöhnen, auf die besagten Tasten zu verzichten!



Verwenden Sie nicht die Tasten ENTF und EINFG

Falls Sie ein Feld löschen möchten, verwenden Sie stattdessen den Buchstaben D im Attributbyte der Konstanten oder des Feldes. Mit den Zeichen < bzw. > können Sie die Bildschirmelemente nach rechts oder links verschieben und blockweise können Sie kopieren oder verschieben, wenn Sie den Block mit zwei Minuszeichen umschließen und die Zielposition anschließend durch ein (=verschieben) bzw. zwei (=kopieren) Gleichheitszeichen bestimmen.



Blockweise verschieben oder kopieren

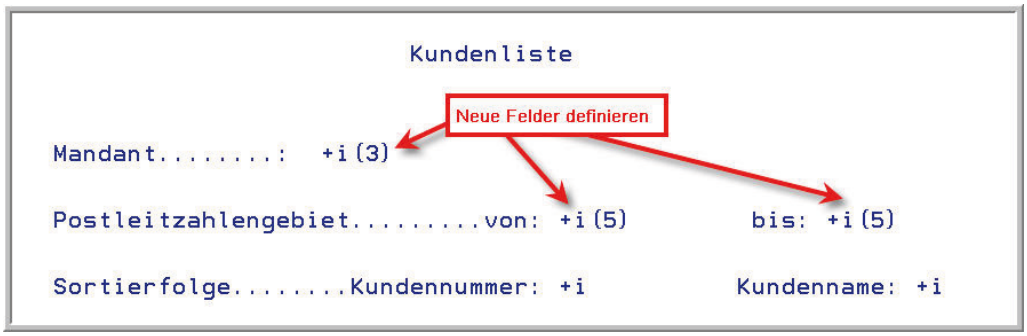
Als Nächstes möchte ich ein Systemdatum, die Uhrzeit, den Benutzer- und Systemnamen im Bildschirm anzeigen, wie es allgemein üblich ist. Dafür stehen mir folgende Synonyme zur Verfügung:

Synonym	Bedeutung
*SYSNAME	Servername
*USER	aktueller Benutzer
*DATE	aktuelles Systemdatum
*TIME	aktuelle Systemzeit

Systemdatum, Uhrzeit, Benutzer und Systemname einfügen

2.12.8.8.1
Seite 6

Fügen wir nun das erste „richtige“ Feld in den Satz ein, indem wir hinter der Konstanten „Mandant…..:“ den Wert +i (3) eingeben. Das Pluszeichen bedeutet, dass ein neues Feld in den Satz aufgenommen werden soll. Der Buchstabe i bestimmt, dass es sich bei dem neuen Feld um ein Eingabefeld handelt. Mit der Angabe (3) bestimmen wir die Länge des neuen Feldes. In unserem Beispiel legen wir also ein 3-stelliges alphanumerisches Input-Feld an. Die Anzeige sollte dann so aussehen wie in der folgenden Abbildung dargestellt:

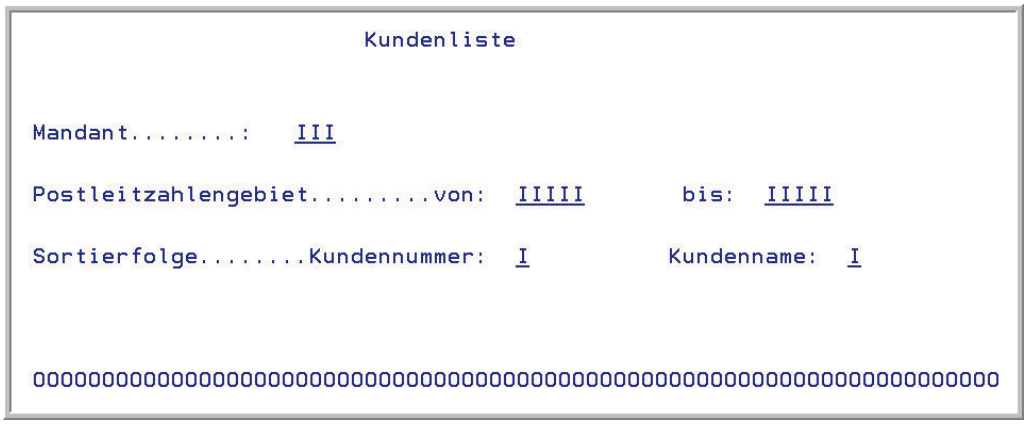


Neue Felder im SDA-Bildschirm definieren

Sobald ich den gezeigten Bildschirm bestätige, fügt SDA dem Satz 5 Eingabefelder hinzu. Wir können einem Satz natürlich auch andere Arten von Feldern hinzufügen. Mit der Angabe +B definieren wir ein neues alphanumerisches Ein-/Ausgabefeld; +O definiert ein Ausgabefeld. Numerische Felder werden mit +3 (Eingabefeld) bzw. +6 (Ausgabefeld) definiert. Numerische Ein-/Ausgabefelder definieren wir mit der Angabe von +9. Nachfolgend einige Beispiele:

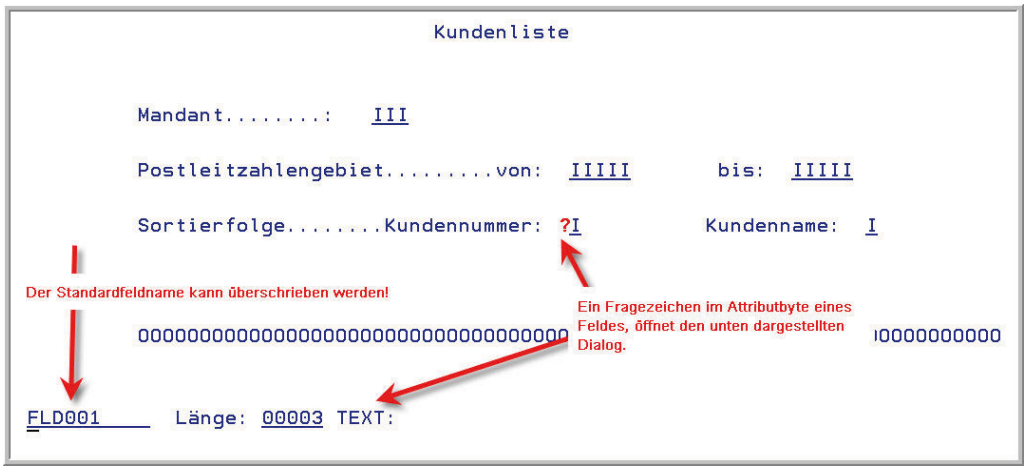
Definition	Erläuterung
+I(4)	4-stelliges eingabefähiges alphanumerisches Feld
+O(75)	75-stelliges ausgabefähiges alphanumerisches Feld
+B(10)	10-stelliges ein- und ausgabefähiges alphanumerisches Feld
+3(4)	4-stelliges eingabefähiges numerisches Feld ohne Nachkommastellen
+6(7,2)	numerisches Ausgabefeld mit 5 Vorkommastellen und 2 Nachkommastellen
+999999,99	10-stelliges ein- und ausgabefähiges numerisches Feld

Ich füge abschließend meinem Bildschirm noch ein 70-stelliges Ausgabefeld hinzu, über das ich später Meldungen ausgeben werde. Wenn ich alle Eingaben bestätige, sieht der Bildschirm folgendermaßen aus:



SDA-Bildschirm mit verschiedenen Feldern

Damit haben wir dem Bildschirm einige Felder hinzugefügt. Allerdings besitzt keines der Felder einen sprechenden Namen. Die einfachste Möglichkeit, den Feldnamen anzupassen, ist die Eingabe eines Fragezeichens (?) im Attributbyte des jeweiligen Feldes. Daraufhin erscheint am unteren Bildschirm ein neuer Dialog. Angezeigt wird hier der von SDA vergebene Standardname, der überschrieben werden kann.



Feldnamensänderung in SDA

2.12.8.8.1
Seite 8

Ich ändere jetzt die Feldnamen entsprechend ab. Anschließend möchte ich aber auch noch weitere Feldattribute ändern. Ich würde gern andere Farben zuordnen und dem Feld KDNR einen Standardwert zuweisen. Hierzu muss ich im Attributbyte der jeweiligen Felder oder der Konstanten das Asterixzeichen (*) verwenden. Wieder öffnet sich ein neuer Dialog:

```
                    Feld-Schlüsselwörter auswählen

Feld . . . . . :   FLD001              Verwendung:  I
Länge . . . . . :  3                   Zeile . . :  8    Spalte . . . :  30

Auswahl eingeben und Eingabetaste drücken.
                                  J=Ja    Für Feldart
  Anzeigeattribute . . . . . .    _       Alle außer verdeckt
  Farben . . . . . .              _       Alle außer verdeckt
  Eingabe-Auswahlmöglichkeiten . .  _     Verdeckt, Eingabe, Verdeckt/Eingabe
  Gültigkeitsprüfung . . . . . .  _       Eing. od. Ein-/Ausg., k. Gleitkomma
  Eingabe-Schlüsselwörter . . . .  =      Eingabe oder Ein-/Ausgabe
  Allgemeine Schlüsselwörter . . .  _     Alle Arten
```

Feld-Schlüsselwörter auswählen

Über die Feld-Schlüsselwörter können Sie grundsätzlich Darstellungsattribute wie Unterstreichungen, Blinken, Inversdarstellungen oder Farben zuweisen. Allerdings können die angezeigten Auswahlen variieren, abhängig davon, ob Sie ein Feld oder eine Bildschirmkonstante bearbeiten. So können Sie beispielsweise für ein Feld auch Gültigkeitsprüfungen oder Standardwerte vereinbaren, für eine Konstante natürlich nicht.

Probieren wir es aus. Ich gebe im Attributbyte der Überschrift ein Asterix-Zeichen (*) ein und wähle im anschließenden Dialog zunächst die Auswahl „Anzeigeattribute“.

```
                    Anzeigeattribute auswählen

Konstante . . . :  Kundenliste
Länge . . . . . :  11                  Zeile . . :  5    Spalte . . . :  35

Auswahl eingeben und Eingabetaste drücken.
                                  Schl.-Wort J=Ja   Bezugszahlen/+
  Feldbedingungen definieren . . . . . .                ___ ___ ___
  Programm-System-Übergabefeld . . . . . .  _________   ___ ___ ___
  Anzeigeattribute:               DSPATR
    Intensivanzeige . . . . . .     HI       _        ___ ___ ___
    Umkehranzeige . . . . . .       RI       _        ___ ___ ___
    Spaltentrennzeichen . . . . . .  CS      _        ___ ___ ___
    Blinken . . . . . .             BL       _        ___ ___ ___
    Anzeigeunterdrückung . . . . . .  ND     _        ___ ___ ___
    Unterstreichen . . . . . .      UL       J        ___ ___ ___
    Positionsanzeiger setzen . . . . . .  PC  _        ___ ___ ___
```

Anzeigeattribute auswählen

Durch die Eingabe von „J“ weise ich der Konstanten das Attribut „Unterstreichen“ zu. Viele der Attribute sind meiner Meinung nach selbsterklärend und müssen somit nicht näher erklärt werden. Außerdem finden Sie in SDA auch einige Erläuterungen, wenn Sie mit der Taste F1 die Hilfe aufrufen. Wenn auch das nicht ausreichend sein sollte, probieren Sie die verschiedenen Attribute einfach aus.

Die Verwendung des Asterix-Zeichens ist sicherlich einfach, aber auch aufwendig, wenn Sie vielen Bildschirmkonstanten und -feldern Anzeige- und/oder Farbattribute zuweisen wollen. In diesen Fällen geht es auch schneller: Sie können einige Attribute durch entsprechende Kurzformen direkt im SDA-Arbeitsbildschirm zuordnen:

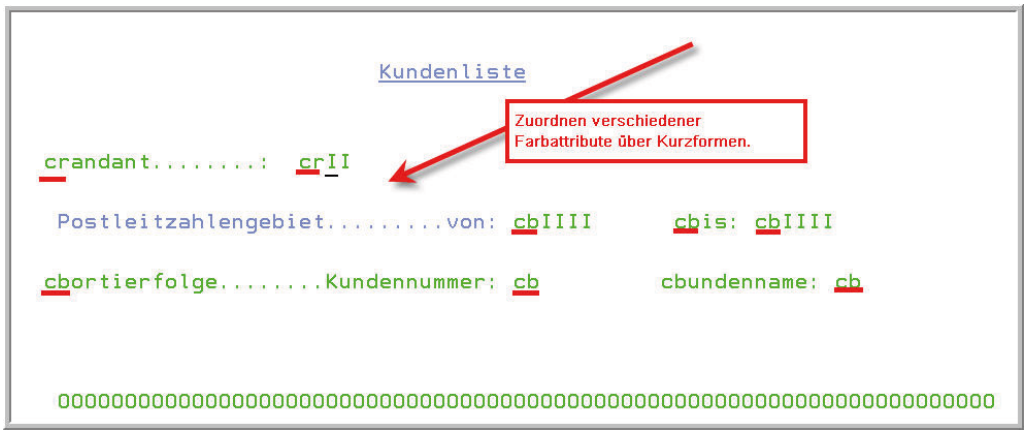
Kurzformen für Anzeigeattribute verwenden

Die Kurzform für die Unterstreichung ist der Buchstabe U. Ich kann jetzt einfach die Kurzformen im Attributbyte verwenden, um das Attribut dem Feld zuzuweisen; durch die Eingabe –U entferne ich das jeweilige Attribut wieder. Die nachfolgende Tabelle enthält eine Übersicht der vorhandenen Kurzformen und deren Bedeutung:

Attribut	Kurzform
Intensivanzeige	H
Umkehranzeige (Inversdarstellung)	R
Blinken	B
Anzeigeunterdrückung	N
Unterstreichen	U
Grün	CG
Blau	CB
Rosa	CP
Rot	CR
Türkis	CT
Weiß	CW

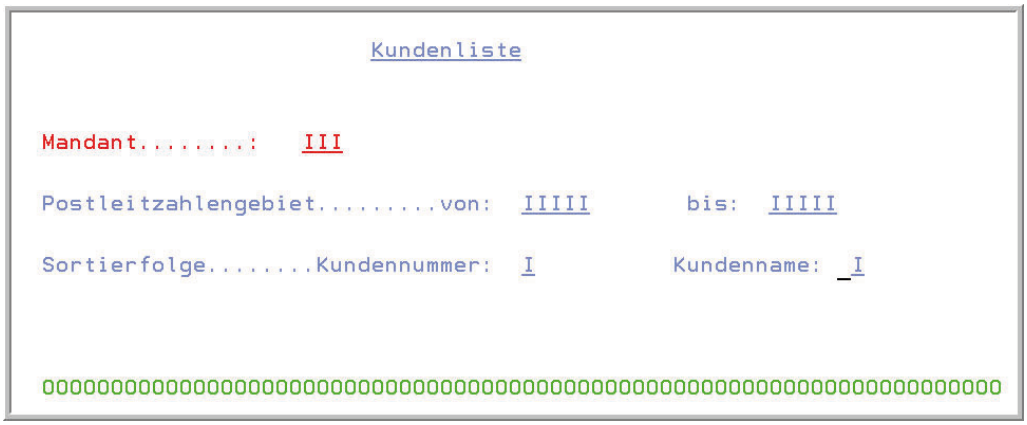
Auf diese Weise lassen sich insbesondere Farbattribute sehr einfach zuweisen und auch wieder entfernen.

2.12.8.8.1
Seite 10



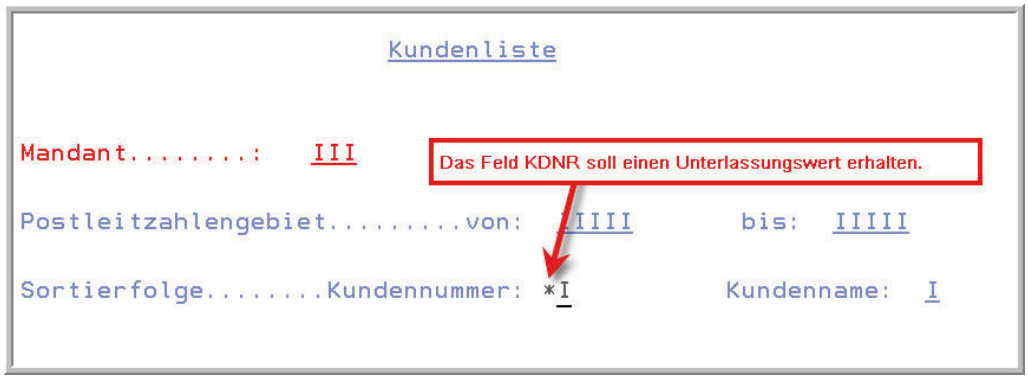
Zuordnen von Farbattributen über Kurzformen

Sobald ich meine Eingaben bestätige, sieht mein Bildschirm folgendermaßen aus:



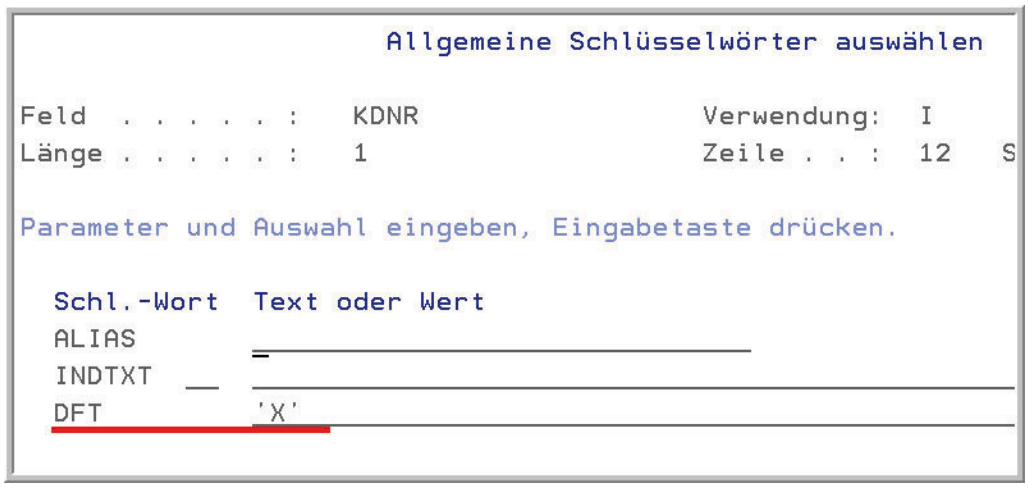
Das Zwischenergebnis

Unter anderem lässt sich mit Hilfe der Feldschlüsselwörter auch ein Defaultwert für ein Feld definieren. Damit wird diesem Feld bereits beim Aufruf der Anzeige ein Unterlassungswert zugewiesen, der selbstverständlich später bei Bedarf überschrieben werden kann. Ich will dem Feld KDNR das Zeichen X zuweisen, um die Standardsortierfolge zu definieren. Dazu gebe ich das Asterix-Zeichen im Attributbyte des Feldes KDNR ein.



Feldattribute zuweisen

Den gezeigten Bildschirm bestätige ich, so dass wieder der Bildschirm „Feld-Schlüsselwörter auswählen“ erscheint. Hier wähle ich dieses Mal die Option „Allgemeine Schlüsselwörter“. Daraufhin öffnet sich der folgende Dialog:



Standardwerte festlegen

2.12.8.8.1
Seite 12

Nachdem ich meine Eingaben bestätigt habe, kehre ich wieder in meinen Arbeitsbildschirm zurück. Hier können wir mit der F4-Taste auch eine Übersicht aller Felder aufrufen, die für den Satz definiert wurden. Ich nutze jetzt die Funktionstaste F4 und erhalte daraufhin die folgende Übersicht:

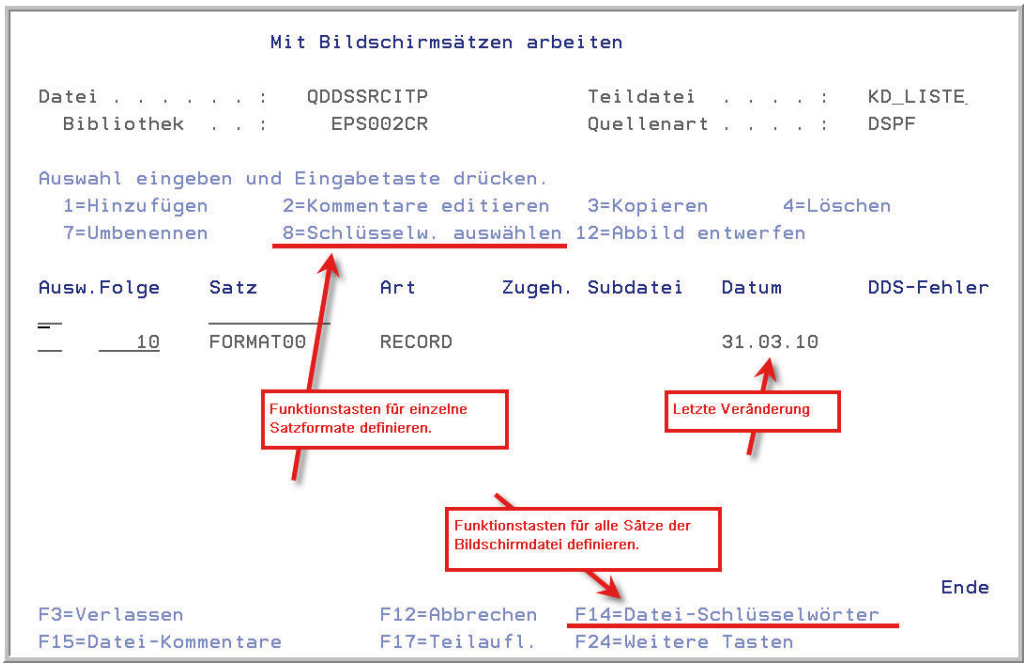
```
                              Mit Feldern arbeiten

Satz . . . . :   FORMAT00

Daten eingeben und Eingabetaste drücken.
  Anzahl der Felder für Blättern . . . . . . . . . . . . . . . .   4

Auswahlmöglichkeiten eingeben, Werte ändern und Eingabetaste drücken.
  1=Schl.wörter ausw.  2=Komm. editieren  4=Feld löschen

Auswahl  Folge   Feld        Art  Verw Länge   Zeil/Sp  Ref Bedingung  Überlapp
  =        30    Postleitza   C           33   10 011
  _        40    bis:         C            4   10 058
  _        50    Sortierfol   C           33   12 011
  _        60    Kundenname   C           11   12 057
  _        70    MANDANT      A   I        3   08 030
  _        80    VON          A   I        5   10 046
```

F4 Mit Feldern arbeiten

Wie Sie in der Abbildung sehen können, verfügt unser Satz derzeit über mehrere Konstanten und Felder. Die Folgenummern 30 bis 60 enthalten jeweils Konstanten – zu erkennen an dem „C“ in der Spalte „Art“ – während in der Folgenummer 70 ein Datenfeld „MANDANT“ definiert wurde. Dieses Feld wurde als Eingabefeld definiert, was wir an dem Eintrag „I“ in der Spalte „Verwendung“ erkennen. Außerdem finden wir in dieser Übersicht auch die Längen der einzelnen Felder und deren Position in dem ausgewählten Satz. Aus dieser Feldübersicht heraus können wir ebenfalls die Attribute aller Felder ändern, indem wir das gewünschte Feld mit der Auswahl „1“ selektieren.

Außerdem lassen sich an dieser Stelle auch die Feldnamen verändern, und die Reihenfolge der Felder lässt sich durch das Anpassen der Folgenummern verändern. Ich nutze diese Anzeige, um für meinen Postleitzahlenbereich den Wertebereich „00000“ bis „99999“ festzulegen und entsprechende Defaultwerte zu hinterlegen.

Jetzt ist mein Bildschirm fertig und kann umgewandelt werden. Mit der F12-Taste teile ich deshalb dem System mit, dass ich die Arbeit mit dem Satz „FORMAT00“ beenden will und kehre daraufhin zur Übersicht aller Bildschirmsätze zurück. Beachten Sie, dass jetzt unser neuer Satz in der Übersicht enthalten ist. Unter anderem sehen Sie hier auch, wann dieses Satzformat letztmalig verändert wurde. Schauen Sie sich dazu die folgende Abbildung an.



Übersicht mit Bildschirmsätzen arbeiten

SDA ist ein sehr mächtiges Tool, dass neben den Feldschlüsselwörtern auch noch Schlüsselwörter für das Satzformat und für die gesamte Bildschirmdatei kennt. Viele dieser Satz- und Dateischlüsselwörter sind sehr spezifisch und für CL-Lösungen häufig irrelevant, aber eine Angabe ist auch für unsere Lösung unumgänglich: Wir benötigen später die Funktionstaste F3, um unser Programm zu beenden. Die geplanten Funktionstasten müssen wir daher jetzt definieren. Dazu stehen zwei Alternativen zur Verfügung:

- Wir können die F3-Taste auf Dateiebene bekanntgeben. In diesem Fall gilt die Funktionstaste für alle Satzformate dieser Bildschirmdatei.
- Oder wir können die F3-Taste auf Satzebene – also für unser Format FORMAT00 – definieren, so dass die Funktionstaste nur bei der Verarbeitung des Satzformats FORMAT00 gültig ist.

Für meine Bildschirmdatei ist es unerheblich, ob ich die Funktionstaste auf Datei- oder auf Satzebene definiere, da die Bildschirmdatei lediglich ein einziges Satzformat beinhaltet. Ich entscheide mich für die Dateischlüsselwörter und drücke daher die Funktionstaste F14 im Bildschirm „Mit Bildschirmsätzen arbeiten“. Es öffnet sich ein weiteres SDA-Menü:

```
                         Datei-Schlüsselwörter auswählen

Teildatei   . :    KD_LISTE_2

Auswahl eingeben und Eingabetaste drücken.

                                        J=Ja
  Allgemeine Schlüsselwörter   . . .    _
  Bezugszahlen-Schlüsselwörter   . .    J
  Druck-Schlüsselwörter . . . . . .
```

„Datei-Schlüsselwörter auswählen“

Sie können sehen, dass ich jetzt den Punkt „Bezugszahlen-Schlüsselwörter“ wähle und daraufhin die folgende Anzeige erscheint:

```
                     Bezugszahlen-Schlüsselwörter definieren

Teildatei   . :    KD_LISTE

Schlüsselwörter und Parameter eingeben, Eingabetaste drücken.
  Bedingte Schlüsselwörter:    CFnn CAnn CLEAR PAGEDOWN/ROLLUP
                               HOME HELP HLPRTN
                                                                 T
Funktionstaste F3   Zugeordneter Indikator/Bezugszahl   Kommentar

Schl.-W.   Bezugszahl/+ Antw Text
CF03       ___ ___ ___   03  Programmende
_________  ___ ___ ___   __  ____________________________________
```

„Bezugszahlen-Schlüsselwörter definieren“

Zunächst gebe ich bekannt, dass die Funktionstaste F3 in dieser Bildschirmdatei definiert werden soll. Alle anderen Funktionstasten sind automatisch gesperrt und können später vom Anwender nicht genutzt werden. Außerdem muss natürlich mein CL-Programm die Funktionstaste F3 „gemeldet“ bekommen, wenn der Anwender die Taste F3 drückt. Die Kommunikation zwischen Bildschirmdatei und Programmobjekt kann auf einem System i Server über sogenannte Bezugszahlen bzw. Indikatoren erfolgen. Bezugszahlen sind nichts anderes als boolesche Variablen, die nur den Zustand „AN“ bzw. „AUS“ kennen. Eine Bezugszahl ist „AUS“, wenn Sie den Wert ‚0‘ enthält und sie ist „AN“, wenn der Wert ‚1‘ in der logischen Variablen steht. Ich kann also später

im Programm die logische Variable „03“ prüfen, um festzustellen, ob der Anwender das Programm beenden oder weitere Listen drucken will. Abschließend bestätige ich meine Eingaben und kehre auf diesem Weg zur Anzeige mit „Bildschirmsätzen arbeiten“ zurück.

Wenn keine weiteren Arbeiten in SDA erforderlich sind, dann können wir SDA jetzt mit einem erneuten Drücken der F12-Taste verlassen. Damit wird folgende Anzeige ausgegeben:

```
                  DDS sichern - Bildschirmdatei erstellen

Auswahl eingeben und Eingabetaste drücken.

  DDS-Quellenbestimmungen sichern  . . . .   J            J=Ja
    Quellendatei . . . . . . . . . . . . .   QDDSSRCITP    F4=Liste
      Bibliothek . . . . . . . . . . . . .   EPS002CR     Name, *LIBL ...
    Teildatei  . . . . . . . . . . . . . .   KD_LISTE     F4=Liste
    Text . . . . . . . . . . . . . . . . .   Kapitel 2.12.8.8.1
_____________

  Bildschirmdatei erstellen  . . . . . .   J            J=Ja
    Bedienerführung für Parameter  . . . .   _            J=Ja
    Bildschirmdatei  . . . . . . . . . . .   KD_LISTE     F4=Liste
      Bibliothek . . . . . . . . . . . . .   EPS002CR     Name, *CURLIB
    Bestehende Datei ersetzen  . . . . . .   J            J=Ja

  Job im Stapelbetrieb erstellen . . . . .   J            J=Ja

  Weitere Angaben zum
    Sichern oder Erstellen festlegen . . .   _            J=Ja
```

DDS sichern und Bildschirmdatei erstellen

SDA übernimmt für uns zwei Aufgaben:

- Einerseits wird aufgrund unserer Angaben im SDA die DDS-Quelle generiert und natürlich auch gesichert. Für die DDS-Quelle können wir u. a. einen beschreibenden Text angeben, der auch in der Übersicht der Quellen in PDM angezeigt wird und uns später die Arbeit erleichtert.
- Andererseits können wir mit dem Parameter „Bildschirmdatei erstellen“ direkt ein ausführbares Bildschirmobjekt erstellen, die SDA-Quelle also umwandeln. Der Standardwert für diesen Parameter wird mit „J“ vorbelegt.

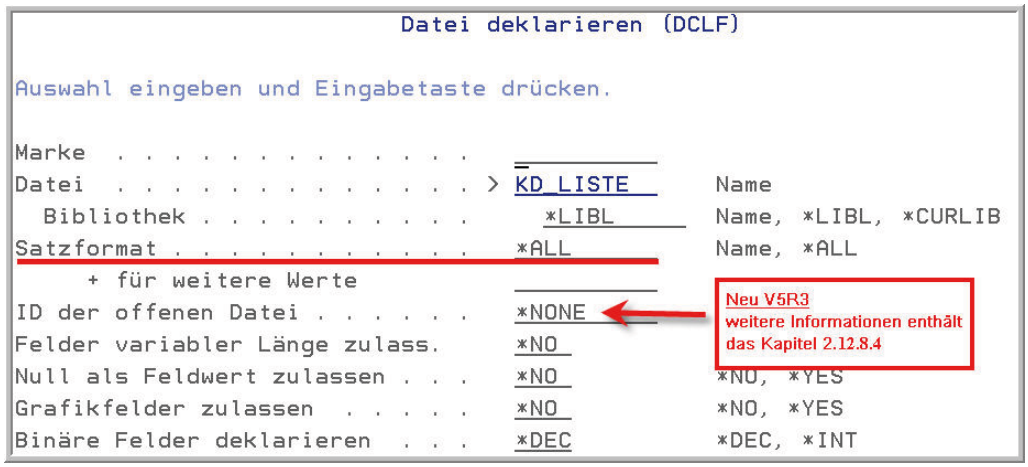
Wenn ich meine Eingaben jetzt mit der Entertaste bestätige, werden die Änderungen an der Quelle gesichert und das zugehörige Objekt wird generiert. Ist dies nicht gewünscht, kann ich die Umwandlung mit der F3-Taste abbrechen oder kehre mit der Taste F12 wieder zurück in den SDA-Definitionsbereich.

2.12.8.8.1

Seite 16

Verwendung von Bildschirmdateien im CL-Programm

Egal auf welche Weise wir die Bildschirmdatei erstellt haben, jetzt muss das erstellte Objekt im CL-Programm eingebunden werden. Ebenso wie die Datenbankdateien muss die Bildschirmdatei mit dem Befehl DCLF (declare file) deklariert werden, bevor sie verarbeitet werden kann.



Bildschirmdatei deklarieren

Jeder DCLF-Befehl enthält mindestens den Namen der Bildschirmdatei. Der zweite Parameter „Satzformat (RCDFMT)“ listet die Satzformatnamen auf, die das CL-Programm verarbeiten wird. Zur Vereinfachung ist *ALL als Standardwert vorgegeben, da in den meisten Fällen die im CL-Programm zu verarbeitenden Bildschirme lediglich ein Satzformat enthalten – so wie auch unser Beispiel. Falls Sie aber einmal eine Bildschirmdatei verarbeiten, die mehr als ein Satzformat enthält, von denen Sie aber nur ein Format benötigen, ist es vorteilhafter, im Parameter „Satzformat“ nur die entsprechend benötigten Formatnamen anzugeben. Der Parameter „ID der offenen Datei“ ist optional und identifiziert die Instanz der geöffneten Datei eindeutig. Vor dem Release V5R3 konnte nur eine einzige Datei je CL-Programm verwendet werden, d. h. der DCLF-Befehl konnte nur ein einziges Mal codiert werden. Seit dem Release V5R3 können bis zu fünf Dateien in einem CL-Programm angesprochen werden.

Es ist somit jetzt auch in einem CL-Programm möglich, neben einer Bildschirmdatei zusätzlich Datenbankdateien oder weitere Bildschirmdateien zu verwenden. Weitere Informationen zur Verwendung der Datei-ID enthält das Kapitel 2.12.8.4 dieses Buchs.

23. Ergänzung 6/2010

2.12.8.8.2
Seite 2

Wenn die Bildschirmdatei mit DDS erstellt wurde, kennt das CL-Programm alle in der Bildschirmdatei enthaltenen Felder, da der Compiler auf die externe Dateibeschreibung zugreift, um die Feldbeschreibungen abzurufen. Dies lässt sich sehr schnell feststellen, wenn Sie Ihr CL-Programm nur mit dem entsprechenden DCLF-Befehl umwandeln und sich anschließend das Umwandlungsprotokoll ansehen:

```
PGM

      DCLF       FILE(EPS002CR/KD_LISTE)
QUALIFIZIERTER DATEINAME - EPS002CR/KD_LISTE
   SATZFORMATNAME - FORMAT00
     CL-VARIABLE                ART       LÄNGE      GENAUIGKEIT   TEXT
     &IN03                      *LGL          1                    Programmende
     &MANDANT                   *CHAR         3
     &VON                       *CHAR         5
     &BIS                       *CHAR         5
     &KDNR                      *CHAR         1
     &KDNAM                     *CHAR         1
     &MSG                       *CHAR        70
Das CL-Programm kennt unsere Bildschirmfelder!
```

Das Umwandlungsprotokoll

In meinem Bildschirm habe ich u. a. das Feld MANDANT definiert, deshalb finde ich auch im CL-Programm die Variable &MANDANT, die ich – wie jede andere CL-Variable – uneingeschränkt nutzen kann. Außerdem müssen Sie wissen, dass alle in der Bildschirmdatei enthaltenen Bezugszahlen im CL-Programm als logische Variablen &IN01 bis &IN99 angesprochen werden. In meinem Bildschirm habe ich beispielsweise die Bezugszahl 03 verwendet, um die Funktionstaste F03 zu deklarieren. Dies ist der Grund, warum der Compiler mir die logische Variable &IN03 zur Verfügung stellt.

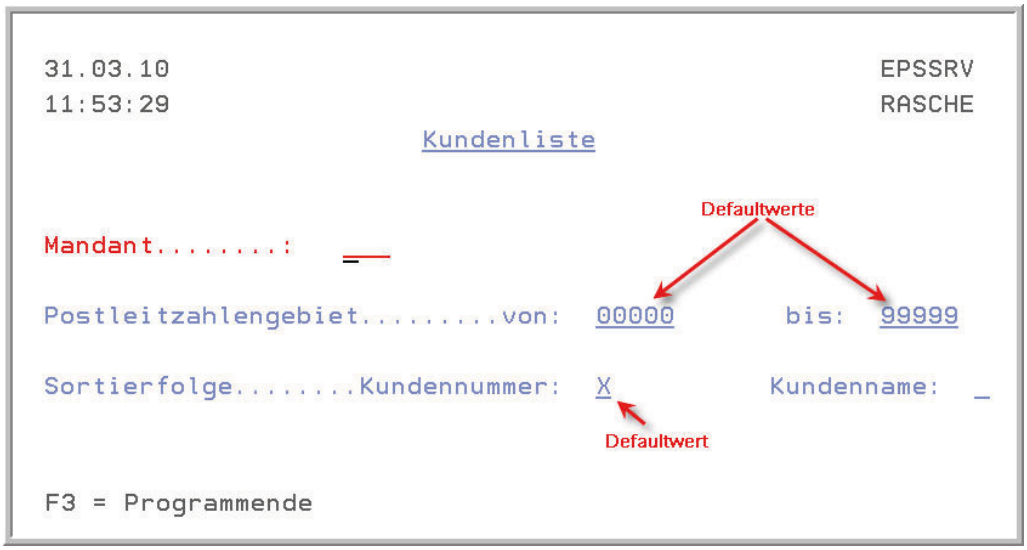
Nun brauchen wir noch den CL-Befehl SNDRCVF (send receive file), um den Bildschirm anzuzeigen und die Benutzereingaben zu verarbeiten. Mit dem Befehl SNDRCVF wird ein Bildschirm angezeigt, der auf die Benutzereingaben wartet und sie anschließend einliest.

```
PGM
      DCLF       FILE(EPS002CR/KD_LISTE)   Die Bezugszahl 03 ist AN, wenn die
                                           Funktionstaste F3 gedrückt wurde!
      DOUNTIL    COND(&IN03 = '1')  /* bis die Taste F3 gedrückt wird */

      SNDRCVF                       /* wird der Bildschirm gesendet   */
                                    /* und wieder eingelesen          */

      ENDDO
```

Bildschirm senden und empfangen

Unser Bildschirm sieht aber auch die Möglichkeit „F3 = Programmende“ vor. Daher wiederhole ich die Anzeige des Bildschirms, bis die Funktionstaste F3 vom Anwender gedrückt wird. Da es sich bei &IN03 um eine logische Variable handelt, ist die Codierung DOUNTIL COND(&IN03 = '1') nicht unbedingt notwendig. Ich könnte stattdessen auch einfach DOUNTIL COND(&IN03) codieren. Es bleibt Ihnen überlassen, welche Methode Sie bevorzugen.

Jetzt ist es an der Zeit, unser Programm ein erstes Mal zu testen:

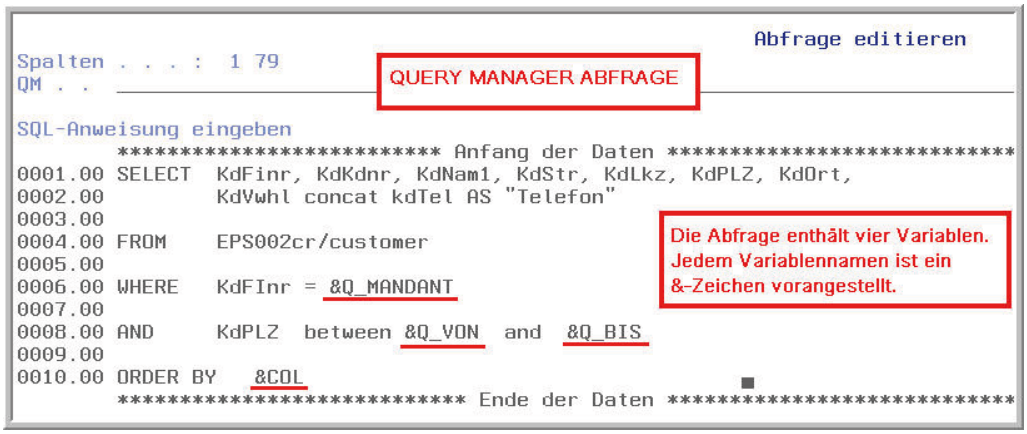


Das CL-Programm

Der Bildschirm wird wie erwartet angezeigt, meine Variablen KDNR, VON und BIS sind mit entsprechenden Werten vorbelegt und der Anwender kann jetzt die Mandantennummer eingeben oder die entsprechend vorbelegten Werte nach eigenen Wünschen ändern. Die Bildschirmeingaben stehen uns anschließend in den CL-Variablen zur weiteren Verarbeitung zur Verfügung.

Erinnern wir uns jetzt an die ursprüngliche Aufgabenstellung: Wir wollen eine Kundenliste entsprechend der Auswahlen im CL-Programm drucken. Was wir zusätzlich benötigen, ist eine entsprechende SQL-Abfrage oder besser gesagt eine QUERY-Abfrage, die Variablen enthält:

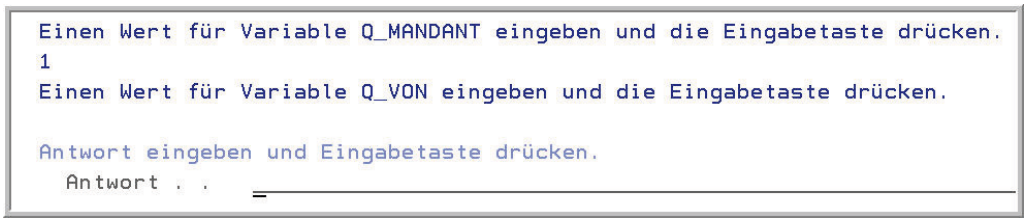
2.12.8.8.2
Seite 4



Die QUERY-Manager-Abfrage

Beachten Sie bitte, dass es sich bei der gezeigten Abfrage um ein mit dem Query Manger erstelltes Objekt handelt. Der Query Manger ist keinesfalls gleichzusetzen mit den Ihnen vielleicht bekannten QUERY/400-Objekten. Den Query Manger starten Sie mit dem CL-Befehl STRQM (start query manger); Sie erzeugen so ein Objekt des Typs *QMQRY. Innerhalb der Abfrage kann ich die WHERE-Bedingungen und die ORDER BY- Spalten parametrisieren. Zu beachten ist das vorangestellte &-Zeichen im Variablennamen. Außerdem ist zu beachten, dass die Variablennamen in Großbuchstaben eingegeben werden sollten, um später Fehler zu vermeiden.

Die so erstellte Abfrage kann ich anschließend mit dem CL-Befehl STRQMQRY ausführen. Aufgrund der verwendeten Variablen erhalte ich bei der Ausführung auf Befehlszeile zunächst entsprechende Abfragenachrichten, die es mir erlauben, die verwendeten QUERY-Variablen zu „füllen“.



Abfragenachrichten, um die Query-Variablen zu füllen

Es würde an dieser Stelle zu weit führen, die Funktionalität des Query Managers ausführlich vorzustellen. Hier verweise ich auf die IBM-Literatur oder andere Veröffentlichungen. Sie sollten aber auf jeden Fall vor der Verwendung des Querys im CL-Programm sicherstellen, dass Sie die Abfrage interaktiv auf der Befehlszeile ausführen können, bevor Sie den Aufruf in das CL-Programm verlagern.

Ich habe meine Abfrage entsprechend getestet und rufe jetzt das Query in meinem CL-Programm auf.

```
     DOUNTIL    COND(&IN03 = '1')  /* bis die Taste F3 gedrückt wird */

     SNDRCVF
     IF COND(&IN03) THEN(RETURN)
     STRQMQRY   QMQRY(EPS002CR/KUNDEN_LS) OUTPUT(*PRINT) +
                SETVAR((Q_MANDANT &MANDANT) (Q_VON &VON) +
                       (Q_BIS &BIS) (COL &COL))
     ENDDO
```

Variable Query

CL-Variable

Query in das CL-Programm integrieren

Zum Aufruf verwende ich – wie bereits erwähnt – den CL-Befehl STRQMQRY. Allerdings nutze ich den Parameter SETVAR, um die Query-Variablen durch die CL-Variablen meines Bildschirms zu füllen.

Zusätzlich möchte ich eine Meldung im Bildschirm anzeigen, dass die Abfrage erfolgreich ausgeführt wurde. Meldungen kann ich jetzt nicht mehr über die Befehle SNDPGMMSG oder SNDUSRMSG verschicken, sondern muss den Nachrichtentext über die Bildschirmvariable &MSG anzeigen. Hierzu verwenden wir einfach den CL-Befehl CHGVAR (change variable), den wir bereits aus Kapitel 2.12.6.2 kennen:

```
CHGVAR &MSG VALUE('Kundenliste für Mandant' *BCAT &MANDANT *BCAT +#
'wurde erfolgreich gedruckt!')
```

```
31.03.10                                                EPSSRV
16:24:29                                                RASCHE
                              Kundenliste


Mandant.......:  _

Postleitzahlengebiet.........von:  00000       bis:  99999

Sortierfolge........Kundennummer:  X          Kundenname:  _


KUNDENLISTE FÜR MANDANT 01 WURDE ERFOLGREICH GEDRUCKT!


F3 = Programmende
```

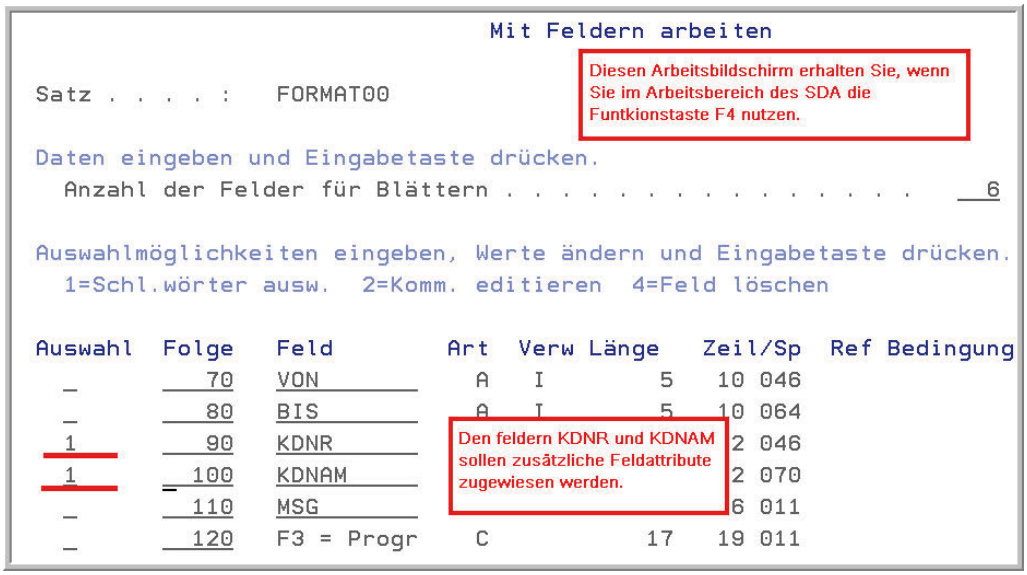
Das ausführbare CL-Programm

Das Programm scheint fertig zu sein; es sei denn, dass der Anwender Fehler beim Ausfüllen der Maske macht: Was passiert, wenn der Anwender keine Mandantennummer einträgt oder für die Sortierfolge beide Felder markiert? Wenn man noch ein wenig nachdenkt, fallen einem sicherlich schnell weitere Fehlerquellen ein.

Im Folgenden soll es nicht darum gehen, alle Fehlersituationen abzufangen. Aber meistens wollen wir den Cursor in diesen Fällen auf den fehlerhaften Wert positionieren und das entsprechende Feld ggf. invers darstellen, um den Anwender bei der Korrektur seiner Fehler zu unterstützen. Diese Wirkung erreichen Sie über entsprechende Feldattribute in der Bildschirmdatei. Da aber die Feldattribute nur bedingt gültig sind, müssen die Attribute dieses Mal durch entsprechende Bezugszahlen bedingt werden, so dass im Programm das Attribut durch Setzen der zugehörigen Bezugszahl aktiviert oder deaktiviert werden kann. Verwirrend? Ich zeige Ihnen, was gemeint ist.

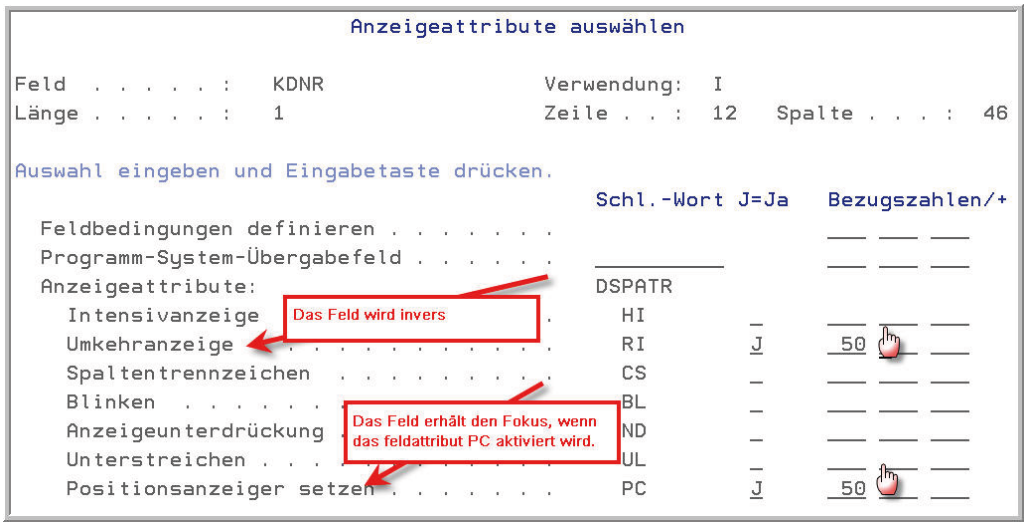
Ich möchte im Programm prüfen, ob entweder das Feld „Kundennummer“ oder das Feld „Kundenname“ als Sortierfolge gewählt wurde. Im Fehlerfall werde ich den Cursor auf das Feld Kundennummer platzieren und beide Felder – also die Kundennummer und den Kundennamen – invers darstellen.

Als Erstes öffne ich wieder unsere Bildschirmdatei in SDA und wähle für das Feld „KDNR“ die Feldattribute:



Mit Feldern arbeiten

Beginnen wir mit dem Feld KDNR. Es erhält die Cursorposition und soll invers dargestellt werden:



Bedingte Feldattribute

2.12.8.8.2
Seite 8

Sie sehen in der Abbildung oben, dass ich dem Feld KDNR die Attribute „Umkehranzeige (RI)“ und „Positionsanzeiger setzen (PC)“ zuweise. Dieses Mal verwende ich aber zusätzlich einen Eintrag in der Spalte Bezugszahlen. Durch den Eintrag gebe ich bekannt, dass die Attribute nur gültig sind, wenn die Bezugszahl 50 „AN“ gesetzt ist. Für das Feld KDNAM verfahre ich ähnlich. Allerdings verzichte ich auf das Attribut „Positionsanzeiger setzen“ und wähle nur die Inversdarstellung. Die Bezugszahl behalte ich aber auch für das Attribut des Feldes KDNAM bei. Schließlich will ich alle Attribute gemeinsam im CL-Programm aktivieren oder deaktivieren.

Jetzt muss ich noch die Prüfung in mein CL-Programm integrieren.

Als Erstes codiere ich eine Prüfroutine:

```
**********************UNTERROUTINEN*****************************
 SUBR PRUEFEN
  IF COND(&KDNR = ,X‘ *AND &KDNAM = ,X‘) THEN(DO)
   CHGVAR VAR(&MSG) VALUE(‘Sie können nur ein Felder +
                           als Sortierfolge auswählen!‘)
   CHGVAR &IN50 VALUE(,1‘)
  ENDDO
  IF COND(&KDNR = ,X‘) THEN(CHGVAR &COL VALUE(,KDKDNR‘))
  IF COND(&KDNAM = ,X‘) THEN(CHGVAR &COL VALUE(,KDNAM1‘))
 ENDSUBR
ENDPGM
```

In dieser Unterroutine prüfe ich zunächst, ob beide Felder markiert wurden. Ist dies der Fall, fülle ich mein Nachrichtenfeld und setze die Bezugszahl 50 auf den Wert ‚1‘; ich schalte die Bezugszahl an.

Wenn nur eines der beiden Felder markiert wurde, weise ich meiner Variablen &COL den Spaltennamen zu, nach welchem die Liste sortiert werden soll. Die Variable &COL bestimmt später im Query die ORDER BY-Klausel meiner SQL-Abfrage.

Nun muss ich nur noch das Query bedingt ausführen. Schauen wir uns hierzu den Hauptteil des CL-Programms an:

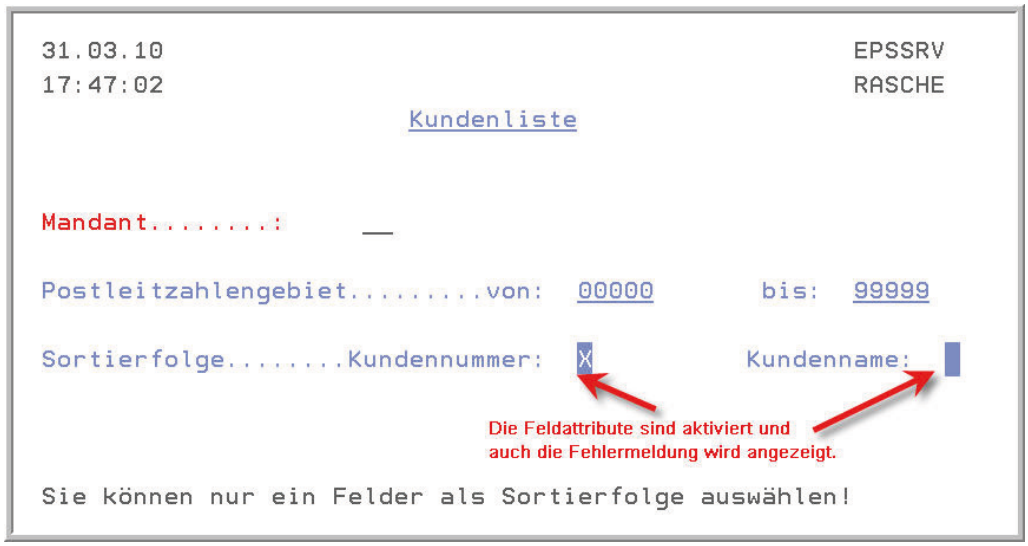
```
CALLSUBR PRUEFEN
IF COND(*NOT &IN50) THEN(DO)
  STRQMQRY QMQRY(EPS002CR/KUNDEN_LS) OUTPUT(*PRINT) +
            SETVAR((Q_MANDANT &MANDANT) (Q_VON &VON) +
                   (Q_BIS &BIS) (COL &COL))
  CHGVAR &MSG VALUE('KUNDENLISTE FÜR MANDANT' *BCAT &MANDANT +
          *BCAT 'WURDE ERFOLGREICH GEDRUCKT!')
  ENDDO
ENDDO
```

Wenn ich meine Unterroutine verlasse, ist die Bezugszahl &IN50 gesetzt, falls der Anwender beide Sortierfelder ausgewählt hat, so dass ich diese Bezugszahl jetzt nutzen kann, um die Ausführung der Abfrage im Fehlerfall zu verhindern. Sie sehen, man kann sehr geschickt Bezugszahlen einsetzen, um Eingabefehler der Anwender zu behandeln. Sie dürfen nur nicht vergessen, sowohl die Fehlermeldungen als auch die Bezugszahl wieder zurückzusetzen. Der günstigste Zeitpunkt hierfür ist stets nach dem SNDRCVF-Befehl. Dies ist der Zeitpunkt, wo der Anwender die Nachricht gelesen hat und die Fehlersituation ggf. behoben wurde:

```
SNDRCVF
IF COND(&IN03) THEN(RETURN)
CHGVAR &IN50 VALUE('0') /* Meldungen und Fehlerattribute werden
                                                    aufgehoben */
CHGVAR &MSG VALUE(, ,)
CALLSUBR PRUEFEN
```

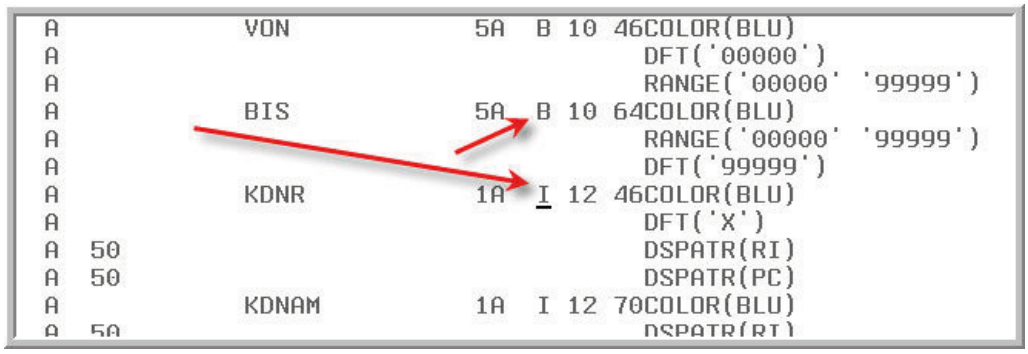
Nun müssen wir natürlich noch prüfen, ob unser Programm tatsächlich wie geplant reagiert.

2.12.8.8.2
Seite 10



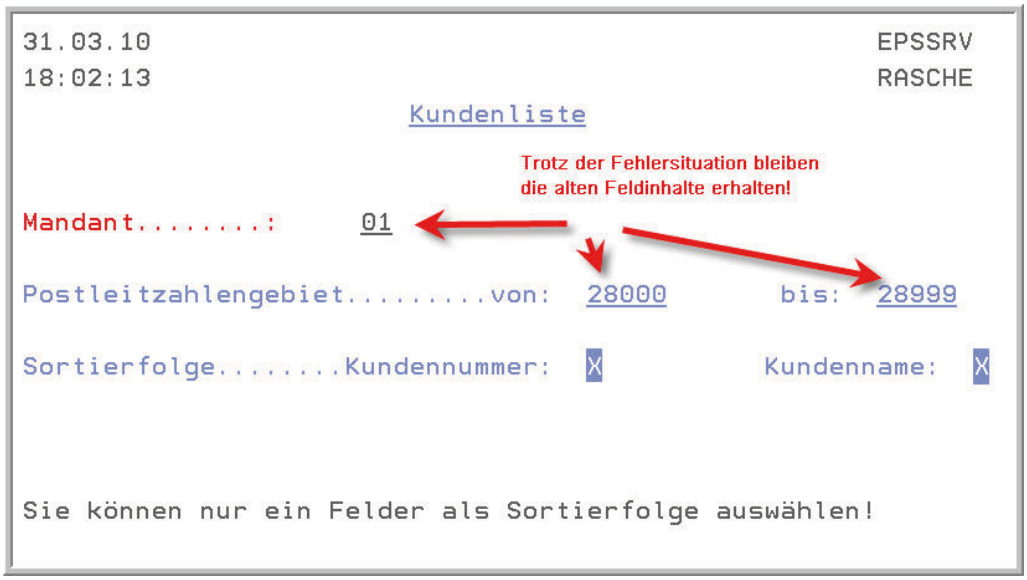
Das Programm mit bedingten Feldattributen

Die Fehlersituation konnte abgefangen werden. Etwas störend ist nur, dass die zuvor eingegebenen Werte nicht mehr angezeigt werden. Das lässt sich schnell ändern, wenn Sie die Feldart entsprechend ändern. Zu diesem Zweck müssen Sie natürlich nicht zwingend SDA aufrufen. Derartige Änderungen lassen sich auch schnell und einfach in der DDS-Quelle durchführen.



DDS-Quelle der Bildschirmdateien bearbeiten

Manchmal geht es einfach schneller, die DDS-Quelle direkt im Editor zu öffnen, um kleinere Änderungen vorzunehmen. Ich ändere meine Input-Felder auf Input/Output-Felder ab. Sie dürfen anschließend nur nicht vergessen, sowohl die Bildschirmdatei als auch das CL-Programm neu zu erstellen. Schauen wir uns abschließend die Wirkung unserer Änderungen an.



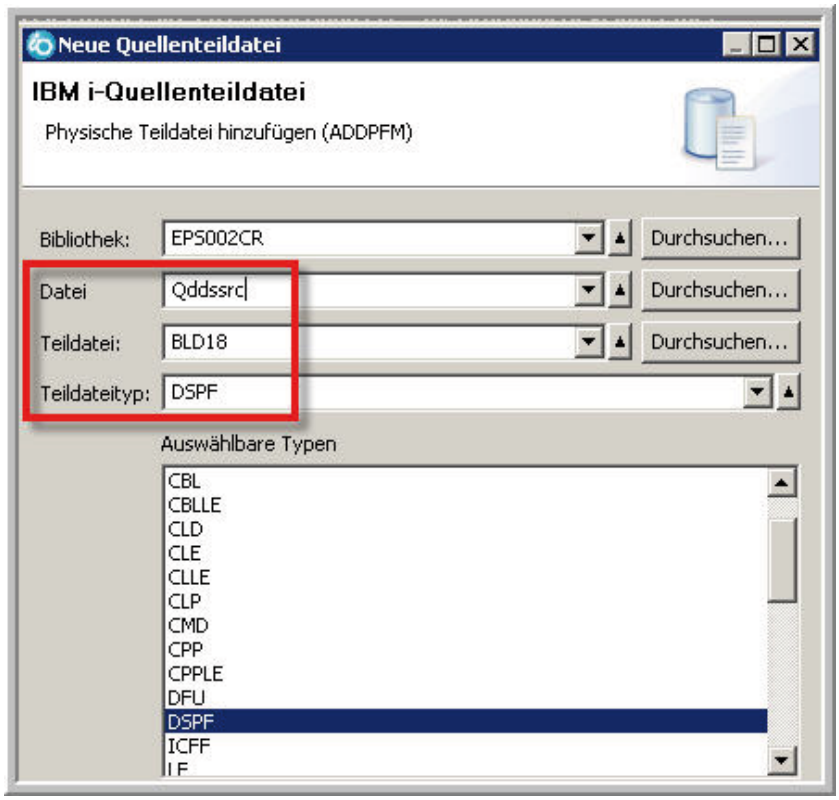
Das fertige CL-Programm

2.12.8.8.2

Seite 12

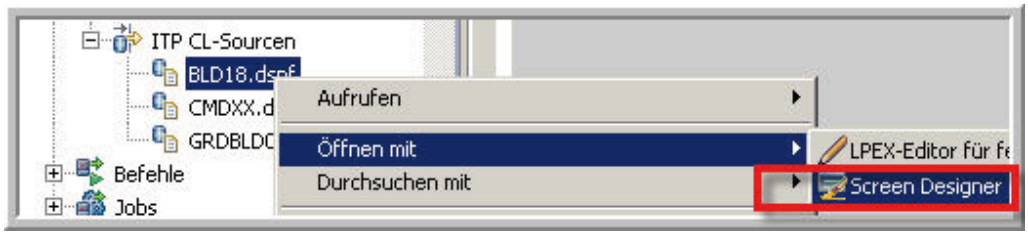
RDI – Screen Designer

Der eben verwendete Bildschirm kann natürlich auch mit RDI oder besser gesagt mit dem Screen Designer entworfen werden. An der Verarbeitung im CL-Programm wird sich nichts ändern. Der Entwurf des Bildschirms gestaltet sich allerdings ganz anders. Um den Screen Designer zu starten, benötigen Sie zunächst einmal eine leere Teildatei. Wurde bereits ein Filter definiert, ist dies einfach über die entsprechenden Kontextfunktionen Ihres Teildateifilters zu realisieren.



Neue DSPF-Teildatei erstellen

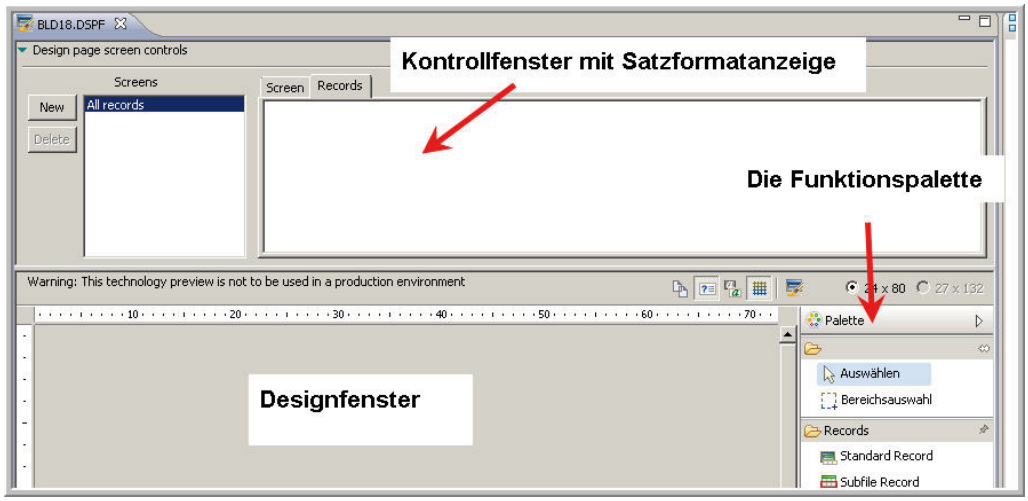
Wie mit SDA müssen Sie die Quellendatei benennen, um die DDS-Sourcen zu speichern. Außerdem braucht die Teildatei einen Namen, damit der Bildschirm im CL-Programm angesprochen werden kann. Die Teildateiart DSPF kennzeichnet letztendlich, dass es sich um eine Bildschirmdefinition handelt. Anschließend müssen Sie die Teildatei mit dem Screen Designer öffnen.



Die Teildatei wird mit dem Screen Designer geöffnet

2.12.8.8.3
Seite 2

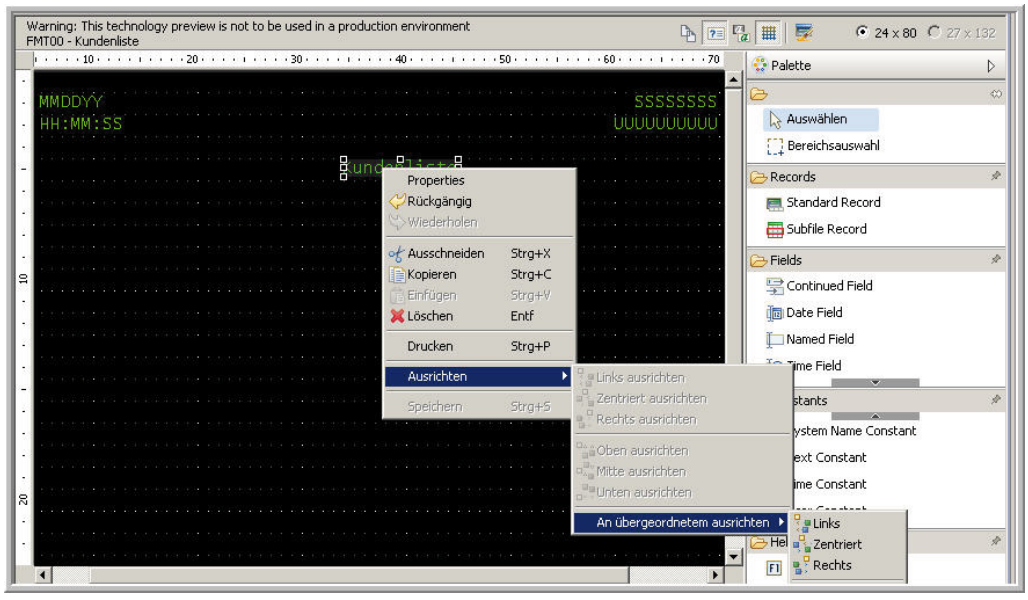
Daraufhin sollte sich der Screen Designer öffnen. Der Screen Designer mit seinem grafischen Interface ist natürlich ganz anders aufgebaut als SDA.



Grundaufbau – Screen Designer

Im Kontrollfenster werden die verschiedenen Satzformate angezeigt. Der markierte Satzname wird im Designfenster angezeigt und kann manipuliert werden. Für die Bearbeitung des jeweiligen Satzformats stellt die Funktionspalette verschiedene Objekte zur Verfügung.

Wir haben eine leere Bildschirmdatei geöffnet und müssen daher erst einmal das Satzformat definieren. Dafür klicke ich auf den Eintrag „Standardformat“ und bewege den Mauszeiger anschließend in das eigentliche Designfenster. Wenn alles funktioniert hat, wird der neu erstellte Satzname im oberen Fensterbereich angezeigt. Unsere Bildschirmdatei enthält jetzt ein Satzformat. Im nächsten Schritt müssen wir die Konstanten und Variablen hinzufügen.

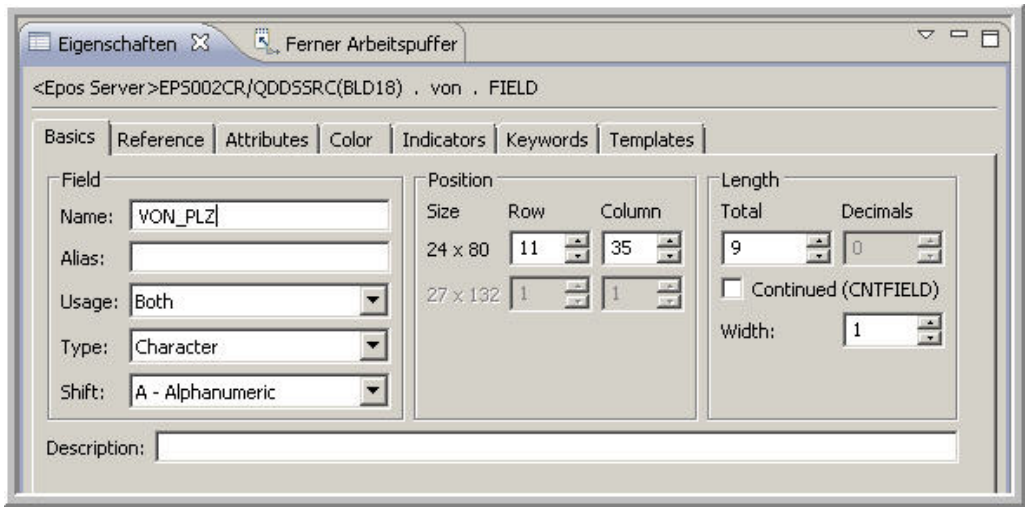


Konstanten definieren

Die Funktionspalette enthält eine Rubrik „Constants“. Hier finden Sie verschiedene Einträge um Datum (Date Constant), Uhrzeit (Time Constant), Benutzername (User Constant), Systemname (System Name Constant) oder einfache Textkonstanten (Text Constant) auf dem Bildschirm zu platzieren. Klicken Sie dafür einfach auf eine der Konstanten und ziehen Sie die Konstante dann auf das Design Fenster. Mit Hilfe des Kontextmenüs der jeweiligen Konstante können Sie anschließend die Konstanten auf dem Bildschirm ausrichten.

Fügen wir nun das erste „richtige“ Feld in den Satz ein, indem wir hinter die Konstante „Mandantennummer:“ ein benanntes Feld (Named Field) platzieren. Klicken Sie auf den Eintrag „Named Field“ in der Funktionspalette. Bewegen Sie anschließend wieder den Cursor an die gewünschte Zielposition. Wenn Sie erneut klicken, haben Sie ein benanntes Feld, also eine Bildschirmvariable definiert. Sollten Sie Felder und Konstanten vom Bildschirm entfernen wollen, markieren Sie das Objekt und drücken die ENTF-Taste. Probleme mit der ENTF-Taste oder der EINFG-Taste kennt der Screen Designer nicht.

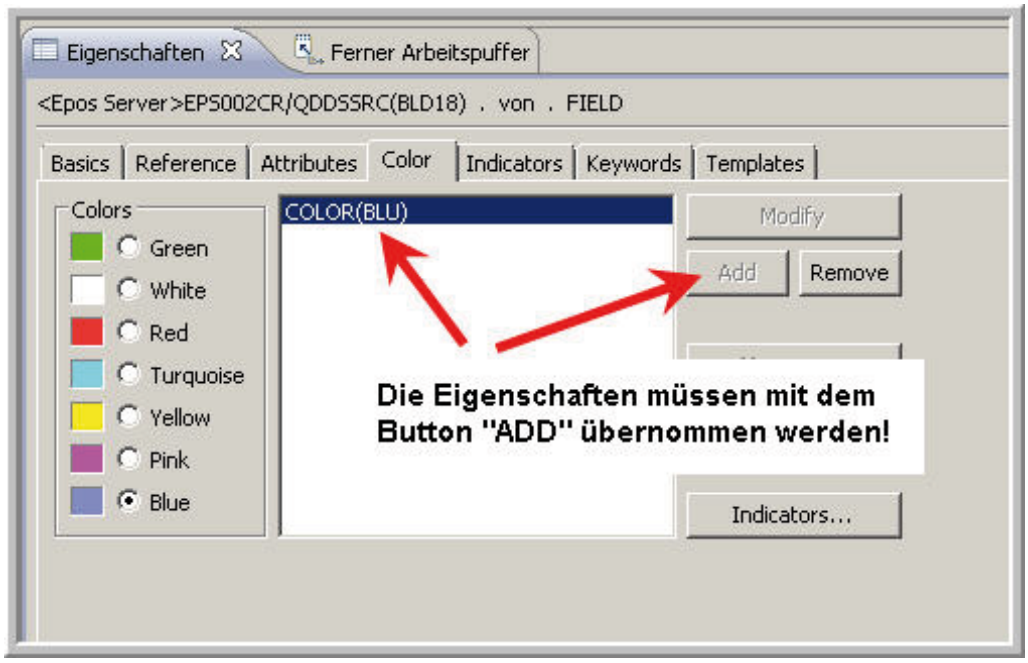
Jetzt markieren Sie das Feld. Im Register „Eigenschaften“ werden daraufhin die Feldeigenschaften angezeigt:



Eigenschaften des Feldes VON_PLZ

Auf der ersten Seite können Sie die Grundeigenschaften bestimmen: den Namen des Feldes, seinen Datentyp und natürlich die Länge. Auf den anderen Seiten des Registers „Eigenschaften“ lassen sich weitere Attribute wie beispielsweise blinkende Darstellung, verdeckte Anzeige und die Feldfarben zuweisen.

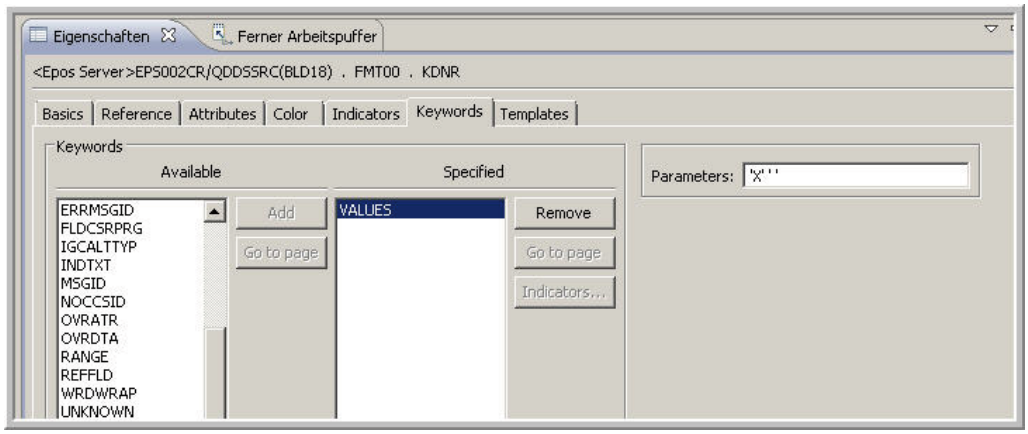
2.12.8.8.3
Seite 4



Farbattribute auswählen

Die Arbeit mit den Attributen ist einfach. Sie dürfen nur nicht vergessen, Ihre Auswahl mit dem Button „ADD“ zu bestätigen. Soll eine Eigenschaft gelöscht werden, nutzen Sie den Button „REMOVE“.

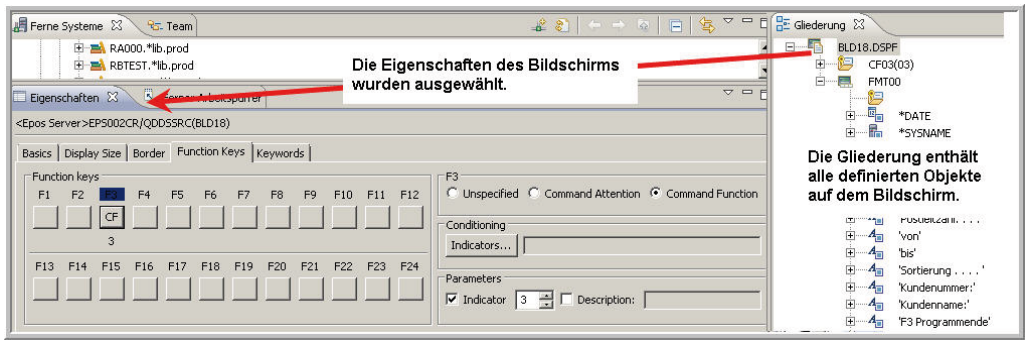
Etwas schwieriger ist die Eingabe von Gültigkeitswerten. Ich will beispielsweise sicherstellen, dass der Anwender später nur den Buchstaben X oder ein Leerzeichen eingeben kann, um die Sortierfolge zu bestimmen. Dafür markiere ich zunächst das Feld KDNR und wechsle in das Register „Keywords“ dieses Felds. Hier wähle ich das Schlüsselwort VALUES aus:



Gültigkeitsprüfungen nutzen

Die Definition ist etwas umständlich. Sie setzt immerhin eine gewisse Kenntnis der Feldschlüsselfelder voraus. Wir nutzen zur Lösung das Schlüsselwort VALUES und vereinbaren im Feld „Parameters:“ die gültigen Eingabewerte. Sind mehrere Werte gültig, dürfen Sie in der Werteliste keine Kommas verwenden.

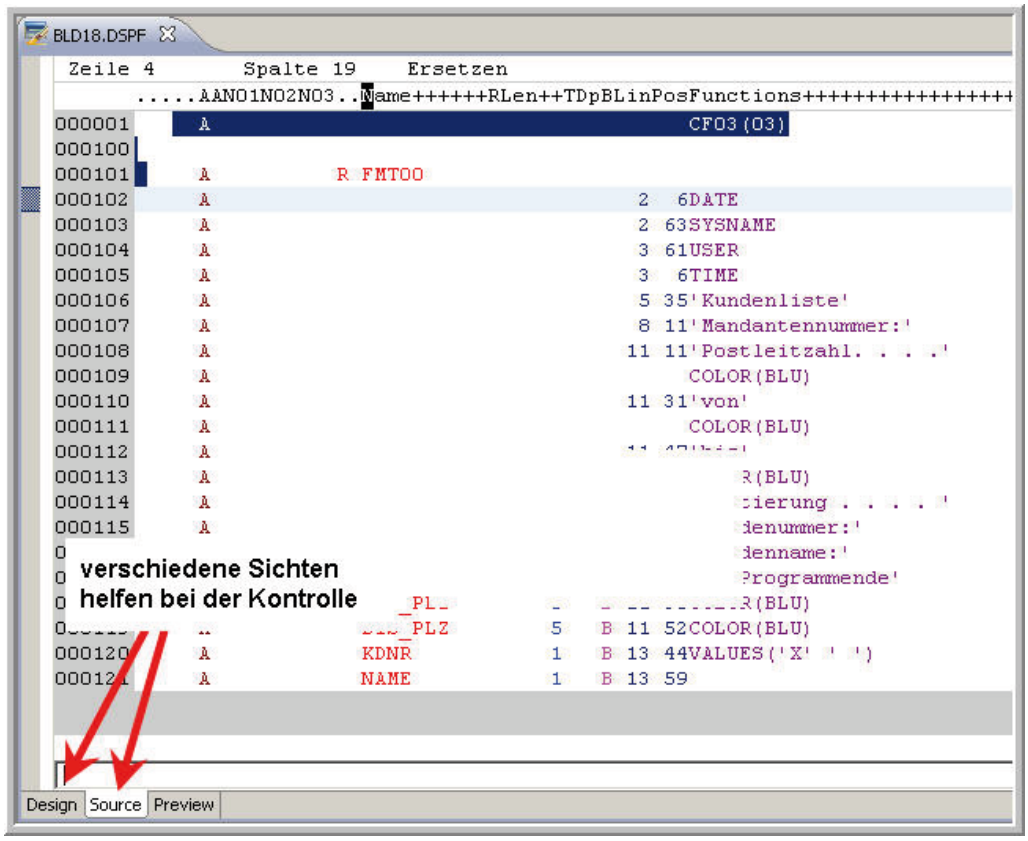
Wir haben bislang Felder definiert und diesen entsprechende Attribute zugewiesen, aber es fehlen die Funktionstasten. Nun wissen wir aus dem vorangegangenen Kapitel, dass es Satz- und Dateischlüsselwörter für Bildschirmdateien gibt. Im RDI treffen Sie die Unterscheidung am einfachsten, wenn Sie die Gliederung zu Hilfe nehmen. Die folgende Abbildung zeigt, dass in der Gliederung der Dateiname markiert ist. Deshalb werden die Dateieigenschaften angezeigt. Sie könnten ebenso gut den Satz oder ein Feld der Bildschirmdatei über die Gliederung auswählen, um die Eigenschaften zu bearbeiten. Die Gliederung ist mit der Funktionstaste F4 im Arbeitsbildschirm einer SDA-Sitzung zu vergleichen. Nur ist die Gliederung natürlich sehr viel übersichtlicher und daher auch einfacher zu handhaben.



Dateischlüsselwörter hinterlegen

Bleiben wir aber bei den gezeigten Dateiattributen. Um die Funktionstasten zu hinterlegen, öffne ich das Register „Function Key“ und wähle einerseits den „Function Key“ F3 und über die Schaltfläche „Parameters“ die Bezugszahl 03 als Antwortbezugszahl aus. Zuvor habe ich für die F3-Taste die Eigenschaft „Command Function“ gewählt. Der oben gezeigte Dialog erscheint komplex und fehleranfällig; er ist es auch. Schnell könnte man Bedingungsbezugzahlen anstatt der notwendigen Antwortbezugszahlen hinterlegen. Außerdem sollte man nicht vergessen, dass die Firma Rational den Screen Designer V7.5 als Preview herausgegeben hat; mit dem ausdrücklichen Hinweis, dieses Tool noch nicht im Produktivumfeld zu nutzen. Es bleibt also abzuwarten, wie die Entwicklung des Screen Designers weitergeht. Zurzeit ist es gut, dass wir jederzeit zwischen dem grafischen Interface und der original DDS-Beschreibung wechseln können:

2.12.8.8.3
Seite 6



Verschiedene Sichten stehen zur Verfügung

In der DDS-Source lässt sich schnell prüfen, ob die Definitionen korrekt sind. Nur brauchen wir dazu leider ein wenig DDS-Kenntnisse.

Wenn unser Bildschirm fertig ist, müssen wir auch im RDI die Bildschirmdatei kompilieren. Hier gibt es keine Besonderheiten. Markieren Sie einfach die entsprechende Bildschirmdatei in der Anzeige „Ferne Systeme“, öffnen Sie das Kontextmenü und wählen Sie den Eintrag „Kompilieren“. Den Umwandlungsprozess, die Fehlermeldung und deren Handhabung kennen Sie ja bereits durch die CL-Programmerstellung.

Kommunikation zwischen Programmobjekten

Wenn Sie auf einem System-i-Server komplexe Applikationen wie beispielsweise ein ERP-System einsetzen, dann besteht die Applikation aus einer Vielzahl von unterschiedlichen Programmen. Wenn viele unterschiedliche und „kleine" Programme entwickelt werden, werden die Programmerstellung und auch die Programmwartung wesentlich einfacher als bei der Verwendung von großen und unübersichtlichen Mammut-Programmen, in denen alle möglichen Verarbeitungen in einem einzigen Programm codiert sind.

Beachtet man die Richtlinien der modularen Anwendungsentwicklung, werden mehrere Programme verwendet, um eine Aufgabenstellung zu realisieren, so dass die einzelnen Abläufe auf unterschiedliche Programme verteilt werden. Dabei müssen natürlich Informationen zwischen den Programmen ausgetauscht werden. Ein Programm muss also in der Lage sein, mit einem anderen Programm zu kommunizieren. Dies kann auf unterschiedliche Arten realisiert werden:

– Man kann Parameter zwischen den Programmen austauschen. Dies ist vermutlich die einfachste und gebräuchlichste Art des Informationsaustausches. Allerdings funktioniert diese Art der Kommunikation nur, wenn die beteiligten Programmobjekte synchron, also zum gleichen Zeitpunkt ausgeführt werden.

– Sollen die Informationen zwischen zwei asynchron – also zu unterschiedlichen Zeiten – ausgeführten Programmen ausgetauscht werden, bieten sich sicherlich Nachrichtenwarteschlangen an. Mittels der Nachrichtenwarteschlangen kann das erste Programm Informationen in Form einer Nachricht senden und ein zweites Programm empfängt die Nachricht zu einem beliebigen Zeitpunkt. Über das Senden und Empfangen von Nachrichten haben wir bereits ausführlich gesprochen, so dass dieses Thema in diesem Kontext nicht wiederholt werden muss.

– Aber es gibt auch andere „Datenaustauschobjekte". Hierzu gehören sicherlich Datenbankdateien. Programm A schreibt Daten in eine Datenbankdatei, die anschließend zu einem beliebigen Zeitpunkt das Programm B wieder lesen und verarbeiten kann. Auf diese Methode müssen wir mit einem CL-Programm allerdings verzichten, denn CL-Programme können keine Informationen in Datenbankdateien schreiben, so dass wir auch diese Methode an dieser Stelle nicht weiter berücksichtigen werden.

– CL-Programme können aber Datenbereiche inklusive des lokalen Datenbereichs (Local Data Area – LDA) verwenden und so ebenfalls eine asynchrone Kommunikation realisieren.

– Zusätzlich zu den lokalen Datenbereichen verfügt jeder Job über 8 Schalter in Form von logischen Variablen. CL-Programme können die Schalter auf „ON“ oder „OFF“ setzen und somit Informationen basierend auf dem Status dieser Schalter weitergeben.

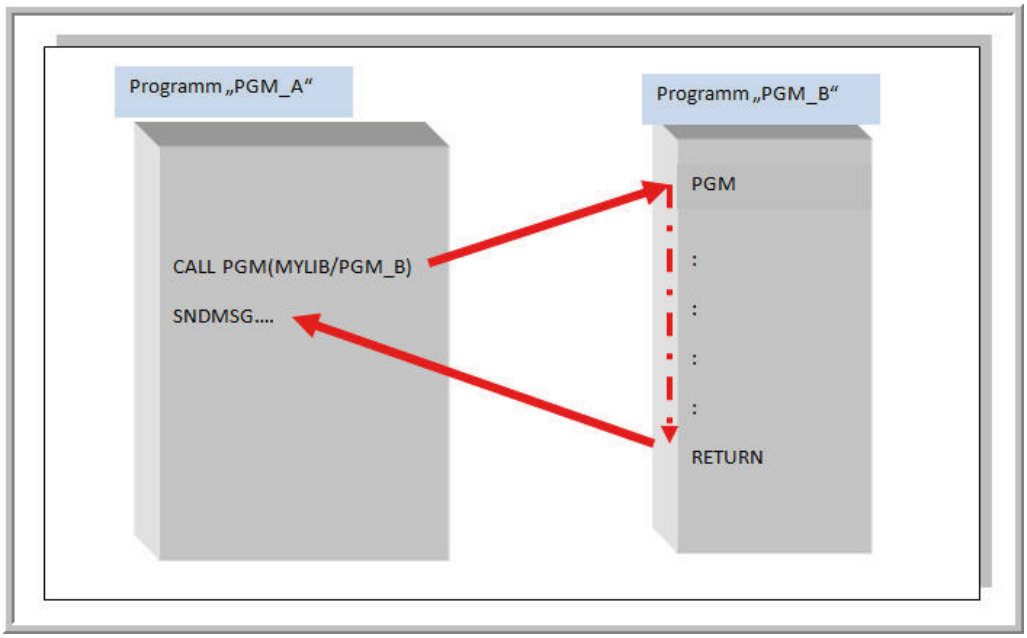
– Aber auch Datenwarteschlangen (Objektart *DTAQ) eignen sich asynchron Daten zwischen Programmen oder Jobs auszutauschen. Das Konzept der Warteschlangen ist Ihnen mittlerweile sicherlich hinlänglich bekannt. Datenwarteschlangen funktionieren ähnlich wie Nachrichtenwarteschlangen, Jobwarteschlangen oder Ausgabewarteschlangen, mit dem Unterschied, dass Sie selbst bestimmen, was in den Warteschlangen enthalten sein soll und wann Sie geleert werden.

Schauen wir uns die verschiedenen Methoden nun im Detail an.

Parameterverwendung in der Anweisung CALL

Sie können in CL die Befehle CALL, CALLPRC (Call Procedure) und TRFCTL (Transfer Control) dazu verwenden, um aus einem CL Programm ein weiteres Programm oder eine eingebundene Prozedur aufzurufen. Der Befehl CALLPRC kann nicht in OPM Programmen verwendet werden und wird von daher zu einem späteren Zeitpunkt in einem separaten Kapitel behandelt.

Mit der Anweisung CALL kann innerhalb eines CL-Programms ein weiteres Programm gestartet werden. Dabei ist es unerheblich, ob das zweite Programm in CL oder einer anderen Sprache geschrieben wurde. Mit der Anweisung CALL kann innerhalb eines Programms die Ausführung eines weiteren Programms gestartet werden.



Wirkungsweise der Anweisung CALL

Wird das zweite aufgerufene Programm beendet, wird die Steuerung wieder an das erste Programm zurück gegeben und die Anweisung, die der CALL-Anweisung folgt, kommt zur Ausführung.

Der CL-Befehl CALL hat nur zwei Parameter:

Der Parameter PGM
Er enthält den qualifizierten Namen des auszuführenden Programms. Wenn kein Bibliotheksname angegeben wird, dann wird das Programm über die Bibliotheksliste gesucht.

Der Parameter PARM
Dieser Parameter enthält die Parameter, die das aufrufende Programm anschließend empfängt.

2.12.9.1
Seite 2

Ein Beispiel für einen Programmaufruf ohne Parameterübergabe sehen Sie in der oberen Abbildung. Sollen Parameter übergeben werden, sieht der Programmaufruf wie folgt aus:

```
********** Datenanfang *************************************************
PGM
             DCL        VAR(&VAR01) TYPE(*CHAR) LEN(60)  VALUE('INHALT VAR01')
             DCL        VAR(&VAR02) TYPE(*DEC) LEN(6 0)  VALUE(02)

             CALL EPS002CR/PGM_B  PARM(&VAR01 'XX' &VAR02 )


             SNDPGMMSG  MSG('Zurück im Hauptprogramm') TOUSR(RASCHE)

ENDPGM
**********Datenende ***************************************************
```

Das aufrufende Programm

In diesem Beispiel wird das Programm PGM_B in der Bibliothek EPS002CR gestartet. Dabei werden drei Parameter übergeben, die für den Aufruf des Programms PGM_B erforderlich sind. Der erste Parameter ist die Variable „&VAR01“, der zweite Parameter ist der konstante, alphanumerische Wert „10“ und der dritte Parameter ist eine numerische Variable.

Wenn Sie Variablen als Parameter übergeben wollen, dann müssen Sie sicherstellen, dass die Parameter in beiden Programmen – dem aufrufenden und dem aufgerufenen Programm – zueinander passen:

- Der Name der Variablen muss in den beiden Programmen nicht übereinstimmen. Die Parameter werden bei einem Programmaufruf sequentiell übergeben und nicht in alphabetischer Form, wie es manchmal bei anderen Programmiersprachen der Fall ist.
- Hingegen müssen in den meisten Fällen die Variablen bezüglich des Datentyps und der Länge übereinstimmen.
- Außerdem muss auch die Parameteranzahl übereinstimmen. Andernfalls erhalten Sie folgende Fehlermeldung:

```
Nachricht . . . :   An CALL übergebene Parameter entsprechen nicht den
  angeforderten.
Ursache  . . . . :  Die an den Befehl CALL (Programm aufrufen) übergebenen
  Parameter entsprechen nicht den von Programm PGM_B in Bibliothek EPS002CR
  angeforderten Parametern. Das richtige Programm konnte nicht gefunden werden
  oder es wurden falsche Parameter übergeben. Die Anzahl der Parameter oder
  die Art der an den Befehl CALL übergebenen Parameter muss vom aufgerufenen
  Programm erwarteten entsprechen.
```

Anzahl der übergebenen Parameter stimmen nicht überein

Schauen wir uns jetzt das aufgerufene Programm PGM_B an:

```
PGM      PARM(&V1 &V2 &V3 )

             DCL        VAR(&V1) TYPE(*CHAR) LEN(20)
             DCL        VAR(&V2) TYPE(*CHAR) LEN(10)
             DCL        VAR(&V3) TYPE(*DEC) LEN(6 0)

             DCL        VAR(&V3_A) TYPE(*CHAR) LEN(6)

             CHGVAR    &V3_A VALUE(&V3)

             SNDPGMMSG  MSG(&V1 *CAT &V2 *CAT &V3_A  ) TOUSR(RASCHE)

RETURN
```

Das Programm PGM_B

Die Variablen müssen in jedem Programm zunächst deklariert werden – auch im aufgerufenen Programm. Beachten Sie, dass die Variablennamen in beiden Programmen unterschiedlich gewählt wurden, lediglich die Datentypen und verwendeten Längen spielen hierbei eine Rolle. Natürlich können Sie alle Datentypen für die Parameter verwenden. Hierbei gibt es keine Restriktionen. Eine Besonderheit in diesem Zusammenhang bilden lediglich Gleitkommazahlen, weil diese Datenart dem CL-Programm unbekannt ist. Sie können aber trotzdem ein CL-Programm aufrufen und als Parameter eine Gleitkommavariable übergeben. Die Gleitkommazahl muss vom CL-Programm als 8-stellige alphanumerische Variable empfangen werden. Das CL-Programm kann dann seinerseits ein weiteres Programm aufrufen, das Gleitkommazahlen verarbeiten kann, und den entsprechenden Wert weitergeben.

Wer genau hinschaut bemerkt vielleicht, dass ich den ersten Parameter &VAR01 und den ersten Empfangsparameter &V1 mit unterschiedlichen Längen definiert habe. Alphanumerische Parameter können im aufgerufenen und aufrufenden Programm unterschiedliche Längen haben. Sie sollten diese Situation aber – wenn möglich – vermeiden, da das Ergebnis vielleicht nicht immer zufriedenstellend ausfällt.

Schauen wir uns das Ergebnis der Übergabe an! Zu diesem Zweck verwende ich den Befehl SNDPGMMSG im aufgerufenen Programm und lasse mir den Inhalt der Empfangsparameter als Nachricht ausgeben. Zuvor muss ich aber den Inhalt der Dezimalvariablen konvertieren, da ich im Parameter MSG lediglich alphanumerische Werte verwenden kann. Dies gelingt mit der CHGVAR-Anweisung, die Sie im Programm sehen. Nun aber das Ergebnis:

```
Von . . . . :    RASCHE          28.01.10    12:18:53
INHALT VAR01         XX          000002  <-- Meldung PGM_B
Von . . . . :    RASCHE          28.01.10    12:18:53
Zurück im Hauptprogramm  <------------------ Meldung PGM_A
```

Das Ergebnis

2.12.9.1
Seite 4

Alle Werte wurden korrekt übergeben und auch das Hauptprogramm (=aufrufendes Programm) arbeitet nach der Rückkehr aus dem Unterprogramm (= aufgerufenes Programm) einwandfrei.

In meinem kleinen Beispiel übergebe ich auch einen konstanten Wert als Parameter.

Die Verwendung erfordert allerdings ein wenig Aufmerksamkeit, denn Sie müssen dabei ein paar Regeln beachten:

Alphanumerische Konstanten (**> 32 Byte**) müssen dieselbe Länge wie die im PGM_B deklarierte Parameter-Variable haben. Diese Regel gilt allerdings nur, wenn ein Zeichenwert mit einer Länge von mehr als 32 Byte übergeben werden soll.

Ist der konstante alphanumerische Wert <= 32 Byte, wird die Empfangsvariable mit Leerzeichen oder wie in meinem Beispiel mit Nullen aufgefüllt oder sie wird rechts abgeschnitten, wenn die Empfangsvariable zu klein ist.

Dezimale Konstanten müssen immer mit einer Länge (15 5) übergeben werden. Dazu muss Programm „B“ diese in einer Variablen aufnehmen, die mit dem Typ Dezimal (*DEC) und einer Länge LEN(15 5) definiert wurde. In diesen Fällen können Sie einfach im Aufruf eine dezimale Konstante ohne Anführungszeichen im CALL-Befehl angeben. Der Wert wird vom System richtig umgesetzt und dezimalstellengenau übergeben. Wird aber der Wert mit irgendeiner anderen Genauigkeit erwartet, muss die dezimale Konstante in hexadezimaler Form übergeben werden. Anderenfalls erhalten Sie Fehler!

Hexadezimale Konstanten müssen in jedem Fall die korrekte Länge haben. Wenn unser Programm PGM_B entweder einen Zeichenwert mit 4 Bytes oder einen dezimalen Wert mit 6 oder 7 Ziffern unabhängig von der Anzahl der Dezimalstellen erwartet, dann muss eine hexadezimale Konstante mit 8 Halbbytes übergeben werden. Dieser Wert würde vom aufgerufenen Programm PGM_B je nach Definition der Parameter-Variablen interpretiert werden.

Schauen wir uns an, was gemeint ist:

```
CALL EPS002CR/PGM_B2 PARM(X'0000067F' X'0000067F')
```

Übergabe hexadezimaler Konstanten

Das Unterprogramm empfängt die konstanten Werte in numerischen Variablen:

```
PGM      PARM(&V1 &V2)

             DCL        VAR(&V1) TYPE(*DEC) LEN(6 0)
             DCL        VAR(&V2) TYPE(*DEC) LEN(6 2)

ENDPGM
```

Variablendeklaration im aufgerufenen Programm

Nach der Übergabe enthält die Variable &V1 den Wert 67. Die Variable wurde ohne Nachkommastellen deklariert. Die Variable &V2 enthält hingegen den Wert 0,67. Die Deklaration der Empfangsvariablen bestimmt somit letztendlich den Wert der hexadezimalen Konstanten. Dieses Vorgehen ist mühsam und zudem auch noch fehleranfällig. Ich denke, Sie sind besser beraten, eine Variable für die Übergabe Ihrer dezimalen Konstanten zu verwenden!

Logische Konstanten müssen entweder mit dem Wert '1' oder '0' übergeben werden; benötigen ansonsten aber keine weitere Aufmerksamkeit.

Gleitkomma-Konstanten wie zum Beispiel 2.5E+73 können auch problemlos an ein CL-Programm übergeben werden. Allerdings können diese Werte nicht in einem CL-Programm genutzt werden. Die Konstante muss einer Variable mit der Definition Typ (*CHAR) und einer festen Länge LEN(8) zugewiesen werden, die das CL-Programm anschließend ohne Probleme an ein weiteres Programm übergeben kann. Spezialwerte wie zum Beispiel *NAN (keine Zahl), *INF (unendlich) und *NEGINF (negativ unendlich) können auf dieselbe Weise an das CL-Programm übergeben werden.

Sie haben sicherlich bemerkt, dass ich sowohl die Anweisung RETURN als auch ENDPGM verwende, um das physische Ende einer Quell-Teildatei zu definieren. Den Befehl RETURN können Sie sich ähnlich vorstellen wie die Anweisung ENDPGM. Mit dem Unterschied, dass ENDPGM in früheren Releases stets am Ende des Programm-Codes angegeben werden musste. Diese Restriktion ist mittlerweile aufgehoben. Ansonsten beendet auch der Befehl RETURN die Verarbeitung des Programms und gibt die Steuerung wieder an das aufrufende Programm zurück. Viele CL-Programmierer folgen aber der Konvention und kodieren weiterhin am Ende des Quellcodes die Anweisung ENDPGM und dokumentieren dadurch das Programmende. Für vorzeitige Programmabbrüche oder beim Verlassen eines Unterprogramms verwenden die meisten Programmierer hingegen die Anweisung RETURN.

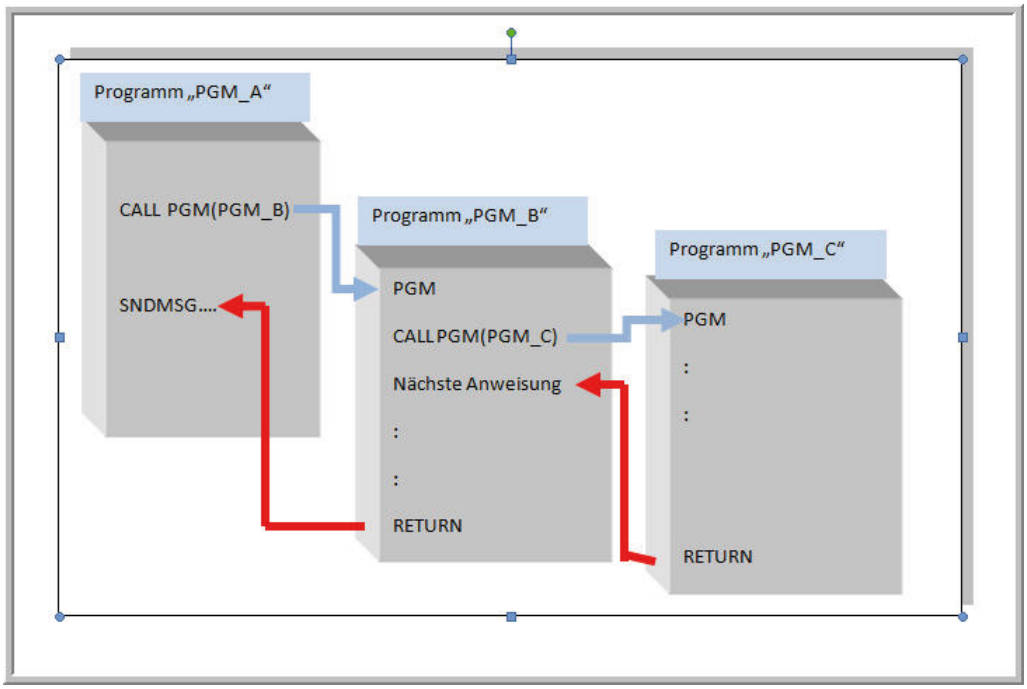
2.12.9.1
Seite 6

Der Befehl TFRCTL

Der Befehl TFRCTL (Transfer Control) arbeitet ähnlich wie der Befehl CALL. Beide Befehle unterbrechen die Ausführung eines Programms und starten ein weiteres Programm.

Wo ist der Unterschied zwischen den beiden Befehlen?

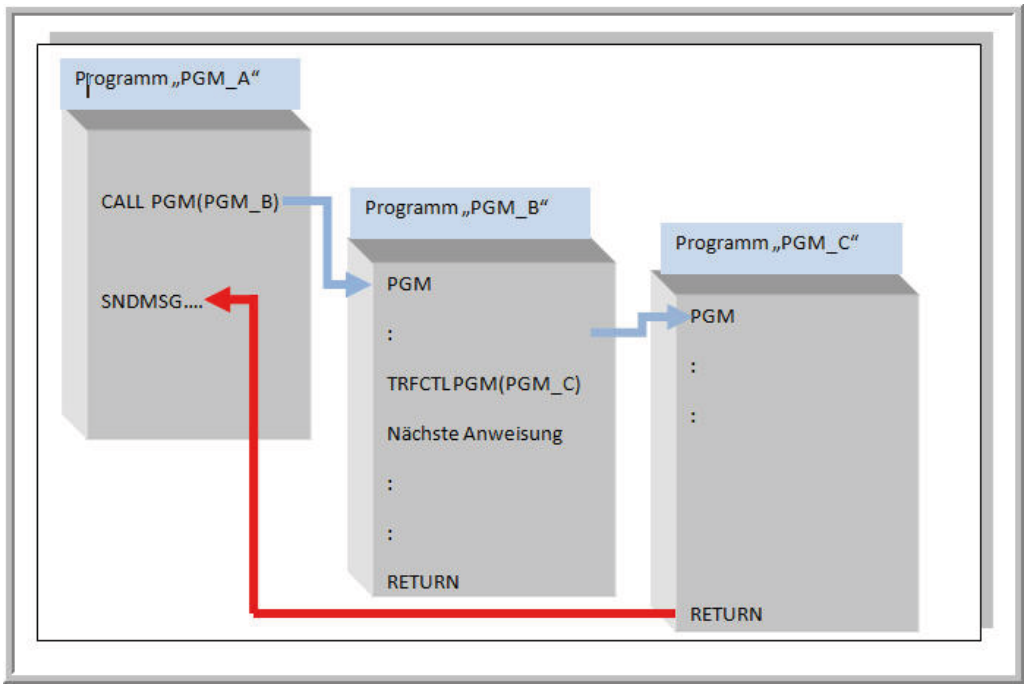
Um den entscheidenden Unterschied zu erkennen, muss man sich anschauen, was nach der Beendigung eines Unterprogramms passiert:



Das Hauptprogramm erhält stets die Kontrolle zurück

2.12.9.2
Seite 2

Verwenden Programme die Anweisung CALL, um ein entsprechendes Unterprogramm aufzurufen, erhält der Aufrufer immer die Kontrolle zurück, sobald das entsprechende Unterprogramm beendet wurde. Anders sieht es hingegen aus, wenn der Aufruf mittels TFRCTL erfolgt:



Wirkungsweise der Anweisung TFRCTL

Im letzten Beispiel ruft Programm A zunächst das Programm B auf und führt dann den Befehl TFRCTL aus, um das Programm C zu starten. Sobald Programm C endet, wird die Kontrolle nicht wieder an das Programm B zurückgegeben, sondern an Programm A, also das erste Programm. Oder anders gesagt, das Programm, aus dem heraus der Befehl TFRCTL ausgeführt wird, wird in der Rückkehr übergangen. Wenn Sie Programm B über eine 5250-Befehlszeile gestartet hätten, dann würde nach Beendigung von Programm C auch wieder die Kontrolle direkt an die Datenstation und die Tastatur zurückgegeben werden – ohne Berücksichtigung von Programm C. TFRCTL hat allerdings einige Einschränkungen, die zu beachten sind:

– TFRCTL wird nicht in ILE-Programmen unterstützt.

– Es können keine Konstanten als Parameter verwendet werden.

Wenn Programm B das Programm C mit dem Befehl TFRCTL aufruft, dann können auch Parameter an Programm C übergeben werden. Allerdings nur dann, wenn diese Parameter zuvor von Programm A empfangen wurden. Programm B kann keine Parameter übergeben, die es nicht selbst als Eingangsparameter empfangen hat!

Trotz dieser Einschränkungen ist TFRCTL eine gute Alternative für einen Programmaufruf, wenn keine Notwendigkeit besteht, in das aufrufende Programm zurückzukehren.

2.12.9.2

Seite 4

Informationsaustausch – Datenbereiche

Der Vorteil bei der Verwendung von Parametern liegt in der einfachen, schnellen Verwendung, denn sie erfordern keine externen Objekte. Der Nachteil bei der Verwendung von Parametern ist aber auch offensichtlich. Es können nur dann Daten übergeben werden, wenn das empfangende Programm gestartet wird. Die Übergabe eines Parameters an ein bereits laufendes Programm ist dagegen nicht möglich. Auch der Informationsaustausch mit Programmen, die auf einem anderen Server ausgeführt werden, ist auf diese Art und Weise nicht zu realisieren. Und was passiert, wenn das aufgerufene Programm mehrere Stunden für die Verarbeitung der empfangenen Daten benötigt? Soll dann das aufrufende Programm solange im Arbeitsspeicher warten?

In diesen Situationen nützen die Parameterübergaben wenig. Wir benötigen zur Lösung dieser Anforderung andere Objekte.

Datenbereiche könnten an dieser Stelle eine Lösung sein. Datenbereiche sind Objekte vom Typ *DTAARA, die Sie nach Bedarf erstellen und auch wieder löschen können. Bei einem Datenbereich handelt es sich um einen Speicherbereich mit einem Namen, aber ohne feste Struktur. Die Angabe von Feldern innerhalb eines Datenbereiches ist daher auch nicht möglich. Aber Datenbereiche können zum Speichern von Informationen verwendet werden. Ein Programm kann zum Beispiel Informationen in einen Datenbereich schreiben, während ein zweites Programm diese Informationen zu einem späteren Zeitpunkt auslesen kann. In diesen Fällen sprechen wir von einer asynchronen Verarbeitung oder asynchronen Programmkommunikation.

Datenbereiche ermöglichen es aber auch, dass die Informationen zwischen zwei oder mehr Programmen ausgetauscht werden, selbst dann, wenn diese Programme in unterschiedlichen Jobs ausgeführt werden. Dies ist deshalb möglich, weil es sich bei Datenbereichen um permanente Objekte handelt, die nicht einem speziellen Job zugeordnet sind, sondern mit dem Befehl CRTDTAARA (Create Data Area) erstellt werden.

```
                    Datenbereich erstellen (CRTDTAARA)

Auswahl eingeben und Eingabetaste drücken.

Datenbereich . . . . . > CL_AREA      Name
  Bibliothek . . . . . .   *CURLIB    Name, *CURLIB
Art  . . . . . > *CHAR                *DEC, *CHAR, *LGL, *DDM
Länge:
  Länge  . . . . . > 2000             1-2000
  Dezimalstellen . . . .   ______     0-9
Anfangswert  . . . . . .   ____________________________
Text 'Beschreibung'  . . . . > 'Beispiel für ITP "Systemmangement"'
______
```

Der Befehl CRTDTAARA (Create Data Area)

2.12.9.3
Seite 2

Ich erstelle in dem Beispiel den alphanumerischen Datenbereich „CL_AREA“ in meiner Standardbibliothek. Beim Erstellen muss ein qualifizierter Name, die Datenart und die Länge angegeben werden. Sie können für alphanumerische Datenbereiche maximal 2.000 Zeichen reservieren, für numerische Datenbereiche sind maximal 24 Stellen mit 9 Nachkommastellen erlaubt und logische Datenbereiche können maximal 1 Zeichen haben. Verzichten Sie auf eine Längenangabe, wird der Datenbereich mit seinen Standardlängen entsprechend des verwendeten Datentyps erstellt. Die nachfolgende Tabelle gibt Ihnen einen Überblick über die Standarddefinitionen im Release V6R1:

Datentyp	Maximum in Byte	Standardlänge in Byte
*CHAR	2000	32 Byte
*DEC	24, 9	15, 5
*LGL	1	1

Die Angabe eines Initialisierungswertes und eines bis zu 50-stelligen Beschreibungstextes ist optional.

Datenbereiche können mit dem Befehl DLTDTAARA („Datenbereich löschen“) wieder gelöscht werden. Dabei spielt es keine Rolle, ob der Datenbereich Daten enthält oder leer ist. Bei der Ausführung dieses Befehls ist lediglich der qualifizierte Name des zu löschenden Datenbereiches anzugeben. Beide Befehle – CRTDTAARA und DLTDTAARA – können entweder manuell ausgeführt oder in ein CL-Programm eingebunden werden.

Sobald ein Datenbereich erstellt ist, kann sein Inhalt mit dem Befehl CHGDTAARA (Datenbereich ändern) geändert werden. CHGDTAARA erwartet neben dem qualifizierten Namen des zu ändernden Datenbereiches die Position, an der Daten geändert werden sollen, die Länge der zu ändernden Daten und den neuen Wert der Daten.

Nehmen wir einmal an, wir wollen den eben erstellten Datenbereich „CL_AREA“ ändern, indem wir die 10. und 11. Stelle durch die Ziffern „8“ und „9“ ersetzen. Der Befehl sähe folgendermaßen aus:

```
                    Datenbereich ändern (CHGDTAARA)

Auswahl eingeben und Eingabetaste drücken.

Datenbereichsbestimmung:
  Datenbereich . . . . . .   CL_AREA      Name, *LDA, *GDA, *PDA
    Bibliothek . . . . . .     *LIBL       Name, *LIBL, *CURLIB
  Unterzeichenfolgebestimmung:
  Unt.zeichenfolge Startposition   10          1-2000
  Unterzeichenfolgenlänge . . .    2           1-2000
Neuer Wert . . . . . .             '89'
```

Der Befehl CHGDTAARA (Change Data Area)

Mit Hilfe des Befehls DSPDTAARA (Display Data Area) lässt sich das Ergebnis sehr schnell überprüfen:

```
                    Datenbereich anzeigen (DSPDTAARA)

Auswahl eingeben und Eingabetaste drücken.

Datenbereich . . . . . > CL_AREA      Name, *LDA, *GDA, *PDA
  Bibliothek . . . . . .   *LIBL      Name, *LIBL, *CURLIB
Ausgabe  . . . . . .     *           *, *PRINT
Ausgabeformat  . . . . . *CHAR       *CHAR, *HEX
```

Der Befehl DSPDTAARA (Display Data Area)

Der Befehl benötigt wiederum den qualifizierten Objektnamen des Datenbereichs. Die Ausgabe kann auf dem Bildschirm oder als Druckausgabe erfolgen und auch bzgl. der Anzeige haben Sie die Wahl zwischen Zeichen- oder Hexdarstellung. Schauen wir uns die Eingabe an:

```
Wert
 *...+....1....+....2....+....3....+....4....+....5
'          89                                        '
'                                                    '
'                                                    '
```

Das Ergebnis

Nun müssen wir nur noch die Informationen aus dem Datenbereich herausholen – also lesen. Der Befehl RTVDTAARA (Datenbereich auffinden) kann in einem CL-Programm dazu verwendet werden, den Inhalt eines Datenbereiches in eine CL-Variable zu stellen. Sie können zum Beispiel den soeben eingefügten Inhalt in einem CL-Programm in die Variable „&INFO“ stellen.

```
****************** Datenanfang ********************************
PGM
     DCL &INFO  *CHAR 2

             RTVDTAARA  DTAARA(CL_AREA (10 2)) RTNVAR(&INFO)

             SNDPGMMSG   MSG(&INFO)

ENDPGM
******************Datenende ********************************
```

Der Befehl RTVDTAARA

2.12.9.3
Seite 4

Der Befehl RTVDTAARA ist durchaus vergleichbar mit dem Befehl CHGDTAARA. Auch im Falle des Datenabrufs müssen Sie die Startposition, die Länge des zu übertragenden Wertes und natürlich eine Zielvariable benennen. Beachten Sie, dass bedingt dadurch, dass der Datenbereich CL_AREA mit dem Typ *CHAR definiert wurde, die in dem Programm verwendete Variable ebenfalls vom Typ *CHAR“ sein muss.

Abschließend ein paar kurze Überlegungen zu den notwendigen Objektsperren. Die Befehle RTVDTAARA und DSPDTAARA nutzen die Sperrstufe *SHRRD (shared for read). Der CL-Befehl CHGDTAARA sperrt den Datenbereich im Modus *SHRUPD (shared for update). Andere Jobs könnten somit den Inhalt des Datenbereichs lesen, während Sie den Inhalt ändern.

Wenn Sie dieses Szenario unterbinden wollen, könnte es sinnvoll sein, den Befehl ALCOBJ (Allocate Object) zu verwenden, um anderen Benutzern den Zugriff auf den Datenbereich zu verbieten, während Sie mit dem Objekt arbeiten. Wenn Sie in einem Job ein Objekt – beispielsweise eine Datei oder wie in unserem Fall einen Datenbereich – zur Eingabe öffnen, wird das Objekt dem Job automatisch zugeordnet. Allerdings können auch andere Jobs noch auf das Objekt zugreifen – es können sowohl Lese- als auch Update-Zugriffe von anderen Jobs erfolgen. Allerdings können die anderen Jobs das Objekt zu diesem Zeitpunkt nicht umbenennen oder löschen. Diesen Vorgang nennt man automatische Zuordnung, weil der Vorgang automatisch vom System ausgelöst wird.

In bestimmten Situationen benötigen Sie aber vielleicht ein Objekt exklusiv. Stellen Sie sich vor, dass wir ein CL-Programm erstellen, mit dem wir im Stapelbetrieb umfangreiche Änderungen an einer Datenbankdatei durchführen oder eine ungewisse Zeit warten müssen, bevor wir die Informationen im Datenbereich ändern können. Um sicherzustellen, dass kein anderer Job oder Benutzer das Objekt während unserer Verarbeitung verwendet, kann noch vor der Ausführung des entsprechenden Programms das Objekt exklusiv für die spätere Verarbeitung zugeordnet werden. Diese manuelle Zuordnung wird mit dem Befehl „Objekt zuordnen“ (ALCOBJ) durchgeführt. Diese Zuordnung bzw. Sperre bleibt solange bestehen, bis der Job beendet wird oder die manuelle Sperre mit dem Befehl „Objekt freigeben“ (DLCOBJ) wieder aufgehoben wird.

Die nachfolgende Abbildung enthält einen Überblick über die verschiedenen Sperrzustände und die Auswirkungen auf Ihren Job und den Job anderer Benutzer.

	Andere Job-Attribute			Ihre Job-Attribute		
Lock Status	Obj Mgt	Lesen	Update	Obj Mgt	Lesen	Update
*EXCL	Yes	Yes	Yes	No	No	No
*EXCLRD	No	Yes	Yes	No	Yes	No
*SHRUPD	No	Yes	Yes	No	Yes	Yes
*SHRNUP	No	Yes	No	No	Yes	No
*SHRRD	No	Yes	No	No	Yes	Yes

Übersicht der Sperrstufen

Das nachfolgende Programm veranschaulicht sowohl die Verwendung des CL-Befehls ALCOBJ als auch den Einsatz der Datenbereiche.

Stellen wir uns vor, dass zu bestimmten Zeiten Transaktionsdaten in verschiedene DB2-Dateien übernommen werden. Während die Transaktion noch ausgeführt wird, dürfen nachfolgende Verarbeitungsjobs keinesfalls auf die Daten zugreifen. Erst wenn die Datenübernahme vollständig abgeschlossen ist, sollen die nachfolgenden Programme die Verarbeitung beginnen und dokumentieren nun ihrerseits die Verarbeitung der Daten durch einen entsprechenden Eintrag im Datenbereich. Wir benötigen also drei verschiedene Transaktionsstati:

– Daten vollständig verarbeitet
– Transaktion ist aktiv
– Daten vollständig übernommen, aber nicht verarbeitet

2.12.9.3
Seite 6

Um dieses Szenario abzubilden, könnte ich den Namen der Transaktionsdateien und den jeweiligen Transaktionsstatus in einem Datenbereich hinterlegen. Die Struktur des Datenbereichs könnte in diesem Fall folgendermaßen aussehen:

Position	Bedeutung
1 – 10	Name der ersten Transaktionsdatei
11	Status: 1 (Daten verarbeitet) 2 (Verarbeitung gestartet) 3 (Transaktion vollständig abgeschlossen)
12 – 21	Name der zweiten Transaktionsdatei
22	Status: 1 (Daten verarbeitet) 2 (Verarbeitung gestartet) 3 (Transaktion vollständig abgeschlossen) : :

Schauen wir uns nun den Datenbereich an:

```
                          Datenbereich anzeigen

Datenbereich  . . . . . :   TRANS_DTA
  Bibliothek  . . . . . :     EPS002CR
Art . . . . . . . . . . :   *CHAR
Länge . . . . . . . . . :   44
Text  . . . . . . . . . :

           Wert
Offset      *...+....1....+....2....+....3....+....
    0      'KONTAKTE  1ARTIKEL   1KUNDEN    1
```

Die Ausgangssituation

Für die Verarbeitung benötigen wir jetzt zwei CL-Programme:

Das erste Programm übernimmt die Daten der Datei KONTAKTE vom PC:

```
PGM
 /* Datentransfer KONTAKTE */

     DCL &STATUS *CHAR 1

     ALCOBJ OBJ((EPS002CR/TRANS_DTA *DTAARA *EXCL)) WAIT(30)
     MONMSG CPF0000 EXEC(DO)

        SNDPGMMSG MSG(,Datenbereich kann nicht zugeordnet +
           werden. Programm wird beendet!‘)
        RETURN
        ENDDO

        RTVDTAARA DTAARA(TRANS_DTA (11 1)) RTNVAR(&STATUS)

        IF (&STATUS *NE ‚2‘) THEN(DO)
           CHGDTAARA DTAARA(TRANS_DTA (11 1)) VALUE(‚2‘)
           DLCOBJ OBJ((TRANS_DTA *DTAARA *EXCL))
```

Bevor der eigentliche Datentransfer gestartet wird, ändert das CL-Programm den Status der Transaktionsdatei KONTAKTE auf den Wert „2“, wenn derzeit keine Verarbeitung durch ein entsprechendes HLL-Programm aktiv ist. Zu diesem Zweck wird der Datenbereich zunächst dem aktuellen Job exklusiv zugeordnet, um eine Konkurrenzsituation in diesem Moment zu verhindern. Für den Befehl ALCOBJ wurde zudem der Parameter WAIT(30) angegeben, um festzulegen, dass das Programm 30 Sekunden warten soll, falls momentan keine exklusive Zuordnung möglich ist. Der Befehl MONMSG überwacht diese Fehlersituation und sorgt dafür, dass ggf. eine Programmnachricht gesendet und das Programm beendet wird. Wenn der Datenbereich hingegen zugeordnet werden kann, ändert das CL-Programm den Statuswert der Datei KONTAKTE auf den Wert „2“; die Übernahme der Datei KONTAKTE ist aktiviert. Anschließend wird der Datentransfer als Batchjob auf dem PC gestartet.

```
STRPCCMD PCCMD(-
,START \\172.16.110.22\ROOT\HOME\RASCHE\EPSW17CR\KONTAKTE.BAT‘) +
PAUSE(*NO)
```

2.12.9.3
Seite 8

Der CL-Befehl STRPCCMD erlaubt mir die Ausführung eines beliebigen PC-Befehls. Ich habe mich für die Anweisung START entschieden, die eine ausführbare PC-Datei ausführt. Ausgeführt wird die Datei KONTAKTE.BAT, eine DOS-Batch-Datei, die den eigentlichen Transfer-Befehl beinhaltet:

Die Batch-Datei KONTAKTE.BAT

Ist die Datenübertragung abgeschlossen, wird die Datenübernahme dokumentiert, indem der Status der Datei KONTAKTE auf den Wert „3“ geändert wird.

```
ALCOBJ OBJ((EPS002CR/TRANS_DTA *DTAARA *EXCL)) WAIT(30)
CHGDTAARA DTAARA(TRANS_DTA (11 1)) VALUE(‚3‘)
DLCOBJ OBJ((TRANS_DTA *DTAARA *EXCL))
```

Ein zweites CL-Programm könnte anschließend für die weitere Verarbeitung der Daten sorgen, indem es in regelmäßigen Abständen anhand des Datenbereichs prüft, ob neue Transferdaten eingetroffen sind und verarbeitet werden müssen.

Das zweite Programm ist etwas aufwändiger zu kodieren: In einem ersten Schritt muss das Programm den Status der entsprechenden Einträge im Datenbereich ermitteln. Da ein Datenbereich keine Felder kennt, müssen die Daten anhand der Offset-Position innerhalb einer Schleife ermittelt werden. Die Startposition zur Ermittlung des Statuswertes muss dabei variabel gehalten werden, was mir mit Hilfe der Variablen &N gelingt.

```
START:
CHGVAR &N VALUE(11)


RTVDTAARA DTAARA(TRANS_DTA (&N 1)) RTNVAR(&STATUS)
DOWHILE (&STATUS *NE , ,)

/* Verarbeitung des Statuswertes 3 */

IF (&STATUS = ,3‘) THEN(DO)
        CALLSUBR STATUS_3
ENDDOe

CHGVAR &N VALUE(&N + 11)
RTVDTAARA DTAARA(TRANS_DTA (&N 1)) RTNVAR(&STATUS)
MONMSG CPF1088 EXEC(GOTO ENDE)


ENDDO

CHGVAR &N VALUE(&N + 11)
RTVDTAARA DTAARA(TRANS_DTA (&N 1)) RTNVAR(&STATUS)
MONMSG CPF1088 EXEC(GOTO ENDE) /* Datenbereich-Ende erreicht */

ENDDO
```

Falls der gesamte Datenbereich belegt ist, muss zudem das Ende des Datenbereichs durch eine entsprechende MONMSG-Anweisung abgefangen werden. Die eigentliche Verarbeitung der Transaktionsdaten darf anschließend nur stattfinden, wenn der Statuswert „3“ ermittelt wurde.

Diese Verarbeitungslogik lässt sich gut in einer Unterroutine abbilden:

```
SUBR STATUS_3
ALCOBJ OBJ((EPS002CR/TRANS_DTA *DTAARA *EXCL)) WAIT(30)
CHGDTAARA DTAARA(TRANS_DTA (&N 1)) VALUE(,2‘)
DLCOBJ OBJ((TRANS_DTA *DTAARA *EXCL))
```

Damit während der Verarbeitung der Transaktionsdatei keine weiteren Datenübernahmen stattfinden können, setze ich den Status für die entsprechende Transaktionsdatei zunächst auf den Wert „2“, um die laufende Verarbeitung zu dokumentieren.

```
CHGVAR &N VALUE(&N – 10) /* Startposition Dateiname */
RTVDTAARA DTAARA(TRANS_DTA (&N 10)) RTNVAR(&DATEI)
```

In einem nächsten Schritt muss der Name der Transaktionsdatei ermittelt werden, um dann anhand des Dateinamens zu entscheiden, welches Verarbeitungsprogramm aufgerufen werden soll. Natürlich wäre es auch denkbar, den Dateinamen an das HLL-Programm zur weiteren Verarbeitung zu übergeben. Hier wurde aber eine andere Lösung gewählt.

```
SELECT
WHEN COND(&DATEI = ‚ARTIKEL‘) THEN(CALL PGM(PGM_ART))
WHEN COND(&DATEI = ‚KUNDEN‘) THEN(CALL PGM(PGM_KUN))
WHEN COND(&DATEI = ‚KONTAKTE‘) THEN(CALL PGM(PGM_KON))
ENDSELECT

CHGVAR &N VALUE(&N + 10) /*Startposition ursprünglicher Statuswert
*/
```

Abschließend – nach erfolgreicher Verarbeitung der Daten durch entsprechende Hochsprachenprogramme – wird der Status der Verarbeitung auf den Wert „1“ gesetzt. Denkbar wäre in diesem Fall auch, dass das HLL-Programm den Statuswert selbst setzt, denn auch HLL-Programme sind in der Lage Datenbereiche zu nutzen.

```
ALCOBJ OBJ((EPS002CR/TRANS_DTA *DTAARA *EXCL)) WAIT(30)
CHGDTAARA DTAARA(TRANS_DTA (&N 1)) VALUE(‚1‘)
DLCOBJ OBJ((TRANS_DTA *DTAARA *EXCL))

ENDSUBR
```

Entscheidet man sich für die oben dargestellte Lösung, muss allerdings noch geklärt werden, was passieren soll, wenn das HLL-Programm Fehler meldet und vor allem wie das HLL-Programm Fehler an das CL-Programm weiterleitet. Dies könnte beispielsweise durch entsprechende Nachrichten, eine MONMSG-Anweisung oder andere Techniken erfolgen. Das soll aber zum jetzigen Zeitpunkt nicht im Fokus der Betrachtung stehen.

Ich denke, das Beispiel zeigt deutlich die Schwächen und Stärken der Datenbereiche.

Vorteilhaft ist sicherlich die einfache Handhabung der Datenbereiche. Datenbereiche können interaktiv manipuliert und betrachtet werden. So lässt sich schnell feststellen, welche Inhalte in einem Datenbereich gespeichert wurden.

Auch ist es positiv zu bewerten, dass nicht nur CL-Programme mit Datenbereichen arbeiten können, sondern unsere HLL-Programme ebenfalls.

In meinem Beispiel ist außerdem deutlich zu erkennen, dass die Informationen nach dem Auslesen im Datenbereich erhalten bleiben. Dies kann vorteilhaft sein, wenn verschiedene Programme die Informationen immer wieder benötigen. Es könnte sich aber auch negativ auswirken, wenn Informationen abgelegt werden, die nach der Verarbeitung nicht mehr benötigt werden. In diesen Fällen müssten Sie manuell bzw. im CL-Programm für die Bereinigung der Daten sorgen.

Und hier finden wir auch eine weitere Einschränkung der Datenbereiche: Datenbereiche erweitern sich nicht dynamisch. Die Größe ist – einmal festgelegt – konstant. Soll ein Datenbereich erweitert werden, müssen Sie den alten Datenbereich zunächst löschen und anschließend mit neuer Längenangabe erneut erstellen.

Hinzu kommt, dass die Größe der Datenbereiche eher klein ist. In 2.000 Byte können nur begrenzt Informationen abgelegt werden.

Und auch die Arbeit mit den Offset-Positionen ist sicherlich nicht jedermanns Sache.

22. Ergänzung 3/2010

2.12.9.3

Seite 12

Spezielle Datenbereiche (*LDA *GDA *PDA)

Auf einem System-i-Server existieren noch spezielle Ausprägungen der Datenbereiche, die nur einigen Jobs oder temporär zur Verfügung stehen. Diese Sonderformen ermöglichen weitere Verarbeitungsmöglichkeiten.

Ferne Datenbereiche
Sie können auch ferne (= Remote) Datenbereiche erstellen. Sie erstellen den Remote-Datenbereich ganz einfach mit dem bereits bekannten CL-Befehl CRTDTAARA.

```
                    Datenbereich erstellen (CRTDTAARA)

Auswahl eingeben und Eingabetaste drücken.

Datenbereich . . . . . > LCL_AREA      Name
  Bibliothek . . . . . >   EPS002CR    Name, *CURLIB
Art . . . . . > *DDM           *DEC, *CHAR, *LGL, *DDM
Ferner Datenbereich . . . . . > RMT_AREA2     Name
  Bibliothek . . . . . >   RASCHE      Name, *LIBL, *CURLIB
Ferner Standort . . . . . > *RDB       Name, *RDB
Relationale Datenbank . . . . > ITPSERVER
APPC-Einheitenbeschreibung . . . *LOC        Name, *LOC
Lokaler Standort . . . . . *LOC       Name, *LOC, *NETATR
Modus . . . . . *NETATR     Name, *NETATR
Ferne Netzwerk ID . . . . . *LOC       Name, *LOC, *NETATR, *NONE
Text 'Beschreibung' . . . . . *BLANK
```

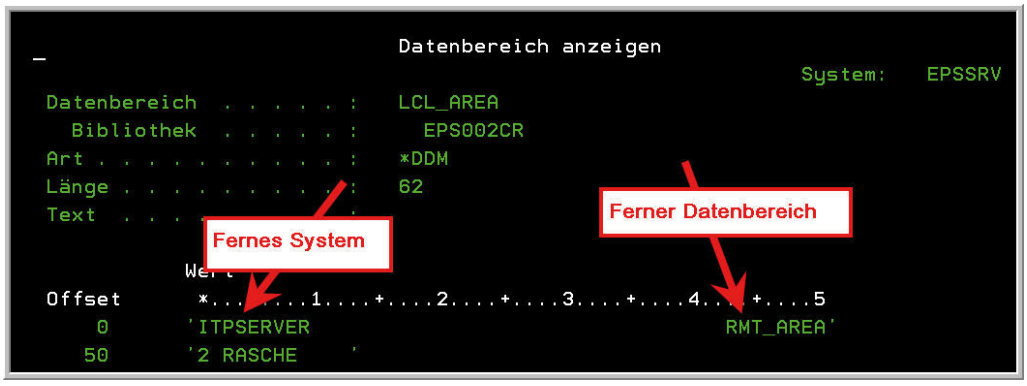
Remote-Datenbereiche erstellen

Sie definieren den fernen Datenbereich, indem Sie für den Parameter ART den Wert *DDM wählen. Daraufhin erhalten Sie die Möglichkeit den Namen des fernen Datenbereichs einzugeben und müssen in einem weiteren Schritt noch den fernen Standort definieren. Hierzu können Sie entweder den Netzwerknamen des fernen Standorts nutzen oder den Namen der fernen Datenbank. Die Datenbank müsste in diesem Fall allerdings zuvor auf dem lokalen System mit dem Befehl WRKRDBDIRE hinterlegt worden sein:

2.12.9.3.1
Seite 2

Mit Einträgen im RDB-Verzeichnis arbeiten

Sobald Sie die Eingaben getätigt haben, wird der Datenbereich erstellt. Der erstellte Datenbereich ist ein DDM-Datenbereich. Der lokale Datenbereich – in unserem Beispiel der Datenbereich LCL_AREA – enthält den Namen des fernen Datenbereichs, auf den zugegriffen wird, sowie den Namen des fernen Zielsystems, auf dem sich der Datenbereich befindet.



Inhalt des „fernen" Datenbereichs

Soll jetzt ein Wert aus dem fernen Datenbereich im lokalen CL-Programm verarbeitet werden, nutzen Sie wie gewohnt den CL-Befehl RTVDTAARA und geben im Befehl einfach den Namen des lokalen Datenbereichs an. Das System stellt anhand der Informationen im lokalen Datenbereich eine DDM-Verbindung zum fernen Server her und liest die Daten aus dem fernen Datenbereich.

Wenn Sie diesen Wert nun in Ihrem Programm verändern und in den fernen Datenbereich übertragen wollen, nutzen Sie wie gewohnt den Befehl CHGDTAARA. Auch jetzt wird eine Verbindung zum fernen System anhand der Informationen in Ihrem lokalen Datenbereich hergestellt und die Daten im fernen Datenbereich abgelegt. Wenn Ihre Programme bislang mit lokalen

Datenbereichen gearbeitet haben und jetzt auf die fernen Datenbereiche zugreifen sollen, müssen Sie die bestehenden Programme von daher weder modifizieren noch kompilieren. Sie müssen lediglich dafür sorgen, dass der DDM-Datenbereich korrekt adressiert wird. Dies erreichen Sie, indem Sie entweder den „alten" lokalen Datenbereich löschen oder umbenennen. Natürlich können Sie auch Ihre Bibliothekssuchliste entsprechend ändern, wenn Sie in Ihren Programmen unqualifiziert adressiert haben.

Wenn Sie allerdings den CL-Befehl DSPDTAARA verwenden, wird zunächst der lokale DDM-Datenbereich angezeigt. Sie können aber im DSPDTAARA-Befehl den Parameter SYSTEM auf den Wert *RMT ändern und erhalten in diesem Fall ebenfalls den fernen Datenbereich angezeigt:

```
                    Datenbereich anzeigen (DSPDTAARA)

Auswahl eingeben und Eingabetaste drücken.

Datenbereich . . . . . > RMT_AREA     Name, *LDA, *GDA, *PDA
  Bibliothek . . . . . .   *LIBL      Name, *LIBL, *CURLIB
Ausgabe  . . . . . .     *           *, *PRINT
Ausgabeformat  . . . . . *CHAR       *CHAR, *HEX

                Zusätzliche Parameter

System . . . . . *rmt          *LCL, *RMT
```

Fernen Datenbereich anzeigen

Lokale Datenbereiche

Für jeden Job erstellt das System automatisch einen speziellen lokalen Datenbereich (*LDA) mit einer fixen Länge von 1.024 Bytes. Dieser Datenbereich ist verfügbar, solange der Job aktiv ist, und kann genauso verarbeitet werden wie jeder andere Datenbereich. Mit einer Ausnahme: Lokale Datenbereiche können nicht explizit erstellt oder gelöscht werden. Außerdem kann der lokale Datenbereich nicht von anderen Jobs verwendet werden. Dies könnte uns zu der Annahme verleiten, dass diese lokalen Datenbereiche in der Bibliothek QTEMP erstellt werden. Diese Annahme ist allerdings falsch, denn dem lokalen Bereich ist keine Bibliothek zugeordnet. Sie können die lokalen Datenbereiche daher auch nicht qualifiziert ansprechen, sondern müssen den Sonderwert *LDA verwenden, um den lokalen Datenbereich des jeweiligen Jobs anzusprechen:

```
CHGDTAARA DTAARA(*LDA (1 100)) VALUE(,der neue Wert')
```

Und eine weitere Besonderheit ist zu beachten: Wenn Sie einen Batch-Job mit dem CL-Befehl SBMJOB starten, wird der Inhalt des lokalen Datenbereichs in den lokalen Datenbereich des Stapeljobs kopiert.

2.12.9.3.1
Seite 4

Gruppendatenbereiche
Zusätzlich werden von dem System automatisch 512 Bytes große Gruppendatenbereiche (*GDA) erstellt, wenn der interaktive Job zum Gruppenjob wird. Für die Verarbeitung gilt dasselbe wie für den lokalen Datenbereich: Ein Gruppendatenbereich kann weder explizit gelöscht noch erstellt werden. Vielmehr wird der Gruppendatenbereich gelöscht, sobald ein Job nicht mehr als Gruppenjob aktiv ist.

Datenbereiche für Programm-Initialisierungsparameter
Vorgestartete Jobs (PJ) können einen speziellen Datenbereich für Programm-Initialisierungsparameter (PIP) verwenden. Dieser Datenbereich wird unter dem Namen *PDA angesprochen. Auch diesen Datenbereich erstellt das System automatisch mit einer Länge von 2.000 Zeichen.

Die Funktion „%SST“ akzeptiert im Übrigen die Sonderwerte *LDA, *GDA und *PDA anstelle der Verwendung einer Variablen, was die Arbeit mit diesen Datenbereichen vereinfacht. Die folgende Anweisung überträgt die ersten zehn Zeichen des Gruppendatenbereichs in die ersten zehn Zeichen des lokalen Datenbereichs:

```
CHGVAR %SST(*LDA 1 10) VALUE(%SST(*GDA 1 10))
```

Programmkommunikation – Jobschalter

Zusätzlich zu den lokalen verfügt jeder Job über 8 Schalter in Form von logischen Variablen. CL-Programme können diese Schalter ein- und ausschalten und auf ihrem Zustand basierend Entscheidungen treffen. Wenn ein Job gestartet wird, werden diese Schalter mit den Werten initialisiert, die in der Job-Beschreibung angegeben sind. Wenn in der Job-Beschreibung für die Initialisierung der Schalter „OFF“ – also „00000000“ angegeben wurde, so haben alle Job-Schalter zu Programmbeginn den Status „OFF“. Sie sollten sich darauf allerdings nicht verlassen und in jedem Fall die Jobbeschreibung prüfen, um die Schalterstellungen zu überprüfen. Für meine Sitzung ist die Jobbeschreibung QDFTJOBD maßgebend. Schnell lässt sich mit dem CL-Befehl DSPJOBD (Display Jobdescription) die Schalterstellung prüfen:

```
                          Jobbeschreibung anzeigen

 Jobbeschreibung:   QDFTJOBD       Bibliothek:   QGPL

 Benutzerprofil . . . . . . . . . . . . . . . . . :   *RQD
 CL-Syntaxprüfung . . . . . . . . . . . . . . . . :   *NOCHK
 In Jobwarteschlange anhalten . . . . . . . . . . :   *NO
 Abbruchbewertung . . . . . . . . . . . . . . . . :   30
 Jobdatum . . . . . . . . . . . . . . . . . . . . :   *SYSVAL
 Jobschalter  . . . . . . . . . . . . . . . . . . :   00000000
 Antwort auf Anfragenachricht . . . . . . . . . . :   *RQD
```

Jobschalter anhand der Jobbeschreibung prüfen

Sie können die Schalter aus- und einschalten, indem Sie den Befehl CHGJOB (Change Job) aufrufen und die neue Schalterstellung im Parameter SWS angeben:

```
                          Job ändern (CHGJOB)

 Auswahl eingeben und Eingabetaste drücken.

 Zeittrennzeichen . . . . . .   ':'          *SAME, *SYSVAL, *BLANK, :...
 Jobschalter  . . . . . . . .   00000010     Zeichenwert, *SAME
 Zeitscheibe  . . . . . . . .   2000         Millisekunden, *SAME
 Auswählbar für Auslagern . .   *YES         *SAME, *YES, *NO
```

Neue Schalterstellung wurde hinterlegt.

Der Befehl CHGJOB – Jobschalter ändern

Den Parameter SWS finden Sie in den erweiterten Parametern auf der zweiten Seite. Er fordert einen 8-stelligen Zeichenwert an, der nur die Zeichenwerte „0“ (OFF), „1“ (ON) oder „X“ (aktuellen Wert beibehalten) enthalten darf. Mit dem folgenden Befehl werden beispielsweise die ersten zwei Schalter auf „ON“ gestellt, die nächsten drei Schalter erhalten den Wert „OFF“ und die letzten Schalter bleiben unverändert.

2.12.9.4
Seite 2

```
CHGJOB SWS('11000XXX')
```

Wollen Sie lediglich einen Schalter setzen, geben Sie folgenden Befehl ein:

```
CHGJOB SWS('XXXX0XXX')
```

Der fünfte Schalter erhält den Wert „0", alle anderen Schalter bleiben unverändert stehen.

Die Schalter hätten wenig Sinn, wenn Sie deren Status nicht in einem CL-Programm abfragen könnten. Dies kann auf zwei unterschiedlichen Wegen erfolgen:

– Mit dem Befehl „Jobattribute auffinden" (RTVJOBA)

– oder mit der Built-in-Funktion %SWITCH in einer IF-Anweisung.

Die erste Variante ist ein wenig umständlich und setzt voraus, dass Sie eine 8-stellige Variable haben – führt aber zum Ziel:

```
DCL     &switches        *CHAR  8

RTVJOBA SWS(&SCHALTER)

        IF (&SCHALTER    *EQ    '00011100') THEN(DO)
               :
               :
        ENDDO


IF (%SST(&SCHALTER 1 1) *EQ '1')     THEN(DO)
           :
           :
ENDDO
```

Das Beispiel zeigt zwei IF-Anweisungen. Die erste IF-Anweisung prüft, ob die Schalterstellung dem Vergleichswert „00011100" entspricht. Die zweite IF-Anweisung prüft, ob der erste Schalter auf „ON" gestellt ist.

Wie gesagt, diese Methode ist etwas aufwändig. Sie können aber alternativ auch die Built-In-Funktion %SWITCH verwenden. Die beiden kodierten IF-Anweisungen sähen dann folgendermaßen aus:

```
IF      COND(%SWITCH(00011100) ) THEN(DO) …..

IF      COND(%SWITCH(1XXXXXXX)) THEN(DO) …...
```

Drei Dinge sind bei diesem letzten Beispiel erwähnenswert:

– Die Funktion benötigt keine Variablen.
– Der Vergleichswert wird nicht in Hochkommata eingeschlossen.
– Die IF-Anweisungen enthalten lediglich die Anweisung %SWITCH, aber keinen Vergleichsoperator.

Was fängt man nun mit diesen Schaltern an? Schalter eignen sich, ähnlich wie lokale Datenbereiche, zur asynchronen Übergabe von Informationen an ein anderes Programm innerhalb desselben Jobs. Da die Anzahl der Schalter auf acht begrenzt ist, ist der Umfang der übertragbaren Informationen sehr stark eingeschränkt. Sollten Sie Gefallen an den Schaltern finden, aber mehr als acht Schalter benötigen, so können Sie entweder Teile der *LDA als Schalter verwenden oder aber einen eigenen temporären Datenbereich in der Bibliothek QTEMP für diese Anforderung erstellen. Bedenken Sie aber bitte, dass nicht alle HLL-Programme mit den Jobschaltern arbeiten können. RPG und Cobol unterstützen jedoch die Verwendung der Programm-Schalter.

2.12.9.4

Seite 4

Programmkommunikation – Nachrichten

Auch Nachrichten können zur Übergabe von Informationen zwischen zwei Programmen verwendet werden. Die Programme können entweder in demselben oder auch in einem anderen Job ausgeführt werden. Obwohl ich bereits ausführlich im Kapitel 2.12.7 über Nachrichten und den Befehl SNDPGMMSG geschrieben habe, erwähne ich die CL-Befehle SNDPGMMSG und RCVMSG der Vollständigkeit halber ein zweites Mal, weil beide Befehle letztendlich dazu dienen, Informationen, also Daten, zwischen den Programmen weiterzugeben.

Wie bereits im Kapitel 2.12.7 erläutert, können Nachrichten an eine Programm-Nachrichtenwarteschlange oder an eine permanente Nachrichtenwarteschlange (Objektart: *MSGQ) gesendet werden. Allerdings können Programm-Nachrichtenwarteschlangen nur dann für den Informationsaustausch zwischen zwei Programmen verwendet werden, wenn die beiden kommunizierenden Programme im selben Programmstapel (=Stack) aktiv sind. Andernfalls müssen permanente Nachrichtenwarteschlangen verwendet werden.

Als Beispiel können wir unser Programm, das die Transaktionsdaten verarbeiten soll, verwenden. Dieses Programm sendet den Wert „erfolgreich abgeschlossen“, wenn alle Transaktionen erfolgreich verarbeitet wurden. Im Fehlerfall erhalten wir die Nachricht „Transaktion mit Fehlern abgebrochen“.

Der zugehörige SNDPGMMSG-Befehl, den das verarbeitende HLL-Programm absetzt, sieht folgendermaßen aus:

```
SNDPGMMSG MSG(‚erfolgreich abgeschlossen‘) +
TOPGMQ(*SAME TRANSFER_B) MSGTYPE(*INFO)
```

Hochsprachen-Programme können einen CL-Befehl natürlich nicht direkt ausführen. Doch es gibt sowohl im RPG- als auch im Cobol-Umfeld Mittel und Wege dies zu realisieren. Eine Möglichkeit ist beispielsweise der Aufruf des APIs QCMDEXEC. Doch die RPG- und Cobol-Syntaxen sollen uns an dieser Stelle nicht auch noch interessieren.

2.12.9.5
Seite 2

Grundsätzlich ist diese Lösung nur möglich, wenn das empfangende Programm TRANSFER_B und das aufgerufene Programm PGM_KON im selben Programmstapel ausgeführt wird, was in meinem Beispiel allerdings der Fall ist, da ich das HLL-Programm mit der Anweisung CALL aufrufe. Muss diese Einschränkung umgangen werden, weil das Unterprogramm beispielsweise über den Befehl SUBMITJOB gestartet wird oder die Jobs asynchron ausgeführt werden müssen, können Sie alternativ jederzeit eine permanente Nachrichtenwarteschlange verwenden:

```
SNDPGMMSG MSG(‚erfolgreich abgeschlossen‘) TOMSGQ(TRANSFER) +
MSGTYPE(*INFO)
```

Ich verwende in dem Beispiel eine speziell für diese Aufgabe erstellte Nachrichtenwarteschlange TRANSFER und nicht etwa die Nachrichtenwarteschlange QSYSOPR, um die Programmkommunikation nicht mit den Betriebssystemnachrichten zu vermengen.

Jetzt muss natürlich das aufrufende Programm TRANSFER_B die Nachricht empfangen und auswerten. Das Programm TRANSFER_B könnte dafür folgendermaßen modifiziert werden:

```
SUBR SUBR(STATUS_3)
:
/* Das Programm PGM_KON bestätigt die Transaktion positiv oder
negativ*/

WHEN COND(&DATEI = ‚KONTAKTE‘) THEN(CALL PGM(PGM_KON))
ENDSELECT
CHGVAR &N VALUE(&N + 10)
/* Anschließend Nachricht auswerten                          */

RCVMSG MSGTYPE(*INFO) MSG(&MSG) WAIT(0)
SNDPGMMSG MSG(&datei *CAT &MSG) TOUSR(*SYSOPR)

IF (&MSG = ‚erfolgreich abgeschlossen‘) THEN(DO)
ALCOBJ OBJ((EPS002CR/TRANS_DTA *DTAARA *EXCL)) WAIT(30)
CHGDTAARA DTAARA(TRANS_DTA (&N 1)) VALUE(‚1‘)
DLCOBJ OBJ((TRANS_DTA *DTAARA *EXCL))
ENDDO

ELSE CMD(DO)
ALCOBJ OBJ((EPS002CR/TRANS_DTA *DTAARA *EXCL)) WAIT(30)
CHGDTAARA DTAARA(TRANS_DTA (&N 1)) VALUE(‚4‘)
DLCOBJ OBJ((TRANS_DTA *DTAARA *EXCL))
ENDDO
ENDSUBR
```

Nachdem das Programm PGM_KON die Verarbeitung beendet und den Status der Verarbeitung übermittelt hat, verwende ich den Befehl RCVMSG, um die Nachricht im aufrufenden Programm zu empfangen. Wird eine Nachricht empfangen, so wird der Nachrichtentext in die Variable &MSG gestellt. Die Variable wurde als alphanumerische Variable mit 30 Stellen deklariert. Sie könnte auch kürzer oder länger sein, solange sie lang genug ist, den Nachrichtentext aufzunehmen. Sie können den Parameter WAIT des Befehls RCVMSG auf einen bestimmten Wert (Sekunden) einstellen, um zu steuern, wie lange das Programm auf eine Nachricht warten soll. Die Angabe 0 wie in meinem Beispiel veranlasst das Programm hingegen nicht zu warten. Ist bereits eine Nachricht vorhanden, wird sie in die Variable &MSG übertragen, falls nicht, bleibt &MSG leer. Der gezeigte Befehl RCVMSG empfängt eine Nachricht aus der Nachrichtenwarteschlange des aktuellen Programms. Sendet das andere Programm eine Nachricht nicht an eine Job-Nachrichtenwarteschlange, sondern an eine permanente Nachrichtenwarteschlange, dann muss der Befehl RCVMSG natürlich entsprechend geändert werden:

```
RCVMSG MSGQ(TRANSFER) MSGTYPE(*INFO) MSG(&MSG) WAIT(0)
```

Eine Anmerkung zum Schluss: Ich verwende in meinem Beispiel einen Text, um die Ausführung zu überprüfen. Für die Dokumentation in Form einer Nachricht an den Systemoperator ist dies sicherlich ein guter Weg. Aber ich nutze den konstanten Text ebenfalls in meiner Bedingungsabfrage und dies ist natürlich fehleranfällig und daher nicht empfehlenswert. In der Praxis würde ich mich sicherlich für einen anderen Wert oder einen anderen Weg entscheiden, die Informationen auszutauschen. Hier handelt es sich aber um ein einfaches Beispiel, das den Einsatz von Nachrichten im Kontext der Programmkommunikation verdeutlichen soll!

2.12.9.5

Seite 4

Programmkommunikation – Datenwarteschlangen

Datenwarteschlangen sind permanente Objekte des Betriebssystems (Objektart *DTAQ), die ebenfalls dazu verwendet werden können, asynchron Informationen zwischen Programmen oder Jobs auszutauschen. Die asynchrone Kommunikation über Datenwarteschlangen ist bei weitem die performanteste Lösung.

Das Konzept der Warteschlangen ist Ihnen durch die vorangegangenen Kapitel hinlänglich bekannt. Datenwarteschlangen funktionieren ähnlich, mit dem Unterschied, dass Sie bestimmen, was darin enthalten ist und wann sie geleert werden.

Datenwarteschlangen können problemlos mit den CL-Befehlen CRTDTAQ (Create Dataqueue) erstellt und mit dem Befehl DLTDTAQ (Delete Dataqueue) gelöscht werden. Der Befehl CRTDTAQ hat folgende Grundparameter:

```
                  Datenwarteschlange erstellen (CRTDTAQ)

Auswahl eingeben und Eingabetaste drücken.

Datenwarteschlange . . . . . > TEST        Name
  Bibliothek . . . . . .       *CURLIB     Name, *CURLIB
Art  . . . . . .               *STD        *STD, *DDM
Maximale Eintragslänge . . . . > 20        1-64512
Auf Zusatzspeicher sichern . . .  *NO      *NO, *YES
Folge  . . . . . .             *FIFO       *FIFO, *LIFO, *KEYED
```

Grundparameter – CRTDTAQ

Datenwarteschlange
Der Parameter enthält den qualifizierten Namen der Datenwarteschlangen.

Art
Inzwischen unterscheidet man zwischen einer Standarddatenwarteschlange (*STD), die auf dem lokalen System erstellt wird, und einer RMTDTAQ, die als DDM-Datenwarteschlange (*DDM) erstellt wird. Für eine ferne Datenwarteschlange müssen Sie den Namen der RMTDTAQ, auf die zugegriffen wird, und den Namen des fernen Ziel-Systems (Parameter RMTLOCNAME), in dem sich die Datenwarteschlange befindet, nennen. Es ist ähnlich wie wir es bei den fernen Datenbereichen kennengelernt haben. Wir werden uns fürs Erste auf die lokalen Datenwarteschlangen konzentrieren.

Maximale Eintragslänge
Der Parameter „Maximale Eintragslänge“ ist nur für lokale Datenwarteschlangen erforderlich. Er bestimmt die maximale Länge der Einträge in der Datenwarteschlange. Dieser Wert sollte mindestens so groß sein wie der längste mögliche Eintrag in der Datenwarteschlange.

22. Ergänzung 3/2010

2.12.9.6
Seite 2

Dieser Wert ist in etwa vergleichbar mit der Feldlänge eines Feldes innerhalb einer Datei. Die Größe eines Eintrages muss mindestens 1 sein und kann maximal 64.512 Zeichen lang sein.

Auf Zusatzspeicher sichern
Hier wird festgelegt, ob die Datenwarteschlange grundsätzlich physikalisch gespeichert werden soll, wenn ein Eintrag hinzugefügt oder entnommen wird. Der Unterlassungswert ist *NO.

Folge
Der Parameter FOLGE legt die Organisationsform der Datenwarteschlange fest. **FIFO (first in, first out)* ist die typische Organisationsform (Unterlassungswert). Das ist vergleichbar mit einer Reihe wartender Menschen an einem Schalter. Die Wartenden kommen am Ende der Schlange an und verlassen sie am anderen Ende. Derjenige, der als erster angekommen ist, geht auch als erster wieder.

**LIFO (Last in, first out)* ist die typische Organisationsform eines Stapels (Stack). Vergleichbar ist die Organisationsform mit einem Stapel Tabletts in einer Kantine. Leere saubere Tabletts werden oben auf den Stapel gelegt, der immer höher wird. Der nächste Gast, der ein Tablett benötigt, nimmt das Tablett, das oben aufliegt. Das Tablett, das als letztes auf den Stapel gelegt wurde, wird somit als erstes benutzt. Das erste Tablett bleibt unter Umständen lange Zeit unbenutzt liegen.

**KEYED (geschlüsselt)* ist eine ganz andere Organisationsform. Die Einträge können willkürlich in die Datenwarteschlange gestellt werden und werden wahlfrei anhand bestimmter Kriterien, in der Regel anhand eines Schlüsselwertes, wieder entnommen.

Sobald die Grundparameter bestätigt werden, erhalten Sie in Abhängigkeit der Einträge in den Grundparametern weitere Parameter zur Eingabe:

```
Absender-ID hinzufügen . . . . .   *NO           *NO, *YES
Größe der Warteschlange:
  Maximale Anzahl an Einträgen     *MAX16MB      Zahl, *MAX16MB, *MAX2GB
  Anfangsanzahl an Einträgen . .   1000          Zahl
Automatisches Zurückfordern  . .   *NO           *NO, *YES
```

Erweiterte Parameter – CRTDTAQ

Absender-ID hinzufügen
Jeder Eintrag kann mit zusätzlichen Informationen gespeichert werden. Die Einträge können mit dem Namen des Jobs, der Bibliothek und des jeweiligen Benutzerprofils gekennzeichnet werden.

Größe der Warteschlange
Dieser Parameter gibt die Größe des Speicherbereichs an, der für die Datenwarteschlange maximal zugeordnet wird. Einerseits bestimmen Sie damit die maximale Größe der Datenwarteschlange von bis zu 2 Gigabyte. Andererseits legen Sie die Anfangsgröße des Speicherbereichs fest. Die Datenwarteschlange wird mit einer Größe erstellt, die ausreicht, um die anfängliche Anzahl an Einträgen mit der maximalen Eintragslänge aufzunehmen. In meinem Beispiel also mit einer Größe von 10 Byte * 1.000 Einträge = 10.000 Byte.

Automatisches Zurückfordern
Dieser Parameter gibt an, ob der für die Datenwarteschlange zugeordnete Speicherbereich automatisch freigegeben wird, wenn die Datenwarteschlange leer ist. Wenn Sie den Wert *YES wählen, wird der zugeordnete Speicherbereich freigegeben, sobald die Datenwarteschlange leer ist. Der Speicherbereich für die anfängliche Anzahl an Einträgen bleibt jedoch zugeordnet. Das automatische Zurückfordern des nicht belegten Speichers kann allerdings auch viel Zeit in Anspruch nehmen in Abhängigkeit der Größe Ihrer Datenwarteschlange. Daher ist es empfehlenswert, die Anfangsanzahl an Einträgen möglichst realitätsnah zu bestimmen. Ist die Anfangsanzahl zu gering gewählt, wird das System den Speicher entsprechend häufig zurückfordern und belastet dadurch das System.

Ähnlich wie bei Nachrichtenwarteschlangen können Einträge an die Datenwarteschlange gesendet und daraus empfangen werden. Ein Unterschied zu anderen Warteschlangen besteht aber vor allem darin, dass es keine CL-Befehle für diese Aufgaben gibt. Stattdessen müssen Sie von IBM zur Verfügung gestellte Programme (APIs) aufrufen.

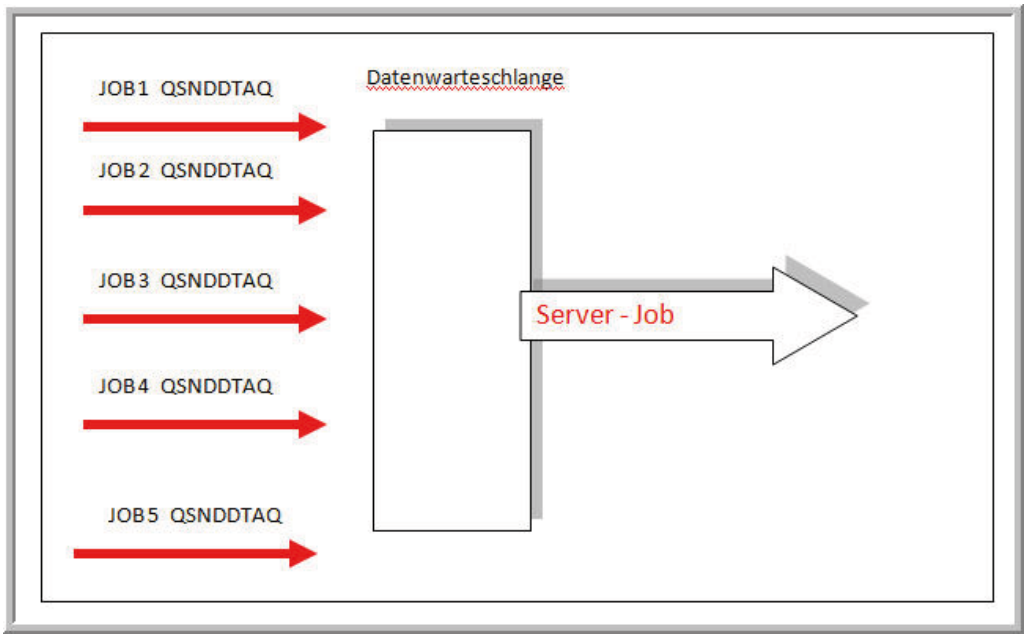
Manch ein Leser wird zurückschrecken, wenn er das Wort API liest. Doch dieses Verfahren ist nicht so schlecht, wie es auf den ersten Blick scheinen mag. Datenwarteschlangen werden u. a. auch von HLL-Programmen verwendet und diese Programme können viel einfacher ein API aufrufen als einen CL-Befehl absetzen. Datenwarteschlangen können dadurch problemlos aus jedem Programmumfeld verwendet werden. Auch zum Löschen der Einträge gibt es keinen CL-Befehl, sondern Sie nutzen die entsprechenden APIs:

QSNDDTAQ	Dateneinträge hinzufügen
QRCVDTAQ	Dateneinträge lesen
QCLRDTAQ	Dateneinträge löschen
QMHQRDQD	Datenwarteschlangenbeschreibung abrufen

Datenwarteschlangen sind nützlich und werden für eine Vielzahl unterschiedlicher Lösungen benötigt:

Asynchrone Kommunikation zweier parallel laufender Jobs
Arbeiten zwei Jobs parallel, von denen einer Daten in anderen Abständen sendet, als der andere Job diese verarbeiten kann, ist der Einsatz von Datenwarteschlangen bestens geeignet. Grundsätzlich wäre auch eine Lösung über Nachrichtenwarteschlangen denkbar, aber Datenwarteschlangen haben einen wesentlich höheren Durchsatz. Stellen Sie sich einfach folgendes Szenario vor: Es gibt zwei Jobs: einer gibt Daten in eine Druckdatei aus und der zweite Job überträgt die Daten zum Drucker. Der zweite Job ist normalerweise langsamer als der erste. Da die Druckausgaben allerdings in einer Ausgabewarteschlange gesammelt werden, spielt dies zunächst keine Rolle. Stellen Sie sich aber nun vor, dass nur wenige Drucker Etiketten drucken können und der Druckjob nicht nur das Etikett erstellt, sondern auch die Verteilung an den gerade zur Verfügung stehenden Etikettendrucker übernehmen soll. Der Druckjob würde permanent auf die Verfügbarkeit eines Druckers warten, bevor er wieder neue Daten übertragen könnte.

Ein anderes Beispiel sind Buchungsdaten. Stellen Sie sich vor, den ganzen Tag über werden Lagerbewegungen erfasst. Einige Lagerbewegungen sind schnell gebucht, weil vielleicht nur der verfügbare Bestand korrigiert werden muss. Andere Lagerbewegungen erfordern wesentlich mehr Arbeitsschritte, weil zunächst umfangreiche Prüfungen durchgeführt werden müssen, bevor die Buchung in den verschiedenen Dateien erfolgen kann. Die interaktive Eingabe könnte langsam werden, wenn viele Anwender gleichzeitig diese Arbeiten durchführen. Um die interaktive Eingabe zu beschleunigen könnte das Erfassungsprogramm sich auf die Prüfungen beschränken und anschließend einen Buchungssatz in eine Datenwarteschlange stellen. Ein zweiter Job mit niedriger Priorität empfängt die Daten aus der Warteschlange und führt die Buchungen durch. Das Fortschreiben der Bestände ist meistens sehr ressourcenintensiv; vor allem, wenn auch noch entsprechende Protokolldateien geführt werden müssen. Die Transaktionen könnten sich in der Datenwarteschlange sammeln und werden nacheinander abgearbeitet. Sobald bei der Erfassung eine Pause eintritt, wird das Buchungsprogramm den Rückstand wieder aufholen.



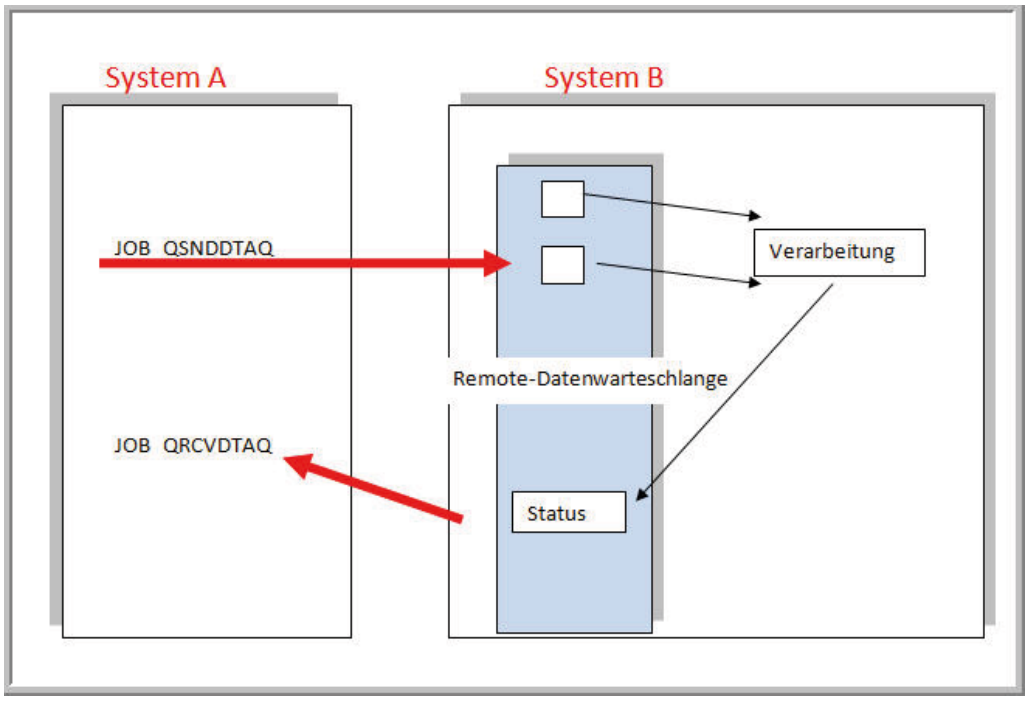
Wirkungsweise Datenwarteschlangen

Auch gibt es Situationen, in denen der Zeitpunkt unklar ist, wann die Daten zur Verarbeitung eintreffen. Auch in diesen Fällen können Datenwarteschlangen helfen. Wenn Sie Daten aus einer Datenwarteschlange empfangen, können Sie dem API mitteilen, dass es eine gewisse Zeit auf das Eintreffen der Daten warten soll.

Hinzu kommt, dass Sie Datenwarteschlangen auch verwenden können, wenn Sie beispielsweise auf die Eingabe innerhalb eines Bildschirms warten oder auf das Eintreffen einer Druckausgabe warten. Bildschirmdateien und auch Ausgabewarteschlangen können mit einer Datenwarteschlange verbunden werden.

2.12.9.6
Seite 6

Außerdem können Sie auch Remotedatenwarteschlangen erstellen und darüber die Kommunikation zwischen zwei Systemen realisieren.



Wirkungsweise ferner Datenwarteschlangen

Aber es gibt noch eine weitere Eigenschaft, die man kennen sollte, bevor man sich detailliert mit Datenwarteschlangen beschäftigt. Das größte Problem bei der Arbeit mit Datenwarteschlangen ist, dass ein Eintrag gelöscht wird, wenn er mit dem API QRCVDTAQ gelesen wird. Stürzt ein Verarbeitungsprogramm ab, bevor es die gelesenen Daten verarbeiten konnte, sind die Daten verloren. Allerdings können Sie mittlerweile Ihre Datenwarteschlangen journalisieren, so dass eine Wiederherstellung der Daten in letzter Konsequenz über das Journal denkbar wäre.

Und IBM hat es einfach versäumt eine komfortable Benutzeroberfläche für die Arbeit mit Datenwarteschlangen zu schaffen. Wenn Sie viel mit Datenwarteschlangen arbeiten wollen oder müssen, sollten Sie vielleicht darüber nachdenken, sich einige CL-Befehle selbst zu definieren, was die tägliche Arbeit sicherlich erleichtern wird.

Sie sehen, es gibt viele Einsatzmöglichkeiten für Datenwarteschlangen mit vielen Vorteilen, aber auch einigen Nachteilen. Letztendlich kommt es also – wie immer – auf die konkrete Aufgabenstellung an.

Der Befehl RUNSQL in der CL-Programmierung (neu TR4)

Wie wir gesehen haben, übernehmen CL-Programme heute vielfältige Aufgaben, und manchmal würden wir dabei gerne Fehler oder andere Informationen in Datenbankdateien protokollieren. Doch obwohl die IBM CL in letzter Zeit stark aufgewertet hat und viele neue Befehle integriert wurden, gibt es bislang keinen Schreibbefehl für Datenbankdateien.

Können wir also tatsächlich keine Informationen aus einem CL-Programm heraus in Datenbankdateien ablegen? Irrtum!

Ich schreibe schon seit vielen Jahren CL-Programme, die physische Datenbanklogdateien nutzen. Nur benutze ich dabei innerhalb des CL-Programms halt SQL, um diese Aufgabe zu lösen. Einziger Wehmutstropfen hierbei: Bislang mussten IBM DB2 Query Manager und das SQL Development Kit for i (5761ST1 bzw. 5770-ST1) installiert sein, damit die Programme laufen. Außerdem waren der Einsatz der Befehle RUNSQLSTM oder die Query Manager-Abfragen häufig etwas umständlich.

Doch diese Zeiten sind vorbei: IBM hat den Betriebssystembefehl **RUNSQL,** der es ermöglicht, SQL-Statements direkt aus einem CL-Programm heraus auszuführen oder – falls erforderlich – natürlich auch interaktiv auf einer 5250-Befehlszeile zu nutzen, in Release 6.1 und 7.1 integriert.

Folgende Voraussetzungen müssen allerdings erfüllt sein:

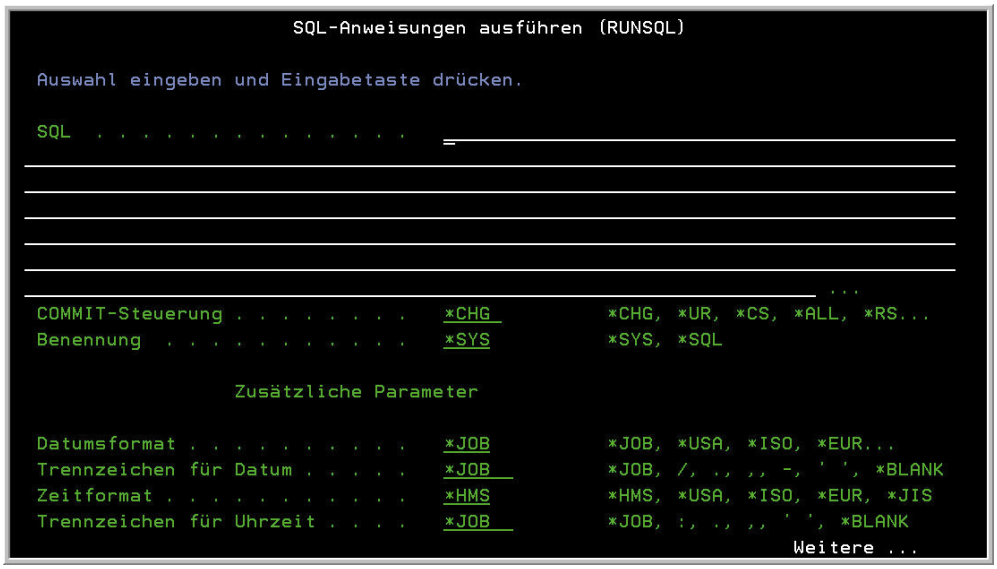
Release i6.1
- Datenbankgruppen-PTF SF99601 Level 25

Release i7.1
- Datenbankgruppen-PTF SF99701 Level 14

2.12.10
Seite 2

Datenbankmodifikationen mit RUNSQL

Bevor wir den Befehl in einem Beispielprogramm integrieren, wollen wir uns kurz mit den Befehlsparametern auseinandersetzen.



Der Befehl RUNSQL

Der Befehl RUNSQL hat mehrere Parameter:

Der wichtigste Parameter ist sicherlich **der** Parameter **„SQL“**: Ein 5.000 Byte großes Zeichenfeld, das ein **beliebiges** SQL-Statement enthalten kann. Wir könnten also neue Tabellen erstellen (CREATE TABLE), logische Sichten aufbauen (CREATE VIEW) oder bestehende Tabellen manipulieren (INSERT, UPDATE, DELETE). Aber auch Selektionen sind möglich, erfordern aber etwas mehr Codierung, da wir im CL-Programm bislang nicht mit Cursormengen arbeiten können.

Der Parameter **„COMMIT-Steuerung“** (**COMMIT**) bestimmt die Transaktionsstufe. Sie sehen in der Abbildung oben, dass die Standardeinstellung *CHG lautet. Sie müssen daher diesen Parameter unbedingt ändern, wenn Sie in Ihren Programmen nicht mit COMMIT und ROLLBACK-Operationen arbeiten wollen. Andernfalls würden alle Operationen rückgängig gemacht werden, wenn Sie Ihr CL-Programm beenden. Nutzen Sie die Einstellung *NONE, wenn keine Transaktionen gewünscht sind.

Bekannt dürfte auch der Parameter **„Benennung“** (**NAMING**) sein, wenn Sie bereits mit SQL arbeiten. Mit Hilfe dieses Parameters bestimmen Sie, welche Namenskonvention Sie im SQL-Statement für die Adressierung Ihrer Objekte verwenden.

Die weiteren Parameter **„Datumsformat“** (**DATFMT**), **„Trennzeichen für Datum“** (**DATSEP**), **„Zeitformat“** (**TIMFMT**) und **„Trennzeichen für Datum“** (**TIMSEP**) sind sicherlich selbsterklärend.

2.12.10.1
Seite 2

Das Gleiche gilt auch für den Parameter **„Trennzeichen für Uhrzeit“** (**TIMSEP**), den sie auf der folgenden Seite finden.

```
Trennzeichen für Uhrzeit . . . .   *JOB          *JOB, :, ., ,, ' ', *BLANK
Standarddatensammlung  . . . . .   *NONE         Name, *NONE
Dezimalzeichen . . . . . . . . .   *JOB          *JOB, *SYSVAL, *PERIOD...
Sortierfolge . . . . . . . . . .   *JOB          Name, *JOB, *LANGIDUNQ...
  Bibliothek . . . . . . . . . .                 Name, *LIBL, *CURLIB
Sprachen-ID  . . . . . . . . . .   *JOB
Daten kopieren zulassen  . . . .   *OPTIMIZE     *OPTIMIZE, *YES, *NO
Blocken zulässig . . . . . . . .   *ALLREAD      *ALLREAD, *NONE, *READ
SQL-Regeln . . . . . . . . . . .   *DB2          *DB2, *STD
Optionen für Dezimalergebnisse:
  Maximale Genauigkeit . . . . .   31           31, 63
  Maximale Anzahl Kommastellen     31           0-63
  Mindestzahl Kommastellen (Div)   0            0-9
Auflösung gleichzeit. Zugriff  .   *DFT         *DFT, *CURCMT, *WAIT
```

Der Befehl RUNSQL – erweiterte Parameter

Die Standarddatensammlung (**DFTRDBCOL**) spezifiziert das Standardschema für unqualifiziert adressierte Tabellen, Views, Indizes und SQL-Packages. Den Parameter benötigen Sie allerdings nur, wenn Sie mit der Namenskonvention *SQL arbeiten, ansonsten verwenden Sie – wie gewohnt – die Bibliotheksliste zur Qualifizierung Ihrer Objekte.

Dezimaltrennzeichen (**DECMPT**), **Sortierfolgen** (**SRTSEQ**) **und Sprachen-ID** (**LANGID**) kennen Sie von vielen anderen i5-Befehlen.

Der Parameter **Daten kopieren zulassen** (**ALWCPYDTA**) ist Ihnen vielleicht in anderen Interfaces schon begegnet. Letztendlich bestimmt der Parameter, ob SQL Ihr Result Set in Form einer Datenkopie zur Verfügung stellen darf oder ob Sie stets mit „Live“-Daten arbeiten müssen. Der Eintrag *OPTIMZE überlässt die Entscheidung dem SQL-Optimizer, der abhängig von Syntax, Interface und Datenzugriff fallweise entscheidet!

Spannend könnte auch der Parameter **„SQL-Regeln“** sein – zumindest für all diejenigen, die den SQL Ansi-Standard einhalten wollen oder müssen. Der Standardwert *DB2 erlaubt alle i5-spezifischen SQL-Syntaxen, die nicht unbedingt dem Standard entsprechen müssen. Verwenden Sie hingegen den Wert *STD, muss der SQL-Standard eingehalten werden.

Der Parameter **„Auflösung gleichzeit. Zugriff“** setzt eine relativ neue SQL-Eigenschaft. *WAIT ist die Standardeinstellung. Bislang sind wir es gewohnt, dass wir im Programm auf die Freigabe einzelner Sätze warten, wenn diese durch andere Jobs gesperrt sind. Dieses Verhalten entspricht dem Wert *WAIT. Abweichend hiervon könnten Sie auch den Wert *CURCMT verwenden. In diesem Fall würden Sie nur die festgeschriebenen bzw. freigegebenen Zeilen einer Tabelle erhalten; alle gesperrten Zeilen würden ignoriert.

Soweit die Theorie! Nun wollen wir den neuen Befehl im CL-Programm verwenden:

Das Programm DELJRNRCV löscht bei uns im Haus nicht mehr benötige Journalreceiver unserer Schulungsbibliotheken.

Zu diesem Zweck werden zunächst alle Receiver aller Bibliotheken mit dem Befehl DSPOBJD ermittelt. Das Ergebnis wird in die Datenbankdatei (*OUTFILE) JRNRCV in der Bibliothek QTEMP gestellt. Anschließend wird die Unterroutine READJRNRCV für die weitere Verarbeitung aufgerufen:

```
PGM

     DCLF       FILE(QADSPOBJ) OPNID(JRN)
     DCL        VAR(&EOF) TYPE(*LGL) VALUE('0')

     DSPOBJD    OBJ(*ALL/*ALL) OBJTYPE(*JRNRCV) OUTPUT(*OUTFILE) +
                OUTFILE(EPSYS/JRNRCV)

     CALLSUBR   SUBR(READJRNRCV)

     RETURN
```

In der Unterroutine nutze ich zunächst eine Datenbanküberschreibung: Da ich im DCLF-Eintrag die IBM-Datei QADSPOBJ verwende, meine Daten aber in die temporäre Datei QTEMP/JRNRCV ablege, muss ich vor dem Datenzugriff mit Hilfe des Befehls OVRDBF die Datei QADSPOBJ durch die Datei JRNRCV überlagern. Dieses Prozedere ist nur erforderlich, weil ich meine Daten temporär innerhalb des Jobs aufbauen möchte. Genauso gut könnten Sie eine permanente physische Datei verwenden und am Ende des Programms den Inhalt der Datei löschen.

```
SUBR       READJRNRCV

OVRDBF     FILE(QADSPOBJ) TOFILE(QTEMP/JRNRCV)
RCVF       OPNID(JRN)
MONMSG     MSGID(CPF0864) EXEC(CHGVAR VAR(&EOF) VALUE('1'))
```

Anschließend werden die Daten gelesen und verarbeitet bis keine weiteren Informationen mehr vorhanden sind. Um von den „alten“ Syntaxen wegzukommen, benutze ich hierfür eine DOWHILE-Schleife (DOWHILE) und die logische Hilfsvariable &EOF, die WAHR wird, wenn das EOF der Datei JRNRCV erreicht ist.

2.12.10.1
Seite 4

```
DOWHILE    COND(*NOT &EOF)

    IF COND(( &JRN_ODLBNM = 'EPS006CR') *OR +
            (&JRN_ODLBNM = 'EPS008CR')) THEN(DO)

       DLTJRNRCV  JRNRCV(&ODLBNM/&ODOBNM) DLTOPT(*IGNINQMSG)
       MONMSG     MSGID(CPF7022) EXEC(ITERATE)
    ENDDO

RCVF       OPNID(JRN)
MONMSG     MSGID(CPF0864) EXEC(CHGVAR VAR(&EOF) VALUE('1'))

ENDDO

DLTOVR     FILE(QADSPOBJ)
ENDSUBR
```

Das CL-Programm ist soweit fertig.

Jetzt möchte ich in einem nächsten Schritt alle gelöschten und nicht gelöschten Receiver in der Datenbankdatei DELJRNRCV protokollieren. Der Dateiinhalt soll später folgendermaßen aussehen:

Inhalt von EPSYS.DELJRNRCV - 172.16.111.23(Itraining)

	RCVNAME	RCVLIB	USER	DATETIME	DEL_JN
1	QSQJRN4097	EPS008CR	RASCHE	2012-06-29 16:02:45.018239	J
2	QSQJRN5097	EPS008CR	RASCHE	2012-06-29 16:02:45.024384	J
3	QSQJRN5098	EPS008CR	RASCHE	2012-06-29 16:02:45.047963	J
4	QSQJRN5099	EPS008CR	RASCHE	2012-06-29 16:02:45.053994	J
5	QSQJRN0001	EPSW07	RASCHE	2012-07-06 11:51:43.531236	J
6	QSQJRN1001	EPSW07	RASCHE	2012-07-06 11:51:43.543514	J
7	QSQJRN1002	EPSW07	RASCHE	2012-07-06 11:51:43.573149	J
8	QSQJRN1003	EPSW07	RASCHE	2012-07-06 11:51:43.580402	J
9	QSQJRN2003	EPSW07	RASCHE	2012-07-06 11:51:43.587391	J
10	QSQJRN2004	EPSW07	RASCHE	2012-07-06 11:51:43.594433	J
11	QSQJRN2005	EPSW07	RASCHE	2012-07-06 11:51:43.602141	J
12	QSQJRN2006	EPSW07	RASCHE	2012-07-06 11:51:43.609166	J
13	QSQJRN0001	EPSW07	RASCHE	2012-07-06 11:57:28.785504	J
14	QSQJRN1001	EPSW07	RASCHE	2012-07-06 11:57:28.891700	J
15	QSQJRN1002	EPSW07	RASCHE	2012-07-06 11:57:28.976177	J
16	QSQJRN1003	EPSW07	RASCHE	2012-07-06 11:57:29.007122	J
17	QSQJRN2003	EPSW07	RASCHE	2012-07-06 11:57:29.058805	J
18	QSQJRN2004	EPSW07	RASCHE	2012-07-06 11:57:29.116604	J
19	QSQJRN2005	EPSW07	RASCHE	2012-07-06 11:57:29.158157	J
20	QSQJRN2006	EPSW07	RASCHE	2012-07-06 12:52:08.043601	N
21	QSQJRN0002	EPSYS	RASCHE	2012-07-06 12:52:08.049621	N
22	QSQJRN0001	EPSYS	RASCHE	2012-07-06 12:43:58.305242	J

Die Protokolldatei DELJRNRCV

Die Datei hat fünf Felder: den Receivernamen, die zugehörige Bibliothek, den verantwortlichen Anwender, einen Zeitstempel und den Vermerk, ob der Receiver gelöscht wurde bzw. nicht gelöscht wurde. Die Datei können Sie mit DDS oder mit SQL erstellen; das spielt eigentlich keine Rolle. Ich habe mich hier allerdings für eine SQL-Tabelle entschieden, da ich mich so um den aktuellen Benutzer und den aktuellen Zeitstempel nicht kümmern muss. Das übernimmt SQL für mich. Die notwendigen SQL-Befehle zur Tabellenerstellung sehen folgendermaßen aus:

Das Statement DROP TABLE löscht eine bereits bestehende Tabelle.

```
DROP TABLE EPSYS/DELJRNRCV;
```

Mit dem nachfolgenden CREATE TABLE-Statement kann die in der oberen Abbildung gezeigte Tabelle erstellt werden:

```
CREATE TABLE EPSYS/DELJRNRCV (
       RCVNAME CHAR(10) NOT NULL DEFAULT '' ,
       RCVLIB CHAR(10)   NOT NULL DEFAULT '' ,

-- Standardwert, wenn Spalte beim Insert leer bleibt, ist der aktu-
elle Benutzer
       "USER" CHAR(18)   NOT NULL DEFAULT USER ,

--  Standwert, wenn die Spalte leer bleibt, ist der aktuelle Zeit-
stempel
       DATETIME TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP ,
       DEL_JN CHAR(1)   NOT NULL DEFAULT 'J' );

LABEL ON TABLE EPSYS/DELJRNRCV IS 'Dokumentation gelöschter Recei-
ver' ;

LABEL ON COLUMN EPSYS/DELJRNRCV
( RCVNAME TEXT IS 'Receiver-Name' ,
       RCVLIB TEXT IS 'Receiver-Bibliothek' ,
       "USER" TEXT IS 'Benutzername' ,
       DATETIME TEXT IS 'Datum-Uhrzeit' ,
       DEL_JN TEXT IS 'Receiver_gelöscht Ja/Nein' ) ;
```

In der Tabelle müssen nur der Reveiver-Name und die Receiver-Bibliothek eingefügt werden. Alle anderen Werte werden als Standardwert eingefügt.

```
INSERT INTO DelJrnRcv (RCVNAME, RCVLIB)
VALUES ('RCVNAME', 'RCVBIB');
```

Falls der Receiver nicht gelöscht werden konnte, muss ich mein Statement noch um einen weiteren Wert ergänzen:

```
INSERT INTO DelJrnRcv (RCVNAME, RCVLIB, DEL_JN)
VALUES ('RCVNAME', 'RCVBIB' , 'N');
```

2.12.10.1
Seite 6

Und genau diesen INSERT-Befehl verwende ich jetzt in meinem CL-Programm. Als Erstes definiere ich eine zusätzliche Variable. Die Variable &SQLSTM soll das auszuführende SQL-Statement enthalten:

```
DCL        VAR(&SQLSTM) TYPE(*CHAR) LEN(5000)
```

Die Variable erhält die Länge 5000, analog zum Parameter SQL des Befehls RUNSQL.

In diese Variable übertrage ich jetzt das SQL-Statement. Dabei ersetze ich natürlich die Konstanten durch meine CL-Variablen. Auf diesem Weg kann ich zur Laufzeit des Programms jedes SQL-Statement dynamisch aufbauen:

```
CHGVAR  &SqlSTM     VALUE (  +
```

Der erste Teil des Statements ist konstant, endet aber mit einem Hochkomma. Bedenken müssen Sie, dass wir im CL-Programm ein Hochkomma als Teil einer Zeichenkette nur definieren können, indem wir es verdoppeln.

```
 ' Insert into  DelJrnRcv(RCVNAME, RCVLIB)  values(''  '     +
```

Jetzt verbinden wir diese Zeichenkette mit dem Inhalt unserer CL-Variablen &JRN_ ODOBNM.

```
*CAT  &JRN_ODOBNM  +
```

Die Value-Werte der INSERT-Anweisung müssen in Hochkomma eingeschlossen und durch Komma getrennt werden; also fügen wir ein Hochkomma, das Komma und anschließend wieder ein Hochkomma an:

```
*TCAT   ''', '''        +
```

Nun kommt die zweite CL-Variable, die den Namen der Bibliothek enthält.

```
*CAT     &JRN_ODLBNM    +
```

Abgeschlossen wird das SQL-Statement mit einem schließenden Hochkomma und der schließenden Klammer.

```
*TCAT   ' '')'
 )
```

Etwas aufwendig ist hierbei sicherlich die Integration der einfachen Anführungszeichen, weil wir die einfachen Anführungszeichen doppeln müssen, wenn sie Bestandteil der Zeichenkette selbst sein sollen. Doch dies hat nichts mit den SQL-Aufrufen zu tun, sondern ist seit eh und je ein kleines Ärgernis in CL-Programmen.

Wir können dieses Problem aber auch umgehen, wenn wir einfach eine Konstante definieren, die ein Hochkomma enthält und anschließend die Konstante im Aufbau des SQL-Statements verwenden. Das folgende Beispiel zeigt Ihnen, was gemeint ist:

```
DCL        VAR(&SQLSTM) TYPE(*CHAR) LEN(5000)
DCL       VAR(&HK)            TYPE(*CHAR)  LEN(1)    VALUE('''')
```

Wenn wir jetzt das SQL-Statement aufbauen, können wir folgende Syntax verwenden:

```
CHGVAR  &SqlSTM     VALUE (  +
' Insert into  DelJrnRcv(RCVNAME, RCVLIB) values(' *CAT &HK *CAT  +
&JRN_ODOBNM  *TCAT &HK  *TCAT   ',  ' *CAT &HK *CAT  &JRN_ODLBNM  +
*CAT &HK  *CAT  ' )'
)
```

Nun muss das SQL-Statement mit Hilfe des neuen Befehls RUNSQL ausgeführt werden:

```
RUNSQL     SQL(&SQLSTM)  COMMIT(*NONE)  NAMING(*SYS)
```

Etwas aufpassen müssen wir nur noch mit der COMMIT-Steuerung. Der Standardwert ist *CHG, so dass wir alle Manipulationen an unserer Tabelle ausdrücklich mit der SQL-Anweisung COMMIT bestätigen müssten, spätestens wenn wir unser CL-Programm beenden. Daher habe ich mich entschieden, den Parameter COMMIT(*NONE) zu verwenden.

2.12.10.1
Seite 8

Integriere ich die oben erläuterten Zeilen in mein CL-Programm, sieht meine Unterroutine READJRNRCV folgendermaßen aus:

```
      SUBR       READJRNRCV

      DSPOBJD    OBJ(*ALL/*ALL) OBJTYPE(*JRNRCV) OUTPUT(*OUTFILE) +
                    OUTFILE(QTEMP/JRNRCV)

      OVRDBF     FILE(QADSPOBJ) TOFILE(QTEMP/JRNRCV)
      RCVF       OPNID(JRN)
      MONMSG     MSGID(CPF0864) EXEC(CHGVAR VAR(&EOF) VALUE('1'))

      DOWHILE    COND(*NOT &EOF)
         CHGVAR   &ERR VALUE('0')

           DLTJRNRCV  JRNRCV(&JRN_ODLBNM/&JRN_ODOBNM) +
              DLTOPT(*IGNINQMSG)

/* Receiver angehängt - kann nicht gelöscht werden
           MONMSG     MSGID(CPF7022) EXEC(DO)

              CHGVAR  &SqlSTM     +
               VALUE('Insert into deljrnrcv(RCVNAME, RCVLIB, DEL_
JN)    +
                                values(''' *CAT &JRN_ODOBNM *TCAT ''',
''' *CAT   +
                                &JRN_ODLBNM *TCAT  ''', ''N'')   ' )
              RUNSQL     SQL(&SQLSTM) COMMIT(*NONE)
              CHGVAR     &ERR Value('1')

           ENDDO

/* Kein Fehlerfall
          IF (*NOT &ERR)  then(DO)
              CHGVAR  &SqlSTM     +
              VALUE('Insert into deljrnrcv(RCVNAME, RCVLIB) +
                    values(''' *CAT &JRN_ODOBNM *TCAT ''', ''' *CAT
+
                           &JRN_ODLBNM *TCAT  ''')' )
             RUNSQL     SQL(&SQLSTM) COMMIT(*NONE)
          ENDDO

/* Nächsten Datenbankeintrag lesen
       RCVF       OPNID(JRN)
       MONMSG     MSGID(CPF0864) EXEC(CHGVAR VAR(&EOF) VALUE('1'))

    ENDDO
```

Sie sehen, dass es fortan sehr einfach ist, jeden SQL-Befehl direkt aus einem CL-Programm abzusetzen. Und Sie müssen natürlich auch nicht immer mit einer Hilfsvariablen (&SQLSTM) arbeiten. Allerdings ist es einfacher, wenn sie relativ komplexe SQL-Anweisungen mit vielen integrierten Hochkommata ausführen wollen. Sind die SQL-Anweisungen etwas einfacher, kann das SQL-Statement auch direkt im Parameter SQL codiert werden. Schauen Sie, was gemeint ist:

Ich integriere in das Programm eine weitere Funktion:

Alle Protokolleinträge, die älter als einen Monat sind, sollen aus der Protokolldatei gelöscht werden. Das notwendige SQL-Statement ist einfach:

```
DELETE FROM DelJrnRcv
WHERE DATE(datetime) <= CURDATE() - 30 DAYS
```

In diesem SQL-Statement gibt es keine integrierten CL-Variablen und auch keine integrierten Hochkommata, so dass ich den SQL-Befehl direkt im CL-Befehl verwende:

```
  /* Protokolleinträge löschen */
SUBR       DLTPRTE

          RUNSQL        SQL(' DELETE FROM DelJrnRcv   WHERE +
                              DATE(datetime) <= CURDATE() - 30
DAYS ') +
                              COMMIT(*NONE)

ENDSUBR
```

Und die Auswahl, welche Funktion aufgerufen wird, steuere ich mit einer entsprechenden SELECT-Anweisung im Hauptteil:

```
PGM     PARM(&FUNC)

         DCLF       FILE(QADSPOBJ)  OPNID(JRN)
         DCL        VAR(&EOF) TYPE(*LGL) VALUE('0')
         DCL        VAR(&FUNC) TYPE(*CHAR) LEN(1)
         DCL        VAR(&SQLSTM) TYPE(*CHAR) LEN(5000)


         SELECT
             WHEN  COND(&FUNC = '1') THEN(CALLSUBR
SUBR(READJRNRCV))
             WHEN  COND(&FUNC = '2') THEN(CALLSUBR SUBR(DLTPRTE))
             OTHERWISE  CMD(SNDPGMMSG MSG('Funktion' *BCAT &FUNC +
                          *BCAT 'nicht definiert!'))


         ENDSELECT
```

2.12.10.1
Seite 10

Auf diesem Weg können Sie jedes SQL-Statement interaktiv oder im CL-Programm nutzen.

Auch SELECT-Statements dürfen Sie aufrufen. Mit dem Result Set können wir nur im CL-Programm nicht unmittelbar etwas anfangen. Welche Möglichkeiten hierfür zur Verfügung stehen, lesen Sie im nächsten Kapitel.

Datenbankselektion mit RUNSQL

Im vorangegangenen Kapitel habe ich Ihnen gezeigt, welche neuen Möglichkeiten Sie mit dem Befehl RUNSQL haben, um Daten zu manipulieren. Wir können jetzt problemlos neue Datensätze in unsere Datenbankdateien einfügen (INSERT), aktualisieren (UPDATE) und löschen (DELETE).

Jetzt wollen wir uns anschauen, welche Möglichkeiten uns die Datenselektion bietet. Für die Ausgabe der Daten zeige ich Ihnen zwei unterschiedliche Lösungsansätze:

- Im ersten Beispiel geben wir die Daten auf einen Bildschirm aus.
- Im zweiten Beispiel werden wir die Informationen drucken ohne ein RPG-Programm einzusetzen.

Für beide Beispiele ist eigentlich nur noch die Fehlerroutine meines CL-Programms relevant.

```
/* Fehlerroutine - Receiver wurden nicht gelöscht - Vermerk in
Datenbankdatei   */

DLTJRNRCV  JRNRCV(&JRN_ODLBNM/&JRN_ODOBNM) DLTOPT(*IGNINQMSG)
MONMSG     MSGID(CPF7022) EXEC(do)

   CHGVAR  &SqlSTM     +
                VALUE('Insert into deljrnrcv(RCVNAME, RCVLIB, DEL_
JN) +
                      values(''' *CAT &JRN_ODOBNM *TCAT ''', ''' 
*CAT   +
                             &JRN_ODLBNM *TCAT  ''', ''N'')   ' )

    RUNSQL     SQL(&SQLSTM) COMMIT(*NONE)
```

Sicherlich können Sie diese Fehlerroutine weiter ausgestalten und beispielsweise die MSGID oder andere Informationen in der Datenbank speichern. Ich habe mich entschlossen, lediglich die Spalte DEL_JN zu verwenden, um nicht gelöschte Receiver zu dokumentieren.

Die nicht gelöschten Receiver kann ich sehr einfach mit folgendem SQL-Statement ermitteln:

```
SELECT     *
FROM  DelJrnRcv
WHERE DEL_JN = 'N'
AND DATE(datetime) = current_date - 1 DAY
```

2.12.10.2
Seite 2

Ich selektiere auf den gestrigen Tag, da die Bereinigungsfunktion bei uns vor 24:00 Uhr läuft, die anschließenden Protokolle aber erst einen Tag später geprüft werden. Als Ergebnismenge habe ich heute die folgenden Daten erhalten:

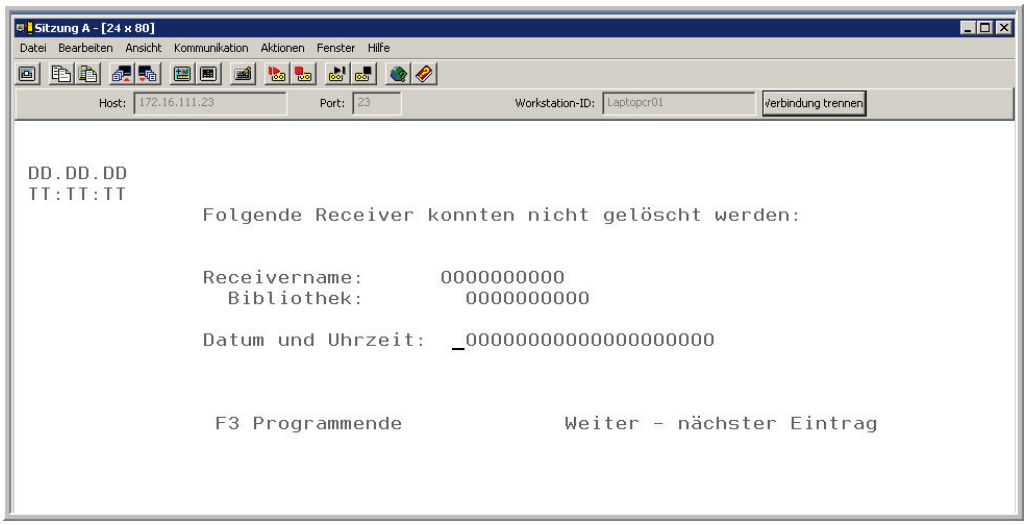
QSQJRN0002	EPSYS	RASCHE	2012-08-02 12:52:08.04...	N
QSQJRN5099	EPS008CR	RASCHE	2012-08-02 16:02:45.05...	N
RCVNAME	RCVLIB	USER	DATETIME	DE...

Die Ergebnismenge

Diese Ergebnismenge könnte ich jetzt am Bildschirm darstellen oder als Protokoll drucken.

Datenbankselektion mit Bildschirmausgabe

Nutzen wir zunächst eine einfache Bildschirmausgabe. Der Bildschirm ist schnell erstellt und könnte folgendermaßen aussehen:



Einzelsatzanzeige im Bildschirm

Nun brauchen wir eine Routine, um die notwendigen Daten zu selektieren. Natürlich könnte ich die Datei DELJRNRCV einfach Satz für Satz lesen und die notwendigen Prüfungen vornehmen. Doch bei der Datumsprüfung wird es dann schon kniffliger!

Ich greife daher in einer neuen Unterroutine GETDATEN auf das gezeigte SQL-Statement zurück und selektiere so nur Daten des Vortags. Diese Informationen stelle ich in die Tabelle QTEMP/DELJRNRCV durch einen vorangestellten CREATE TABLE-Befehl:

```
CREATE TABLE  QTEMP/DELJRNRCV
AS
(
SELECT *
FROM DelJrnRcv
WHERE DATE(datetime) =  CURDATE() - 1 day and DEL_JN = 'N'
)
WITH DATA
```

Das Statement verpacke ich natürlich wieder in den RUNSQL-Befehl:

```
SUBR GetDaten

DLTF  Qtemp/DelJrnRcv   /* Löschen falls bereits vorhanden */
MONMSG CPF0000
                /* Variable mit SQL-Statement füllen */
CHGVAR     VAR(&STM)    +
```

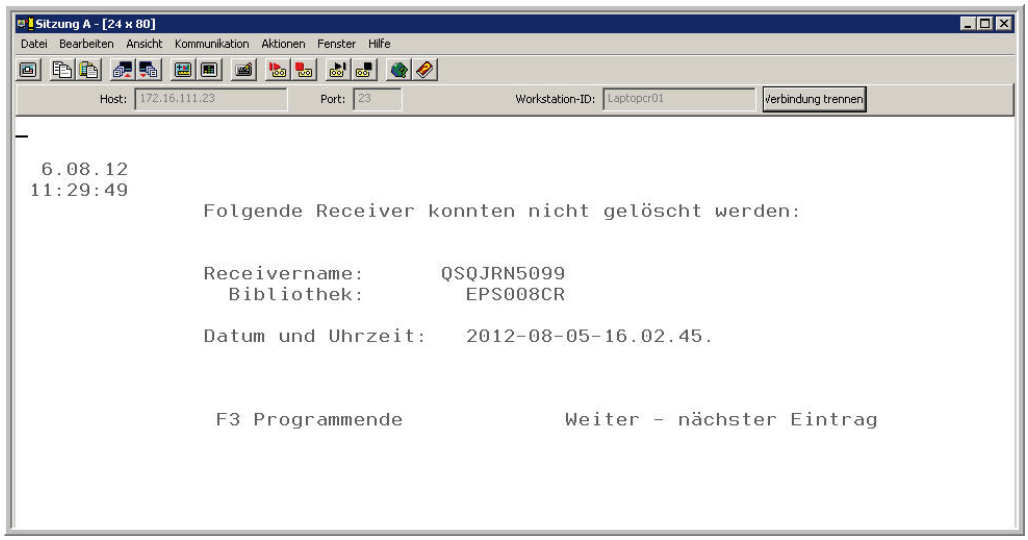
2.12.10.2.1
Seite 2

```
VALUE  ('CREATE TABLE QTEMP/DELJRNRCV AS    +
(SELECT *                                    +
                    FROM  EPSYS/DELJRNRCV    +
               WHERE date(datetime) = +
          CURRENT_DATE - 1     +
          AND  DEL_JN = ''N'')  +
          WITH  DATA ')
                 /* Statement ausführen */

RUNSQL SQL(&STM)  COMMIT(*NONE)

/* überschreibe die Originaldatei mit temporärer Datenmenge!
*/

OVRDBF     FILE(DELJRNRCV) TOFILE(QTEMP/DELJRNRCV)     LVLCHK(*NO)

ENDSUBR
```

Den Inhalt dieses Result Sets können wir jetzt problemlos als Einzelsätze am Bildschirm darstellen. Voraussetzung ist allerdings Release i6.1, denn wir müssen mit mehreren Dateien arbeiten, die anhand der zugewiesenen OPNID später angesprochen werden:

```
DCLF       FILE(epsys/deljrnrcv) OPNID(JRN)
 DCLF       FILE(epsys/JRNRCV) OPNID(DSP)

 DCL        VAR(&EOF) TYPE(*LGL) VALUE('0')
 DCL        VAR(&STM) TYPE(*CHAR) LEN(5000)

DCL        VAR(&Data) TYPE(*CHAR) LEN(20)
   DCL  VAR(&JRNRCV) TYPE(*CHAR) STG(*DEFINED) LEN(10)
DEFVAR(&DATA)
   DCL  VAR(&JRNLIB) TYPE(*CHAR) STG(*DEFINED) LEN(10) DEFVAR(&DATA
11)

  CALLSUBR   GetDaten                /* Daten selektieren */

  RCVF       OPNID(JRN)
   MONMSG   CPF0864 exec(chgvar &EOF Value('1'))

   DOWHILE (*NOT &DSP_IN03 *and *Not &EOF)
           chgvar &DSP_JRNRCV Value(&JRN_RCVNAME)
           chgvar &DSP_LIB Value(&JRN_RCVLIB)
           chgvar &DSP_DatTim Value(&JRN_DateTime)

        SNDRCVF    OPNID(DSP)
        RCVF   opnid(jrn)
         MONMSG CPF0864 exec(chgvar &EOF Value('1'))
   ENDDO

   DLTOVR     FILE(DELJRNRCV)
   DLTF Qtemp/DelJrnRcv

RETURN
```

Und so sieht das Ergebnis aus:



Das Ergebnis

Besser wäre es natürlich, wenn wir die Informationen in einer Subfile anzeigen könnten. CL kann aber keine normalen Subfiles verarbeiten. Doch wir könnten für diese Aufgabenstellung einfach eine Nachrichten-Subfile verwenden. Die Ausgabe sieht dann folgendermaßen aus:

Die Ergebnismenge im Nachrichtensubfile

Um die Ausgabe zu realisieren, brauchen wir zunächst eine DDS-Beschreibung für die Nachrichten-Subfile:

2.12.10.2.1
Seite 4

```
                                    DSPSIZ(24 80 *DS3)
                                    CA03(03)

                                    ERRSFL
R  MSGSFL                           SFL  SFLMSGRCD(9)
    MSGKEY                          SFLMSGKEY
    MSGQ                            SFLPGMQ

           R MSGCTL                 SFLCTL(MSGSFL)
                                    SFLSIZ(0011)
                                    SFLPAG(0010)
                                    OVERLAY
                                    SFLDSP  SFLDSPCTL
                                    SFLINZ
MSGQ                                SFLPGMQ
                     2  1 DATE     EDTCDE(Y)
                     2 54 TIME
                     5 25          'Nicht gelöschte Receiver'
                     6  1  '-----------------------------------'
```

Kurz einige Erläuterungen zur DDS-Definition:

ERRSFL kennzeichnet die Bildschirmdatei als Nachrichten-Subfile.

SFLMSGRCD bestimmt die Anfangszeile für den Subfile-Bereich.

SFLMSGKEY ist der Ausgabebereich im Subfile; hier werden später die Informationen dargestellt.

SFLPGMQ Nachrichten-Subfiles beziehen ihre Werte ausschließlich aus der Nachrichtenwarteschlange eines Programms. Welche Programmnachrichtenwarteschlange verwendet wird, bestimmt zur Programmlaufzeit der Inhalt der Variablen MSGQ.

Alle weiteren Definitionen in der DDS-Beschreibung sollten Ihnen aus der Subfile-Programmierung bekannt sein.

Das Programm GETDATEN wird jetzt ein wenig modifiziert. Die anzuzeigenden Daten werden dieses Mal nicht an den Bildschirm übertragen, sondern in die Nachrichtenwarteschlange des aufrufenden Programms SHOWDATEN gestellt. Mehr Änderungen sind im Programm nicht notwendig!

```
     DOWHILE (&EOF *NE '1')
         SNDPGMMSG  MSGID(CPF9897) MSGF(QCPFMSG) +
                      MSGDTA(&JRN_RCVNAME  *CAT ' ' *CAT +
                      &JRN_Rcvlib *cat &jrn_user)
     RCVF  opnid(jrn)
     MONMSG CPF0864 EXEC(chgvar &EOF Value('1'))
     ENDDO
```

Jetzt brauchen wir nur noch das aufrufende Programm:

```
PGM
          DCLF       FILE( JRNRCVSFL)      /* Nachrichtensubfile */
          RMVMSG     MSGQ(SHOWDATEN) CLEAR(*ALL)
          CHGVAR     VAR(&MSGQ) VALUE(SHOWDATEN)
          CALL GETDATEN
          SNDRCVF RCDFMT(MSGCTL)
         ENDPGM
```

Dieses Programm ist kurz. Es enthält die Deklaration der Nachrichtensubfile, löscht alle vorhandenen Nachrichten in der programmeigenen Nachrichtenwarteschlange und hinterlegt die Nachrichtenwarteschlange im Feld &MSGQ der Bildschirmdatei. Anschließend erfolgen der Programmaufruf und die Anzeige der Nachrichtensubfile.

Die Ergebnisse stehen in einer Nachrichtensubfile

2.12.10.2.1

Seite 6

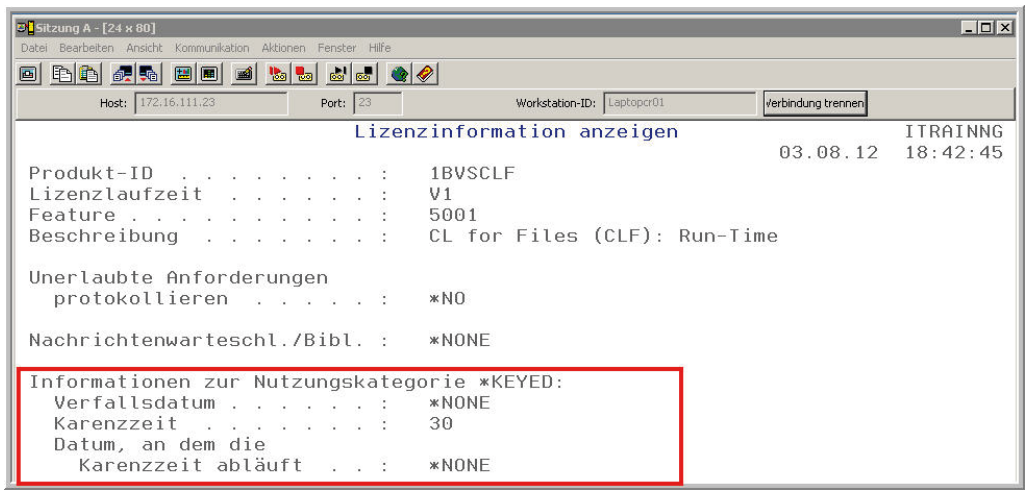
Datenbankselektion mit Druckausgabe

Und jetzt wollen wir unsere Daten nicht am Bildschirm ausgeben, sondern drucken. Natürlich geht das nicht mit Bordmitteln allein. Wir können aber das kostenlose Tool „Control Language for Files (CLF) 1BVSCLF“ von Bruce Vining Services, LLC verwenden. Detaillierte Informationen finden Sie auf der entsprechenden Webseite:

Webseite von Bruce Vining Services, LLC

Im Download-Bereich stehen Ihnen verschiedene Varianten zur Verfügung:
- als ZIP-Datei
- per FTP
- als E-Mail

Zusätzlich finden Sie dort einen guten Installationsleitfaden, so dass weder der Download noch die Installation Probleme bereiten sollten. Das Tool besteht aus mehreren Modulen (Optionen). Die Basislizenz, die ich in meinem nachfolgenden Beispielprogramm verwende, ist im Gegensatz zu den weiteren Optionen kostenlos und benötigt keinen Lizenzschlüssel. *(The base runtime (base option) and sample programs (option 4) do not require a license key).*

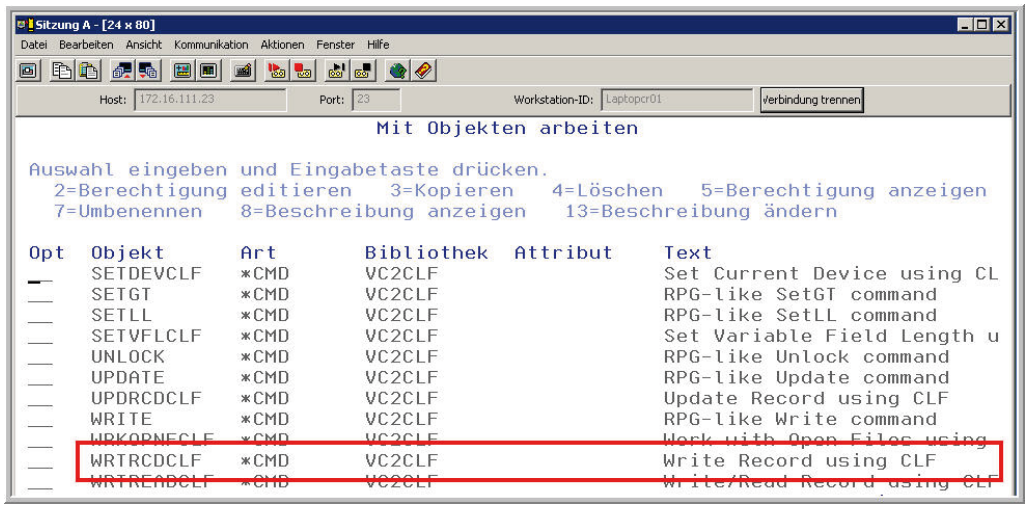


Die Basisoption benötigt keinen Lizenzschlüssel

2.12.10.2.2
Seite 2

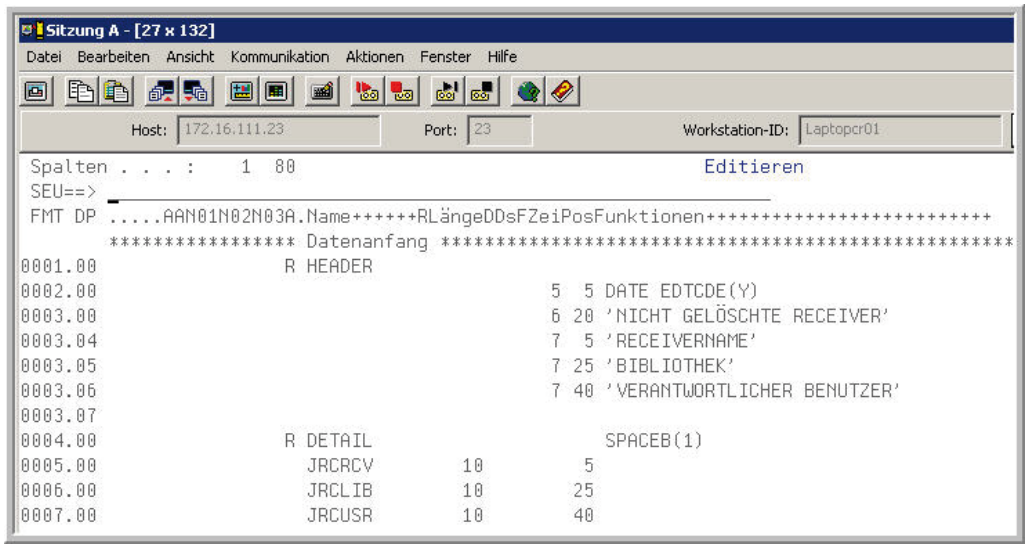
Die anderen Module sind dagegen kostenpflichtig, stehen aber ebenfalls erst einmal als 30-Tage-Testversion zur Verfügung.

Wenn die Basisoption installiert ist, finden Sie auf Ihrem System die Bibliothek VC2CLF. Darin befinden sich eine Vielzahl neuer CL-Befehle.



Neue Commands in der Bibliothek VC2CLF

U. a. enthält die Bibliothek den Befehl WRTRCDCLF, den ich gleich im CL-Programm verwende, um meine Daten zu drucken. Bevor es soweit ist, benötigen wir aber erst einmal eine externe Druckdatei. Die Druckdatei nenne ich DRKJRNRCV Sie könnte folgendermaßen aussehen:

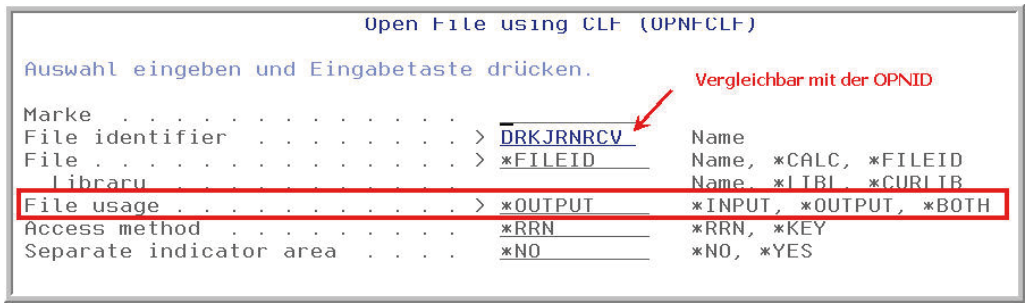


Die Druckdatei DRKJRNRCV

Meine Druckdatei kennt lediglich zwei Formate, eine Überschrift HEADER und die Datenzeile DETAIL.

Im CL-Programm kann ich diese Datei mit Standard-CL-Befehlen normalerweise nicht ansprechen. Jetzt stehen mir aber zwei neue CL-Befehle zur Verfügung, die es mir erlauben, die externe Druckdatei DRKJRNRCV zu öffnen und Zeilen zu drucken.

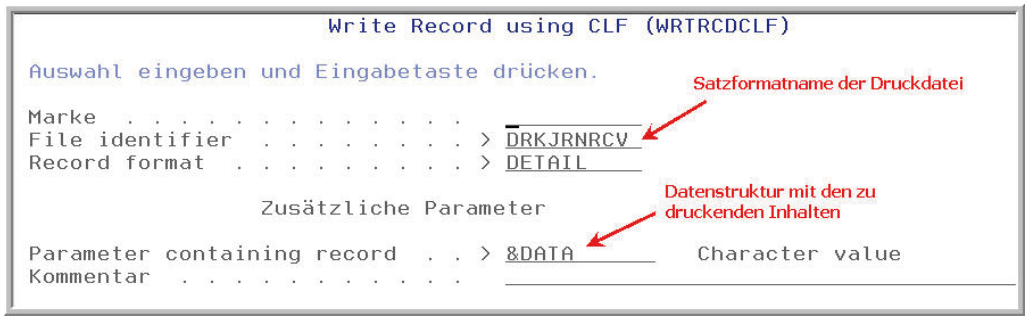
Öffnen kann ich die Druckdatei mit dem Befehl OPNFCLF:



Der Befehl OPNFCLF

Die Druckdatei wird nicht implizit vom CL-Programm geöffnet, sondern muss explizit von uns im Programm geöffnet werden.

Wenn wir Zeilen drucken wollen, benötigen wird den CLF-Befehl WRTRCD-CLF:



Der Befehl WRTRCDCLF

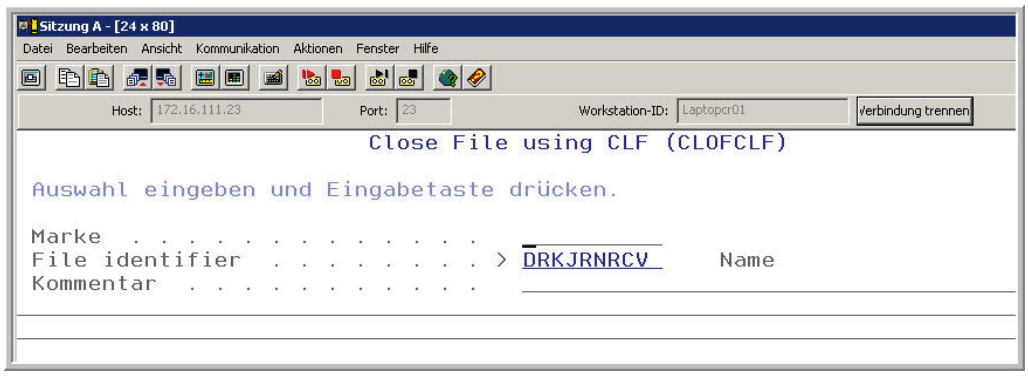
Mit Hilfe dieses Befehls verbinden wir einen Datenpuffer in unserem Programm mit einem Satzformat der Druckdatei. Sie müssen bei der Deklaration des Puffers lediglich darauf achten, dass Sie den Puffer bzgl. Variablentyp und Länge korrekt definieren. Ansonsten ist die Deklaration heute ein Kinderspiel, da wir ja jetzt auch im CL-Programm Datenstrukturen verwenden können:

```
DCL    VAR(&Data) TYPE(*CHAR) LEN(20)
    DCL  VAR(&JRNRCV) TYPE(*CHAR) STG(*DEFINED)   LEN(10)
DEFVAR(&DATA )
    DCL  VAR(&JRNLIB) TYPE(*CHAR) STG(*DEFINED) LEN(10)
DEFVAR(&DATA 11)
    DCL  VAR(&JRNUSR) TYPE(*CHAR) STG(*DEFINED) LEN(10)
DEFVAR(&DATA 11)
```

2.12.10.2.2
Seite 4

Der Befehl WrtRcdClf erzeugt also die eigentliche Druckausgabe, wobei die zu druckenden Daten später im Parameter RCDBUF (Record Buffer) stehen und das zu verwendende Druckformat durch den Parameter RCDFMT gewählt wird.

Abschließend benötigen wir noch den Befehl CLOFCLF:



Der Befehl CLOFCLF

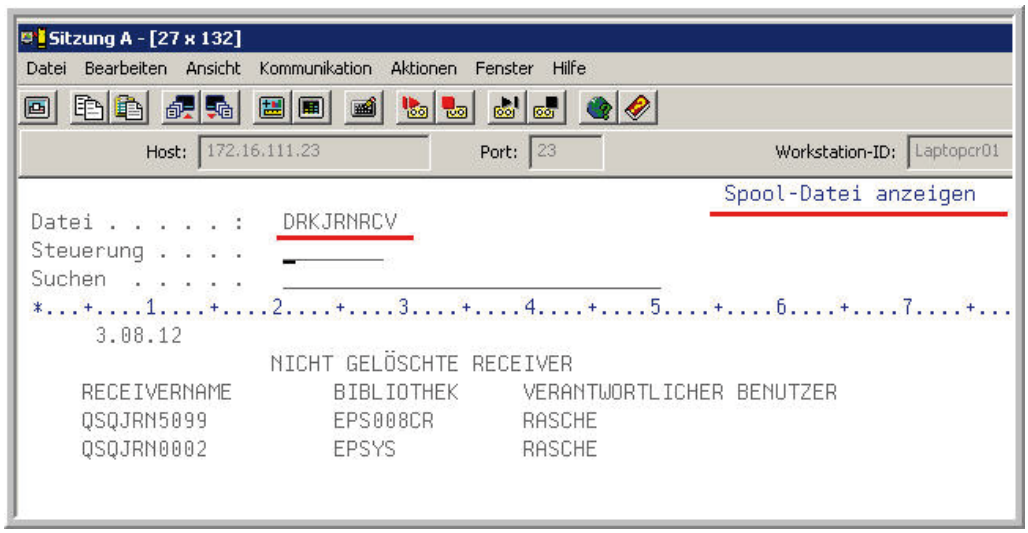
Am Ende des CL-Programms müssen wir die Druckdatei wieder schließen. Mehr als diese drei neuen Befehle sind nicht notwendig, um aus einem CL-Programm heraus Protokolle zu drucken.

```
PGM

     DCLF       FILE( deljrnrcv) OPNID(JRN)

     DCL        VAR(&EOF) TYPE(*LGL) VALUE('0')
     DCL        VAR(&STM) TYPE(*CHAR) LEN(5000)
 /* für Druckausgabe */

     DCL        VAR(&NoData) TYPE(*CHAR) LEN(1)
     DCL        VAR(&Data) TYPE(*CHAR) LEN(20)
     DCL  VAR(&JRNRCV) TYPE(*CHAR) STG(*DEFINED) LEN(10)
DEFVAR(&DATA)
     DCL  VAR(&JRNLIB) TYPE(*CHAR) STG(*DEFINED) LEN(10)
DEFVAR(&DATA 11)
    DCL  VAR(&JRNusr) TYPE(*CHAR) STG(*DEFINED) LEN(10)
DEFVAR(&DATA 21)


     callsubr   GetDaten

     OPNFCLF    FILEID(DRKJRNRCV) USAGE(*OUTPUT) LVLCHK(*NO)
     WRTRCDCLF  FILEID(DRKJRNRCV) RCDFMT(HEADER) RCDBUF(&NODATA)


     RCVF       OPNID(JRN)
     monmsg CPF0864 exec(chgvar &EOF Value('1'))
     dowhile (*Not &EOF)
            chgvar &JRNRCV Value(&JRN_RCVNAME)
```

```
             chgvar &JrnLIB Value(&JRN_RCVLIB)
chgvar &JrnusrValue(&JRN_user)
             WRTRCDCLF  FILEID(DRKJRNRCV) RCDFMT(DETAIL)
RCDBUF(&DATA)


             rcvf opnid(jrn)
             monmsg CPF0864 exec(chgvar &EOF Value('1'))
    enddo
    CLOFCLF    FILEID(DRKJRNRCV)
    DLTOVR FILE(DELJRNRCV)
    Dltf qtemp/DelJrnRcv
    Return
    /* Unterroutinen */
    subr GetDaten
      dltf qtemp/deljrnrcv
      monmsg cpf0000
      CHGVAR     VAR(&STM) VALUE('CREATE TABLE QTEMP/DELJRNRCV AS +
           (SELECT *                                        +
            from epsys/deljrnrcv where date(datetime) = +
            curdate() and DEL_JN = ''N'') with data ')
      RUNSQL sql(&STM) commit(*NONE)
      OVRDBF     FILE(DELJRNRCV) TOFILE(QTEMP/DELJRNRCV) +
                         LVLCHK(*NO)
   Endsubr


       ENDPGM
```

2.12.10.2.2
Seite 6

Sind alle Programme und die Druckdatei kompiliert, erhalte ich als Ergebnis den gewünschten Report:



ILE CL-Programme

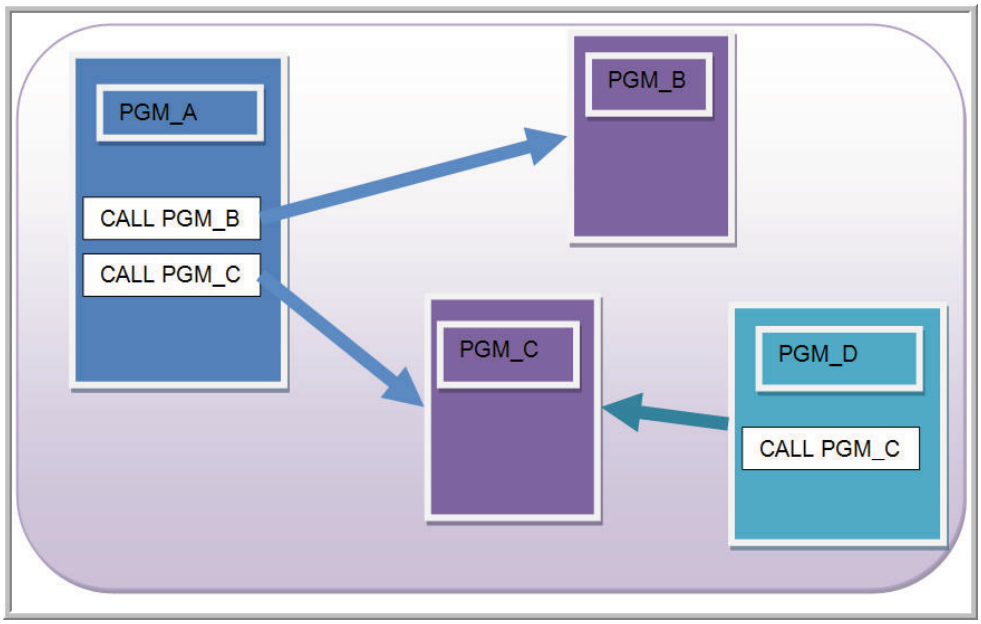
1993 führte IBM ein neues Modell für Programmiersprachen auf der damaligen AS/400 ein. Integrated Language Environment (ILE). Wie der Name impliziert, bietet ILE eine integrierte Schnittstelle für verschiedene Programmiersprachen. ILE sollte die Anwendungsentwicklung, Qualität, Zuverlässigkeit, Wartungsfreundlichkeit sowie die Performance der Programme verbessern. ILE sollte damit die Grundlage bilden für die Anwendungsentwicklung der Zukunft auf einer AS/400. Um das ILE-Konzept und dessen Neuerungen aber zu verstehen, müssen wir uns zunächst ein wenig mit der Vergangenheit beschäftigen.

24. Ergänzung 9/2010

2.13
Seite 2

Original Program Model (OPM)

Obwohl wir seit langem die Konzepte der modularen Programmierung kennen und auch wissen, dass das Erstellen kleiner Programme mit einzelnen Funktionen lobenswert ist, erreicht dieses Vorgehen schnell seine Grenzen. Kleine ausführbare Programme zu entwickeln, bedeutet in der Praxis, dass mehrere Programme andere externe Programme aufrufen. Das jedoch beeinträchtigt schnell die Performance, denn vor ILE konnten externe Programme nur dynamisch über CALL aufgerufen werden. Wir sprechen in diesen Fällen auch vom externen Programmaufruf.



Das Original Program Model

Beim dynamischen Aufruf muss jedoch der Bezug zum aufrufenden Programm zunächst einmal hergestellt werden, bevor das eigentliche Programm verwendet werden kann. Je mehr Programmaufrufe aber in einer Applikation stattfinden, desto mehr wird bei diesem Konzept die Performance leiden. Aus diesem Grund müssen viele Funktionen in einem Programmobjekt kombiniert werden, um die Aufrufe externer Programme auf ein erträgliches Maß zu begrenzen; mit der Folge, dass die einzelnen Programme zahlreiche Funktionen enthalten und dadurch in der Regel ziemlich umfangreich sind oder einzelne Funktionen in unterschiedlichen Programmen mehrfach codiert werden.

Die in der Abbildung beschriebene Umgebung wird als OPM (Original Program Model) bezeichnet. Es ist die Umgebung, die wir seit der Geburt der AS/400 kennen. Durch die zunehmende Komplexität der Anwendungen aufgrund grafischer Schnittstellen, ereignisgesteuerter Programmierung und Client-Server-Programmierung wuchs aber die Bedeutung der Programmmodularität und das oben dargestellte Konzept war nicht mehr zeitgemäß.

2.13.1 Seite 2

Integrated Language Environment (ILE)

ILE konnte und kann den Problemen der OPM-Architektur gerecht werden, indem einige Dinge von Grund auf geändert wurden. Das ILE-Konzept erlaubt es, Programme zunächst als nicht-ausführbare Module zu kompilieren und diese anschließend zu verbinden, um ein ausführbares Programmobjekt zu erhalten. Dieses modulare Vorgehen erscheint auf den ersten Blick umständlich, muss man doch zwei Kompilierungsschritte durchführen, um ein lauffähiges Programm zu erzeugen.

Doch gerade dieses modulare Vorgehen ermöglicht es beispielsweise, wiederverwendbare Komponenten zu erstellen, ohne die Performance-Einbußen externer Programmaufrufe hinnehmen zu müssen. Aufrufe eines Programms oder besser gesagt eines Moduls, das mit dem aufrufenden Programm verbunden ist, können mit der gleichen Geschwindigkeit wie eine Subroutine ausgeführt werden.

Zudem kann darauf verzichtet werden, dieselbe Routine in mehreren Programmen zu codieren. Ein und dasselbe Modul kann jederzeit in verschiedene Programme eingebunden werden.

Außerdem müssen die Programmmodule nicht in einer einheitlichen Sprache geschrieben werden. Module unterschiedlicher Sprachen können zu einem Programmobjekt verbunden werden, ohne dass Änderungen in der Weitergabe der Parameter der Laufzeit-Attribute sowie der Fehlerbearbeitung erforderlich wären. Jede Sprache kann so die individuellen Stärken ausspielen und letztendlich ein gemeinsames Programmobjekt bilden.

Hinzu kommt, dass die Ressourcen, welche das Programm benötigt, effektiver genutzt werden, indem der Job in Aktivierungsgruppen aufgeteilt wird und das Öffnen der Dateien, die gemeinsame Nutzung von Dateien und das Laden von Programmen innerhalb einer Aktivierungsgruppe gesteuert wird.

24. Ergänzung 9/2010

2.13.2
Seite 2

Integrated Language Environment (ILE) und CL-Programmierung

Obwohl IBM bereits ILE CL mit Release V3R1 auf dem System-i-Server einführte, werden die Begriffe ILE und CL nicht häufig in einem Satz genannt, denn in der Regel wird ILE mit ILE RPG gleichgesetzt und manchmal ist ILE RPG dann auch noch identisch mit der Programmiersprache RPG IV. Beides ist falsch. Während RPG IV nur eine Weiterentwicklung der Programmiersprache RPG III mit neuen Bestimmungsarten, Built-in-Funktionen und Free-Format-Codierung ist, handelt es sich bei ILE um ein neues Entwicklungskonzept, ein neues Architekturmodell. ILE erlaubt die Kombination unterschiedlicher Programmiersprachen in einem Programm und ermöglicht vor allem einen modularen Programmaufbau, wie er mit OPM-Programmen nicht möglich war.

Ein neues Konzept erfordert aber auch neue Denkweisen und bringt neue Objektarten und unbekannte Begriffe und ungewohnte Erstellungsprozesse mit sich. Wir wollen daher in diesem Kapitel ein OPM CL-Programm Schritt für Schritt zu einem ILE-Programm „umbauen“ und dabei sowohl die neue Architektur beleuchten als auch neue Begriffe und Objekttypen verstehen. Schauen wir uns dafür als erstes das Originalprogramm OPM_ORIGIN an:

Dem Programm OPM_ORIGIN wird der Name eines SQL-Skriptes zur Ausführung übergeben.

```
PGM PARM(&SQLSTM)
DCL &SQLSTM *CHAR 10
```

Mit Hilfe der SQL-Skripte sollen Datentransferanforderungen realisiert werden. Wir müssen uns zunächst ein solches SQL-Skript anschauen, um das zugehörige CL-Programm zu verstehen:

```
/* Sicht zum Aufteilen der Daten wird erstellt */
 CREATE VIEW QTEMP.ORDERHDR AS (SELECT
     ORDER_NUMBER,
     CUSTOMER_NUMBER,
     ORDER_DATE,
     ORDER_DELIVERY,
     ORDER_TOTAL,
     ORDER_SALESREP,
     INSERT_DATE,
     DATUM_CHAR
 FROM EPS0042CR.ORDERHDR
 WHERE YEAR(ORDER_DATE) = '2010');
```

Im SQL-Skript wird eine SQL-View in der Bibliothek QTEMP erstellt; im Beispiel mit den Umsatzdaten des aktuellen Geschäftsjahres. Die SQL-Abfrage ist hierbei allerdings unerheblich. Es kann jedes SQL-Statement verwendet

2.13.2.1
Seite 2

werden, solange die Regeln für das Erstellen einer SQL-View beachtet werden.

Der zweite Befehl des SQL-Skripts erledigt den Datentransfer. Mit Hilfe des CL-Befehls CPYTOIMPF (Copy to Importfile) können System-i-Daten auf den PC transferiert werden. Dabei wird nicht nur die Zeichenumsetzung von EBCDIC nach ASCII realisiert, sondern es können zudem Informationen über Satzendekennzeichen, Feldbegrenzer und Darstellungsformate von Datums-, Zeit- und Dezimalzahlen bestimmt werden:

```
/* Daten exportieren aus EPS0042CR.ORDERHDR */

 CL: CPYTOIMPF  FROMFILE(QTEMP/ORDERHDR)
        TOSTMF('/home/rasche/OrderHDR.csv')
        STMFCCSID(*PCASCII) STMFAUT(*DFT)
        MBROPT(*REPLACE) FROMCCSID(*FILE)
        RMVBLANK(*BOTH)   NUMFLDPAD(*ZERO)
        RCDDLM(*CRLF) DTAFMT(*DLM)
        STRDLM(*DBLQUOTE) FLDDLM(';')
STRESCCHR(*STRDLM)
        DECPNT(*COMMA) DATFMT(*EUR) TIMFMT(*EUR);
```

Neu seit Release V6R1 ist es, dass SQL-Skripte neben den eigentlichen SQL-Abfragen auch CL-Befehle enthalten können.

Kommen wir nun wieder zurück zu unserem CL-Programm OPM_ORIGIN. In einem ersten Schritt wird geprüft, ob die SQL-View, die im Skript erstellt wird, in der Bibliothek QTEMP bereits existiert. Hierbei muss allerdings die Konvention gelten, dass die Namen der exportierten Dateien als Skriptnamen verwendet werden.

```
CHKOBJ     OBJ(QTEMP/&SQLSTM) OBJTYPE(*FILE)
        MONMSG     MSGID(CPF9801) EXEC(DO)
                                  CALLSUBR TRANS
                                  RETURN
                   ENDDO
```

Ist die Datei bislang nicht vorhanden, wird die Unterroutine TRANS aufgerufen, die das SQL-Skript ausführt. Wird hingegen im CL-Programm festgestellt, dass die SQL-View bereits in der Bibliothek QTEMP existiert, löscht das CL-Programm zuerst die SQL-View und ruft danach die Unterroutine auf.

```
        DLTF  FILE(QTEMP/&SQLSTM)
        CALLSUBR TRANS

 RETURN
```

Sicherlich könnten wir auch einen DROP VIEW-Befehl in das SQL-Skript integrieren. Doch dann würden wir Laufzeitfehler mit der Bewertung 20 erhalten, wenn wir versuchen, ein nicht vorhandenes Objekt zu löschen. Mit der Folge, dass wir unser SQL-Skript mit einem ERRLVL (Error Level) 20 ausführen müssten.

Innerhalb der Unterroutine TRANS wird der Anwender als erstes über den Beginn der Transaktion durch eine Statusmeldung informiert. Für die Statusmeldungen benötigen wir den SNDPGMMSG-Befehl, da nur dieser Befehl STATUS-Meldungen (MSGTYPE *STATUS) versenden kann. Gesendet wird die Status-Nachricht an die *EXT-Nachrichtenwarteschlange des Aufrufers. Die Kombination von TOPGMQ(*EXT) MSGTYPE(*STATUS) sendet eine Echtzeitnachricht an die Arbeitsstation, die Sie lesen können, während das Programm ausgeführt wird.

Wirkungsweise von Statusmeldungen

Da Statusmeldungen aber nur als vordefinierte Nachrichten gesendet werden dürfen, wenden wir einen kleinen Trick an. Wir nutzen die MSGID CPF9897 der IBM-Nachrichtendatei QCPFMSG. Diese vordefinierte Nachricht enthält keinen Text, sondern lediglich einen Platzhalter, der bis zu 512 Zeichen Text darstellen kann. Den anzuzeigenden Nachrichtentext können wir anschließend mit Hilfe des Parameters MSGDTA frei formulieren. Sie müssen nur daran denken, die Nachricht vor dem auszuführenden Befehl zu senden. Weitere Informationen zur Verwendung von Nachrichtendateien enthält das Kapitel 2.12.7.4.

```
SUBR TRANS

   SNDPGMMSG  MSGID(CPF9898) MSGF(QCPFMSG) +
         MSGDTA('Datentransfer startet') +
         TOPGMQ(*EXT) MSGTYPE(*STATUS)
   DLYJOB DLY(2)
```

Den DLYJOB-Befehl verwende ich im Beispielprogramm lediglich, um genügend Zeit zu haben, die Statusmeldungen zu lesen. Anschließend wird das SQL-Skript mit dem CL-Befehl RUNSQLSTM ausgeführt.

```
RUNSQLSTM  SRCFILE(EPS002CR/QSQLSRC)  +
           SRCMBR(&SQLSTM) NAMING(*SQL)
```

2.13.2.1
Seite 4

Viele Informationen sind hierbei nicht notwendig; lediglich die verwendete Namenskonvention *SQL muss bekannt gegeben werden.

Nach der Skriptausführung erhält der Anwender erneut eine Meldung, die ihm mitteilt, dass die Skriptausführung beendet ist.

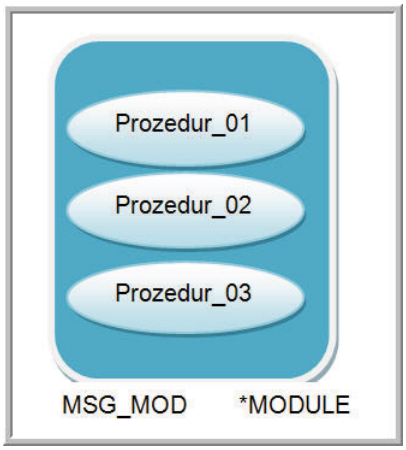
```
    SNDPGMMSG  MSGID(CPF9898) MSGF(QCPFMSG) +
       MSGDTA('Datentransfer beendet') +
       TOPGMQ(*EXT) MSGTYPE(*STATUS)
    DLYJOB DLY(2)
ENDSUBR
```

Das Programm ist einfach aufgebaut und unbekannte Funktionen sollten nicht vorhanden sein. Nun soll dieses OPM-Programm entsprechend neu strukturiert werden, indem wir ein ILE-Programmobjekt erstellen.

ILE-Module erstellen

Um unser Programm OPM_ORIGIN jetzt zu einem ILE-Programm „umzubauen“, müssen wir die Architektur des Programms ändern und zunächst einmal mehrere Module erstellen, die anschließend zum Programm verbunden werden.

ILE-Programme bestehen in der Regel aus verschiedenen Modulen. Bei einem Modul handelt es sich um ein Objekt der Objektart *MODULE. Solch ein Modul kann mehrere Prozeduren enthalten, die in CL, RPG oder Cobol geschrieben sind.



Standardaufbau eines Moduls

Der Begriff der Prozedur ist im CL-Umfeld nicht einfach zu verstehen, da es innerhalb eines CL-Moduls bislang nicht möglich ist, mehrere Prozeduren zu hinterlegen. Auch ist es nicht möglich, CL-Funktionen, also Prozeduren mit Rückgabewert zu definieren. Hier muss man abwarten, was die Zukunft bringt, da CL jetzt bereits Unterroutinen kennt, die Returnwerte liefern, wird es vielleicht irgendwann auch möglich sein, CL-Funktionen zu schreiben. Bis zum Release V7R1 ist diese Vision allerdings noch nicht wahr geworden.

Jedes CL-Modul enthält daher bislang immer genau eine Prozedur, wodurch der Begriff der Prozedur im CL-Umfeld ein wenig ad absurdum geführt wird. Dennoch bleiben wir bei den geläufigen Termini und sprechen einfach auch im CL-Umfeld von Prozedur – und nicht vom Modulaufruf.

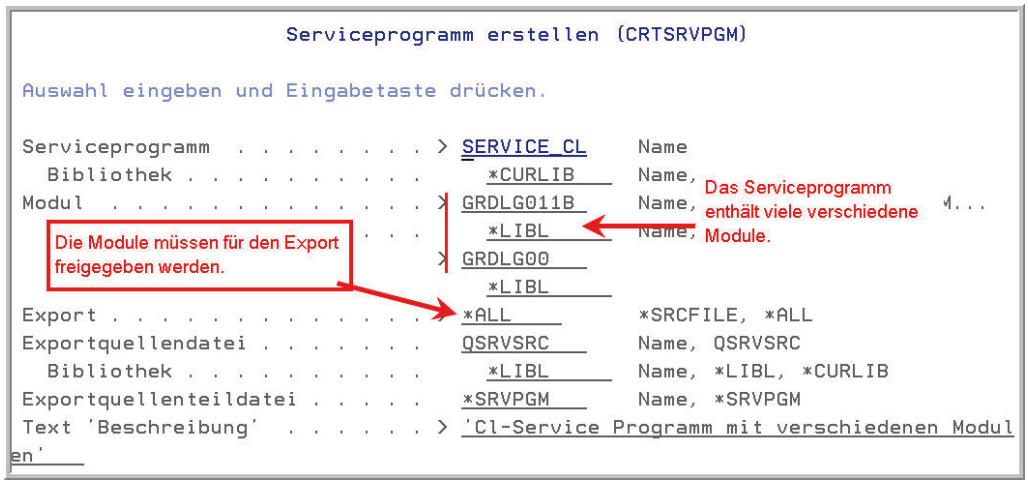
Wenn wir verstanden haben, was ein Modul ist, stellt sich die Frage: „Welche Module müssen wir erstellen?“ Um hier den entsprechenden Ansatz zu finden, sollten wir uns die Frage stellen: „Was kann das Programm OPM-ORIGIN?“

- Das Programm kann ein SQL-Skript ausführen.
- Das Programm kann die Existenz eines Objekts prüfen.
- Das Programm kann Statusmeldungen senden.

2.13.2.2
Seite 2

Wir könnten also drei eigenständige Module erstellen, die diese Funktionen übernehmen:

Das Modul oder besser gesagt die Prozedur MSG_STATUS übernimmt die Zustellung der Statusmeldung.



Die Prozedur MSG_STATUS

Dazu erhält das Modul MSG_STATUS den zu sendenden Text als Eingangsparameter übergeben, führt den SNDPGMMSG-Befehl aus und wird nach einer Sekunde Wartezeit beendet.

Die zweite Prozedur CHKOBJ prüft die Existenz eines beliebigen Objektes.

```
pgm parm(&LIB &OBJ &ART &Return)
    DCL &LIB *CHAR 10
    DCL &OBJ *CHAR 10
    DCL &art *CHAR 10
    DCL &RETURN  *uint   2

    chgvar &Return Value(1)
    CHKOBJ     OBJ(&LIB/&OBJ) OBJTYPE(&ART)
    MONMSG     MSGID(CPF9801) EXEC( CHGVAR &RETURN VALUE(0) )

endpgm
```

Die Prozedur CHKOBJ

Der qualifizierte Objektname und Objekttyp werden dafür an das Modul übergeben. Die Antwort wird anhand des Parameters &RETURN übermittelt.

Die dritte Prozedur übernimmt die Ausführung des SQL-Skripts und enthält den einzigen „neuen“ Befehl, den es im ILE-Umfeld gibt – den Befehl CALL-PRC.

```
PGM  PARM(&Sqlstm)

     DCL &sqlstm  *CHAR 10
     DCL &MSG *CHAR 50

          chgvar &MSG Value('SQL-Skript startet in 1 Sekunde!')
          CALLPRC    PRC(MSG_STATUS) PARM(&MSG)

          RUNSQLSTM  SRCFILE(EPS002CR/QSQLSRC) SRCMBR(&SQLSTM) +
                       NAMING(*SQL)
          chgvar &MSG Value('SQL-Skript endet in 1 Sekunde!')
          CALLPRC    PRC(MSG_STATUS) PARM(&MSG)


ENDPGM
```

Die Prozedur RUNSQLSTM

Wir können bereits erahnen, dass die erstellten Module – wie externe Programme – aufgerufen werden müssen und dabei die Parameterübergaben zu realisieren sind. Der Aufruf von Prozeduren wurde in CL bereits vor langer Zeit über den Befehl CALLPRC (Call Procedure) realisiert.

```
                    Binderprozedur aufrufen (CALLPRC)

Auswahl eingeben und Eingabetaste drücken.

Prozedur . . . . . .   MSG_STATUS


Parameter:
  Parameter . . . . . .   &MSG
  Wurde übergeben . . . .   *BYREF          *BYREF, *BYVAL
    + für weitere Werte
CL-Var f. zurückgegebenen Wert   *NONE
```

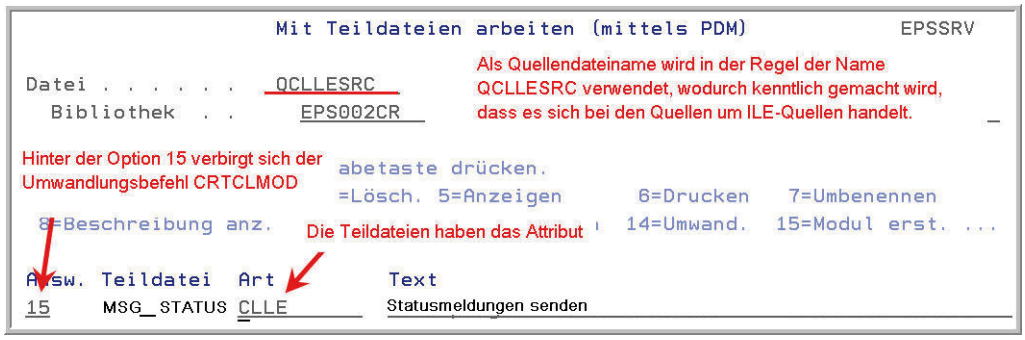
Bedienerführung des Befehls CALLPRC

Der Parameter „Prozedur (PRC)“ enthält den Namen der aufzurufenden Prozedur und der Parameter PARM die zu übergebenden Parameterwerte, wie wir es vom CALL-Befehl bereits kennen. Vor Release V5R3 konnten Parameter im CL-Modul nur „By Reference(*BYREF)“ – also als Speicheradresswert – übergeben werden. Seit V5R3 ist auch die Übergabe „By Value(*BYVAL)“ möglich. Der Unterschied hierbei besteht in der Rückgabe. Wenn ein Parameter „By Reference“ übergeben wird und dieser Parameter anschließend durch die aufgerufene Prozedur verändert wird, ist diese Änderung anschließend auch in der aufrufenden Prozedur sichtbar. Schließlich wird die Speicheradresse des Parameters und nicht eine Kopie der aufgerufenen Prozedur zur Verfügung gestellt. Anders verhält es sich, wenn der Parameter „By Value“ übergeben wird. In diesem Fall arbeitet die aufgerufene Prozedur mit einer Kopie und eventuelle Änderungen des Parameters wirken sich nicht in der aufrufenden Prozedur aus. Sieht man einmal von den Klauseln *BYREF und *BYVAL ab, ähnelt der Befehl CALLPRC dem CL-Befehl CALL und benötigt keine weiteren Erläuterungen.

2.13.2.2
Seite 4

Weitere ILE-Befehle wurden im CL-Umfeld nicht etabliert, so dass der Quellcode ansonsten keine Überraschungen enthält.

Nun kommen wir zur technischen Umsetzung. Um CL-Module zu erstellen, ergibt sich eine erste Änderung: Die ILE CL-Teildatei benötigt die Teildateiart CLLE, wenn aus der Teildatei ein Modul erstellt werden soll. Häufig ändert sich darum auch der Name der Quellendatei. Unsere „normalen" Teildateien haben wir in der Quellendatei QCLSRC gespeichert, für die ILE-Quellendateien nutzen wir meistens den Namen QCLLESRC. Die Kompilierung der Teildateien zum Modul erfolgt durch den jeweiligen Compiler und wird über den Befehl CRTxxxMOD ausgelöst, wobei xxx für die verwendete Sprache steht. Eine Teildatei, die CL-Anweisungen enthält, muss also mit dem Befehl CRTCLMOD (Create CL Modul) zum Modul umgewandelt werden. Der Befehl ähnelt in seinen Parametern dem Befehl CRTCLPGM und muss daher auch nicht weiter erläutert werden.



Die Modulerstellung – CRTCLMOD

In der Abbildung können Sie erkennen, dass PDM alternativ zum Befehl CRTCLMOD die Option 15 zur Verfügung stellt, um ILE-Module umzuwandeln. Wie bei der Programmumwandlung erhalten Sie auch hier anschließend ein Umwandlungsprotokoll, das Ihnen mitteilt, ob das Modul erstellt werden konnte oder ob evtl. Fehler zu korrigieren sind.

Ist das Modul erst einmal erstellt, stehen viele CL-Befehle für die Arbeit mit Modulen zur Verfügung:

CL-Befehl	Bedeutung
CRTxxxMOD	Modul erstellen
DLTMOD	Modul löschen
DSPMOD	Modulinformationen anzeigen
CHGMOD	Module ändern
WRKMOD	Mit Modulen arbeiten

Ein Kennzeichen aller Module ist es, dass sie nicht direkt aufgerufen werden können. Das in der oberen Abbildung erstellte Modul MSG_STATUS können Sie also nicht mit dem Befehl CALL aufrufen, sondern es muss als nächstes in

ein ILE-Programm eingebunden werden. Zuvor müssen wir aber auch noch die Module CHKOBJ und RUNSQLSTM erstellen:

```
CRTCLMOD MODULE(CHKOBJ) SRCFILE(QCLLESRC)
CRTCLMOD MODULE(RUNSQLSTM) SRCFILE(QCLLESRC)
```

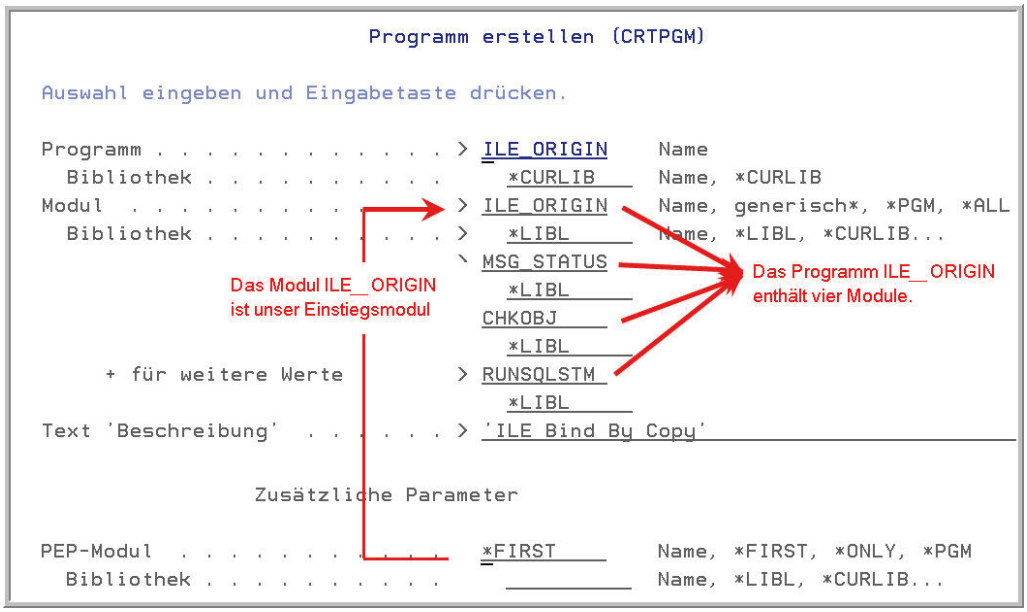
2.13.2.2

Seite 6

ILE-Programmobjekte erstellen – Methode BIND BY COPY

Unsere Module werden jetzt in einem zweiten Schritt zu einem ILE-Programm verbunden. Ein ILE-Programm kann dabei aus bis zu 300 Modulen bestehen, die zudem in unterschiedlichen Sprachen geschrieben sein können.

Diese ILE-Programmobjekte werden mit dem Befehl CRTPGM (Create Program) erzeugt. Ein solches Programm kann später wieder direkt mit der Anweisung CALL ausgeführt werden.



Der CL-Befehl CRTPGM

In der Abbildung können Sie sehen, dass die Module ILE_ORIGIN, MSG_STATUS, CHKOBJ und RUNSQLSTM zum Programm verbunden werden. Das ILE-Programm selbst erhält den Namen ILE_ORIGIN. Der gezeigte CRTPGM-Befehl wird im Übrigen vom PDM nicht über eine entsprechende Option unterstützt. Mit PDM kann man mit Hilfe der Option 15 zwar Module erstellen, aber eine Option für die anschließende Programmerstellung fehlt.

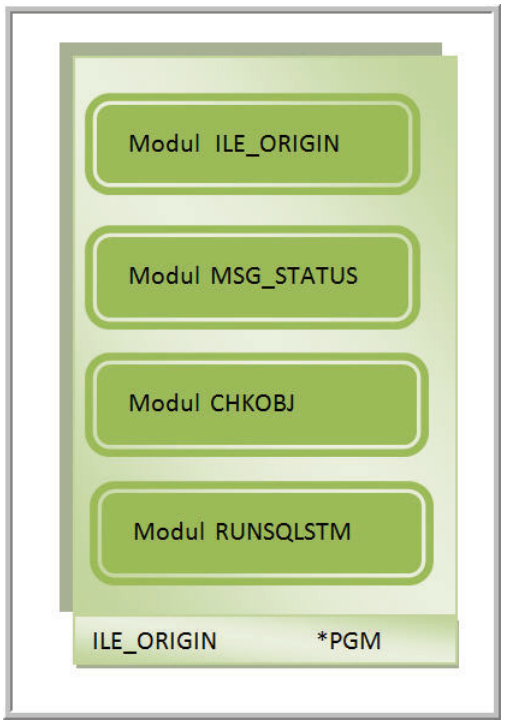
Unterstützung wird es in diesem Bereich mit Sicherheit auch niemals geben. IBM hat in diesem Zusammenhang bereits früh auf die Möglichkeiten des Remote System Explorer (RSE) verwiesen. Will man also in der gewohnten PDM-Umgebung ILE-Objekte erstellen, bleibt nur die Möglichkeit, die entsprechenden CL-Befehle interaktiv auf einer Befehlszeile auszuführen oder die Befehle in einem CL-Programm zu hinterlegen. Nur, warum sollte man derartige Anstrengungen für eine Entwicklungsumgebung unternehmen, die seitens IBM nicht mehr gewollt ist?

Kommen wir noch einmal auf den Erstellungsbefehl unseres Programms zurück. Im CRTPGM-Befehl wird das Modul ILE_ORIGIN als Einstiegsmo-

2.13.2.3
Seite 2

dul definiert. Jedes ILE-Programm benötigt ein solches PEP-Modul (Program Entry Point), also ein Modul, dem beim Aufruf des Programms die Kontrolle übertragen wird. Dieser Einstiegspunkt wird in dem Parameter „PEP-Modul (ENTMOD)“ des Befehls CRTPGM spezifiziert. Als Standardwert wird der Wert *First verwendet. Das erste Modul in der Modulliste wird damit zum Entry Point des ILE-Programms. Soll ein anderes Modul den Einstiegspunkt bilden, müssen Sie den Parameter ENTMOD entsprechend abändern. Auch hier stellt sich die Frage, warum sollten Sie das tun?

Wenn wir den in der Abbildung gezeigten Dialog bestätigen, erhalten wir das Programmobjekt ILE_ORIGIN mit folgendem Aufbau:



Das fertige Programmobjekt – BIND BY COPY

Auch für die Arbeit mit ILE-Programmen stehen natürlich anschließend entsprechende CL-Befehle zur Verfügung:

CL-Befehl	Bedeutung
CRTPGM	Programm erstellen
DLTPGM	Programm löschen
DSPPGM	Programminformationen anzeigen
CHGPGM	Programmattribute ändern
WRKPGM	Mit Programmen arbeiten
UPDPGM	Programmmodule modifizieren

Einige Leser, die bereits eigene ILE-Programme erstellt haben, werden sich jetzt vielleicht fragen, warum ich nicht einfach die Option 14 im PDM bzw.

den Befehl CRTBNDCL zur Programmerstellung verwende? Tatsächlich läuft auch bei der Kompilierung über CRTBNDCL ein zweistufiger Prozess ab, wenn die Teildatei die Art CLLE aufweist. Zuerst wird das Modul in der Bibliothek QTEMP erstellt. Dann wird aus dem Modul ein Programm generiert und das Modul anschließend in QTEMP gelöscht. Bei dieser Umwandlung geht aber das Modul verloren; es kann kein zweites Mal in einem anderen Programm verwendet werden. Außerdem wird auf diesem Weg aus jeder Teildatei wieder genau ein Programmobjekt. Die Verwendung mehrerer Module, die in unterschiedlichen Sprachen geschrieben wurden, ist nicht möglich. Machen Sie es sich lieber zur Regel, zunächst den Befehl CRTCLMOD zu verwenden, um ein Modul zu erstellen, und binden Sie dieses Modul anschließend in ein ILE-Programm ein. Der Befehl CRTBNDPGM (Create Binding CL) bzw. die Option 14 steht nur zur Verfügung, damit ohne viel Aufwand bestehende OPM-Programme schnell in ILE-Programme umgesetzt werden können. Wenn Sie jedoch auch weiterhin aus einer Teildatei ein Programmobjekt erstellen wollen, stellt sich die Frage, warum wechseln Sie zum ILE-Konzept?

Fassen wir kurz zusammen: Um ILE-Programme zu erstellen, ist immer ein zweistufiger Prozess erforderlich, denn bevor ein Programm erstellt werden kann, müssen zunächst einmal alle Module erstellt worden sein. Nur bei dem ersten Schritt – der Modulerstellung – spricht man im ILE-Umfeld von Kompilierung. Im zweiten Schritt – dem Binden – werden alle verwendeten Module über den CL-Befehl CRTPGM (=Programm erstellen) zu einem Programm verbunden. Diese Methode nennt man „Bind by Copy“, da alle Module, die in einem ILE-Programm verwendet werden, hierbei vollständig in das Programmobjekt eingebunden oder besser gesagt hineinkopiert werden. Wird das Modul mehrfach an unterschiedlichen Stellen im Programm aufgerufen oder in unterschiedlichen Programmen verwendet, wird das Modul auch mehrfach kopiert. Die Einbindung über Kopie (Bind by Copy) hat somit zwei Nachteile:

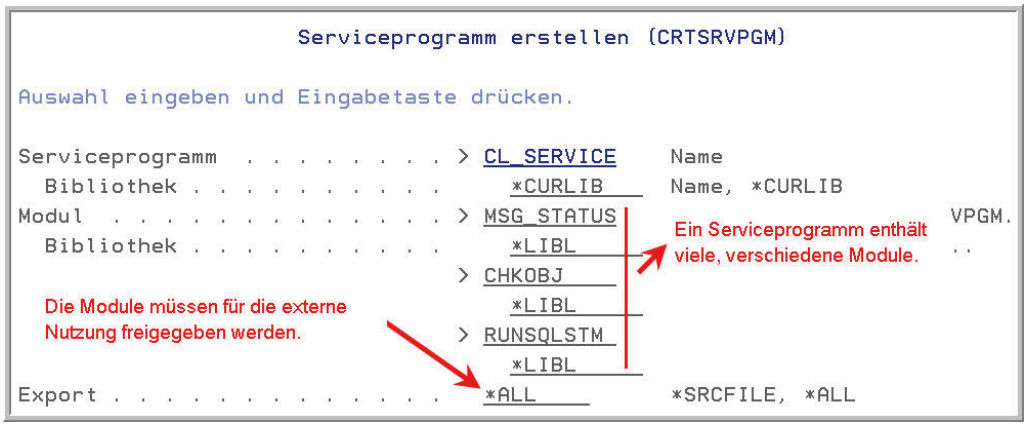
Zum einen werden die Programm-Objekte aufgebläht, zum anderen müssen Sie, sobald ein Modul geändert wird, alle Programme, die dieses Modul verwenden, neu erstellen. Sie müssen also immer wissen, welche Programme welche Module beinhalten.

2.13.2.3
Seite 4

ILE-Programmobjekte erstellen – BIND BY REFERENCE

Dieses Szenario klingt angesichts der Vielzahl an Programmobjekten nicht sehr verlockend. Gott sei Dank gibt es Alternativen in Form von Serviceprogrammen, die wir nutzen können. Serviceprogramme sind neue ILE-Objekte der Art *SRVPGM. Ein Serviceprogramm kann aus einem oder mehreren Modulen bestehen und stellt eine Sammlung von Modulen zur Verfügung. Ein Serviceprogramm besitzt jedoch keinen eigenen Einstiegspunkt und kann von daher auch nicht eigenständig aufgerufen werden. Serviceprogramme können wir uns als eine Art Baukasten vorstellen. ILE-Programme können bei Bedarf einzelne „Bauklötze“ eines Serviceprogramms nutzen, ohne dass das Modul oder die Prozedur im ILE-PGM selbst enthalten sein muss.

Erstellt werden Serviceprogramme über den Befehl CRTSRVPGM (Create Service Program):



*Der Befehl CRTSRVPGM – EXPORT *ALL*

Wie beim CRTPGM-Befehl müssen die verwendeten Module aufgelistet werden, damit sie dem Serviceprogramm hinzugefügt werden. Es versteht sich von selbst, dass Sie ein Modul nur in ein Serviceprogramm einbinden sollten. Auch dieses Mal werden die verschiedenen Module in das Serviceprogramm hineinkopiert, wobei jedes Modul von außen – also durch entsprechende ILE-Programmobjekte – später aufgerufen werden kann. Der Parameter EXPORT(*ALL) gibt die Module zur externen Nutzung frei. Jedes Modul, das im Serviceprogramm liegt, erhält grundsätzlich eine eindeutige Signatur innerhalb des Serviceprogramms. Anhand dieser Signatur kann das Modul später eindeutig identifiziert werden, d. h. aber auch, dass die aufrufenden ILE-Programme die Signaturen kennen müssen, um die Module zu nutzen.

2.13.2.4
Seite 2

Auch für die Arbeit mit Serviceprogrammen gibt es entsprechend neue CL-Befehle:

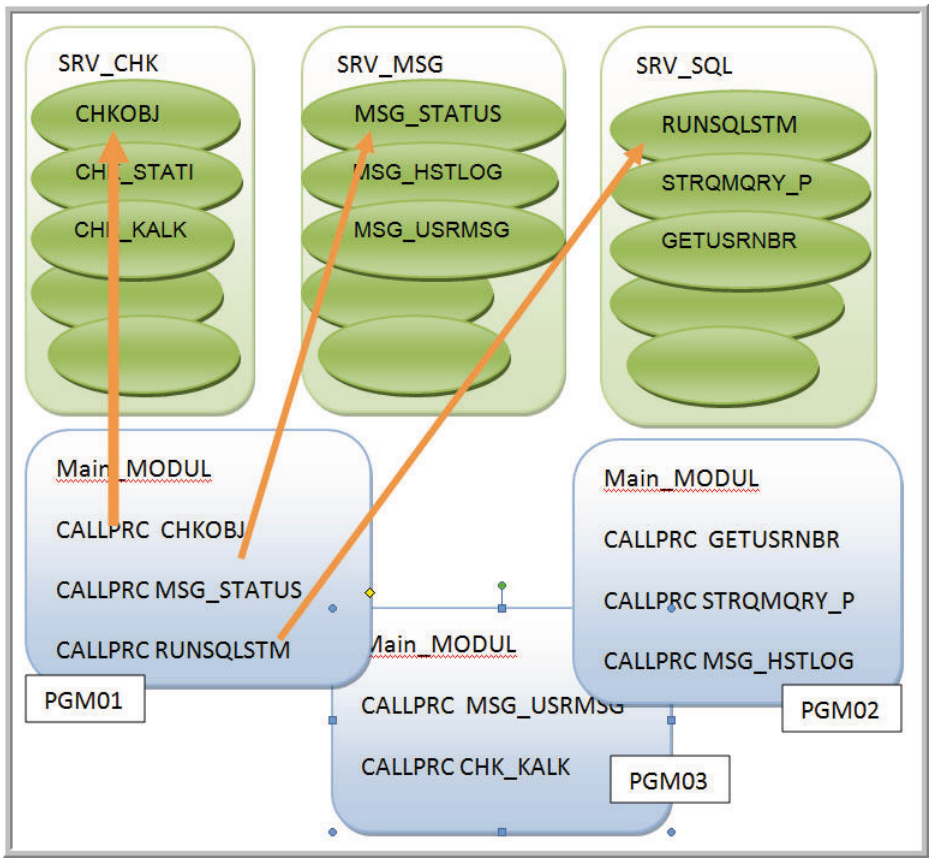
CL-Befehl	Bedeutung
CRTSRVPGM	Serviceprogramm erstellen
DLTSRVPGM	Serviceprogramm löschen
DSPSRVPGM	Serviceprogramminformationen anzeigen
CHGSRVPGM	Serviceprogrammattribute ändern
WRKSRVPGM	Mit Serviceprogrammen arbeiten
UPDSRVPGM	Serviceprogrammmodule modifizieren

Wie viele Serviceprogramme Sie erstellen und wie Sie die Serviceprogramme gliedern, ist allein Ihre Entscheidung. Häufig enthalten Serviceprogramme in der Praxis nur Module, die lediglich in einer Programmiersprache geschrieben wurden und wenn man Funktionalitäten aus anderen Programmiersprachen braucht, wird dafür halt ein weiteres Serviceprogramm erstellt. Hier wäre die verwendete Programmiersprache das Ordnungskriterium. Ob diese Strategie falsch oder richtig ist, können wir jetzt nicht diskutieren. Nur wäre nicht eine Aufteilung gemäß der Funktionalität – also Prüfroutinen, Datumsroutinen – übersichtlicher? Und vielleicht stammt der Ansatz, Serviceprogramme anhand der gewählten Programmiersprache aufzubauen, einfach nur aus „alten“ Zeiten?

Folge ich also dem Ordnungskriterium „Modulfunktionalität“, ergeben sich bei meinem Beispiel drei eigenständige Serviceprogramme:

- SRV_MSG: Das Nachrichten-Serviceprogramm
- SRV_SQL: Das SQL-Serviceprogramm
- SRV_CHK: Das Prüfroutinen-Serviceprogramm

Sicherlich ist der Aufwand für dieses kleine Beispiel zu hoch. Doch in der Realität hätte ich auch nicht nur drei Prozeduren! Wenn Sie sich ein wenig Zeit nehmen, fallen Ihnen sicherlich viele verschiedene Prüfalgorithmen oder SQL-Zugriffe ein, die sich genauso gut als Prozeduren abbilden lassen wie die gezeigten. Außerdem müssen Sie bedenken, dass die Möglichkeiten wachsen, wenn Sie nicht nur über CL-Prozeduren nachdenken. Nicht ohne Grund darf ein Serviceprogramm bis zu 300 Module enthalten und so könnte sich sehr schnell das folgende Szenario entwickeln:



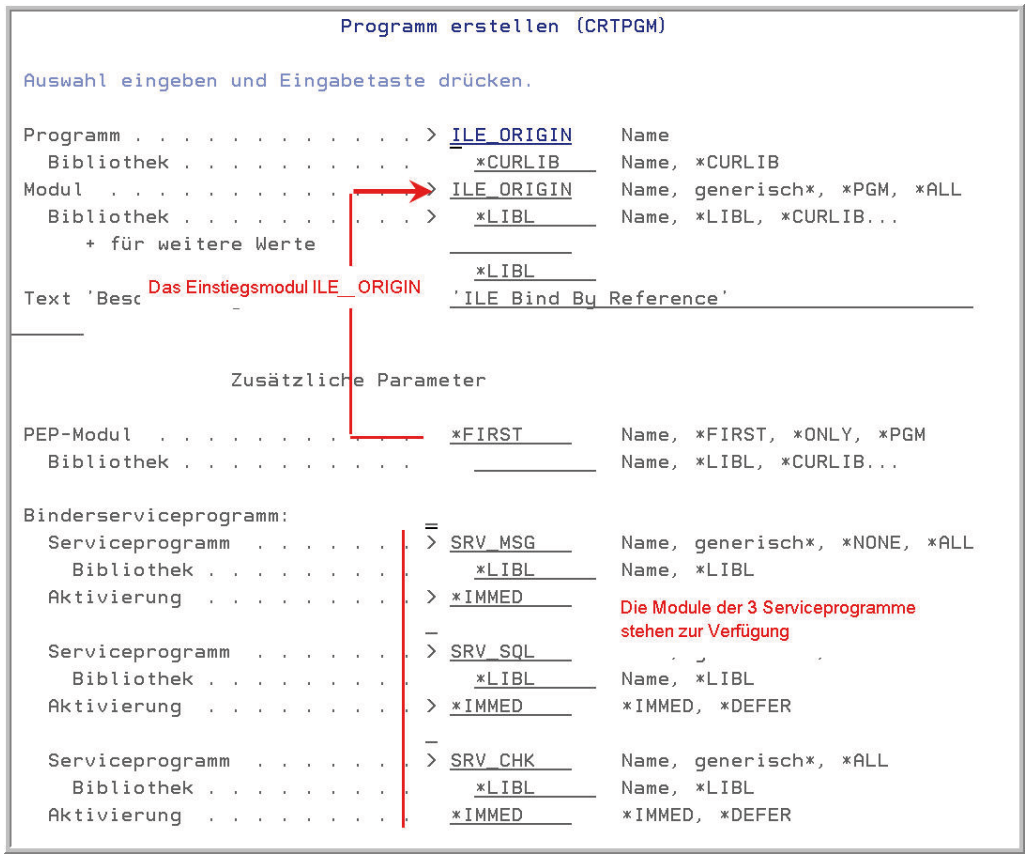
ILE-Architektur

Jedes meiner Serviceprogramme muss ich jetzt natürlich noch erstellen.

```
CRTSRVPGM  SRVPGM(SRV_MSG) MODULE(MSG_STATUS) EXPORT(*ALL)
CRTSRVPGM  SRVPGM(SRV_SQL) MODULE(RUNSQLSTM) EXPORT(*ALL)
CRTSRVPGM  SRVPGM(SRV_CHK) MODULE(CHKOBJ) EXPORT(*ALL)
```

Auch jetzt muss wieder – in einem zweiten Schritt – das ILE-Programmobjekt gebunden werden. Dafür geben wir jetzt allerdings im CRTPGM-Befehl anstatt der einzelnen Module die zu verwendenden Serviceprogramme an:

2.13.2.4
Seite 4



Der Befehl CRTPGM mit Serviceprogramm

Die Namen der Serviceprogramme werden im Parameter „Binderserviceprogramm (BNDSRVPGM)“ hinterlegt. Bei dieser Vorgehensweise wird nicht das Serviceprogramm selbst als Objekt in das Programm kopiert, sondern nur eine eindeutige Signatur der entsprechend im Programm verwendeten Prozeduren. Diese Methode der Moduleinbindung nennt man „Copy By Reference“, da erst zur Ausführungszeit das Serviceprogramm in den Speicher kopiert und die Signatur aufgelöst wird. Sie können zusätzlich über das Element „Aktivierung“ steuern, ob das Serviceprogramm bereits gemeinsam mit dem ILE-Programm aktiviert – also in den Arbeitsspeicher verlagert – werden soll oder ob die Aktivierung verzögert wird, bis tatsächlich ein Modul des Serviceprogramms genutzt wird.

Der Vorteil der Binding-Methode „Copy By Reference“ wird deutlich, wenn einzelne Module nachträglich modifiziert werden. Wird ein bestehendes Modul des Serviceprogramms geändert, ist es ausreichend, das entsprechende Modul zu bearbeiten und neu zu kompilieren. Das Serviceprogramm wird anschließend mit dem Befehl UPDSRVPGM entsprechend aktualisiert. Solange die Signatur unverändert bleibt, müssen die Programmobjekte, die das Modul aufrufen, nicht neu erstellt werden.

Was jedoch häufig unerwähnt bleibt: die Signatur eines Serviceprogramms kann sich aufgrund von Modifikationen durchaus ändern! Bei der Erstellung eines Serviceprogramms wird dessen Signatur über die Anzahl vorhandener Prozeduren, die Namen und die Reihenfolge der exportierten Prozeduren ermittelt. Das heißt, dass sich die Signatur des Serviceprogramms ändert, sobald eine exportierte Prozedur hinzugefügt oder gelöscht wird. Dabei spielt es keine Rolle, ob die Prozedur in einem bestehenden Modul hinzugefügt oder ob das Serviceprogramm um ein komplettes Modul erweitert oder verringert wird. In der Konsequenz bedeutet das, dass Sie, wenn das Serviceprogramm – wie in meinem Beispiel – mit dem Parameter EXPORT(*ALL) erstellt wurde, alle ILE-Programme, in denen das Serviceprogramm eingebunden wurde, neu binden müssen, wenn sich die Anzahl oder die Reihenfolge der Module innerhalb eines Serviceprogramms ändert, da in diesem Fall auch die eindeutigen Signaturen neu aufgebaut werden.

2.13.2.4

Seite 6

Exportquellendateien

Wir wissen jetzt, dass die Signatur der Serviceprogramme für den späteren Modulzugriff benötigt wird und dass sich die Signaturen innerhalb des Serviceprogramms aufgrund von Modifikationen durchaus ändern können! Um zu verhindern, dass wir unsere ILE-Programmobjekte neu binden müssen, nur weil wir die Module innerhalb unseres Serviceprogramms verändern, können wir anstelle des bisher verwendeten Parameters EXPORT(*ALL) bei der Serviceprogrammerstellung Exportquellendateien verwenden. Folgende Abbildung zeigt die für unser Beispiel notwendigen Exportteildateien.

```
Teildatei SRV_ SQL  Teildateiart: BND
STRPGMEXP  PGMLVL(*CURRENT) LVLCHK(*YES)
EXPORT     SYMBOL(RUNSQLSTM)
ENDPGMEXP

Teildatei SRV_ MSG  Teildateiart: BND
STRPGMEXP  PGMLVL(*CURRENT) LVLCHK(*YES)
EXPORT     SYMBOL(msg_status)
ENDPGMEXP

Teildatei SRV_ CHK  Teildateiart: BND
STRPGMEXP  PGMLVL(*CURRENT) LVLCHK(*YES)
EXPORT     SYMBOL(CHKOBJ)
ENDPGMEXP
```

Die Exportteildateien

Jede dieser Teildateien erhält den Namen des entsprechenden Serviceprogramms und ist in der Quellendatei QSRVSRC mit der Teildateiart BND gespeichert. Natürlich können Sie andere Namen wählen oder eine andere Quellendatei benutzen; nur erhöht sich dadurch später der Aufwand bei der Erstellung unserer Serviceprogramme aufgrund der Standardvorgaben im Befehl CRTSRVPGM.

Sie können in der oberen Abbildung sehen, dass für die Beschreibung der Exporte in den Teildateien eine spezielle Bindersprache verwendet wird. Diese Bindersprache umfasst nur drei Anweisungen, die nachfolgend kurz erläutert werden:

Die **Anweisung STRPGMEXP** kennzeichnet den Beginn einer Export-Liste und hat insgesamt drei Parameter:

PGMLVL(*CURRENT) bezeichnet die aktuelle Export-Liste. **PGMLVL (*PRV)** wird für alle vorhergehenden Versionen angegeben. Die aktuelle Export-Liste sollte an erster Stelle in der Binder-Quelle stehen. Sollten mehrere Export-Listen mit *CURRENT hinterlegt worden sein, wird trotzdem

2.13.2.4.1
Seite 2

nur der erste Eintrag als aktuelle Liste verarbeitet, die folgenden Einträge werden als „alte“ Varianten interpretiert.

Der Parameter **LVLCHK** bestimmt die Aktualitätsprüfung. Der Unterlassungswert ist *YES. Wenn hingegen *NO angegeben wird, dann wird die Signaturprüfung außer Kraft gesetzt, was zu unerwünschten Ergebnissen führen kann.

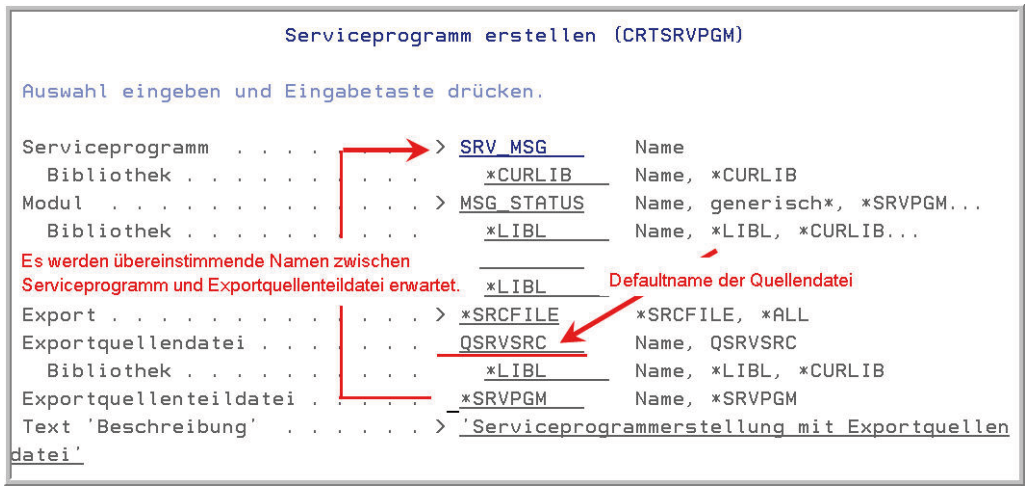
Der Parameter **SIGNATUR** bestimmt die Modulsignatur. Der Unterlassungswert für die Modulkennung ist *GEN, d. h., die Signatur wird vom System erstellt. Es besteht jedoch auch die Möglichkeit, eine eigene Signatur zu benennen. Die Signatur kann entweder als alphanumerischer oder hexadezimaler String angegeben werden.

Die **Anweisung EXPORT** kennt nur den Parameter SYMBOL. Er benennt die zu exportierenden Modulnamen. Werden die Modulnamen ohne Begrenzer aufgelistet, werden die Namen automatisch in Großbuchstaben umgesetzt. Case-sensitive Modulnamen müssen entweder durch einfache oder doppelte Hoch-Kommata begrenzt werden.

Die **Anweisung ENDPGMEXP** kennzeichnet schließlich das Ende der Exportliste und hat keine weiteren Parameter.

In einer solchen Teildatei können Sie neben den aktuell exportierten Prozeduren auch die Vorgängerversionen speichern. Über die verschiedenen Versionen können dann unterschiedliche Signaturen ermittelt und geprüft werden, wenn das ILE-Programm aufgerufen wird, so dass der Programmausführung nichts im Wege steht, wenn die ILE-Programmobjekte alte Signaturen enthalten.

Bevor es aber soweit ist, muss die Exportdatei natürlich erst einmal zur Erstellung des Serviceprogramms genutzt werden. Dafür muss der CRTSRVPGM-Befehl wieder ein wenig verändert werden:



Der Befehl CRTSRVPGM – Export Quellendatei

Der Parameter EXPORT(*SRCFILE) verweist auf die Teildatei SRV_MSG in der Quellendatei QSRVSRC, die die notwendigen Exportbestimmungen enthält.

Da wir für jedes Serviceprogramm eine eigene Teildatei benötigen, müssen wir anschließend auch die anderen Serviceprogramme unter Berücksichtigung dieser Exportteildateien neu erstellen.

```
CRTSRVPGM SRVPGM(SRV_CHK) MODULE(CHKOBJ)
```

```
CRTSRVPGM SRVPGM(SRV_SQL) MODULE(RUNSQLSTM) BNDSRVPGM((SRV_MSG))
```

Sie sehen, dass ich mich in den Erstellungsbefehlen unserer Serviceprogramme nicht um die Exportteildateien kümmern muss. Der Befehl CRTSRVPGM ist mit seinen Standardwerten darauf eingestellt, eine gleichnamige Teildatei in der Quellendatei QSRVSRC zu nutzen.

Leider gibt es aber bislang keine Automatik zum Anlegen und Aktualisieren der Binderquellen. Wird also eine Prozedur zu einem Serviceprogramm hinzugefügt oder entfernt, müssen Sie daran denken, die Binderquelle zu pflegen.

Falls Sie aber bereits mit Serviceprogrammen arbeiten und dabei bisher auf die Binderquellen verzichtet haben, können Sie die Binderteildateien mit Hilfe des Befehls RTVBNDSRC jederzeit nachträglich abrufen.

```
                    Binderquelle abrufen (RTVBNDSRC)

Auswahl eingeben und Eingabetaste drücken.

Modul . . . . . . . . . . . . .   __________    Name, generisch*, *ALL
  Bibliothek . . . . . . . . . .     *LIBL      Name, *LIBL, *CURLIB...
     + für weitere Werte           __________
                                     *LIBL
Serviceprogramm . . . . . . . . > SRV_MSG       Name
  Bibliothek . . . . . . . . . .     *LIBL      Name, *LIBL, *CURLIB
Exportquellendatei . . . . . . . > Qsrvsrc _    Name, QSRVSRC
  Bibliothek . . . . . . . . . . >   EPS002CR   Name, *LIBL, *CURLIB
Exportquellenteildatei . . . . .   *DFT         Name, *DFT
Sätze ersetzen od. hinzufügen  .   *REPLACE     *ADD, *REPLACE
```

Der Befehl RTVBNDSRC

Mit dem Befehl RTVBNDSRC (Binderquelle abrufen) können die Exporte aus einer Gruppe von Modulen, einem Serviceprogramm oder aus beiden abgerufen und in eine angegebene Teildatei gestellt werden. Diese Teildatei kann später als Eingabe für den Parameter SRCMBR des Befehls CRTSRVPGM (Serviceprogramm erstellen) verwendet werden. Mit Hilfe des Befehls RTVBNDSRC kann so die Bindersprache auch nachträglich automatisch erstellt werden.

2.13.2.4.1
Seite 4

Binderverzeichnisse

Auf die ILE-Programmerstellung haben die Quellenexportdateien keinen Einfluss. Wir müssten daher auch weiterhin die Module und Serviceprogramme auflisten, die das Programmobjekt verwendet. Doch gerade diese Aufgabe ist nicht nur eine lästige, sondern auch fehleranfällige Angelegenheit. Denken Sie nur daran, wie schnell Module bei einer erneuten Serviceprogrammerstellung vergessen werden könnten? Hinzu kommt, dass Sie stets die entsprechenden Modul- und Serviceprogrammnamen kennen und schreiben müssen. Dieses Problem kann durch die Verwendung von Binderverzeichnissen umgangen werden.

Ein Binderverzeichnis ist ein Objekt der Art *BNDDIR. Binderverzeichnisse werden nur für die Programm- und Serviceprogrammerstellung benötigt und enthalten eine Auflistung von Modulen und Serviceprogrammen, die in die jeweiligen Programme eingebunden werden sollen. Folgende Tabelle zeigt eine Übersicht der verfügbaren CL-Befehle, um mit Binderverzeichnissen zu arbeiten:

CL-Befehl	Bedeutung
CRTBNDDIR	Binderverzeichnis erstellen
DLTBNDDIR	Binderverzeichnis löschen
DSPBNDDIR	Binderverzeichnis anzeigen
WRKBNDDIR	Mit Binderverzeichnissen arbeiten
ADDBNDDIRE	Binderverzeichniseintrag hinzufügen
RMVBNDDIRE	Binderverzeichniseintrag entfernen
WRKBNDDIRE	Mit Binderverzeichniseinträgen arbeiten

Wenn ein Binderverzeichnis mit entsprechenden Einträgen erstellt wird, müssen die genannten Module oder Serviceprogramme noch nicht existieren. Erst bei der Programmerstellung werden die Binderverzeichnisse durchsucht. Wurde ein Eintrag im Binderverzeichnis gefunden, wird geprüft, ob das entsprechende Objekt vorhanden ist. Die vorhandenen Objekte werden dann je nach Objektart eingebunden. Für Module wird die Methode „Bind by Copy“ verwendet, für Serviceprogramme die Methode „Bind by Reference“.

Über die Organisation von Binderverzeichnissen herrscht geteilte Meinung. Einige Programmierer raten dazu, möglichst wenige Binderverzeichnisse zu erstellen, die zudem nach Möglichkeit nur Serviceprogramme enthalten sollten. Gemäß dieser Strategie sollte kein Serviceprogramm oder Modul in verschiedenen Binderverzeichnissen stehen. Der Vorteil liegt auf der Hand: Es gibt nur wenige *BNDDIR-Objekte und vom Grundsatz her könnten Sie sogar bei der ILE-Programmerstellung alle Binderverzeichnisse dem Programm zuordnen. Es führt nämlich nicht zum Fehler, wenn die Binderverzeichnisse Einträge enthalten, die im Programm nicht benötigt werden. Bei

24. Ergänzung 9/2010

2.13.2.5
Seite 2

der Programmerstellung werden nur die Einträge geprüft, die im Programm benötigt werden.

Der Nachteil an der Methode ist vielleicht nicht sofort ersichtlich: Wenn in einem Binderverzeichnis sehr viele Einträge vorhanden sind, können Sie anhand der Binderverzeichnisse nicht mehr nachvollziehen, welche Module oder Serviceprogramme die jeweiligen ILE-Objekte tatsächlich benötigen.

Aus diesem Grund bevorzugen andere Programmierer für jedes ILE-Programm ein eigenes Binderverzeichnis, welches genau die Module und Serviceprogramme enthält, die im Programm benötigt werden. Gemäß dieser Philosophie ist es leicht, die Binderverzeichnisse den ILE-Programmen zuzuordnen, insbesondere dann, wenn die Binderverzeichnisse den Namen des Programms erhalten. Zusätzliche Binderverzeichnisse werden gemäß dieser Philosophie nur verwendet, wenn externe Quellen eingebunden werden, die bereits eigene Binderverzeichnisse bereitstellen.

Der Nachteil dieser Organisationsform ist darin zu sehen, dass Sie sehr viele Binderverzeichnisse erhalten und ggf. Serviceprogramme und Module in unterschiedlichen Binderverzeichnissen vorhanden sind. Den Überblick über die Einträge in den verschiedenen Binderverzeichnissen zu behalten, könnte eine gewisse Herausforderung darstellen. Probleme könnten auch dann auftreten, wenn Sie Ihrem ILE-Programm zwei Binderverzeichnisse zuordnen würden, die beide das gleiche Modul enthalten. Hier wären Namenskonflikte vorprogrammiert. Nur warum sollten Sie dies tun, wenn Sie Ihrer Philosophie treu bleiben und für jedes Programm ein eigenes Binderverzeichnis erstellen?

Wie immer gibt es nicht den einen richtigen Weg und Sie müssen wieder einmal selbst entscheiden, welcher Philosophie Sie folgen wollen.

Ich entscheide mich für ein Binderverzeichnis, das den Namen des Programmobjektes erhält und erstelle daher zunächst einmal das Binderverzeichnis ILE_ORIGIN mit folgendem Befehl:

```
                  Binderverzeichnis erstellen (CRTBNDDIR)

Auswahl eingeben und Eingabetaste drücken.

Binderverzeichnis  . . . . . > ILE_ORIGIN    Name
  Bibliothek . . . . . .         *CURLIB     Name, *CURLIB
Berechtigung . . . . . .       *LIBCRTAUT    Name, *LIBCRTAUT, *CHANGE...
Text 'Beschreibung'  . . . . . > 'Binderverzeichnis für PGM ILE_ORIGIN'
```

Das Binderverzeichnis ILE_ORIGIN erstellen

Viele Parameter sind nicht notwendig, um ein solches Verzeichnis zu erstellen. Sie geben dem Verzeichnis lediglich einen Namen und bestimmen die öffentliche Berechtigung. Anschließend müssen wir dem Binderverzeichnis

die Namen unserer Serviceprogramme hinzufügen, die das Programm ILE_ORIGIN benötigt:

```
                Binderverz.eintrag hinzufügen (ADDBNDDIRE)

Auswahl eingeben und Eingabetaste drücken.

Binderverzeichnis  . . . . . > ILE_ORIGIN    Name
  Bibliothek . . . . . .         *LIBL       Name, *LIBL, *CURLIB...
Objektspezifikationen:           _
  Objekt . . . . . . . . . . > SRV_SQL       Name, generisch*, *ALL
    Bibliothek . . . . . .       *LIBL       Name, *LIBL
  Objektart  . . . . . .       *SRVPGM       *SRVPGM, *MODULE
  Aktivierung  . . . . . .     *IMMED        *IMMED, *DEFER
                               _
  Objekt . . . . . . . . . . > SRV_CHK       Name, generisch*, *ALL
    Bibliothek . . . . . .       *LIBL       Name, *LIBL
  Objektart  . . . . . .       *SRVPGM       *SRVPGM, *MODULE
  Aktivierung  . . . . . .     *IMMED        *IMMED, *DEFER
     + für weitere Werte       _
```

Der Befehl ADDBNDDIRE

Mit dem Befehl ADDBNDDIRE (Binderverzeichniseintrag hinzufügen) werden unserem Binderverzeichnis zwei Einträge hinzugefügt. Ob alles reibungslos geklappt hat, lässt sich anschließend einfach mit dem Befehl WRKBNDDIRE feststellen:

```
                Mit Binderverzeichniseinträgen arbeiten

Binderverzeichnis:   ILE_ORIGIN      Bibliothek:   EPS002CR

Auswahl eingeben und Eingabetaste drücken.
  1=Hinzufügen   4=Entfernen

Aus-                                                  ------Erstellung------
wahl  Objekt      Art      Bibliothek  Aktivierung    Datum       Zeit
 =    _________   _______  __________  ___________
 _    SRV_SQL     *SRVPGM  *LIBL       *DEFER         01.07.10    22:45:26
 _    SRV_CHK     *SRVPGM  *LIBL       *DEFER         01.07.10    22:43:55
```

Inhalt des Binderverzeichnisses ILE_ORIGIN

Beide Serviceprogramme sind hinterlegt, so dass im letzten Schritt noch das ILE-Programmobjekt erstellt werden muss.

2.13.2.5
Seite 4

```
                        Programm erstellen (CRTPGM)

Auswahl eingeben und Eingabetaste drücken.

Programm . . . . . > ILE_ORIGIN    Name
  Bibliothek . . . . . *CURLIB     Name, *CURLIB
Modul  . . . . . > ILE_ORIGIN    Name, generisch*, *PGM, *ALL
  Bibliothek . . . . . *LIBL       Name, *LIBL, *CURLIB...
     + für weitere Werte
                                   *LIBL
Text 'Beschreibung'  . . . . . > 'ILE_Pgrogrammobjekt LBW ITP'
_____

                Zusätzliche Parameter

Binderverzeichnis  . . . . . > ILE_ORIGIN    Name, *NONE
  Bibliothek . . . . . *LIBL       Name, *LIBL, *CURLIB...
```

ILE-Programm mit Binderverzeichnis erstellen

Dieses Mal können wir auf die Angabe der Serviceprogramme verzichten und verwenden stattdessen den Parameter „Binderverzeichnis (BNDDIR)“, der den Namen des zuvor erstellten Binderverzeichnisses enthält. Auch jetzt nutzen wir wieder das Modul ILE_ORIGIN als Einstiegsmodul für unser Programm.

Das Einstiegsmodul ILE_ORIGIN

Das Modul ILE_ORIGIN wurde bislang in jedem Erstellungsbefehl verwendet, nur haben wir uns nicht die darin enthaltenen Programmanweisungen angesehen. Das wollen wir jetzt nachholen, denn dieses Modul zeigt die eigentlichen Vorteile der ILE-Programmierung:

```
PGM  PARM(&SQLSTM)

DCL &SQLSTM *CHAR 10
DCL &J_N *UINT 2

CALLPRC PRC('CHKOBJ')  +
        PARM(('QTEMP     ') (&SQLSTM ) ('*FILE     ' ) (&J_N))

IF COND(&J_N = 1) THEN(DLTF  FILE(QTEMP/&SQLSTM))

CALLPRC    PRC('RUNSQLSTM') PARM((&SQLSTM))

 ENDPGM
```

Erinnern Sie sich noch an das ursprüngliche OPM-Programm? Übrig geblieben sind drei Anweisungen:

- Der Aufruf der Prüfroutine
- Die Prüfung, ob die Datei gelöscht werden muss
- Und der Aufruf zur Ausführung des SQL-Skriptes

Und hier zeigt sich der Vorteil der neuen Architektur. Sind die Bausteine erst einmal vorhanden, ist die spätere Codierung sehr viel schneller zu erledigen. Kein Programmierer muss mehr darüber nachdenken, wie er Statusmeldungen versendet oder wie er die Existenz eines Objektes überprüfen soll. Und dabei ist es sogar egal, ob ein RPG-Programm, CL-Programm oder Cobol-Programm diese Funktionen nutzen möchte. Sie sind einfach da und stehen jedem Programm und jedem Programmierer fortan zur Verfügung. Der Anfangsaufwand mag hoch sein, doch der spätere Nutzen ist größer!

Sie werden sicherlich verstehen, dass dieses Kapitel keinen Raum für philosophische Betrachtungen der unterschiedlichen Programmarchitekturen zulässt. Eine detaillierte Darstellung der ILE-Architektur mit all seinen Facetten und Einsatzmöglichkeiten würde ebenfalls den Rahmen dieses Buches sprengen und daher muss ich an dieser Stelle auf ein entsprechendes Handbuch der IBM verweisen, wenn Sie mehr über die ILE-Architektur, deren Vor- und Nachteile erfahren wollen:

http://publib.boulder.ibm.com/infocenter/iseries/v5r3/topic/books/sc415606.pdf

2.13.2.6
Seite 2

Ich konnte mich im Rahmen dieses Kapitels nur auf die wesentlichen Begriffe beschränken und wollte Ihnen anhand eines einfachen Beispielprogramms die Architekturunterschiede zwischen einem OPM-Programm und einem ILE-Programm verdeutlichen und dies sollte gelungen sein.

Command-Erstellung

Als Programmierer kennen Sie sicherlich viele i5/OS-Befehle. Diese Befehle stellen Ihnen eine konsistente und einfach zu handhabende Schnittstelle zur Verfügung. Befehle (Commands) sind Objekte der Art *CMD, die in Bibliotheken gespeichert sind. Sie können – wie jedes andere Objekt – gespeichert, verändert oder gelöscht werden. Commands können Sie direkt auf der Befehlszeile eingeben und veranlassen so das System, bestimmte Aktivitäten auszuführen. Doch keiner der IBM-Befehle ruft Ihre eigenen Anwendungsprogramme auf und manchmal gibt es Situationen, in denen man sich nicht vorhandene i5/OS-Betriebssystembefehle wünscht. Doch muss es nicht beim Wunsch bleiben, wir können mit wenig Aufwand eigene Systembefehle erstellen, die entweder die Parameterübergabe an die Anwendungsprogramme steuern oder Betriebssystemfunktionen ergänzen. Der Einsatz solcher Commands hat dabei viele Vorteile:

- Commands generieren eigene Eingabebildschirme, so dass Sie sich nicht mehr mit dem Erstellen von Bildschirmmasken aufhalten müssen. Stellen Sie sich vor, dass regelmäßig Datentransfers erforderlich sind. Dafür soll zunächst ein Query aufgerufen werden, wobei die Sortier- und Selektionskriterien parametrisiert werden. Die Parameterübergabe an das Query kann über einen 5250-Eingabebildschirm erfolgen oder mit Hilfe selbst erstellter Commands. Auf die Erstellung und Handhabung der 5250-Bildschirme könnte daraufhin im CL-Programm verzichtet werden.
- Mit Hilfe von Commands schaffen Sie zudem eine konsistente Benutzeroberfläche und erlauben die Parametereingaben in beliebiger Reihenfolge.
- Oder denken Sie nur an einige Objekte, beispielsweise an Datenwarteschlangen. Für Datenwarteschlangen fehlen ganz einfach die entsprechenden CL-Befehle. Stattdessen müssen APIs verwendet werden und der Aufruf der APIs ist fehleranfällig und nicht unbedingt jedermanns Sache. Die API-Aufrufe wären viel einfacher, wenn wir CL-Befehle zur Verfügung hätten, die die Parametereingabe realisieren. Commands erleichtern zudem die Parametereingabe, weil sie die Parameterwerte als Zeichenketten übernehmen; auch wenn das Programm andere Datentypen erwartet und notfalls können auch Standardvorgaben übernommen werden, falls einer der Parameter nicht eingegeben wird.
- Sie könnten sogar Laufzeitumgebungen für Ihre Commands bestimmen und so dafür sorgen, dass das Command beispielsweise nur innerhalb eines Batch-Programms ausgeführt werden darf.

2.14
Seite 2

- Hinzu kommt, dass in den CL-Programmen teilweise umfangreiche Gültigkeitsprüfungen realisiert werden müssen; diese Prüfungen könnten Commands übernehmen; die Kodierung im CL-Programm würde entfallen.

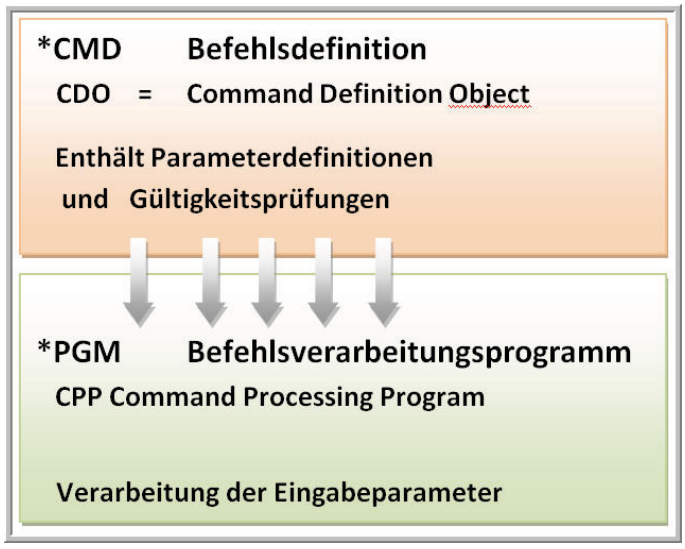
Viele Vorteile! Grund genug also, sich ein wenig detaillierter mit der Command-Erstellung und deren Möglichkeiten zu beschäftigen.

Die Befehlskomponenten

Commands werden wie Programme entworfen, kodiert, umgewandelt und getestet. Nur haben Commands andere Aufgaben als Programme. Programme sind die „Arbeiter“ auf Ihrem Server; sie erledigen die eigentlichen Aufgaben, während Commands Anweisungen an das System sind, eine bestimmte Aufgabe durchzuführen. Ein Command besteht daher immer aus mindestens zwei Objekten:

Befehlsdefinition (CDO)
In der Befehlsdefinition werden die Parameter definiert und Regeln für die Parametereingabe, wie Datentyp und Länge der Parameter, bekannt gegeben. Aber auch die gültigen Eingabewerte und Beziehungen zwischen den Parametern werden in der Befehlsdefinition hinterlegt. Die Befehlsdefinition ist das eigentliche Befehlsobjekt und wird als Objekt der Art *CMD gespeichert.

Befehlsverarbeitungsprogramm (CPP)
Das Verarbeitungsprogramm, welches mit der Befehlsdefinition verbunden ist, führt die angeforderten Aktionen aus. Dieses Programm erhält die in der Befehlsdefinition festgelegten Parameter und kann in jeder verfügbaren Programmiersprache geschrieben sein. Die meisten CPPs sind zwar in CL geschrieben, aber nichts spricht dagegen, beispielsweise ein RPG-Programm einzusetzen.



Aufbau eines Command

Darüber hinaus können Commands zusätzliche Objekte enthalten:

Programme zur Gültigkeitsprüfung (VCP)
Gültigkeitsprüfprogramme sind grundsätzlich optionale Komponenten eines Commands. Das Gültigkeitsprüfprogramm kann als eine Art Vorstufe des Befehlsverarbeitungsprogramms verstanden werden, da es unmittelbar vor dem eigentlichen Befehlsverarbeitungsprogramm mit identischen Parametern ausgeführt wird. Letztendlich soll das Gültigkeitsprüfprogramm die eingegebenen Parameterwerte auf Plausibilität prüfen; es übernimmt alle Prüfungen, die im Command selbst nicht durchgeführt werden können und „entlastet" dadurch das eigentliche Befehlsverarbeitungsprogramm.

Programm zur Überschreibung der Bedienerführung (POP)
Überschreibungsprogramme ermitteln Daten und stellen diese Daten anschließend als Standardwerte der Bedienerführungsanzeige des Commands zur Verfügung. Sie kennen viele Betriebssystembefehle, die mit POPs arbeiten. Denken Sie nur an die vielen CHG-Befehle.

Programme zur Auswahlsteuerung
Auswahlprogramme steuern die Anzeige gültiger Eingabewerte. Das Kapitel 2.14.7 zeigt Aufbau und Wirkung von Programmen, die die Anzeige der Bedienerführung steuern.

Nachrichtendateien
Meistens werden in Commands Textkonstanten angezeigt, wenn die Bedienerführungsanzeige (F4) aktiviert wird. Es können aber auch anstelle der konstanten Texte Nachrichtendateien für die Anzeige verwendet werden. Der Vorteil besteht darin, dass Sie die angezeigten Texte ändern können, ohne den Command neu erstellen zu müssen. Auch wenn Sie mehrsprachig arbeiten, bietet die Verwendung von Nachrichtendateien anstelle von konstanten Texten entsprechende Vorteile. Im Kapitel 2.14.2.1 und 2.14.6 werden Nachrichtendateien eingesetzt.

Grundparameter – Befehlserstellung

Sehen wir uns jetzt den Quellcode einer einfachen Befehlsdefinition etwas genauer an. Sie könnten Ihre Befehlsdefinitionen natürlich in der gleichen Quellendatei wie die CL-Programme speichern. Aus Gründen der Übersichtlichkeit empfiehlt es sich jedoch, eine eigene Quellendatei zu erstellen, die den Namen QCMDSRC erhält und ausschließlich CDO-Quellen – also Befehlsdefinitionen – enthalten sollte. Der folgende Befehl erstellt die Quellendatei QCMDSRC in der Bibliothek ITPCR:

```
CRTSRCPF FILE(ITPCR/QCMDSRC) +
TEXT('eigene Command-Definitionen Beispiele für ITP Fachbuch')
```

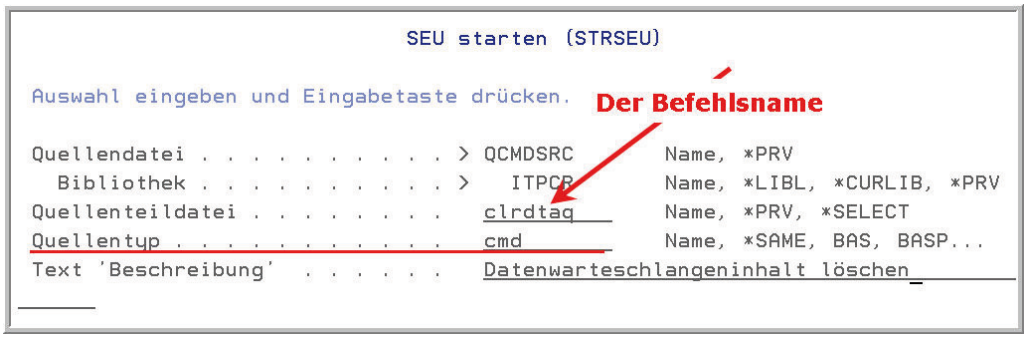
Nehmen wir als Ausgangszenario für unseren ersten Command das CL-Programm CLRDTAQ. Dieses CL-Programm ist unser CPP-Objekt.

```
PGM PARM(&DATAQ &LIB)

    DCL VAR(&DATAQ) TYPE(*CHAR) LEN(10)
    DCL VAR(&LIB) TYPE(*CHAR) LEN(10)

    CALL PGM(QCLRDTAQ) PARM(&DATAQ &LIB )
     MONMSG MSGID(CPF9801) EXEC(DO)
     SNDPGMMSG MSG(,Datenwarteschlange' *BCAT &LIB *TCAT ,/' *CAT
                 &DATAQ *BCAT + 'wurde nicht gefunden!')
     RETURN
     ENDDO

     MONMSG MSGID(CPF9810) EXEC(DO)
     SNDPGMMSG MSG('Bibliothek' *BCAT &LIB *BCAT 'wurde nicht
                 gefunden!')
     RETURN
     ENDDO

     SNDPGMMSG MSG('Datenwarteschlangeninhalt von ' *CAT
                 &DATAQ *BCAT 'in Bibliothek' + *BCAT &LIB
                 *BCAT 'wurde gelöscht!')

ENDPGM
```

2.14.2
Seite 2

Schauen Sie sich den oberen Quelltext an. Das CL-Programm birgt keine Geheimnisse: Als Parameter werden der Name einer Datenwarteschlange und der Name der Bibliothek, die das Objekt enthält, übergeben. Ohne dass weitere Objektprüfungen stattfinden, wird das API QCLRDTAQ aufgerufen, um den Inhalt dieser Datenwarteschlange zu löschen. Falls das Objekt nicht vorhanden sein sollte, werden die entsprechenden Fehlermeldungen im Programm abgefangen und der Anwender enthält entsprechende Fehlermeldungen. Wenn das API keine Fehler auslöst, wird eine positive Nachricht gesendet. Für dieses CL-Programm wollen wir jetzt eine Befehlsdefinition erstellen, damit der Programmaufruf und die Parameterübergabe vereinfacht werden.

Zunächst benötigen wir hierfür wieder einen geeigneten Editor, um den Quellcode für das CDO-Objekt zu erfassen. Es ist natürlich egal, ob Sie PDM oder RDI verwenden. Wichtig ist nur, dass Sie in Ihrer zuvor erzeugten Quellendatei QCMDSRC, ein Member der Teildateiart CMD, erzeugen:



Teildatei der Art CMD erstellen

Der Teildateiname ist eigentlich frei wählbar. Er wird aber in der Regel später als Befehlsname verwendet und sollte daher „sprechend“ sein. Sie sehen in der oberen Abbildung, dass ich den Namen des CL-Programms auch für die Befehlsdefinition verwende. Dies hat Vorteile, weil bei der Befehlserstellung standardmäßig versucht wird, eine Verknüpfung mit einem CL-Programm herzustellen, das den gleichen Namen trägt wie die Befehlsdefinition selbst. Wenn Sie nicht jedes Mal bei der Umwandlung des Commands die Standardeinstellungen überschreiben wollen, sollten Sie auf Namensgleichheit zwischen CDO- und CPP-Objekt achten.

Die Anweisung CMD

Wird für einen i5/OS-Befehl die Bedienerführung (F4) aufgerufen, so erscheint ein standardisierter Eingabebildschirm. Bestandteil des Bildschirms ist u. a. eine Titelzeile, die die Funktion des Commands genauer beschreibt.

Wirkungsweise der Anweisung CMD

Die Titelzeile bestimmen wir mit der Anweisung CMD.

```
CMD PROMPT('Löschen Datenwarteschlangeninhalt')
```

Die Anweisung CMD ist zwingend erforderlich. Das Schlüsselwort PROMPT beinhaltet den Text, der am oberen Bildschirmrand erscheint, wenn Sie die Bedienerführungsanzeige anfordern. Das Schlüsselwort PROMPT generiert somit eine Titelzeile. Als Titel kann entweder eine Textkonstante von maximal 30 Stellen angegeben werden oder alternativ ein Eintrag aus einer Nachrichtendatei verwendet werden, der aber ebenfalls nur 30 Zeichen lang sein sollte. Wenn wir anstelle der konstanten Texte Nachrichtendateien verwenden wollen, müssten wir die Anweisung CMD folgendermaßen ändern:

```
CMD PROMPT(CMD0002) PMTFILE(EPS002CR/CL_MSG *STATIC)
```

Die Überschrift wurde in diesem Beispiel in der Nachrichtendatei CL_MSG hinterlegt und soll statisch (*STATIC) eingebunden werden. Wenn der Befehl kompiliert wird, werden die Nachrichtentexte immer statisch im CDO-Objekt abgelegt.

Ob aber der statische Text tatsächlich zur Ausführungszeit des Commands verwendet wird, entscheidet der Parameterwert *STATIC. Wird der Wert *STATIC verwendet, muss auf die Nachrichtendatei zur Ausführungszeit des Commands nicht zugegriffen werden. Änderungen der Texte in den Nachrichtendateien sind daher bedeutungslos; es sei denn, sie kompilieren das

2.14.2.1
Seite 2

Objekt neu. Anders verhält es sich, wenn Sie den Wert *DYNAMIC verwenden. Bei der Befehlsausführung werden in diesem Fall die Nachrichtentexte anhand der Nachrichten-IDs dynamisch aus der Nachrichtendatei abgerufen. Sie können also jederzeit Ihre Nachrichtentexte ändern, ohne die Befehle neu erstellen zu müssen. Nur wenn zur Ausführungszeit des Commands nicht auf die Nachrichtendatei zugegriffen werden kann, wird der statisch im Objekt hinterlegte Text verwendet.

In früheren Releases hatte die Anweisung CMD keine weiteren Schlüsselwörter; heute enthält sie eine Reihe weiterer Optionen.

```
                        Befehlsdefinition (CMD)

 Auswahl eingeben und Eingabetaste drücken.

 Marke . . . . . .
 Bedienertext od. Nachricht.-ID   > CMD0002
 Nachrichtendatei f. Bed.text:
   Nachrichtendatei . . . . . > CL_MSG
     Bibliothek . . . . . .     *LIBL
   Nachrichtentext  . . . . . > *STATIC      *STATIC, *DYNAMIC
 Nachrichtendatei . . . . . .               Name
   Bibliothek . . . . . .       *LIBL       Name, *LIBL, *CURLIB
 Text 'Beschreibung'  . . . . .
 ______
 Max. Positionsparameter  . . . .             0-99, *NOMAX
 Ausführung wo zulässig . . . . .             *ALL, *BATCH, *INTERACT...
      + für weitere We
 Gültig bei Modus . . . . . .                 *ALL, *PROD, *DEBUG, *SERVICE
      + für weitere Werte
 Benutzer begrenzt zulassen . . .             *NO, *YES
```

Texte für die Überschriften können in Nachrichtendateien abgelegt werden.

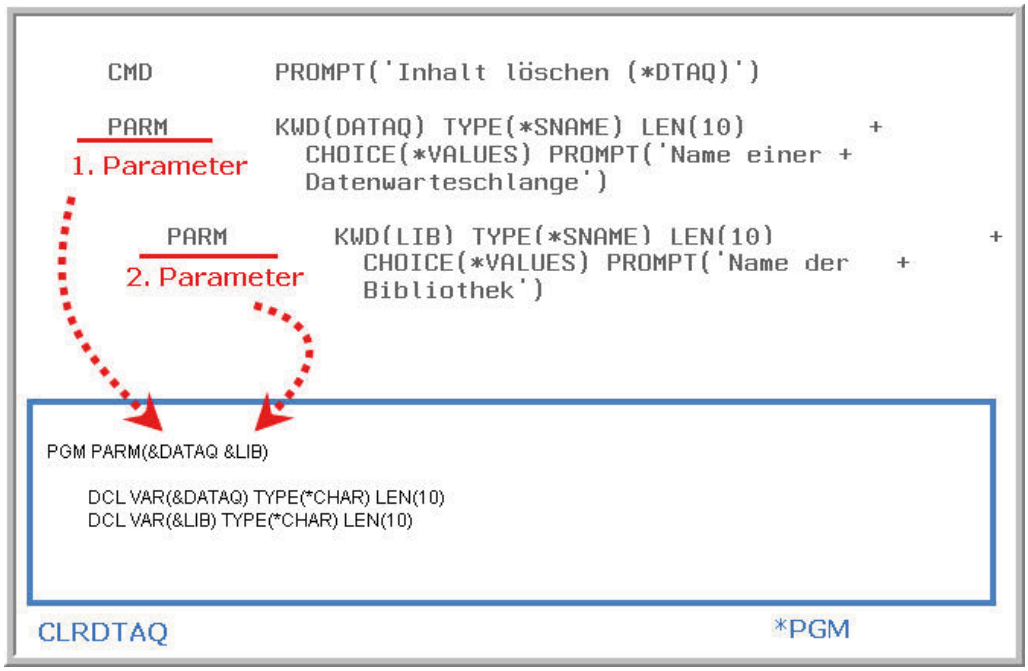
Umwandlungsparameter der Anweisung CMD

Die Anweisung CMD und die Umwandlungsparameter

Normalerweise hinterlegen Sie die Optionen für die Command-Erstellung, wenn Sie Ihr Command mit dem Befehl CRTCMD umwandeln. Doch schnell kann es passieren, dass bei der Modifikation der Objekte Einstellungen vergessen werden, da ist es sicherer, die Erstellungsoptionen im Quelltext zu definieren. Die in der Anweisung CMD hinterlegten Optionen stellen somit sicher, dass beim generierten Objekt die gewünschten Attribute verwendet werden. Für die Optionen der Anweisung CMD gibt es jedoch keine Standardwerte. Werden also in der Anweisung CMD keine Erstellungsoptionen angegeben, werden die Einstellungen des Erstellungsbefehls CRTCMD verwendet. Die Erläuterung der verschiedenen Klauseln finden Sie im Kapitel 2.14.3 im Zusammenhang mit dem CL-Befehl CRTCMD.

Die Anweisung PARM

Mit der Anweisung PARM müssen wir nun die Parameter beschreiben, die das CL-Programm empfangen soll. Für jeden Parameter, der vom CPP-Programm empfangen wird, benötigen wir in der Regel eine PARM-Anweisung. Unser Beispielprogramm CLRDTAQ erwartet den Namen der Datenwarteschlange und den Namen der Bibliothek. Wir müssen also zwei PARM-Anweisungen kodieren:



Die Parameterübergabe an das CPP

Die Klauseln der PARM-Anweisung sind sehr komplex und bieten eine Vielzahl an Möglichkeiten. In der oberen Abbildung verwende ich zunächst nur die Grundparameter KWD, TYPE, LEN, CHOICE und PROMPT der Anweisung PARM. Diese Grundparameter sollten Sie mindestens für die Parameterdefinition verwenden. Schauen wir uns die Grundparameter etwas genauer an:

Das Schlüsselwort KWD

KWD beschreibt das Schlüsselwort bzw. den Namen des Parameters. In unserem Beispiel heißt der Parameter DATAQ. KWD muss einen einfachen Namen enthalten, also eine Zeichenkette, die mit einem Buchstaben beginnt und nur Buchstaben, Zahlen und Unterstreichungszeichen enthält. Die maximale Länge beträgt 10 Stellen. Die Namen von Parametern sollten innerhalb eines Commands eindeutig und informativ sein. Versuchen Sie, verwirrende Benennungen zu vermeiden und einprägsame Namen zu verwenden. Einfache und schlüssige Namensgebungen wirken sich positiv auf die spätere Verwendung und Lesbarkeit der Commands aus.

2.14.2.2
Seite 2

Das Schlüsselwort TYPE
In CL-Programmen werden Variablen mit der Anweisung DCL deklariert. Dabei enthalten die Schlüsselwörter TYPE und LEN die Datenart und die Länge der Variablen. Diese Funktion wird auch benötigt, um Parameter für einen Command zu definieren. Der einzige Unterschied besteht darin, dass bei der Befehlsdefinition eine größere Auswahl an Datentypen zur Verfügung steht. Wir kennen aus der CL-Programmierung zwar Zeichenfelder, numerische Variablen und mittlerweile auch Binärdaten und Pointer. Doch der Datentyp *SNAME – wie ich ihn im Beispiel verwende – ist uns fremd.

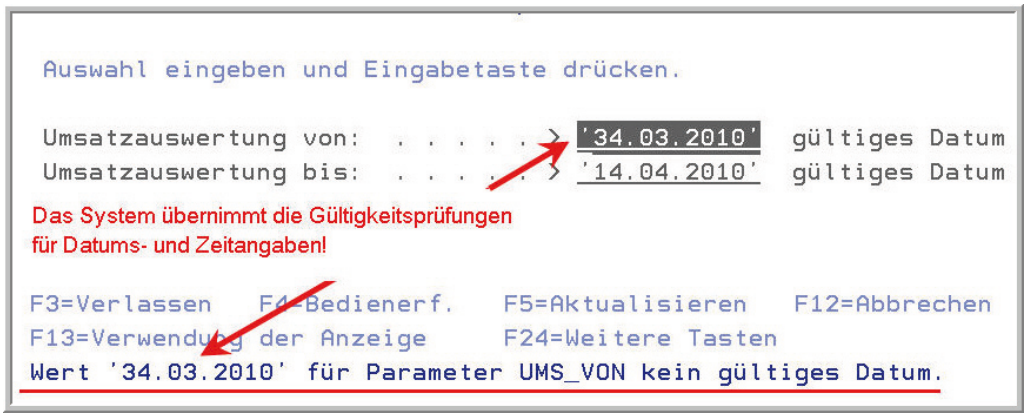
Der Datentyp *NAME, *SNAME, *CNAME
Diese Typen werden verwendet, um Objektnamen zu definieren und helfen, formale Eingabeprüfungen zu realisieren. Der Datentyp *SNAME im Beispielprogramm legt fest, dass nur einfache Namen für die Eingabe akzeptiert werden, die lediglich Buchstaben, Zahlen und Unterstriche enthalten dürfen. Da der Name Datenwarteschlange übergeben werden soll und Objektnamen nur diese Zeichen enthalten dürfen, kann TYPE(*SNAME) bereits als formale Eingabeprüfung angesehen werden. Unabhängig davon, ob Sie den Typen *NAME, *SNAME oder *CNAME verwenden, dürfen die Parameterwerte in Klammern oder einfache Anführungszeichen eingeschlossen werden und können bis zu 256 Byte lang sein. Die Unterscheidung der Typen *NAME, *SNAME und *CNAME liegt nur in der Verwendung zulässiger Sonderzeichen:

Typ	1.Zeichen	Restliche Zeichen
*NAME	Buchstabe, $, §, #	alphanumerische Zeichen, **Punkte, Unterstreichungszeichen,** $, § und # dürfen verwendet werden.
*SNAME	Buchstabe, $, §, #	alphanumerische Zeichen, Unterstreichungszeichen, $, § und # dürfen verwendet werden.
*CNAME	Buchstabe, $, §, #	alphanumerische Zeichen und die Sonderzeichen $, § und # dürfen verwendet werden.

Der Datentyp *PNAME
Der Typ *PNAME repräsentiert einen Pfadnamen.

Der Datentyp *GENERIC
Generische Namen kennen Sie aus vielen Systembefehlen. Der CL-Befehl WRKOBJ erlaubt beispielsweise generische Namen. Ein generischer Name enthält mindestens ein Zeichen, auf das ein Stern (*) folgt. Der Name kennzeichnet in diesem Fall eine Gruppe von Objekten, deren Namen mit den Zeichen vor dem * beginnen. Ist kein * vorhanden, geht das System davon aus, dass es sich bei dem generischen Namen um einen vollständigen Objektnamen handelt. Wenn ein Command allerdings nur ein Objekt verarbeitet – wie in meinem Beispiel, sollten Sie nicht den Typ *GENERIC verwenden, sondern lieber *SNAME oder auch *CNAME einsetzen.

Der Datentyp *DATE und *TIME
Die Typen *DATE und *TIME akzeptieren bei der Eingabe nur gültige Datums- und Zeitangaben. Die Gültigkeitsprüfung wird automatisch vom System durchgeführt. Wenn Sie beispielsweise einen Parameter mit *TIME definieren, wird die Eingabe 097830 abgewiesen. Die Befehlsdefinition wird einen Fehler melden und das CPP-Programm wird nicht ausgeführt werden, weil 09:78:30 keine gültige Zeitangabe ist. Ebenso verhält es sich mit Datumsfeldern wie folgende Grafik zeigt:



*Gültigkeitsprüfungen für *DATE und *TIME*

Der Wert 34.03.2010 ist als Datum ungültig, so dass dieser Wert nicht an das CL-Programm übergeben würde. Sie können also immer davon ausgehen, dass die formale Prüfung der Datums- und Zeitfelder stets das Command übernimmt.

Der Benutzer muss das Datum im Übrigen entsprechend des Jobformats angeben. Das Trennzeichen kann hierbei optional verwendet werden. Die Jahreszahl kann zwei- oder vierstellig angegeben werden. Bei einer zweistelligen Jahreszahl wird angenommen, dass das Datum zwischen dem 1. Januar 1940 und dem 31. Dezember 2039 liegt. Bei einer vierstelligen Jahreszahl liegt das Datum zwischen dem 24. August 1928 und dem 9. Mai 2071. Die Übergabe des Datums an das Befehlsverarbeitungsprogramm (CPP) erfolgt anschließend als 7-stellige Zeichenkette im Format CJJMMTT unabhängig davon, welches Jobformat vereinbart ist und unabhängig von der Benutzereingabe. Zeitwerte werden an das Verarbeitungsprogramm im Format hhmmss als 6-stellige Zeichenkette übergeben.

Der Datentyp *DEC, *LGL, *CHAR
Diese Datentypen kennen Sie aus der CL-Programmierung. Die mit *DEC definierten Parameter müssen in dezimale Variablen übernommen werden und dürfen nur Ziffern, die Vorzeichen Plus und Minus und ein Komma für die Nachkommadarstellung enthalten. Auch der Datentyp *LGL ist uns nicht fremd. Der Parameterwert wird als logischer Wert eins (‚1') oder Null (‚0') an das CPP-Programm übergeben. Ein *CHAR-Parameter kann jedes beliebige Zeichen enthalten. Der Parameterwert ist wie in der CL-Programmierung eine Zeichenfolge. Enthält die Zeichenfolge Sonderzeichen – mit Ausnahme des Sonderzeichens * –, muss sie in Hochkomma eingeschlossen werden.

Der Datentyp *HEX
*HEX legt hexadezimale Werte fest, wobei die Parameter die Zeichen 0 (Null) bis F enthalten können. Diese hexadezimalen Werte werden vom Befehlsverarbeitungsprogramm in EBCDIC-Zeichen umgewandelt, wobei jeweils zwei hexadezimale Zeichen ein EBCDIC-Zeichen ergeben. Der hexadezimale Wert „C1" würde als Buchstabe „A" vom CPP-Programm empfangen werden.

Der Datentyp *CMDSTR
Der Parameterwert ist eine Zeichenkette, die einen ausführbaren Befehl beinhaltet. Sie kennen diese Datenart von der CL-Programmierung: Die Anweisung SBMJOB kann in der Klausel CMD einen ausführbaren Befehl enthalten, d. h. zwei eigenständige Befehle werden verschachtelt. Der große Vorteil von *CMDSTR ist, dass die Bedienerführung für den verschachtelten Befehl aufgerufen werden kann. So wird die Bedienerführung in der Bedienerführung möglich. Außerdem erfolgt eine Gültigkeitsüberprüfung vor der Übergabe an das CPP. Sie sollten dabei beachten, dass bei dieser systemseitigen Gültigkeitsprüfung die Befehlsfolge neu erstellt wird und jedem Parameter Schlüsselwörter hinzugefügt werden, wenn die Parameter positionsgebunden angegeben waren. Die Reihenfolge der Parameter kann hierdurch geändert werden. Außerdem werden die Parameter in Hexadezimalschreibweise umgesetzt. Dadurch kann der neuerstellte Befehl erheblich länger als die ursprüng-

liche Befehlsfolge sein. Überschreitet die Länge des neu erstellten Befehls die zulässige Länge, die im Schlüsselwort LEN dieses Parameters angegeben ist, wird das Command abgebrochen.

Der Datentyp *ZEROELEM
Der Parameter wird immer als eine Liste mit Nullelementen betrachtet, für die kein Wert im Befehl angegeben werden kann. Es handelt sich hierbei also um eine Art DUMMY-Parameter. *ZEROELEM-Parameter sind nützlich, wenn beispielsweise zwei Befehle dasselbe Befehlsverarbeitungsprogramm verwenden. Enthält das Command A dabei weniger Parameter als das Command B, würde der Aufruf schiefgehen. In diesen Szenarien kodiert man einfach im Command A einen DUMMY-Parameter, um die Eingangsschnittstelle des CL-Programms zu bedienen.

Der Datentyp *INT2, *INT4, *UINT2, *UINT4
Der Parameterwert ist eine Ganzzahl, die als 2- oder 4-Byte-Binärzahl mit und ohne Vorzeichen weitergeleitet wird.

Der Datentyp *NULL
Der Parameterwert ist ein Nullzeiger. *NULL-Parameter dürfen Sie natürlich im Command nur verwenden, wenn Ihr Eingangsparameter im CL-Programm ein Pointer ist. Zu beachten ist, dass eine Anweisung DEP oder die Schlüsselwörter REL und RANGE anderer PARM-Anweisungen möglicherweise nicht auf den Wert eines mit *NULL definierten Parameters verweisen können.

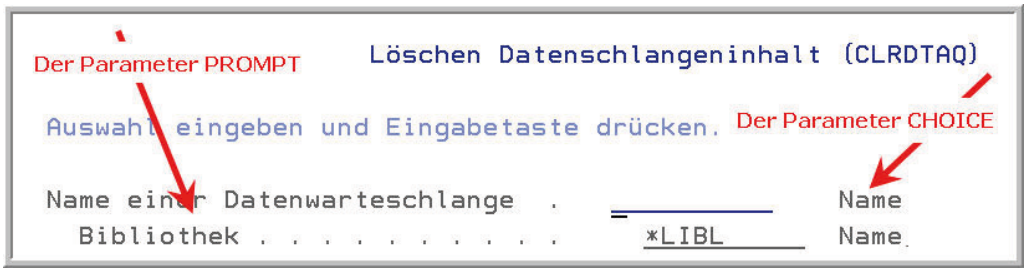
Es gibt weitere Datentypen im Command, die aber teilweise uninteressant sind oder für IBM gelieferte Befehle reserviert wurden. Hierzu gehören u. a. die Typen TYPE(*X) und TYPE(*VARNAME). Eine Erläuterung dieser Datentypen enthält die Bedienerhilfe.

Zusätzlich zur Datenart müssen wir die Länge für unseren Parameter festlegen. Dies geschieht mit Hilfe des Parameters LEN. Der Parameter LEN bestimmt die maximale Parameterlänge und somit auch indirekt die Länge der Variablen im CL-Programm, die den Parameter übernimmt. Für die Datenarten *DATE, *TIME, *INT2, *UINT2, *INT4, *UINT4, *ZEROELEM, *NULL und *CMD kann keine Längenangabe spezifiziert werden. *DEC akzeptiert maximal die Längenangabe LEN(24 9); der Standardwert ist wie in der CL-Programmierung mit LEN(15 5) vorgegeben. Der Typ *CHAR darf höchstens 5000 Byte lang sein, wenn für den CCSID-Parameter *JOB angegeben ist, oder 10.000 Byte, wenn für den CCSID-Parameter *UTF16 verwendet wird. *LGL akzeptiert nur die Längenangabe LEN(1) und *HEX kann mit maximal 256 Byte definiert werden. Weitere Restriktionen entnehmen Sie bitte wieder der Bedienerhilfe.

26. Ergänzung 3/2011

2.14.2.2
Seite 6

Als letztes fehlen die Parameter CHOICE und PROMPT. Schauen Sie sich die folgende Grafik an. Sie zeigt zunächst die Wirkungsweise dieser Parameter:



Wirkungsweise der Parameter PROMPT und CHOICE

Wenn Sie die Bedienerführung eines Befehls öffnen, werden rechts von den Eingabefeldern gültige Eingabewerte angezeigt. Die CHOICE-Klausel bestimmt, was hier gezeigt wird. Der im Beispiel verwendete Standardwert CHOICE(*VALUES) überlässt es dem System, festzustellen, welche Werte angezeigt werden. Dies ist die einfachste Möglichkeit, einen Auswahltext anzuzeigen. Sie können den Auswahltext aber auch selbst bestimmen, indem Sie CHOICE(*NONE) angeben, wenn kein Auswahltext erscheinen soll oder einen maximal 30-stelligen Text in der CHOICE-Klausel hinterlegen, der dann während der Befehlsausführung verwendet wird. Der Auswahltext könnte auch durch eine Nachrichten-ID spezifiziert werden. Eine weitere Möglichkeit haben Sie, wenn Sie CHOICE(*PGM) angeben. In diesem Fall wird der Auswahltext dynamisch von einem zusätzlichen Programm aufgebaut. Das könnte interessant sein, wenn die angezeigten Werte von der Existenz bestimmter Objekte abhängen. Ein entsprechendes Beispiel enthält Kapitel 2.14.7 dieses Buches.

Die Bedienerführungsanzeige enthält zudem links neben den Eingabeparametern einen kurzen beschreibenden Text. Diesen Text hinterlegen Sie mit der Klausel PROMPT. Als Standardwert wird PROMPT(*NONE) angenommen, so dass kein Text angezeigt wird. Sie sollten aber auf Bedienerführungstexte keinesfalls verzichten, wenn die Bedienerführung sinnvoll nutzbar sein soll. Alternativ könnten Sie auch hier wieder eine vordefinierte Nachricht einer Nachrichtendatei verwenden.

Jetzt sind alle Grundparameter hinterlegt. Wir können daher fürs Erste unseren Editor verlassen und müssen anschließend mit Hilfe der Anweisung CRTCMD das ausführbare Befehlsobjekt erstellen.

Die Anweisung CRTCMD

Wie bei der Programmerstellung muss auch der Quellcode der Commands umgewandelt werden, um ein Objekt der Art *CMD zu erstellen. Bei der Umwandlung mit dem Befehl CRTCMD wird der Quellcode auf syntaktische Fehler hin überprüft und das Objekt erstellt, wenn keine Fehler gefunden werden. In diesem Kapitel schauen wir uns den Befehl CRTCMD und seine Einstellungen etwas genauer an:

Zum Erstellen der meisten Commands ist es ausreichend, die Parameter CMD, PGM und SRCFILE anzugeben.

```
                        Befehl erstellen (CRTCMD)

Auswahl eingeben und Eingabetaste drücken.

Befehl . . . . . . . . . . . . . > CLRDTAQ      Name
  Bibliothek . . . . . . . . . . >   ITPCR      Name, *CURLIB
Befehlsverarbeitungsprogramm . . > CLRDTAQ      Name, *REXX
  Bibliothek . . . . . . . . . . >   *LIBL      Name, *LIBL, *CURLIB
Quellendatei . . . . . . . . . . > QCMDSRC      Name
  Bibliothek . . . . . . . . . . >   ITPCR      Name, *LIBL, *CURLIB
Quellenteildatei . . . . . . . . > CLRDTAQ      Name, *CMD
```

Bedienerführung CRTCMD – Grundparameter

Der Parameter „Befehl (CMD)“ ist verbindlich. Geben Sie hier den qualifizierten Namen des zu erstellenden Commands ein.

Der Parameter „Befehlsverarbeitungsprogramm (PGM)“ ist ebenfalls verbindlich und kennzeichnet das Befehlsverarbeitungsprogramm, das die eigentlichen Funktionen ausführt. Durch diesen Parameter werden die Befehlsdefinition und das Befehlsverarbeitungsprogramm miteinander verknüpft. Als Standardwert wird für den Parameter PGM stets der gleiche Name verwendet wie im Parameter CMD. Es ist sinnvoll, die Namen für Befehlsdefinition und Befehlsverarbeitungsprogramm identisch zu wählen, weil Sie so einfacher zuordnen können, welche Befehlsdefinition zu welchem Befehlsverarbeitungsprogramm gehört. Außerdem besteht nicht die Gefahr, dass Sie bei einer erneuten Umwandlung des Commands vergessen, den Namen des Befehlsverarbeitungsprogramms abzuändern.

Der Parameter „Quellendatei (SRCFILE)“ ist bereits optional. Hier kann der qualifizierte Name einer Quellendatei angegeben werden, die den Quellcode der Befehlsdefinition enthält. Als Standardwert wird der Name QCMDSRC verwendet.

26. Ergänzung 3/2011

2.14.3
Seite 2

Für meine Befehlsdefinition sind diese Grundparameter zunächst ausreichend. Ich bestätige daher den oben gezeigten Dialog und kann anschließend die Befehlsdefinition testen:

```
Parameter oder Befehl
===> ITPCR/CLRDTAQ   DATAQ(TesT)  LIB(EPS002CR)
F3=Verlassen      F4=Bedienerführung   F5=Aktualisieren     F6=Erstellen
F9=Auffinden      F10=Befehlseingabe   F23=Weitere Angaben  F24=Weitere Tasten
Datenwarteschlangeninhalt von TEST in Bibliothek EPS002CR wurde gelöscht!
```

Die Befehlsdefinition arbeitet korrekt

Die Abbildung zeigt, dass ich den CL-Befehl CLRDTAQ direkt auf der Befehlszeile verwenden kann und dadurch mein CL-Programm CLRDTAQ aufrufe, das mir fehlerfrei den Inhalt der Datenwarteschlange löscht. Eines fällt sofort auf: Die Groß- und Kleinschreibung der Objektnamen muss nicht mehr beachtet werden; alle Parameter werden an das CPP in Großbuchstaben übergeben. Der Grund hierfür ist die Standardeinstellung der PARM-Klausel „Groß-/Kleinschreib. des Werts (CASE)“. Wünschen Sie eine Unterscheidung zwischen großen und kleinen Buchstaben, müssen Sie diese Klausel in der PARM-Anweisung auf den Wert *MIXED abändern:

```
PARM KWD(DATAQ) TYPE(*SNAME) LEN(10) CASE(*MIXED) CHOICE(*VALUES) +
PROMPT('Name der Datenwarteschlange')
```

Der Befehl CRTCMD hat eine Reihe weiterer zusätzlicher Parameter, die wir an dieser Stelle ebenfalls betrachten wollen:

```
                          Befehl erstellen (CRTCMD)

Auswahl eingeben und Eingabetaste drücken.

Gültig bei Modus . . . . . . . . MODE           *ALL
               + für weitere Werte
Ausführung wo zulässig . . . . . ALLOW          *ALL
               + für weitere Werte
Benutzer begrenzt zulassen . . . ALWLMTUSR      *NO
Max. Positionsparameter  . . . . MAXPOS         *NOMAX
Nachrichtendatei f. Bed.text:    PMTFILE
  Nachrichtendatei . . . . . .                  *NONE
    Bibliothek . . . . . . . .
  Nachrichtentext  . . . . . .
Nachrichtendatei . . . . . . . . MSGF           QCPFMSG
  Bibliothek . . . . . . . . .                    *LIBL
Bücherfach . . . . . . . . . . . HLPSHELF       *NONE
Anzeigengruppe für Hilfetext . . HLPPNLGRP      *NONE
  Bibliothek . . . . . . . . .
Hilfe-ID . . . . . . . . . . . . HLPID          *NONE
```

Bedienerführung CRTCMD Teil 2

Der Parameter „Gültig bei Modus (MODE)“ legt fest, in welchen Umgebungen das Command ausgeführt werden darf. Sie können die Werte *ALL, *PROD (nur Produktionsumgebungen), *DEBUG (nur im Testhilfemodus) und *SERVICE (nur im Wartungsmodus) verwenden. Der Standardwert *ALL kann in den meisten Fällen beibehalten werden.

Der Parameter „Ausführung wo zulässig (ALLOW)“ bestimmt hingegen von wo der Befehl aufgerufen werden darf. Nachfolgende Tabelle enthält die gültigen Werte und ihre Bedeutung:

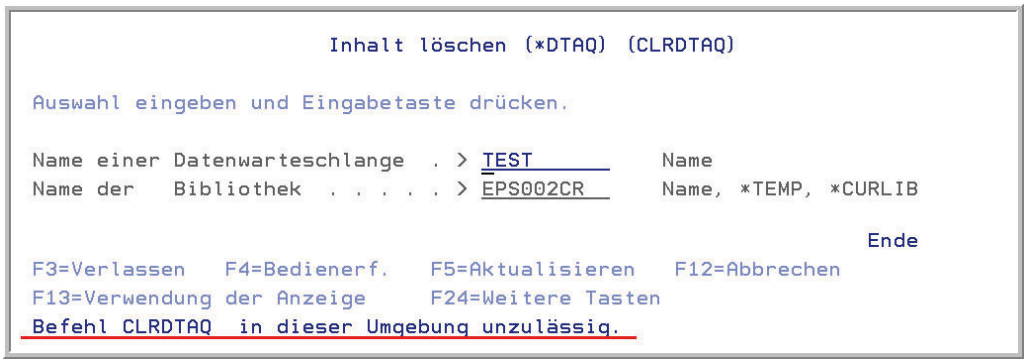
Wert	Bedeutung
*ALL	Die Standardeinstellung *ALL erlaubt uns die Verwendung unserer Commands auf der Befehlszeile, in Modulen und Programmen gleichermaßen. Der Befehl kann somit in allen Umgebungen aufgerufen werden.
*BATCH	Der Befehl kann nur innerhalb eines Batch-Jobs verwendet werden.
*INTERACT	Der Befehl kann nur in interaktiven Jobs verwendet werden.
*BPGM	Der Befehl kann nur innerhalb eines Programms verwendet werden, das als Batchjob ausgeführt wird.
*IPGM	Der Befehl kann nur innerhalb eines Programms verwendet werden, das interaktiv ausgeführt wird.
*EXEC	Der Befehl kann als Zeichenfolge an das API QCMDEXC zur Verarbeitung übergeben werden.
*BMOD	Der Befehl kann nur in einem CL-ILE-Batchprogramm benutzt werden.
*IMOD	Der Befehl kann nur in einem CL-ILE-interaktiven Programm benutzt werden.

Ich könnte also mit Hilfe des Parameters ALLOW verhindern, dass der eben erstellte Befehl CLRDATAQ interaktiv ausgeführt wird:

```
CRTCMD CMD(CRTUSRSPC) PGM(CRTUSRPSC) ALLOW(*BATCH *BPGM *BMOD)
```

2.14.3
Seite 4

Rufe ich anschließend das Command interaktiv auf einer 5250-Befehlszeile auf, erhalten wir folgende Fehlermeldung:



Wirkungsweise des Parameters ALLOW

Diese Einschränkungen der Befehlsverwendung können nützlich sein, um beispielsweise die Ausführung bestimmter CL-Programme interaktiv zu verhindern. Doch es gibt noch weitere Umwandlungsparameter:

Der **Parameter „Benutzer begrenzt zulassen (ALWLMTUSR)“:** Benutzer mit begrenzten Berechtigungen, d. h. mit der Angabe LMTCPB(*YES) im Benutzerprofil, können nur die Befehle DSPJOB, DSPJOBLOG, DSPMSG, SNDMSG und SIGNOFF ausführen, wenn ihnen eine Befehlszeile zur Verfügung steht. Ihre eigenen Commands dürften von diesen Benutzern ebenfalls nicht aufgerufen werden, weil der Parameter „Benutzer begrenzt zulassen (ALWLMTUSR)“ den Standardwert *NO annimmt. Sie müssen daher den Parameter auf den Wert *YES ändern, wenn Benutzer mit eingeschränkten Berechtigungen das Command ausführen sollen.

Der **Parameter „Maximale Positionsparameter (MAXPOS)“** sollte beim Standardwert *NOMAX belassen werden. Ich kenne keinen Grund, warum die Eingabe positionsgebundener Parameter auf einer Befehlszeile beschränkt werden sollte.

Der **Parameter „Nachrichtendatei für Bedienerführungstext (PMTFILE)“** legt die Nachrichtendatei fest, die verwendet wird, wenn Sie in der Klausel PROMPT oder CHOICE auf eine Nachrichten-ID verweisen.

Sie können für Ihr Command aber auch spezielle Fehlermeldungen mit Hilfe sogenannter DEP-Definitionsanweisungen hinterlegen. Wenn Sie diese Technik einsetzen, müssen Sie im **Parameter „Nachrichtendatei (MSGF)“** die Nachrichtendatei hinterlegen, die die im Befehl verwendeten Nachrichten-IDs enthält. **Ein Beispiel im Kapitel 6.6 zeigt, was gemeint ist.**

Sie können für Ihre Befehlsdefinition eigene Hilfetexte als Panelgruppen hinterlegen. Die zugehörige Panelgruppe müssen Sie im **Parameter „Anzeigengruppe für Hilfetext (HLPPNLGRP)“** hinterlegen. Der Parameter HLPPNLGRP nimmt den qualifizierten Namen der Anzeigengruppe auf, die den Hilfetext enthält. Der **Parameter „Hilfe-ID (HLPID)“** kennzeichnet dabei das Hauptmodul in der Anzeigegruppe, das den übergeordneten Hilfetext enthält.

```
Aktuelle Bibliothek . . . . . CURLIB          *NOCHG
Produktbibliothek . . . . . . PRDLIB          *NOCHG
Bedienerf.überschreibungsprog.  PMTOVRPGM     *NONE
  Bibliothek . . . . . .
Berechtigung . . . . . . . . . AUT            *LIBCRTAUT
Befehl ersetzen . . . . . . . REPLACE    >    *NO
```

Bedienerführung CRTCMD Teil 3

Über die **Parameter „Aktuelle Bibliothek (CURLIB)“** und **„Produktbibliothek (PRDLIB)“** können Sie für die Ausführung Ihres Befehls eine bestimmte aktuelle oder Produktbibliothek setzen. Der Parameter PRDLIB wird normalerweise von Systembefehlen verwendet, aber es spricht nichts dagegen, dass auch Ihre Commands mit Produktbibliotheken arbeiten. Die Funktion der Produktbibliotheken entspricht ein wenig der aktuellen Bibliothek mit dem Unterschied, dass nach Beendigung des Befehls wieder die beim Start des Befehls angegebene Produktbibliothek verwendet wird. Die Produktbibliothek steht in der Bibliothekssuchliste zwischen der aktuellen Bibliothek (*CURLIB) und den Benutzerbibliotheken.

Egal für welche Einstellungen Sie sich entscheiden, es besteht bei Veränderung der Standardeinstellungen immer die Gefahr, dass Sie bei einer erneuten Kompilation des Befehls vergessen, die Standardwerte anzupassen. Früher wurden daher die Erstellungsoptionen häufig als Kommentare im Quelltext hinterlegt, heute können wir die Umwandlungsparameter auch in der Anweisung CMD innerhalb der Befehlsdefinition hinterlegen. Nehmen wir das vorangegangene Beispiel:

2.14.3
Seite 6

```
                        Befehlsdefinition (CMD)

Auswahl eingeben und Eingabetaste drücken.

Marke . . . . . .
Bedienertext od. Nachricht.-ID   > 'Inhalt löschen (*DTAQ)'
Nachrichtendatei f. Bed.text:
  Nachrichtendatei . . . . . .                Name, *NONE
    Bibliothek . . . . . .      *LIBL         Name, *LIBL, *CURLIB
  Nachrichtentext  . . . . . .  *STATIC       *STATIC, *DYNAMIC
Nachrichtendatei . . . . . .                  Name
  Bibliothek . . . . . .        *LIBL         Name, *LIBL, *CURLIB
Text 'Beschreibung'  . . . . . .
_____
Max. Positionsparameter  . . . .              0-99, *NOMAX
Ausführung wo zulässig . . . . . > *BATCH     *ALL, *BATCH, *INTERACT...
                                  > *BPGM
     + für weitere Werte          > *BMOD
```

Umwandlungsparameter der Anweisung CMD

Anstatt den Parameter ALLOW des Umwandlungsbefehls CRTCMD zu ändern, hinterlege ich in der oberen Abbildung diese Informationen jetzt direkt im Quellcode. Dafür verwende ich die Anweisung CMD, so kann ich jederzeit das Objekt neu kompilieren, ohne wichtige Umwandlungsparameter zu vergessen.

Nützliche Klauseln der PARM-Anweisung

Hat man erst einmal die ersten Commands erstellt, werden schnell weiter Anforderungen gestellt. Bislang habe ich in meinem Befehl CLRDATAQ beispielsweise nicht berücksichtigt, dass das CPP nicht mit leeren Parameterwerten aufgerufen werden darf. Es blieb bislang auch unberücksichtigt, dass das Command für den Bibliotheksnamen den Wert *CURLIB als Alternative zu einem Bibliotheksnamen an das CPP übergeben könnte. Diese Dinge wollen wir jetzt ändern.

2.14.4
Seite 2

Erforderliche Parameter – Standardwerte

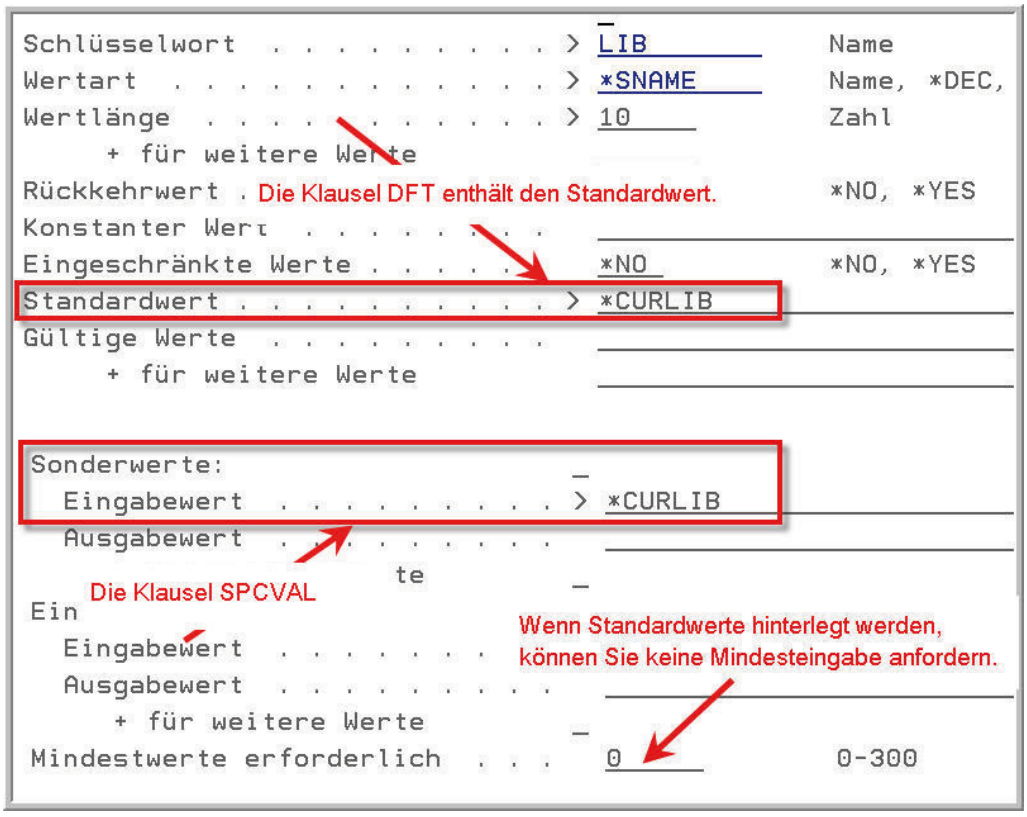
Das CPP soll nicht ausgeführt werden, wenn der Parameter DATAQ leer ist und für die Bibliothek wollen wir den Sonderwert *CURLIB als Standardwert hinterlegen, wenn der Anwender keinen Bibliotheksnamen angibt.

Um einen verbindlichen Parameter zu definieren, benötigen wir lediglich die Klausel MIN in unserer Parameterdefinition:

```
PARM KWD(DATAQ) TYPE(*SNAME) LEN(10) MIN(1) CHOICE(*VALUES) +
PROMPT('Name des User Space')
```

Ein Parameter wird als verbindlich bezeichnet, wenn das System keinen Standardwert verwendet und die Eingabe von Leerzeichen oder Null nicht akzeptiert. In einem Command muss die Klausel MIN der entsprechenden PARM-Anweisung nur den Wert 1 enthalten, damit der Parameter als verbindlich gilt und es ist darauf zu achten, dass alle verbindlichen Parameter vor den optionalen Parametern kodiert werden. Das war einfach!

Nun hinterlegen wir noch für unsere Bibliothek den Standardwert *CURLIB. Hierzu benötigen wir die Klausel DFT – wie folgende Abbildung zeigt.



Standardwerte vereinbaren

2.14.4.1
Seite 2

Es ist bei der Hinterlegung von Standardwerten darauf zu achten, dass der Parameter, der den Standardwert annehmen soll, mit MIN(0) definiert werden muss, um festzulegen, dass keine zwingende Eingabe erforderlich ist. Ändert der Anwender beim Aufruf unseres Commands nicht den Parameter LIB, wird immer der Standardwert *CURLIB an das CPP-Programm übergeben. Somit ist sichergestellt, dass unserem CL-Programm immer der Name einer Bibliothek übergeben wird.

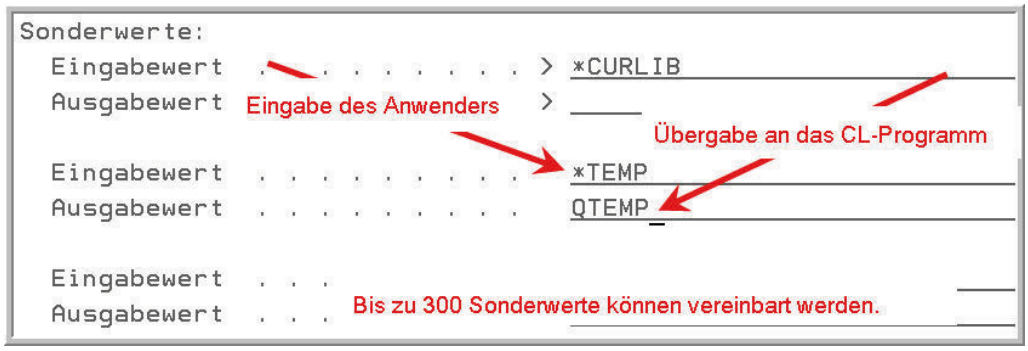
```
                    Inhalt löschen (*DTAQ) (CLRDTAQ1)

Auswahl eingeben und Eingabetaste drücken.

Name einer Datenwarteschlange  . > _________    Name
Name der   Bibliothek  . . . . .   *CURLIB     Name, *CURLIB
```

Wirkungsweise hinterlegter Standardwerte

Wie erwartet wird jetzt als Standardwert *CURLIB für die Objektadressierung verwendet. Beachten müssen Sie in diesem Zusammenhang Datums- und Zeitwerte. Wenn Sie für ein Datumsfeld Standardwerte hinterlegen wollen, muss der Standardwert im Format MMTTJJ, MMTTJJJJ oder im Format CJJMMTT hinterlegt werden, wobei C das Jahrhundert, JJ das Jahr, MM den Monat und TT den Tag repräsentiert. Die Jahrhundertziffer wird für die Jahre 19xx auf 0 und für die Jahre 20xx auf 1 gesetzt.

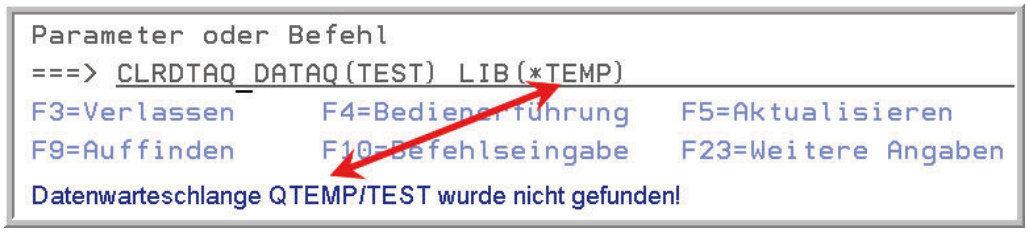
Kommen wir aber noch mal auf das in der Abbildung gezeigte Beispiel zurück. Der Datentyp TYP(*SNAME), den wir für den Parameter LIB verwendet haben, wird keinen Stern (*) innerhalb des Objektnamens akzeptieren. Um die Eingabe dennoch zu ermöglichen, müssen wir für unseren Parameter Sonderwerte hinterlegen.

Mit der Klausel SPCVAL kann eine Liste mit bis zu 300 Sonderwerten für einen Parameter definiert werden. Jeder Eintrag in der Klausel SPCVAL kann aus einem oder zwei Werten bestehen. Der erste Wert, der Eingabewert, enthält stets die Bezeichnung des Sonderwertes wie *SYSVAL, *LIBL oder *CURLIB. Mit dem zweiten Eintrag, dem Ausgabewert, kann anschließend festgelegt werden, welcher Wert tatsächlich an das CL-Programm übergeben wird, wenn der entsprechende Eingabewert ausgewählt wird:



Sonderwerte vereinbaren

Sie können in der oberen Abbildung erkennen, dass ich inzwischen einen weiteren Sonderwert vereinbart habe und den Bibliotheksnamen QTEMP an mein CL-Programm übergeben will, wenn der Sonderwert *TEMP im Befehl CLRDATAQ genutzt wird. Prüfen wir, ob das Vorhaben gelingt:



*Wirkungsweise des Sonderwertes *TEMP*

Der Sonderwert *TEMP wird im Befehl verwendet, aber der Bibliotheksname QTEMP wird an das CL-Programm übergeben. Ist kein Ausgabewert hinterlegt, erhält das CL-Programm immer den Eingabewert übergeben.

2.14.4.1
Seite 4

Konstanten und zusätzliche Parameter

Um Ihnen die Wirkungsweise von Konstanten und zusätzlichen Parametern zu erläutern, verwenden wir im Folgenden ein anderes CPP-Programm. Das CL-Programm CRTUSRSPC erstellt einen User Space (*USRSPC). User Spaces sind Anwenderbereiche, die ebenso wie Datenwarteschlangen (*DTAQ) und Datenbereiche (*DTAARA) dazu genutzt werden können, Informationen abzulegen.

Viele APIs nutzen beispielsweise diese Bereiche für ihre Ausgaben, aber auch Java-Programmierer können mit entsprechenden Methoden der Java-Toolbox darauf zugreifen. Ein Anwenderbereich ist ein Objekt der Art *USRSPC, das erstellt, manipuliert und gelöscht werden kann. Da die User Spaces auf der Festplatte erstellt werden, können diese Objekte – wie alle anderen Objekte auch – gesichert und wiederhergestellt werden. Grundsätzlich lassen sich User Spaces mit Data Areas vergleichen. Es handelt sich auch bei einem User Space zunächst um einen unstrukturierten Datenbereich, in dem beliebige Informationen gespeichert werden können. Im Gegensatz zu den Data Areas, deren Maximalgröße auf 2 KB begrenzt ist, können User Spaces bis zu 16 Megabyte groß sein und ihre Größe dynamisch anpassen.

Für die Erstellung, aber auch für die Verwaltung eines User Space gibt es allerdings keine entsprechenden CL-Befehle. Alle Zugriffe erfolgen ausschließlich über APIs. Es bietet sich daher an, eigene Befehlsdefinitionen für die Arbeit mit User Spaces zu erstellen.

User Spaces können mit Hilfe des APIs QUSCRTUS (Create User Space) erstellt werden. Ein entsprechendes CL-Programm ist schnell geschrieben:

Das API erwartet den qualifizierten Namen des User Space, wobei erst der Objektname und in den letzten 10 Byte der Bibliotheksname genannt werden muss. Als Sonderwert könnte ich später wieder anstelle des Bibliotheksnamens den Wert *CURLIB verwenden. Es ist natürlich auch möglich, den User Space in der Bibliothek QTEMP anzulegen, so dass der Anwenderbereich automatisch gelöscht wird, wenn der Job endet.

```
PGM PARM(&OBJ &LIB &EXT &SIZE &INIT &AUT &TEXT &REPLACE)

       DCL VAR(&USRSPC) TYPE(*CHAR) LEN(20)
       DCL VAR(&OBJ) TYPE(*CHAR) LEN(10)
       DCL VAR(&LIB) TYPE(*CHAR) LEN(10)
```

2.14.4.2
Seite 2

Leider kann ich für den Objektnamen und die Bibliothek keine Datenstruktur einsetzen, da Datenstrukturfelder nicht als Parameter im Aufruf verwendet werden können.

Über den Parameter „Extended Attribute“ können Sie dem User Space ein Zusatzattribut zuweisen – ähnlich einer physischen Datei, die das Zusatzattribut PF trägt, oder wie bei einer Druckdatei, die anhand des Zusatzattributes *PRTF identifiziert werden kann. Ob Sie ein Zusatzattribut verwenden und wenn ja welches, bleibt Ihnen überlassen, solange Sie die gültigen Namenskonventionen des Betriebssystems beachten.

```
DCL VAR(&EXT) TYPE(*CHAR) LEN(10)
```

Nun muss noch die Anfangsgröße des Anwenderbereichs festgelegt werden. Werte zwischen 1 und 16.776.704 sind möglich.

```
DCL VAR(&SIZE) TYPE(*INT) LEN(4)
```

Für den Parameter „Initial Value“ wird empfohlen, den Wert X‘00‘ zu verwenden, um den Bereich entsprechend zu initialisieren.

```
DCL VAR(&INIT) TYPE(*CHAR) LEN(1)
```

Die beiden letzten erforderlichen Parameter bestimmen die Berechtigung der Benutzergruppe *PUBLIC. Hier gelten die allgemeinen Berechtigungsklassen: *ALL, *CHANGE, *USE, *EXCLUDE. Aber auch den Namen einer Berechtigungsliste können Sie einsetzen oder den Sonderwert *LIBCRTAUT verwenden.

```
DCL VAR(&AUT) TYPE(*CHAR) LEN(10)
```

Nun können wir noch einen beschreibenden Text für das Objekt hinzufügen und haben damit alle zwingend erforderlichen Parameter deklariert.

```
DCL VAR(&TEXT) TYPE(*CHAR) LEN(50)
```

Um sicher zu stellen, dass der Anwenderbereich auch neu erstellt wird, wenn er bereits existiert, können Sie den Parameter &REPLACE verwenden.

```
DCL VAR(&REPLACE) TYPE(*CHAR) LEN(10)
```

Zu guter Letzt folgt die Fehlerstruktur, um zu prüfen, ob das API fehlerfrei ausgeführt wurde. Die Struktur muss folgendermaßen deklariert werden:

```
DCL VAR(&ERR_DS) TYPE(*CHAR) STG(*AUTO) LEN(256)
DCL VAR(&LEN) TYPE(*INT) STG(*DEFINED) LEN(4) DEFVAR(&ERR_DS)
DCL VAR(&RTN_LEN) TYPE(*INT) STG(*DEFINED) LEN(4) DEFVAR(&ERR_DS 5)
DCL VAR(&MSGID) TYPE(*CHAR) STG(*DEFINED) LEN(7) DEFVAR(&ERR_DS 9)
DCL VAR(&RESERVED) TYPE(*CHAR) STG(*DEFINED) LEN(1) DEFVAR(&ERR_DS 16)
DCL VAR(&MSGDTA) TYPE(*CHAR) STG(*DEFINED) LEN(240) DEFVAR(&ERR_DS 17)
```

Den Aufbau der Datenstruktur kennen Sie bereits aus dem Kapitel 2.12.6.12.2 dieses Buches und Datenstrukturen haben wir auch bereits kennen gelernt, so dass hier keine weiteren Erläuterungen notwendig sind.

Die Variablendeklariation ist am aufwendigsten. Im Programm müssen wir nur noch den qualifizierten Objektnamen verketten, da ich ja Bibliotheksname und Objektname als einzelne Parameterwerte an das CPP-Programm übergebe. Achten Sie aber darauf, dass erst der Objektname und dann der Bibliotheksname genannt werden muss.

```
CHGVAR &USRSPC VALUE(&OBJ *CAT &LIB)
```

Anschließend erfolgt bereits der Aufruf des APIs QUSCRTUS:

```
CALL PGM(QUSCRTUS) +
PARM(&USRSPC &EXT &SIZE &INIT &AUT &TEXT &REPLACE &ERR_DS)
```

Damit ich erfahre, ob Fehler aufgetreten sind, prüfe ich das Feld &RTN_LEN der Fehlerstruktur. Enthält diese Variable den Wert 0, sende ich die Meldung, dass der User Space fehlerfrei erzeugt wurde. Im Fehlerfall gebe ich die ESCAPE-Nachricht aus, die das API mir über die Fehlerstruktur zurückgibt.

```
IF COND(&RTN_LEN *NE 0) THEN(DO)
   SNDPGMMSG MSGID(&MSGID) MSGF(QCPFMSG) MSGDTA(&MSGDTA)
   ENDDO

ELSE CMD(DO)
   SNDPGMMSG    MSG(,User Space‘ *BCAT &OBJ *BCAT ,in Bibliothek‘
                *BCAT &LIB *BCAT + ‘wurde erstellt’)
   ENDDO
ENDPGM
```

2.14.4.2
Seite 4

Nun ist es an der Zeit, unser CL-Programm CRTUSRSPC zu testen:

```
> CALL PGM(CRTUSRSPC) PARM(MYSPACE QTEMP 'TST' 5000 X'00' '*USE' 'Test für
  CL-Programmierung' '*YES')
  User Space MYSPACE in Bibliothek QTEMP wurde erstellt
```

Testaufruf des CPP-Programms CRTUSRSPC

Für dieses CL-Programm will ich jetzt eine Command-Definition erstellen, denn der CALL-Aufruf meines CL-Programms ist nicht nur sehr aufwendig, sondern auch fehleranfällig. Das Command soll alle angeforderten Parameter an das CPP-Programm übergeben. Neben den bekannten Funktionen will ich allerdings die *PUBLIC-Berechtigung und die Initialisierungswerte als Konstanten übergeben. Die Parameter &REPLACE &SIZE und &EXT hingegen sind optionale, zusätzliche Parameter, wobei darauf zu achten ist, dass die Variable &REPLACE lediglich die Werte *YES und *NO empfangen darf.

Beginnen wir zunächst mit den bekannten Dingen:

```
CMD     PROMPT(,User Space erstellen‘)

PARM    KWD(USRSPC) TYPE(*SNAME) LEN(10) MIN(1) CHOICE(*VALUES) +
        PROMPT(,Name des User Space‘)

PARM    KWD(LIB) TYPE(*SNAME) LEN(10) DFT(*CURLIB) SPCVAL((*CURLIB) +
        (*TEMP QTEMP)) CHOICE(*VALUES) PROMPT(,Bibliothek‘)

PARM    KWD(TEXT) TYPE(*CHAR) LEN(30) MIN(1) CHOICE(*VALUES) +
        PROMPT(‘User Space Verwendung’)
```

Der Objektname und ein beschreibender Text sind Pflichtparameter. Als Bibliotheksname wird wieder der Standardwert *CURLIB verwendet.

Nun wollen wir aber die Initialisierungswerte und die allgemeine Berechtigung (*PUBLIC) als konstante Werte an das CPP-Programm übergeben:

```
PARM KWD(INIT) TYPE(*CHAR) LEN(1) CONSTANT(X’00’)

PARM KWD(AUT) TYPE(*CHAR) LEN(10) CONSTANT(*EXCLUDE)
```

Konstante Parameter werden mit der Klausel CONSTANT deklariert. Sie bestimmt, dass das Befehlsverarbeitungsprogramm immer einen gleichbleibenden Wert erhält. Wenn Sie sich den Quelltext anschauen, werden Sie feststellen, dass ich keine Standardwerte mehr nutze und auch keinen Bedienerführungstext hinterlege. Die Klausel CONSTANT unterbindet die Verwendung der Klauseln DFT und PROMPT. Konstante Parameter werden im Übrigen nie angezeigt, wenn Sie die Bedienerführung eines Befehls anfordern.

Die Parameter &EXT, &SIZE und &REPLACE sollen in einem weiteren Schritt als optionale, zusätzliche Parameter definiert werden. Wir wissen bereits, dass optionale Parameter immer die Klausel MIN (0) enthalten und mit oder ohne Standardwert definiert werden können. Durch die Definition eines Standardwertes wird der Parameter nicht leer an das CPP übergeben, wenn der Anwender keine Eingabe macht. Wir wollen aber jetzt einen Schritt weiter gehen und die optionalen Parameter erst anzeigen, wenn der Anwender die F10-Taste nutzt. Sie kennen diese Wirkung von vielen Systembefehlen.

Selten genutzte Parameter können mit dem Schlüsselwort PMTCTL (*PMTRQS) in der PARM-Anweisung gekennzeichnet werden. Es ist lediglich darauf zu achten, dass diese Parameter unterhalb der verbindlichen Parameter kodiert werden.

```
CMD     PROMPT(,User Space erstellen‘)

/* Grundparameter – verbindliche Parameter */
PARM    KWD(USRSPC) TYPE(*SNAME) LEN(10) MIN(1) CHOICE(*VALUES) +
        PROMPT(,Name des User Space‘)

PARM    KWD(LIB) TYPE(*SNAME) LEN(10) DFT(*CURLIB) SPCVAL((*CURLIB)
        (*TEMP QTEMP)) + CHOICE(*VALUES) PROMPT(,Bibliothek‘)

PARM    KWD(TEXT) TYPE(*CHAR) LEN(30) CHOICE(*VALUES) +
        PROMPT(‘User Space Verwendung’)

/* optionale, zusätzliche Parameter – Taste F10 */

PARM    KWD(SIZE) TYPE(*INT4) DFT(1000) CHOICE(*VALUES)
        PMTCTL(*PMTRQS) + PROMPT(‘Anfangsgröße des User Space’)

PARM    KWD(EXT) TYPE(*CHAR) LEN(10) CHOICE(‘beliebige
        Zeichenfolge(10)’) + PMTCTL(*PMTRQS) PROMPT(,EXTENSION
        des User Space‘)
```

Für den Parameter REPLACE müssen wir noch berücksichtigen, dass lediglich die Werte ‚J‘ oder ‚N‘ für die Übergabe an das CL-Programm verwendet werden dürfen. Die im folgenden Quellcodeausschnitt verwendete Klausel RSTD für den Parameter REPLACE kann ich nutzen, um diese Anforderung zu erfüllen. Durch die Klausel RSTD(*YES) wird festgelegt, dass ein Parameter nur bestimmte Werte annehmen soll. Die gültigen Eingabewerte werden anschließend in den Klauseln VALUES, SPCVAL und SNGVAL hinterlegt, wie der folgende Quellcode zeigt:

```
PARM    KWD(REPLACE) TYPE(*CHAR) RSTD(*YES) DFT(J) VALUES(J N) +
        CHOICE(*VALUES) PMTCTL(*PMTRQS) PROMPT('Anwenderbereich
        ersetzen?[J/N]')
```

Es ist dabei natürlich darauf zu achten, dass auch die verwendeten Standardwerte stets der Plausibilitätsprüfung standhalten müssen.

Die PARM-Anweisung kennt noch viele weitere Klauseln, die Plausibilitätsprüfungen durchführen. Um die Prüfungen zu aktivieren, muss lediglich der Standardwert des Parameters RSTD auf *YES geändert werden. Anschließend könnten wir beispielsweise mit Hilfe der Klausel RANGE Wertebereiche definieren.

```
PARM    KWD(SIZE) TYPE(*INT4) DFT(1000) RANGE(1000 32000)
        CHOICE(*VALUES) +
        PROMPT('Anfangsgröße des User Space')
```

Es kann dabei auch Bezug auf einen anderen Parameter genommen werden, indem der entsprechende Parametername als Unter- bzw. Obergrenze aufgeführt wird.

Die Klausel REL definiert hingegen eine Beziehung zwischen einem Parameter und einem konstanten Wert oder zwei verschiedenen Parametern der Befehlsdefinition:

```
PARM    KWD(SIZE) TYPE(*INT4) DFT(5000) REL(*GT 1000)
        CHOICE(*VALUES) +
        PMTCTL(*PMTRQS) PROMPT('Anfangsgröße des User Space')
```

Wie in CL-Programmen können die Vergleichsoperatoren *EQ, *GT, *GE, *LT, *LE *NE hierbei verwendet werden.

Wird die Klausel FULL(*YES) hingegen genutzt, vereinbaren Sie eine verbindliche Feldfüllung, d. h. der Anwender muss den Parameter in der gesamten Länge ausfüllen.

Schauen Sie sich die entsprechenden Parameter bei Gelegenheit einmal an. Sie werden feststellen, dass vieles selbsterklärend ist.

Qualifizierte Namen verwenden

Von einem qualifizierten Namen sprechen wir, wenn der Parametername sich aus mehreren Elementen zusammensetzt. Ein typisches Beispiel hierfür sind die vielen qualifizierten Objektnamen. Um ein Objekt wirklich eindeutig zu identifizieren, müssen immer der Objektname und die Bibliothek angegeben werden, die das Objekt enthält. Bisher haben wir die beiden Informationen jedoch als eigenständige Parameter übergeben. Dies soll sich jetzt ändern:

Der Parameter USRSPC fordert eigentlich einen qualifizierten Objektnamen an. Schauen Sie sich hierfür den folgenden Quellcode an:

```
CMD PROMPT(,User Space erstellen‘)

/* Qualifizierte Parameternamen */
PARM    KWD(USRSPC) TYPE(Q1) PROMPT(,Name des User Space‘)
```

Für den Parameter USRSPC wurde jetzt die Datenart TYPE(Q1) verwendet. Q1 ist aber keine Datenart, sondern ein Verweis auf ein Label, eine Marke. Die PARM-Anweisung enthält jetzt auch keine Längenangabe mehr, da diese sich später aus der Summe aller Elemente ergeben wird. Der Bedienerführungstext für das erste Element muss dagegen weiterhin in der PARM-Anweisung stehen.

Unser Command muss nun ein Label Q1 enthalten. Dort sind die Anweisungen zu finden, die den qualifizierten Parameter genauer beschreiben.

```
/* Zuerst wird der Parameter USRSPC mit Hilfe der Anweisung QUAL
   spezifiziert */
Q1:   QUAL TYPE(*SNAME) LEN(10) MIN(1)

/* Die zweite QUAL-Anweisung beschreibt die Eigenschaften des
   ursprünglichen Parameters LIB */
QUAL TYPE(*SNAME) LEN(10) DFT(*CURLIB) SPCVAL((*CURLIB) (*TEMP
QTEMP)) +
   CHOICE(*VALUES) PROMPT(‘Bibliothek’)
```

Das Beispiel zeigt, dass wir jetzt eine neue Anweisung benötigen. Die Anweisung QUAL wird benötigt, um den qualifizierten Parameter USRSPC im Command zu definieren. Die erste QUAL-Anweisung bestimmt die Eigenschaften des Parameters USRSPC, sie darf natürlich jetzt keinen Bedienerführungstext mehr erhalten, da dieser ja bereits in der entsprechenden PARM-Anweisung hinterlegt wurde. Ansonsten ändert sich aber nichts; alle Schlüsselworte behalten die bereits erläuterten Funktionen. Auch in der 2. QUAL-Anweisung sind keine neuen Klauseln verborgen. Lediglich das Schlüsselwort KWD wird nicht mehr spezifiziert, da der Bibliotheksname jetzt ein Element des Parameters USRSPC ist und demzufolge keinen eigenen Parameternamen mehr benötigt.

2.14.5
Seite 2

Anschließend folgen die weiteren Parameterdefinitionen im CDO-Objekt:

```
PARM KWD(TEXT) TYPE(*CHAR) LEN(30) MIN(1) CHOICE(*VALUES) +
   PROMPT('User Space Verwendung')

/* zusätzliche Parameter */
PARM KWD(REPLACE) TYPE(*CHAR) RSTD(*YES) DFT(N) VALUES(J N)
     PMTCTL(*PMTRQS) +
     PROMPT('Anwenderbereich ersetzen?[J/N]')

PARM KWD(SIZE) TYPE(*INT4) DFT(1000) RANGE(1000 32000)
     CHOICE(*VALUES) +
     PMTCTL(*PMTRQS) PROMPT('Anfangsgröße des User Space')

PARM KWD(EXT) TYPE(*CHAR) LEN(10) CHOICE('beliebige
     Zeichenfolge(10)') +
     PMTCTL(*PMTRQS) PROMPT('EXTENSION des User Space')

/* Konstante Parameter */
PARM KWD(INIT) TYPE(*CHAR) LEN(1) CONSTANT(X'00')
PARM KWD(AUT) TYPE(*CHAR) LEN(10) CONSTANT(*EXCLUDE)
```

Sobald der Command umgewandelt ist, können Sie die Wirkung qualifizierter Namen erkennen:

```
                    User Space erstellen (CRTUSRSPC)

Auswahl eingeben und Eingabetaste drücken.
Der qualifizierte Parameter USRSPC
User Space Name  . . . . . .   _________     Name
  Bibliothek . . . . . . . .     *CURLIB     Name, *CURLIB, *TEMP
User Space Verwendung  . . . .   ____________________________

                Zusätzliche Parameter      Wirkungsweise der Klausel PMTCTL(*PMTRQS)

Anwenderbereich ersetzen?[J/N]     N
Anfangsgröße des User Space  . .   1000_____   1000-32000
EXTENSION des User Space . . . .   _________   beliebige Zeichenfolge(10)
```

Wirkungsweise qualifizierter Parameter

Das zweite Element des Parameters USRSPC ist jetzt ein wenig eingerückt und wir haben es nicht mehr mit zwei eigenständigen Parametern zu tun, sondern lediglich mit dem Parameter USRSPC.

Demzufolge muss sich unser CL-Programm natürlich ein wenig ändern:

Das CL-Programm darf nur noch den Parameter USRSPC anfordern. Der Programmeingang muss sich also entsprechend ändern.

```
PGM PARM(&USRSPC &TEXT &REPLACE &SIZE &EXT &INIT &AUT)
```

Unser CDO-Objekt übergibt die qualifizierten Objektnamen übrigens in der Reihenfolge:

Objektname Bibliotheksname.

So erklärt sich dann auch der Aufbau der Datenstruktur &USRSPC im CL-Programm:

```
DCL VAR(&USRSPC) TYPE(*CHAR) LEN(20)
DCL VAR(&OBJ) TYPE(*CHAR) STG(*DEFINED) LEN(10) DEFVAR(&USRSPC)
DCL VAR(&LIB) TYPE(*CHAR) STG(*DEFINED) LEN(10) DEFVAR(&USRSPC 11)
```

Alle weiteren Variablen bleiben unverändert:

```
DCL VAR(&EXT) TYPE(*CHAR) LEN(10)
DCL VAR(&SIZE) TYPE(*INT) LEN(4)
DCL VAR(&INIT) TYPE(*CHAR) LEN(1)
DCL VAR(&AUT) TYPE(*CHAR) LEN(10)
DCL VAR(&TEXT) TYPE(*CHAR) LEN(50)
DCL VAR(&REPLACE) TYPE(*CHAR) LEN(10)
… … …
```

Aufgrund der eingesetzten Datenstruktur und der Übergabe des qualifizierten Objektnamens können wir auf die CHGVAR-Anweisung im CPP-Programm verzichten, die bislang den Parameter USRSPC mit Inhalten versorgte. Dies hat das Command bereits erledigt. Wir können also unmittelbar nach der Variablendeklaration das API aufrufen:

```
CALL PGM(QUSCRTUS) PARM(&USRSPC &EXT &SIZE &INIT &AUT &TEXT
     &REPLACE + &ERR_DS)
```

2.14.5
Seite 4

Abhängigkeiten zwischen Parametern

Eigene Commands erleichtern aber nicht nur die Arbeit mit APIs, sondern können für jedes Anwendungsprogramm genutzt werden. Ich nutze im nächsten Beispiel ein CL-Programm, das ein Query ausführen soll und übergebe dabei auch gleich verschiedene Ausgabeoptionen, wobei ich mich im ersten Beispiel zunächst auf den Namen des Querys und den Ausführungszeitpunkt beschränke.

Das CL-Programm ist wieder schnell geschrieben:

Zunächst übergebe ich den qualifizierten Namen des Querys und ein Datum plus Uhrzeit, um den Ausführungszeitpunkt zu spezifizieren.

```
PGM  PARM(&QUERY &DATE &TIME)

DCL  &QUERY *CHAR 20
   DCL VAR(&QRYNAM) TYPE(*CHAR) STG(*DEFINED) LEN(10)
       DEFVAR(&QUERY)
   DCL VAR(&LIB) TYPE(*CHAR) STG(*DEFINED) LEN(10)
       DEFVAR(&QUERY 11)

   DCL VAR(&DATE) TYPE(*CHAR) LEN(7)
   DCL VAR(&DATUM) TYPE(*CHAR) LEN(10)
   DCL VAR(&TIME) TYPE(*CHAR) LEN(6)
```

Den Aufbau entsprechender Bildschirmdateien und die damit verbundene Programmierung kann ich mir sparen, da ich in einem zweiten Schritt ein Command definieren werde, um das CPP-Programm aufzurufen. Für die Übergabe des Datums und der Uhrzeit sollen allerdings folgende Vereinbarungen gelten:

Wenn weder ein Datum noch eine Uhrzeit eingetragen werden, wird das SQL-Skript sofort ausgeführt. Im CPP-Programm muss ich diese Bedingung entsprechend berücksichtigen:

```
    IF  COND(&DATE = ,0000000‘) THEN(+
        ADDJOBSCDE JOB(&QRYNAM) +
                   CMD(STRQMQRY QMQRY(&LIB/&QRYNAM) OUTPUT(*PRINT) +
                                QMFORM(*QMQRY) +
                                DATETIME(*NO) PAGNBR(*NO)) +
       FRQ(*ONCE) SCDDATE(*CURRENT) SCDTIME(*CURRENT))
```

2.14.6
Seite 2

Wenn allerdings ein Datum eingetragen wird, muss auch eine Uhrzeit für die Ausführung an das CL-Programm übergeben werden. Das Datum wird vom Command im Format CJJMMTT übergeben und muss in europäisches Format umgesetzt werden, damit es im CL-Befehl ADDJOBSCDE verwendet werden kann. Dies gelingt durch den Befehl CVTDAT.

```
    ELSE CMD(DO)
         CVTDAT DATE(&DATE) TOVAR(&DATUM) FROMFMT(*CYMD) TOFMT(*EUR)
```

Anschließend können Datum und Uhrzeit im Scedulerjob entsprechend verwendet werden:

```
         ADDJOBSCDE JOB(&QRYNAM) +
                    CMD(STRQMQRY QMQRY(&LIB/&QRYNAM) OUTPUT(*PRINT) +
                    QMFORM(*QMQRY) DATETIME(*NO) PAGNBR(*NO)) +
         FRQ(*ONCE) SCDDATE(&DATUM) SCDTIME(&TIME)

    ENDDO
ENDPGM
```

Die Klausel DEP

Nun muss noch das Command definiert werden. Die Deklaration des qualifizierten Objektnamens ist uns bereits vertraut:

```
        CMD PROMPT('Query ausführen')

        PARM KWD(QUERY) TYPE(Q1) PROMPT('Name des Querys')

Q1:     QUAL TYPE(*SNAME) LEN(10) MIN(1) CHOICE(*VALUES)
        QUAL TYPE(*SNAME) LEN(10) DFT(*CURLIB) SPCVAL((*CURLIB)) +
             PROMPT('Bibliothek')
```

Wir können die Parameter DATE und TIME allerdings nicht als zwingende Parameter definieren, da sie ja leer übergeben werden können, wenn das Query sofort ausgeführt werden soll.

```
PARM    KWD(DATE) TYPE(*DATE) MIN(0) CHOICE(*VALUES)
           PROMPT('Ausführung am:')

PARM    KWD(TIME) TYPE(*TIME) MIN(0) CHOICE(*VALUES)
           PROMPT('Ausführung um:')
```

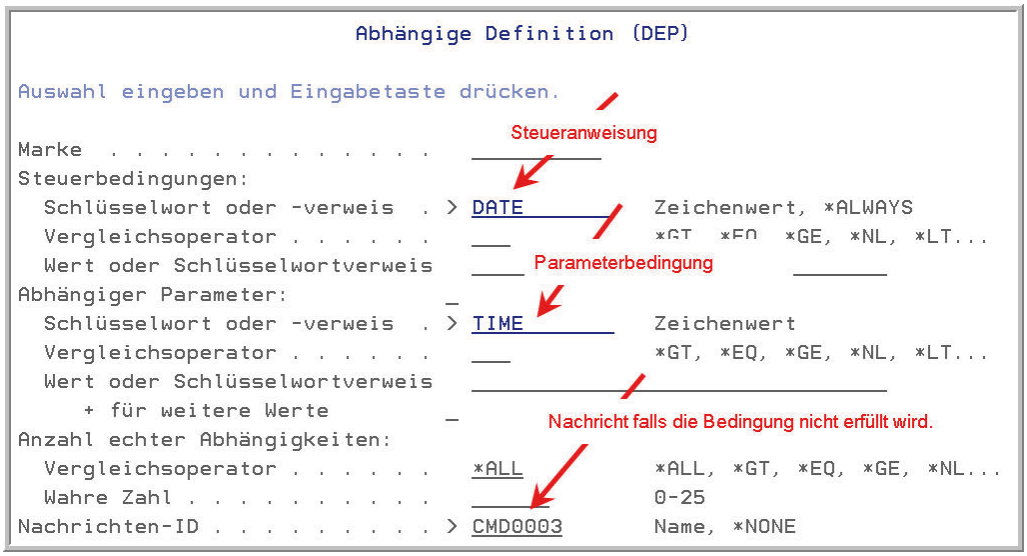
Andererseits muss aber das Command dafür sorgen, dass die Uhrzeit zwingend einzutragen ist, wenn der Anwender einen Ausführungstag bestimmt und umgekehrt. In dieser Situation ist es also nur dann sinnvoll, den Parameter TIME anzufordern, wenn der Parameter DATE genutzt wurde. Die Abhängigkeiten zwischen zwei oder mehreren Parametern werden im Command mit der Anweisung DEP hergestellt.

```
DEP CTL(DATE) PARM(TIME) MSGID(CMD0003)
DEP CTL(TIME) PARM(DATE) MSGID(CMD0003)
```

Die DEP-Anweisung kann dabei nur für Parameter eingesetzt werden, die mit dem Schlüsselwort KWD definiert wurden. Elemente eines qualifizierten Namens, die keinen eigenen Parameternamen haben, können auf diesem Weg nicht geprüft werden.

2.14.6.1
Seite 2

Schauen wir uns die Klauseln einer DEP-Anweisung etwas genauer an:



Bedienerführung der Anweisung DEP

Die DEP-Anweisung besteht immer aus einer Steueranweisung und einer oder mehreren zugehörigen Parameterbedingungen. Die Steuerbedingung enthält dabei die Bedingung, die erfüllt sein muss, damit die abhängigen Parameter erforderlich werden. Ich verwende in meinem Beispiel als Steuerbedingung lediglich den Parameternamen DATE und gebe damit bekannt, dass geprüft werden soll, ob in dem Parameter eine Eingabe erfolgt ist. Wenn eine Eingabe stattgefunden hat, muss auch der Parameter TIME einen Wert erhalten. Ist diese abhängige Bedingung nicht erfüllt, soll die Nachricht CMD0003 gesendet werden. Ich verwende hierfür eine spezifische Nachricht meiner Nachrichtendatei. Das System würde lediglich eine allgemeingültige, nichtssagende Nachricht senden. Die Anweisung DEP hat allerdings keine Klausel, um neben der Nachrichten-ID auch den zugehörigen Namen der Nachrichtendatei zu spezifizieren. Die zugehörige Nachrichtendatei müssen Sie zur Umwandlungszeit bekannt geben – also im Befehl CRTCMD:

```
                        Befehl erstellen (CRTCMD)

Auswahl eingeben und Eingabetaste drücken.

Gültig bei Modus . . . . . .   *ALL_____      *ALL, *PROD, *DEBUG, *SERVICE
     + für weitere Werte       _____
Ausführung wo zulässig . . . .  *ALL_____      *ALL, *BATCH, *INTERACT...
     + für weitere Werte       _____
Benutzer begrenzt zulassen . . .  *NO          *NO, *YES
Max. Positionsparameter  . . . .  *NOMAX       0-75, *NOMAX
Nachrichtendatei f. Bed.text:
  Nachrichtendatei . .  PMTFILE   *NONE_____   Nachrichtendatei für Bedienertext (F4-Taste)
    Bibliothek . . . . . .        _____
  Nachrichtentext  . . . . . .    _____
Nachrichtendatei . . .  MSGF      cmd_msgf__   Nachrichtendatei für DEP-Anweisung
  Bibliothek . . . . . .          *LIBL_____
```

Nachrichtendateien für DEP-Bedingungen

Achten Sie darauf, dass es zwei Einträge für Nachrichtendateien gibt. Für die abhängigen Parameter hinterlegen wir den Namen der entsprechenden Nachrichtendatei im Parameter MSGF des CRTCMD-Befehls. Zur Umwandlungszeit wird nicht geprüft, ob die Nachrichtendatei vorhanden ist. Der Zugriff auf die Nachrichtentexte erfolgt erst zur Ausführungszeit, so dass wir hier auch nicht zwischen statischen und dynamischen Zugriffen unterscheiden können, wie es bei den Bedienerführungstexten möglich war.

Da man allerdings schnell vergisst, diesen Parameter im CRTCMD-Befehl zu modifizieren, wenn das Command neu kompiliert wird, hinterlegen wir die Nachrichtendatei wieder im Quellcode. Diese Möglichkeit existiert allerdings erst seit kurzer Zeit.

```
                        Befehlsdefinition (CMD)

Auswahl eingeben und Eingabetaste drücken.

Marke  . . . . . .              _____
Bedienertext od. Nachricht.-ID   > 'SQL-Ausführung'_____
Nachrichtendatei f. Bed.text:
  Nachrichtendatei . . . . . .  _____      Name, *NONE
    Bibliothek . . . . . .        *LIBL____  Name, *LIBL, *CURLIB
  Nachrichtentext  . . . . . .  *STATIC__  *STATIC, *DYNAMIC
Nachrichtendatei . . . . . .    cl_MSG_____  Name
  Bibliothek . . . . . .          eps002cr__ Name, *LIBL, *CURLIB
```

Nachrichtendatei in der CMD-Anweisung hinterlegen

2.14.6.1
Seite 4

Schauen wir uns nun die Wirkung an:

```
                         Query ausführen (STRQRY)

Auswahl eingeben und Eingabetaste drücken.

Name des Querys  . . . . . . . . > ART_QTY01      Name
  Bibliothek . . . . . . . . . . >   EPS0042CR    Name, *LIBL
Ausführung am: . . . . . . . . . > '10.01.2011'   Datum
Ausführung um: . . . . . . . . .   __________     Zeit
                                                                Ende
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
Geben Sie ein Datum UND eine Uhrzeit für die Ausführung ein!
```

Wirkungsweise der DEP-Klauseln

Ich könnte mithilfe der DEP-Anweisungen aber auch komplexere Bedingungen formulieren. Stellen wir uns vor, dass die Querys nur noch zwischen 3 Uhr und 5 Uhr morgens ausgeführt werden dürfen. Diese Regelung gilt allerdings erst seit dem 01.01.2010. Ob die Bedingung sinnvoll ist, sei dahingestellt. Sie eignet sich aber, um die Definition komplexer Bedingungen zu erläutern:

```
                        Abhängige Definition (DEP)

Auswahl eingeben und Eingabetaste drücken.
Dem Parameter wird jetzt ein
&-Zeichen vorangestellt.
                    . . . . .   __________
Steuerbedingungen:
  Schlüsselwort oder -verweis  . > &DATE        Zeichenwert, *ALWAYS
  Vergleichsoperator . . . . . . > *GE          *GT, *EQ, *GE, *NL, *LT...
  Wert oder Schlüsselwortverweis > 1100101
Abhängiger Parameter:
  Schlüsselwort oder -verweis  . > &TIME        Zeichenwert
  Vergleichsoperator . . . . . . > *GE          *GT, *EQ, *GE, *NL, *LT...
  Wert oder Schlüsselwortverweis > 030000
                                   Es handelt sich immer um UND-Verknüpfungen!
  Schlüsselwort oder -verweis  . > &TIME        Zeichenwert
  Vergleichsoperator . . . . . . > *LE          *GT, *EQ, *GE, *NL, *LT...
  Wert oder Schlüsselwortverweis > 050000
     + für weitere Werte
```

Komplexe Abhängigkeiten formulieren

Die Abbildung zeigt eine komplex formulierte Abhängigkeit zwischen den Parametern DATE und TIME. Wenn ein Datum nach dem 01.01.2010 eingegeben wird, darf für die Uhrzeit nur noch ein Wert zwischen 03 und 05 Uhr verwendet werden. Bei der Bedingungsformulierung gilt es jedoch, einige Spezifika zu beachten:

- Wenn komplexe Bedingungen formuliert werden, muss dem Parameternamen das &-Zeichen vorangestellt werden.
- Wenn nur auf die Eingabe eines Parameters geprüft wird – wie im vorangegangenen Beispiel – darf das &-Zeichen dem Parameternamen nicht vorangestellt werden.
- Wenn Datums- und/oder Zeitfelder in den Bedingungen verwendet werden, darf die Konstante nicht in Hochkomma eingeschlossen werden. Datums- und Zeitfelder werden in diesem Kontext wie numerische Felder behandelt, was ein wenig gewöhnungsbedürftig ist.
- Sollen Datumsfelder mit einer Konstanten verglichen werden, muss die Konstante im Format CJJMMTT geschrieben werden. Andernfalls führen die Eingaben ggf. zu falschen Ergebnissen.

Und nun ist es an der Zeit, das Command zu prüfen:

```
                        Query ausführen (STRQRY)

Auswahl eingeben und Eingabetaste drücken.

Name des Querys  . . . . . > ART_QTY01     Name
  Bibliothek . . . . . . . . >   EPS0042CR   Name, *LIBL
Ausführung am: . . . . . . . > '10.01.2011'  Datum
Ausführung um: . . . . . . . > '22:00:00'    Zeit

                                                                  Ende
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
Seit dem 01.01.2010 werden Ihre Querys nur noch zwischen 3 und 5 Uhr morgens
```

Wirkungsweise komplexer DEP-Anweisungen

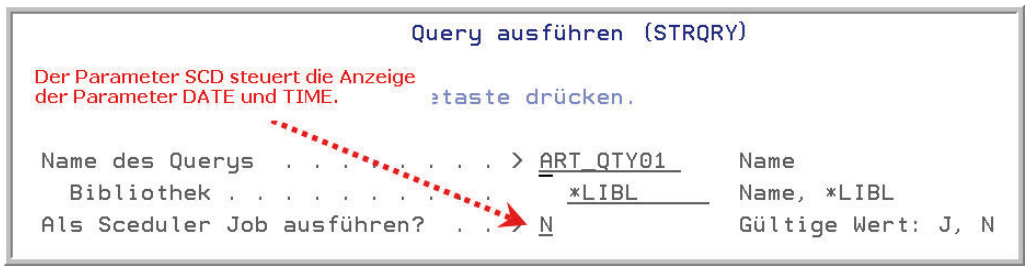
Wie erwartet, wird mein CL-Programm nicht aufgerufen, solange die Bedingungen für die Parameter DATE und TIME nicht erfüllt sind.

2.14.6.1

Seite 6

Die Anweisung PMTCTL

Wenn wir es zulassen wollen, dass die Parameter DATE und TIME für eine sofortige Ausführung des SQL-Skripts leer übergeben werden, ist auch eine weitere Lösung denkbar: Wir könnten im Command einen weiteren Parameter nutzen, dessen Inhalt darüber entscheiden soll, ob das Skript als Sceduler Job ausgeführt wird oder sofort übergeben werden soll.

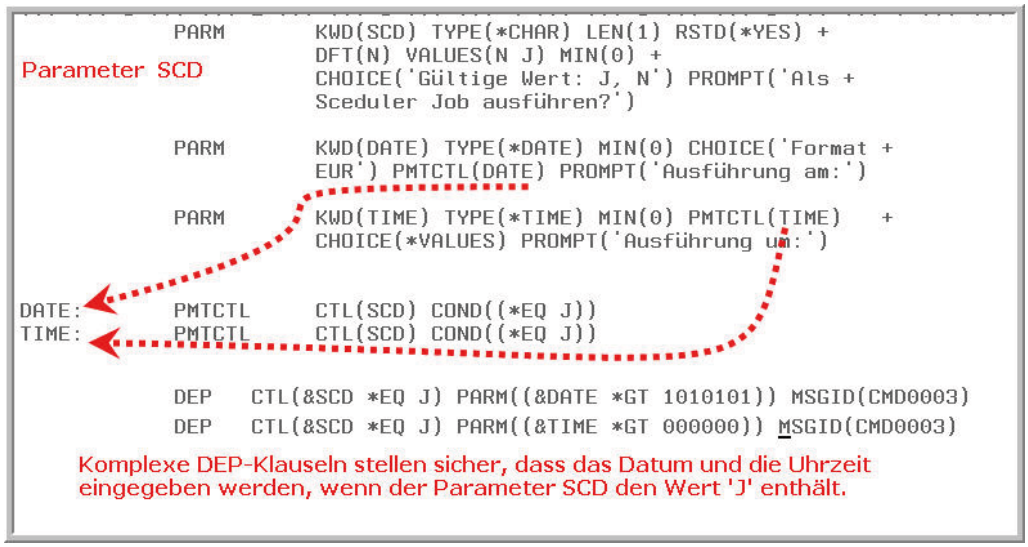


Der neue Parameter SCD

Der neue Parameter SCD soll jetzt die Anzeige der Parameter DATE und TIME steuern. Enthält der Parameter den Wert „J“, werden die beiden Parameter eingeblendet, andernfalls wird das CL-Programm direkt aufgerufen.

Wir können diesen Effekt mit der Klausel PMTCTL der PARM-Anweisung erreichen. Mit dieser Klausel können wir die Anzeige einzelner oder einer Gruppe von Parametern von Bedingungen abhängig machen.

Bevor wir das Beispiel in die Tat umsetzen, sollten Sie einen formalen Sachverhalt beachten: Es gibt die Klausel PMTCTL in der PARM-Anweisung, die wir bereits im Kapitel 2.14.4.2 genutzt haben und es gibt die Anweisung PMTCTL. Die Klausel PMTCTL der PARM-Anweisung kann auf ein Label verweisen, welches eine PMTCTL-Anweisung enthält:



Die Klausel PMTCTL verweist auf ein LABEL

2.14.6.2
Seite 2

Ich habe im Command einiges verändert. Der neue Parameter SCD wurde deklariert. Anschließend wurden die PARM-Anweisungen der Parameter DATE und TIME um das Schlüsselwort PMTCTL ergänzt. Bei dem Parameter DATE können Sie sehen, dass die Klausel PMTCTL(DATE) auf das Label DATE verweist. Sie müssen keine übereinstimmenden Parameter- und Label-Namen verwenden; ich finde allerdings, dass es die Arbeit vereinfacht und das Command übersichtlicher gestaltet.

Hinter dem jeweiligen Label nutze ich die Anweisung PMTCTL. Sie bestimmt, dass der Parameter SCD die Anzeige der Felder DATE und TIME in der Bedienerführung steuert:

```
                    Steuerdefinition anfordern (PMTCTL)

Auswahl eingeben und Eingabetaste drücken.

Marke . . . . . > DATE
Steuerschlüsselwort . . . . . > SCD        Name
Steuerbedingungen:                _
  Vergleichsoperator . . . . . > *EQ       *GT, *EQ, *GE, *NL, *LT..
  Wert d. Steuerschlüsselworts  > J
     + für weitere Werte        _
Anzahl wahrer Bedingungen:
  Vergleichsoperator . . . . .   *ALL       *ALL, *GT, *EQ, *GE, *NL.
  Anzahl wahrer Bedingungen . .  _____      0-25
Logischer Vergleich . . . . .   *AND       *AND, *OR
```

Bedienerführung der Anweisung PMTCTL

Ich musste als Steuerbedingung vereinbaren, dass der Parameter DATE angezeigt werden soll, wenn der Parameter SCD den Wert ‚J' enthält. Eine identische Bedingung formuliere ich mit der Anweisung PMTCTL für den Parameter TIME. Anschließend ergänze ich die Command-Definition um zwei komplexe DEP-Klauseln, damit die Parameter DATE und TIME nicht leer an das CPP-Programm übergeben werden, wenn der Parameter SCD den Wert ‚J' enthält.

Auswahlsteuerung – CHOICEPGM

Nehmen wir unseren letzten Command als Beispiel, um die Wirkung von Auswahlprogrammen zu erläutern. Der Anwender kann im Command den Namen eines Querys eintragen; vorausgesetzt, er kennt den korrekten Namen. Besser wäre es daher, wenn wir dem Anwender die Möglichkeit bieten, die Taste F4 zu nutzen, um sich die zur Verfügung stehenden Querys anzeigen zu lassen:

```
        Wert für Parameter angeben                QUERY
Die Taste F4 wurde gedrückt, um die zur Verfügung
stehenden Querys anzuzeigen.

Art  . . . . . . . . . . . . . . . :   EINFACHER NAME
NAME des Querys  . . . . . . . . .    _____
                                      =

ART_QTY01
ART_QTY02
CUSTOMER
S_KD_LISTE
STOCKLS
TESTBSP01
TESTBSP02
TESTBSP03
TEST00
TEST2010_6
```

Ein Auswahlprogramm stellt die Werte zur Verfügung.

Wirkungsweise eines Auswahlprogramms

In unserem Command müssen wir nur wenige Einstellungen ändern:

```
     CMD        PROMPT('Query ausführen') MSGF(ITPCR/CMD_MSGF)

     PARM      KWD(QUERY) TYPE(*SNAME) LEN(10)  MIN(1) +
                 CHOICE(*PGM)   CHOICEPGM(ITPCR/CHOICEPGM)        +
                 PROMPT('NAME des Querys')


     PARM       KWD(SCD) TYPE(*CHAR) LEN(1) RSTD(*YES) +
                DFT(N) VALUES(N J) MIN(0) +
                CHOICE('Gültige Wert: J, N') PROMPT('Als +
                Sceduler Job ausführen?')
```

Die Klausel CHOICEPGM

2.14.7
Seite 2

Ich habe dieses Mal auf einen qualifizierten Objektnamen verzichtet. Vielmehr möchte ich zur Ausführungszeit des Commands die zur Verfügung stehenden Objekte anhand des Benutzernamens ermitteln und als ausführbare Objekte im Command anzeigen. Dabei gilt für dieses Beispiel die Restriktion, dass jeder Anwender in einer separaten Bibliothek seine Querys ablegt. Der Bibliotheksname ist identisch mit dem Benutzernamen. Diese Restriktion ist natürlich nicht zwingend, letztendlich können Sie die Auswahlwerte anhand eigener Kriterien individuell zusammenstellen.

Alles, was wir in der Befehlsdefinition beachten müssen, ist, die Verwendung CHOICE(*PGM) für den entsprechenden Parameter zu hinterlegen. Zusätzlich muss mit der Klausel CHOICEPGM ein Programm als Auswahlprogramm für den Parameter benannt werden. In meinem Beispiel wird das Programm CHOICEPGM in der Bibliothek ITPCR aufgerufen, wenn der Anwender für den Parameter QUERY die F4-Taste drückt.

Schauen wir uns jetzt das Auswahlprogramm CHOICEPGM an. Das Auswahlprogramm CHOICEPGM tauscht mit dem Command zwei Parameter aus:

```
PGM PARM(&CMDPARM &STRUK)
```

Der erste Parameter – in meinem Beispiel die Variable &CMDPARM – ist eine 21-stellige Zeichenkette, die den Namen des Command, den Namen des Parameters und ein Kennzeichen enthält. Ich kann diesen Parameter durch eine Datenstruktur folgendermaßen abbilden:

```
DCL &CMDPARM *CHAR 21

    DCL VAR(&CMD) TYPE(*CHAR) STG(*DEFINED) LEN(10)
        DEFVAR(&CMDPARM 1)
    DCL VAR(&PARM) TYPE(*CHAR) STG(*DEFINED) LEN(10)
        DEFVAR(&CMDPARM 11)
    DCL VAR(&P_C) TYPE(*CHAR) STG(*DEFINED) LEN(1)
        DEFVAR(&CMDPARM 21)
```

Die Variable &P_C enthält den Buchstaben C, wenn das Command einen einfachen Auswahltext benötigt; der Anwender hat also nicht die F4-Taste gedrückt. Das Command fordert in diesem Fall von dem CL-Programm eine 30-stellige Zeichenkette an, wie sie auch das System erstellen würde:

```
IF (&P_C = 'C') THEN(CHGVAR &VALUE VALUE('Name des Querys oder
F4-Taste'))
```

Enthält aber die Variable &P_C den Buchstaben P, muss eine Zeichenkette erstellt werden, die alle gültigen Auswahlwerte enthält. Der Buchstabe P zeigt somit an, dass die Funktionstaste F4 innerhalb des Commands vom Anwender gedrückt wurde, um eine Auswahl der zur Verfügung stehenden Querys angezeigt zu bekommen.

Die Übergabe der Querynamen – also der Auswahlwerte – an den Command erfolgt als Datenstruktur. Eigentlich handelt es sich hierbei eher um ein Array, was die ganze Sache nicht gerade vereinfacht. Da CL bislang keine Arrays kennt, müssen wir ein Array simulieren. Dafür definieren wir zunächst einmal eine Datenstruktur. Ich nenne diese Struktur &STRUK:

```
DCL &STRUK *CHAR 250
```

Die Länge der Datenstruktur ist davon abhängig, wie viele Werte angezeigt werden sollen und kann daher beliebig variieren. Das erste Feld der Datenstruktur ist ein binärer Zähler der Länge zwei.

Dieser Zähler – in meinem Beispiel die Variable &Z – enthält die Anzahl der zusammengestellten Auswahlwerte:

```
DCL VAR(&Z) TYPE(*INT) STG(*DEFINED) LEN(2) DEFVAR(&STRUK) --
Anzahl der Vorkommen
DCL VAR(&VALUE) TYPE(*CHAR) STG(*DEFINED) LEN(30) DEFVAR(&STRUK 5)
```

Innerhalb der Datenstruktur muss es ein Feld geben, welches den einfachen Auswahltext enthält. Also den Text, der angezeigt werden soll, solange der Anwender nicht die Funktionstaste F4 drückt. Anschließend folgen die Elemente, die gefüllt werden, mit den zur Verfügung stehenden Auswahlwerten:

```
/*1. Wert */
     DCL VAR(&L1) TYPE(*INT) STG(*DEFINED) LEN(2) DEFVAR(&STRUK 3)
     DCL VAR(&VALUE1) TYPE(*CHAR) STG(*DEFINED) LEN(10)
         DEFVAR(&STRUK 5)
```

2.14.7
Seite 4

Für jeden gültigen Auswahlwert benötigen Sie zwei Informationen: die Länge des Auswahltextes als 2 Byte langer Binärwert und den eigentlichen Text. Da wir bislang im CL-Programm nicht mit Tabellen arbeiten können, müssen wir diese Deklarationen für jedes Vorkommen wiederholen, d. h. wenn Sie maximal 20 Auswahlwerte anzeigen wollen, benötigen Sie 20 Elemente in der Datenstruktur. Ich beschränke mich in dem Beispiel auf maximal 10 Einträge und definiere daher die Variablen &L1 und &VALUE1 bis &L10 und &VALUE10 nach folgendem Muster:

```
/*2. Wert */
   DCL VAR(&L2) TYPE(*INT) STG(*DEFINED) LEN(2) DEFVAR(&STRUK 15)
   DCL VAR(&VALUE2) TYPE(*CHAR) STG(*DEFINED) LEN(10)
       DEFVAR(&STRUK 17)

/*3. Wert */
   DCL VAR(&L3) TYPE(*INT) STG(*DEFINED) LEN(2) DEFVAR(&STRUK 27)
   DCL VAR(&VALUE2) TYPE(*CHAR) STG(*DEFINED) LEN(10)
       DEFVAR(&STRUK 29)
… …

/* 10. Vorkommen */
   DCL VAR(&L10) TYPE(*INT) STG(*DEFINED) LEN(2) DEFVAR(&STRUK 111)
   DCL VAR(&VALUE10) TYPE(*CHAR) STG(*DEFINED) LEN(10)
       DEFVAR(&STRUK 113)
```

Im Programm müssen Sie zunächst einmal feststellen, ob die Taste F4 gedrückt wurde oder nicht:

```
IF (&P_C = 'C') THEN(DO)
              CHGVAR &VALUE VALUE('Name eines Querys oder F4')
              CHGVAR &Z VALUE(0)
              ENDDO
```

Dies gelingt anhand der Variablen &P_C. Enthält die Variable den Buchstaben C, benötigen wir lediglich den Standardtext. Wenn die Variable den Wert P enthält, müssen wir hingegen die entsprechenden Auswahlwerte zusammenstellen:

```
IF (&P_C = 'P') THEN(CALLSUBR F4)
```

In der Unterroutine F4 muss ich mir anschließend überlegen, wie die Datenstruktur gefüllt werden kann. Ich habe für das Beispiel eine sehr einfache Lösung gewählt, die folgendermaßen aussieht:

```
SUBR F4
       CHGVAR &STRUK VALUE(' ')
       CLRPFM CHOICE
       RTVJOBA USER(&USER)
       DSPOBJD OBJ(&USER/*ALL) OBJTYPE(*QMQRY) OUTPUT(*OUTFILE) +
                 OUTFILE(ITPCR/CHOICE)
```

In einem ersten Schritt initialisiere ich die Datenstruktur, ermittle den Benutzernamen und stelle die Objektnamen in die physische Datei CHOICE. Voraussetzung für dieses Vorgehen ist die bereits erwähnte Restriktion Benutzername gleich Bibliotheksname.

Anschließend kann ich die physische Datei lesen und dabei meine Datenstruktur füllen:

```
LOOP:   RCVF
             MONMSG CPF0864 EXEC(RETURN)

             SELECT
             WHEN (&VALUE1 = ' ') THEN(DO)
                CHGVAR &L1 VALUE(10)
                CHGVAR &VALUE1 VALUE(&ODOBNM)
                CHGVAR &Z VALUE(1)
             ENDDO

             WHEN (&VALUE2 = ' ') THEN(DO)
                CHGVAR &L2 VALUE(10)
                CHGVAR &VALUE2 VALUE(&ODOBNM)
                CHGVAR &Z VALUE(2)
             ENDDO
… … …

            WHEN (&VALUE10 = ' ') THEN(DO)
               CHGVAR &L10 VALUE(10)
               CHGVAR &VALUE10 VALUE(&ODOBNM)
               CHGVAR &Z VALUE(10)
            ENDDO
       ENDSELECT

       GOTO LOOP
ENDSUBR
```

Wie viele Elemente gefüllt werden und ob Sie mit Datenstrukturen oder einfachen Zeichenketten sowie dem Verkettungsoperator *TCAT arbeiten, bleibt Ihnen überlassen.

Ist der Parameter entsprechend mit Werten versorgt, wird das Auswahlprogramm einfach über die Anweisung ENDPGM oder RETURN verlassen und das Command zeigt – wie zuvor bereits gezeigt – die zusammengestellten Auswahlwerte an.

Listen verwenden

Erinnern Sie sich noch an das API QCLRDTAQ und an unser CPP-Programm CLRDTAQ?

```
PGM PARM(&DATAQ &LIB)
        DCL VAR(&DATAQ) TYPE(*CHAR) LEN(10)
        DCL VAR(&LIB) TYPE(*CHAR) LEN(10)

        CALL PGM(QCLRDTAQ) PARM(&DATAQ &LIB )
             MONMSG     MSGID(CPF9801) EXEC(DO)
             SNDPGMMSG    MSG('Datenwarteschlange' *BCAT &LIB *TCAT '/' +
                          *CAT &DATAQ *BCAT 'wurde nicht gefunden!')
             RETURN
             ENDDO
```

Das CPP-Programm CLRDTAQ

Das CPP-Programm ruft das API QCLRDTAQ auf, welches anschließend den Inhalt einer beliebigen Datenwarteschlange löscht. Dafür wird einerseits der Name der Datenwarteschlange, andererseits die Bibliothek, die das Objekt enthält, angefordert.

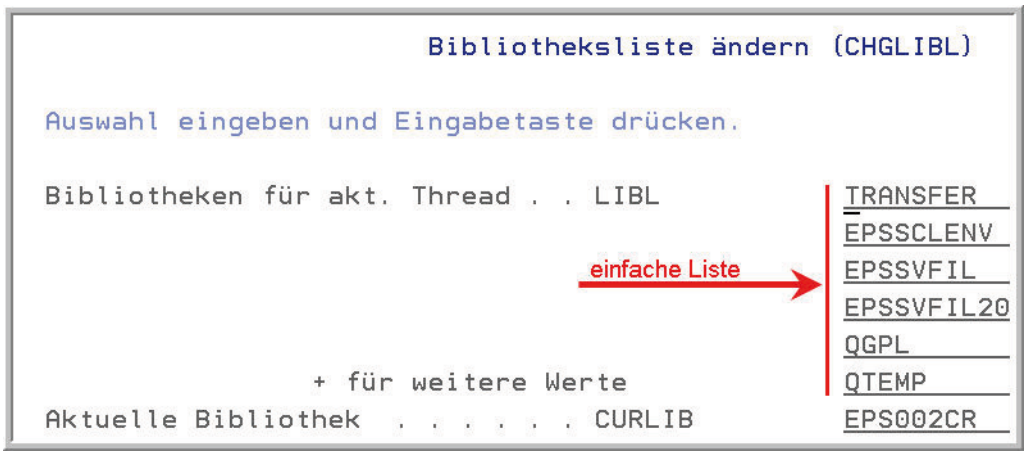
Dieses Programm möchte ich mit Hilfe eines entsprechenden Commands dahingehend ändern, dass das CL-Programm eine Liste mit bis zu 10 Objektnamen entgegennimmt und alle übergebenen Datenwarteschlangen bereinigt.

Für diese Aufgabenstellung bieten sich Listen an. Wir unterscheiden dabei zwischen einfachen, gemischten und verschachtelten Listen.

2.14.8
Seite 2

Einfache Listen

Eine einfache Liste ist nichts weiter als eine Folge von gleichartigen Einzelwerten. Der Befehl CHGLIBL wird gern verwendet, um Aussehen und Funktion einfacher Listen zu veranschaulichen:



Beispiel einer einfachen Liste

In dem Parameter LIBL stehen nur gleichartige Einzelwerte. Ein Pluszeichen am Ende der Liste weist darauf hin, dass zusätzliche Werte eingegeben werden können. Wenn Sie in einer Liste Werte an einer bestimmten Position einfügen wollen, müssen Sie an dieser Listenposition ein Größerzeichen (>) eingeben, den Rest des Feldes löschen und die Eingabetaste drücken. Der überschriebene Wert und die nachfolgenden Elemente der Liste werden daraufhin weiter nach hinten verschoben und Sie haben Platz, um einen neuen Wert einzufügen.

Nehmen wir unser CL-Programm CLRDTAQ als Ausgangsbasis, um einfache Listparameter zu empfangen.

Schauen wir uns aber zuvor das fertige Command in der Ausführung an und anschließend die zugehörigen Befehlsdefinitionen:

Das Command CLRDTAQ

2.14.8.1
Seite 2

Das Command übergibt zwei Parameter, die Namen der Datenwarteschlangen und den Sonderwert *LIBL, um die Objekte zu lokalisieren. Wir definieren wie gewohnt eine Überschrift und anschließend den Listparameter. Dies geschieht, indem ich neben dem MIN-Wert auch einen entsprechend hohen MAX-Wert festlege:

```
PARM   KWD(DATAQ) TYPE(*SNAME) LEN(10) MIN(1) MAX(10)
        CHOICE(*VALUES) + PROMPT('Name einer Datenwarteschlange')
```

Die Klausel MAX(10) bestimmt in meiner Befehlsdefinition also, dass ich bis zu 10 Datenwarteschlangennamen eintragen kann.

Abschließend folgt der Parameter LIB. Ich verzichte fürs Erste auf die Verwendung eines qualifizierten Namens; werde dies aber später entsprechend ändern:

```
PARM KWD(LIB) TYPE(*SNAME) LEN(10) DFT(*LIBL) SPCVAL((*LIBL))
CHOICE(*VALUES) +
PROMPT('Bibliothek')
```

```
 PARM       KWD(LIB) TYPE(*SNAME) LEN(10) DFT(*LIBL)
               SPCVAL((*LIBL)) CHOICE(*VALUES) +
               PROMPT('Bibliothek')
```

Wenn das Command umgewandelt ist, bleibt die Frage zu klären, wie das CL-Programm die Liste empfangen muss. Die Befehlsdefinition übergibt die Liste als einen einzigen Parameter, der allerdings genau 2 Byte länger ist als die Summe aller Elemente.

*Listenlänge = Maximal zulässige Werte * Elementlänge + 2*

Da ich in einer Befehlsdefinition zehn Elemente à 10 Byte definiert habe, ist die Liste 102 Byte lang. Daraus resultiert die entsprechende Variablendeklaration im CL-Programm:

```
PGM PARM(&LISTE &LIB)
          DCL VAR(&LISTE) TYPE(*CHAR) LEN(102)
```

Die beiden zusätzlichen Bytes unseres Parameters &LISTE enthalten eine Binärzahl, die die Anzahl der belegten Listenelemente an das CPP übergibt. Das erste Datenelement der Liste beginnt somit immer an der dritten Listenposition:

```
DCL VAR(&DATAQ) TYPE(*CHAR) LEN(10)
  DCL &IND *INT 2
  DCL &POS *INT 2  VALUE(3)


DOFOR  VAR(&IND) FROM(1) TO(&ANZ)
         CHGVAR &DATAQ  VALUE(%SST(&LISTE &POS 10))
         CALL PGM(QCLRDTAQ) PARM(&DATAQ &LIB)
         CHGVAR &POS VALUE(&POS + 10)
ENDDO
```

Da wir jetzt mit Datenstrukturen und Binärzahlen im CL-Programm arbeiten können, ist der Zugriff auf den „Zähler“ und die eigentlichen Daten einfacher geworden. Leider können wir aber nach wie vor keine Tabellen definieren, so dass uns nichts anderes übrig bleibt, als die einzelnen Einträge innerhalb einer Schleife zu ermitteln. Dafür sind natürlich einige zusätzliche Variablen notwendig. Doch sowohl die DCL-Anweisungen als auch die Schleifenanweisung sind Ihnen aus den vorangegangenen Kapiteln hinlänglich bekannt und müssen daher nicht weiter erläutert werden:

```
  DCL VAR(&DATAQ) TYPE(*CHAR) LEN(10)
  DCL &IND *INT 2
  DCL &POS *INT 2 VALUE(3)

DOFOR VAR(&IND) FROM(1) TO(&ANZ)
     CHGVAR &DATAQ VALUE(%SST(&LISTE &POS 10))
     CALL PGM(QCLRDTAQ) PARM(&DATAQ &LIB)
     CHGVAR &POS VALUE(&POS + 10)
ENDDO
```

2.14.8.1
Seite 4

Natürlich können diese einfachen Listen auch mit unseren qualifizierten Objektnamen aufgebaut werden. Alles, was wir dafür tun müssen, ist die Anweisung QUAL und die Klausel MAX zu kombinieren:

```
        CMD         PROMPT(CMD0002) PMTFILE(EPS002CR/CL_MSG +
                      *STATIC)

        PARM      KWD(DATAQ) TYPE(Q1) MIN(1) MAX(10) +
                  CHOICE(*VALUES) PROMPT('Name einer Datenwarteschlange')

 Q1:    QUAL TYPE(*SNAME) LEN(10)

        QUAL TYPE(*SNAME) LEN(10) DFT(*LIBL) +
             SPCVAL((*LIBL)) CHOICE(*VALUES) +
             PROMPT('Bibliothek')
```

Qualifizierte Objektnamen in einfachen Listen

Das CL-Programm muss natürlich auch ein wenig verändert werden:

```
PGM PARM(&LISTE)
    DCL VAR(&LISTE) TYPE(*CHAR) LEN(202)
        DCL VAR(&ANZ) TYPE(*INT) STG(*DEFINED) LEN(2)  DEFVAR(&LISTE)
        DCL VAR(&DATEN) TYPE(*CHAR) STG(*DEFINED) LEN(100) DEFVAR(&LISTE 3)

    DCL VAR(&QUALOBJ) TYPE(*CHAR) LEN(20)
        DCL VAR(&DATAQ) TYPE(*CHAR) LEN(10) STG(*DEFINED) DEFVAR(&QUALOBJ)
        DCL VAR(&LIB) TYPE(*CHAR) LEN(10) STG(*DEFINED) DEFVAR(&QUALOBJ 11)

    DCL &IND *INT 2
    DCL &POS *INT 2  VALUE(3)

    DOFOR  VAR(&IND) FROM(1) TO(&ANZ)
           CHGVAR &QUALOBJ  VALUE(%SST(&LISTE &POS 20))
           CALL PGM(QCLRDTAQ) PARM(&DATAQ &LIB)
           CHGVAR &POS VALUE(&POS + 20)
    ENDDO
ENDPGM
```

Das modifizierte CL-Programm CLRDTAQ

Die Verarbeitungslogik im CL-Programm bleibt aber die gleiche. Lediglich die Eingangsparameter und die Längenangaben müssen ein wenig geändert werden, damit die qualifizierten Objektnamen korrekt verarbeitet werden können.

Gemischte Listen

Gemischte Listen enthalten im Gegensatz zu einfachen Listen verschiedene Elemente mit unterschiedlichen Attributen. Sie kennen gemischte Listen bereits aus Ihrer täglichen Arbeit. Viele Systembefehle verwenden gemischte Listen:

```
Größe der Teildatei:              SIZE
  Anfangsanzahl Sätze . . . . .          10000
  Satzanzahl für Erweiterung . .         1000
  Maximale Erweiterungen . . . .         3
```

Beispiel einer gemischten Liste

Der Parameter SIZE des Befehls CHGPF (Change Physical File) ist ein solches Beispiel. Der Size-Parameter enthält drei eigenständige Elemente, die unterschiedliche Attribute bzw. unterschiedliche Längendefinitionen haben. Solche gemischten Listen sind schnell definiert. Wir können unser Command STRQRY gut als Beispiel für gemischte Listen nutzen, indem wir das Ausführungsdatum, die Uhrzeit und die Kopienanzahl als gemischte Liste aufbauen. Folgende Grafik zeigt, was gemeint ist:

```
                        Query ausführen (STRQRY)

Auswahl eingeben und Eingabetaste drücken.

Name des Querys . . . . . .          Name eines Querys oder F4
Ausgabeoptionen:
  Ausführen am:  . . . . . .         FORMAT: *EUR
  Ausführen um:  . . . . . .         FORMAT: *EUR
  Kopienanzahl . . . . . .           Ganze Zahl
```

Das Command STRQRY als gemischte Liste

Für die Definition gemischter Listen muss zunächst der Parameter Optionen definiert werden. Der Bedienerführungstext wird als „Überschrift“ verwendet. Die zugeordneten Elemente werden darunter aufgeführt.

```
PARM KWD(OPTIONEN) TYPE(E1) PROMPT('Ausgabeoptionen')
```

2.14.8.2
Seite 2

Sie können im Quellcode erkennen, dass wir auch bei gemischten Listen mit einem Label arbeiten, das ich im Beispiel E1 nenne. Im Unterschied zu den qualifizierten Namen wird jetzt die Anweisung ELEM verwendet, um die verschiedenen Elemente der gemischten Liste zu beschreiben:

```
E1:  ELEM    TYPE(*DATE)            CHOICE('FORMAT: *EUR')
      PROMPT('Ausführen am:')
     ELEM    TYPE(*TIME)            CHOICE('FORMAT: *EUR')
       PROMPT('Ausführen um:')
     ELEM    TYPE(*DEC)    LEN(2 0) CHOICE('Ganze Zahl')
       PROMPT('Kopienanzahl')
```

Weitere Definitionen sind nicht notwendig, so dass wir uns jetzt dem CPP-Programm zuwenden können. Das CL-Programm übernimmt diese gemischte Liste in einer einzigen Variablen. Die Variable muss die Elemente und einen binären Präfix speichern. Dieses Präfix enthält die Anzahl der Elemente. In unserem Beispiel also den Wert 3.

Um die Länge der Datenstruktur zu ermitteln, die wir im CPP-Programm deklarieren müssen, addieren wir die einzelnen Elementlängen: Im Beispiel enthält die gemischte Liste ein Datumsfeld, das 7 Byte benötigt, die Uhrzeit mit 6 Byte Länge und den Parameter Kopienanzahl als 2 Byte lange Dezimalzahl. Die Daten benötigen also 15 Byte. Hinzu kommt das Präfix mit 2 Byte. Die Datenstruktur &OPTION ist daher 17 Byte lang und enthält entsprechend meiner Command-Definition die einzelnen Elemente der gemischten Liste.

```
PGM  PARM(&QUERY &OPTION)
     DCL  &QUERY *CHAR  10

     DCL VAR(&OPTION)  TYPE(*CHAR) LEN(17)
           DCL         VAR(&ANZOPT) TYPE(*INT) STG(*DEFINED) LEN(2) +
                         DEFVAR(&OPTION 1)
           DCL         VAR(&DATE) TYPE(*CHAR) STG(*DEFINED) LEN(7) +
                         DEFVAR(&OPTION 3)
           DCL         VAR(&TIME) TYPE(*CHAR) STG(*DEFINED) LEN(6) +
                         DEFVAR(&OPTION 10)
           DCL         VAR(&COPY) TYPE(*DEC) STG(*DEFINED) LEN(2 0) +
                         DEFVAR(&OPTION 16)
```

Die Datenstruktur im CPP STRQRY

Sicherlich könnte man die gemischte Liste noch um viele Ausgabeoptionen erweitern und auch das Programm könnten wir um entsprechende Fehlerroutinen oder Prüfroutinen ergänzen. Doch all diese Dinge würden an der Verarbeitung der gemischten Liste nichts ändern!

Verschachtelte Listen

Verschachtelte Listen sind eine weitere „Spielvariante“ der Listen. Eine verschachtelte Liste enthält wie eine einfache Liste eine bestimmte Anzahl gleichartiger Elemente, wobei diese Elemente als gemischte Liste definiert werden. Wir haben es also im Grunde genommen mit einer Tabellenstruktur zu tun. Auch wenn es sich kompliziert anhört, sind verschachtelte Listen nicht schwer zu codieren, da es sich ja um eine Kombination bereits bekannter Definitionen und Verarbeitungsalgorithmen handelt. Nehmen wir das vorangegangene Beispiel als Basis für eine verschachtelte Liste:

```
         PARM KWD(OPTIONEN) TYPE(E1) MIN(1) MAX(10) PROMPT('Ausgabeoptionen')

E1:      ELEM    TYPE(*SNAME) LEN(10) MIN(1) PROMPT('SQL-Skript')
         ELEM    TYPE(*DATE) CHOICE('FORMAT: *EUR') PROMPT('AUSFÜHREN AM:')
         ELEM    TYPE(*TIME) CHOICE('FORMAT: *EUR') PROMPT('Ausführen um:')
        Der Element-Block kann 10-mal wiederholt werden.
```

Definition verschachtelter Listen

Alles, was wir tun müssen ist, die Klausel MAX in der PARM-Anweisung unserer Ausgabeoptionen zu verwenden.

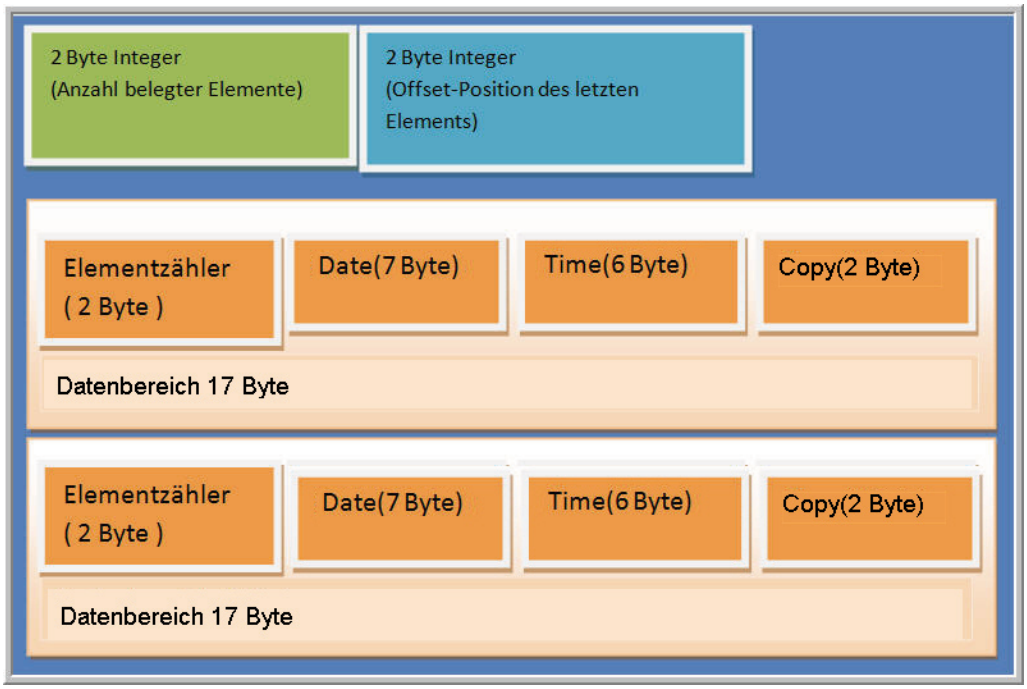
```
                        Query ausführen (STRQRY_V4)

Auswahl eingeben und Eingabetaste drücken.

Name des Querys . . . . . > BEISPIEL01        Name eines Querys oder F4
Ausgabeoptionen:                _
  Ausführen am:  . . . . . > '10.01.2011'  FORMAT: *EUR
  Ausführen um:  . . . . . > '10:00:00'    FORMAT: *EUR
  Kopienanzahl . . . . . > 1___            Ganze Zahl
                                =
  Ausführen am:  . . . . . > '10.02.2011'  FORMAT: *EUR   verschachtelte
  Ausführen um:  . . . . . > '10:00:00'    FORMAT: *EUR   Liste
  Kopienanzahl . . . . . > 1___            Ganze Zahl
                                _
  Ausführen am:  . . . . . > '10.03.2011'  FORMAT: *EUR
  Ausführen um:  . . . . . > '10:00:00'    FORMAT: *EUR
  Kopienanzahl . . . . . > 1___            Ganze Zahl
     + für weitere Werte        _
```

Wirkung verschachtelter Listen

Wir können jetzt mit unserem Command bis zu 10 Ausführungszeitpunkte für das Query hinterlegen.

2.14.8.3
Seite 2

Nun müssen wir nur noch unser CL-Programm ein wenig ändern. Zunächst müssen wir den Aufbau der verschachtelten Liste verstehen. Sehen Sie sich hierfür die folgende Grafik an:



Struktur einer verschachtelten Liste

Der erste Zähler enthält die Anzahl gefüllter Vorkommen. Es handelt sich um eine 2 Byte lange Integerzahl. Das zweite Feld enthält die Anfangsposition des letzten Datenelements.

Es folgen die einzelnen Elemente in Form einer gemischten Liste, deren Aufbau Ihnen bereits bekannt ist.

Wir müssen unser CL-Programm also ein wenig verändern, um die Daten zu verarbeiten:

```
DCL &OPTION *CHAR 174
  DCL VAR(&ANZELEM) TYPE(*INT) STG(*DEFINED) LEN(2) DEFVAR(&OPTION)
  DCL VAR(&POS) TYPE(*INT) STG(*DEFINED) LEN(2) DEFVAR(&OPTION 3)
```

Die Datenstruktur nenne ich wieder &OPTION. Sie enthält die Felder &ANZELEM und &POS. Das Feld &ANZELEM benötige ich, um die Schleife im Programm zu steuern. Das Feld &POS liefert mir die Position des letzten Datenelements. Für die Datenelemente selbst verwende ich die bekannte Struktur aus den vorangegangenen Beispielen; nur nenne ich die Struktur jetzt &DATEN:

```
DCL VAR(&DATEN) TYPE(*CHAR) LEN(17)
 DCL VAR(&ANZOPT) TYPE(*INT) STG(*DEFINED) LEN(2) DEFVAR(&DATEN 1)
 DCL VAR(&DATE) TYPE(*CHAR) STG(*DEFINED) LEN(7)  DEFVAR(&DATEN 3)
 DCL VAR(&TIME) TYPE(*CHAR) STG(*DEFINED) LEN(6)  DEFVAR(&DATEN 10)
 DCL VAR(&COPY) TYPE(*DEC) STG(*DEFINED) LEN(2 0) DEFVAR(&DATEN 16)
```

Letztlich brauche ich wieder einen Index zur Kontrolle der Schleifendurchläufe.

```
DCL VAR(&I)   TYPE(*UINT) LEN(2)
```

Außerdem benötigen wir eine Variable, die die Startposition der verschiedenen Elemente enthält. Die Variable ist bereits vorhanden, denn das Feld &POS enthält bei der Parameterübergabe bereits die Anfangsadresse des letzten Datenelements. Allerdings muss ich diesen Wert um den Wert 1 erhöhen, da wir im CL-Programm nicht mit dem Offset-Wert 0, sondern mit dem Wert 1 starten.

```
CHGVAR &POS VALUE(&POS + 1)
```

Sind die Datenstrukturen mit ihren Startpositionen und Längendefinitionen erst einmal korrekt definiert, ist die weitere Verarbeitung kein Problem mehr:

```
   DOFOR      VAR(&I) FROM(1) TO(&ANZELEM)

         CHGVAR &DATEN VALUE(%SST(&OPTION &POS 17))

 /* hier folgt die Verarbeitung der Daten               */

          CHGVAR &POS VALUE(&POS - 17)
     ENDDO

 ENDPGM
```

Die Zählschleife steuert die Verarbeitung der Datenelemente und der anschließende Übertragungsbefehl versorgt die Datenstruktur &DATEN mit entsprechenden Inhalten. Ist das erste Element verarbeitet, muss vor der Verarbeitung des nächsten Elements nur noch die Offset-Position um die Länge eines Datenelements reduziert werden.

2.14.8.3
Seite 4

Es gäbe sicherlich noch viele Beispiele für weitere Befehlsdefinitionen und viele gute und sinnvolle Anwendungen. Allerdings glaube ich auch, dass Sie selber sehr schnell eigene Ideen entwickeln werden, wenn Sie erst einmal beginnen, eigene Commands zu erstellen.

Application Programing Interface (API)

API – ein Begriff, der in der Welt der Programmierung (nicht nur in der IBM-Welt) immer wieder auftaucht. Fast jeder hat ihn schon einmal gehört, viele wissen jedoch nicht genau, was sich dahinter verbirgt. API ist die Kurzform für Application Programming Interface. Der Begriff wurde schon vor langer Zeit von der IBM eingeführt und bedeutet nichts anderes als – ursprünglich – von der IBM zur Verfügung gestellte Programme, die aus eigenen Programmen heraus aufgerufen werden können. Mit jedem neuen Release steigt die Anzahl der verfügbaren APIs. Heute können wir auf weit über 1.000 dokumentierte APIs zugreifen.

Vielleicht stellen Sie sich jetzt die Frage, warum Sie APIs einsetzen sollten. Dafür gibt es viele Gründe:

– APIs übernehmen beispielsweise Aufgaben, die ohne Maschinen-Interface-Anweisungen nicht zu realisieren sind. So ist es beispielsweise nur mit Hilfe der APIs QSNDDTAQ (Send Data Queue) und QRCVDTAQ (Receive Data Queue) möglich, Informationen in eine Datenwarteschlange zu schreiben oder deren Inhalte zu lesen.

– Aber es gibt auch APIs, die die gleichen Ergebnisse wie CL-Befehle liefern, beispielsweise das API QMHSNDPM (Send Program Message). Jedoch stellen viele dieser APIs – verglichen mit den entsprechenden CL-Befehlen – detailliertere Informationen zur Verfügung, wie beispielsweise das API QUSROBJD (Retrieve Object Description).

– Außerdem sind APIs häufig schneller in der Ausführung, insbesondere wenn die Informationen in einem HLL-Programm verfügbar sein sollen. Der Aufruf eines entsprechenden CL-Programms dauert länger als die Ausführung des entsprechenden APIs, weil die meisten APIs in C geschrieben wurden.

– Hinzu kommt, dass viele CL-Befehle das Ergebnis entweder auf dem Bildschirm oder in eine physische Datei ausgegeben. Auf die Ergebnisdatei kann anschließend zwar problemlos zugegriffen werden, doch das Erstellen der physischen Datei, das Schreiben auf die Platte und das anschließende Lesen kosten Zeit. APIs schreiben die Ergebnisse entweder in eine Empfangsvariable oder geben die Daten in einen Anwenderbereich, einen so genannten User Space aus.

– Bei der Verwendung von CL-Befehlen und der Ausgabe in Dateien besteht zudem nur selten die Möglichkeit, die gewünschten Informationen zu spezifizieren. Und nur selten lassen einem die CL-Befehle die Wahl zwischen detaillierten Informationen und Basisinformationen. Anders die APIs: APIs kennen häufig unterschiedliche Formate, über die Sie den Umfang der Ergebnisinformationen bestimmen können. Und natürlich wird die Verarbeitung umso schneller, je weniger Detailinformationen angefordert werden.

APIs gibt es für die unterschiedlichsten Anforderungen. Hierbei können die folgenden Hauptkategorien in Release V6R1 unterschieden werden:

API-Hauptkategorien

Unabhängsgig von den jeweiligen Hauptkategorien muss man APIs aber v. a. entsprechend Ihrer Laufzeitumgebungen unterscheiden. So gibt es APIs, die ohne Probleme in OPM-Programmen eingesetzt werden können, weil Sie mit der Anweisung CALL aufgerufen werden können. Diese APIs basieren auf einem eigenständigen Programm (programmbasierende APIs). Diese APIs gibt es bereits seit langem. Sie können aus jedem OPM-Programm problemlos aufgerufen werden. Es handelt sich hierbei um echte Programme, die dynamisch mit der Anweisung CALL aufgerufen werden. Die Namen dieser APIs haben maximal 8 Zeichen, bestehen nur aus Großbuchstaben und beginnen stets mit dem Buchstaben „Q“. Die OPM-APIs können grob in drei Kategorien unterteilt werden:

Einfache Programm-APIs haben entweder keine Parameter oder nur eine sehr begrenzte Parameterliste, die in der Regel nur Input-Parameter enthält. Solche APIs sind einfach aufzurufen und können dennoch nützliche Funktionen liefern:

API	Funktion	Aufruf
QUSCMDLN	Das API QUSCMDLN öffnet ein Fenster mit einer Befehlszeile, die auch aus interaktiven Programmen den Zugriff auf alle Betriebssystembefehle ermöglicht.	CALL PGM(QUSCMDLN)
QEZMAIN	Das API QEZMAIN erstellt einen Gruppenjob, um das Assist-Menü darzustellen.	CALL PGM(QEZMAIN)
QEZAST	Dieses API ermöglicht ebenfalls den Zugriff auf das Assist-Menü. Allerdings wird hierfür der Befehl GO ASSIST verwendet.	CALL PGM(QEZAST)
QCMD	Aus einem Programm heraus kann das Systemprogramm QCMD aufgerufen werden. Dadurch wird die Anwendung verlassen und der Benutzer kann auf Systemebene arbeiten.	CALL PGM(QCMD)
QCMDEXC	Das API QCMDEXC ermöglicht es dem Programmierer, aus einem Hochsprachenprogramm CL-Befehle abzusetzen. Das API benötigt hierfür zwei Input-Parameter: – Erster Parameter: Zeichenfeld – Zweiter Parameter: numerisches Feld mit zehn Vorkomma- und fünf Nachkommastellen	

Programm-APIs mit Receiver-Variable

Diese APIs können mit CL-Befehlen verglichen werden, die mit der Zeichenfolge „RTV“ beginnen. Sie stellen das Ergebnis in eine Output-Variable, eine sogenannte Receiver-Variable, wobei für die Receiver-Variable meistens keine feste Länge angegeben wird. Die tatsächlich übergebene Länge muss dem API über einen weiteren Parameter mitgeteilt werden. Das API QRCVDTAQ (Inhalte einer Datenwarteschlange lesen) und das API QWCRSVAL (Systemwerteinstellungen abrufen) gehören beispielsweise in diese Kategorie.

2.15
Seite 4

Programm-List-APIs

List-APIs ergänzen mehr oder weniger einige DSP-Anweisungen des Betriebssystems. Das Ergebnis der List-APIs wird immer in User Spaces ausgegeben. Beispiele für den Einsatz von User Spaces zeigt das Kapitel 2.15.2.

Neben den beschriebenen Programm-APIs existieren aber auch APIs, die nur als Prozeduren oder Funktionen in Serviceprogrammen zur Verfügung stehen. Diese APIs können folglich auch nur in ILE-Umgebungen eingesetzt werden. Diese API-Gruppe kann auch als auf Serviceprogrammen basierende APIs bezeichnet werden. Innerhalb der auf Serviceprogrammen basierenden APIs kann man zwischen Common Execution Environment APIs (CEE-APIs) und den UNIX-Type APIs unterscheiden.

Serviceprogramm-APIs können nur aus ILE-Umgebungen heraus aufgerufen werden, da es sich hierbei nicht um Programme, sondern um Prozeduren und Funktionen handelt, die in Serviceprogrammen eingebunden sind. Es gibt inzwischen auch einige APIs, die doppelt zur Verfügung gestellt werden: als OPM und als ILE-Variante.

Vorsicht ist in diesem Zusammenhang geboten bei den Prozedurnamen: Die ILE-API-Prozedurnamen sind case sensitiv, d. h., es ist unbedingt auf die korrekte Groß- und Kleinschreibung zu achten. Dazu ist es erforderlich, Prozedurnamen in Hochkommata einzuschließen. Die ILE-API-Prozedurnamen beginnen mit dem Buchstaben „Q“ und können aus maximal 30 Zeichen bestehen. Alle Serviceprogramme, die ILE-APIs beinhalten, sind in dem Binderverzeichnis QUSAPIBD hinterlegt.

```
                  Mit Binderverzeichniseinträgen arbeiten

Binderverzeichnis:   QUSAPIBD       Bibliothek:   QSYS

Auswahl eingeben und Eingabetaste drücken.
  1=Hinzufügen   4=Entfernen

Aus-                                                      ------Erstellung------
wahl  Objekt      Art      Bibliothek  Aktivierung  Datum       Zeit
 =    __________  ______   __________  ___________
 _    QANESERV    *SRVPGM  QSYS        *IMMED       03.10.07    10:42:02
 _    QCAVFYNMS   *SRVPGM  QSYS        *IMMED       04.12.07    04:55:46
 _    QCJASTUB    *SRVPGM  QSYS        *IMMED       03.10.07    10:42:06
 _    QCSTCRG3    *SRVPGM  QSYS        *IMMED       03.10.07    10:42:38
```

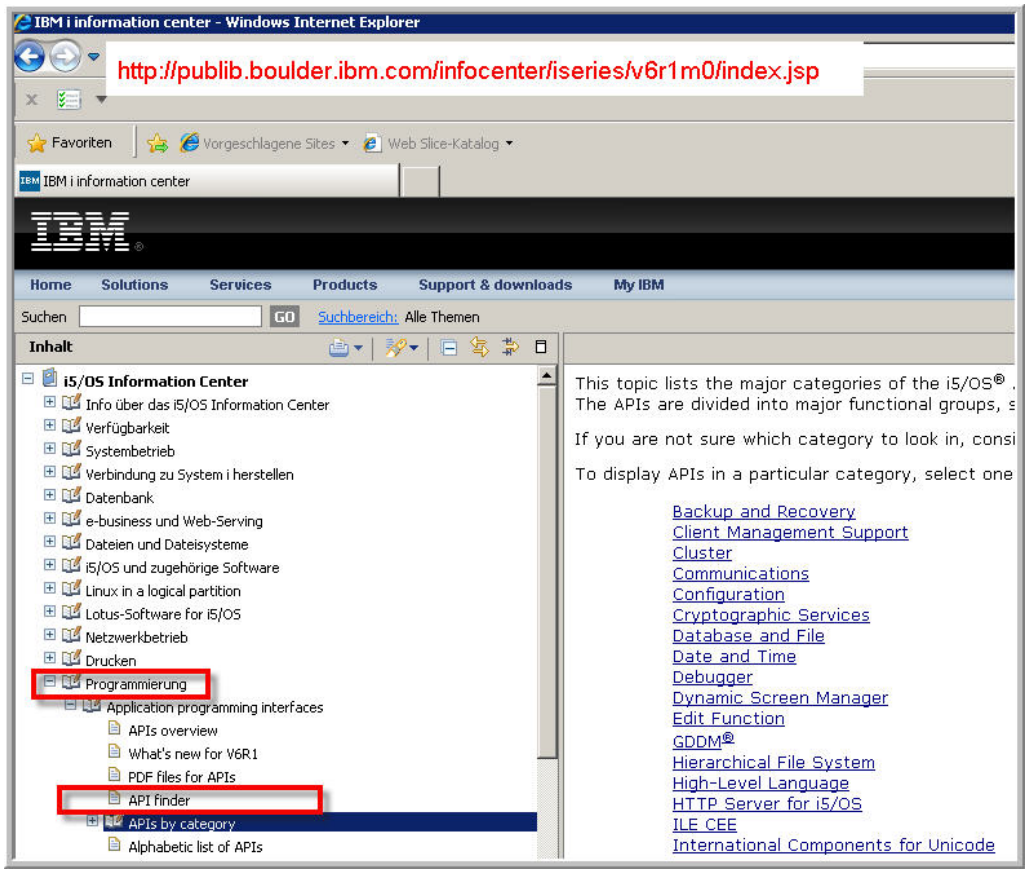
Einträge des Binderverzeichnisses QUSAPIBD

Das Binderverzeichnis QUSAPIBD wird von den ILE-Compilern automatisch herangezogen, wenn Sie im ILE-Programm ILE-APIs verwenden. Sie müssen also bei der Programmerstellung weder das Serviceprogramm, in dem die Prozedur hinterlegt ist, noch das Binderverzeichnis QUSAPIBD angeben. Auch auf Serviceprogrammen basierende List-APIs nutzen für die Ausgabe ihrer Daten User Spaces. Der Aufbau der Informationen im User Space ändert sich beim Einsatz von auf Serviceprogrammen basierenden APIs nicht; lediglich das Header-Format ist ein anderes.

Ich habe bereits erwähnt, dass es sich bei den CEE-APIs um ein Set von Prozeduren handelt, die für alle IBM-Systeme gleichermaßen gelten. Die Prozedurnamen beginnen grundsätzlich mit den Buchstaben „CEE“ und bestehen aus maximal acht Großbuchstaben. CEE steht im Übrigen für „Common Execution Environment“. Die Prozeduren sind im Serviceprogramm QILE hinterlegt, das ebenfalls von den ILE-Compilern automatisch eingebunden wird, wenn Sie ein solches API verwenden. CEE-APIs bieten Funktionen, die in der Regel von den Programmiersprachen nicht abgedeckt werden. Es gibt beispielsweise CEE-APIs, um anhand eines Datumswertes die Kalenderwoche nach ISO-Richtlinien zu ermitteln. Die häufigste Verwendung finden CEE-APIs jedoch in Umgebungen, in denen mit unterschiedlichen Programmiersprachen gearbeitet wird, um ein einheitliches Fehlerhandling für alle Programmierumgebungen zu gewährleisten.

2.15
Seite 6

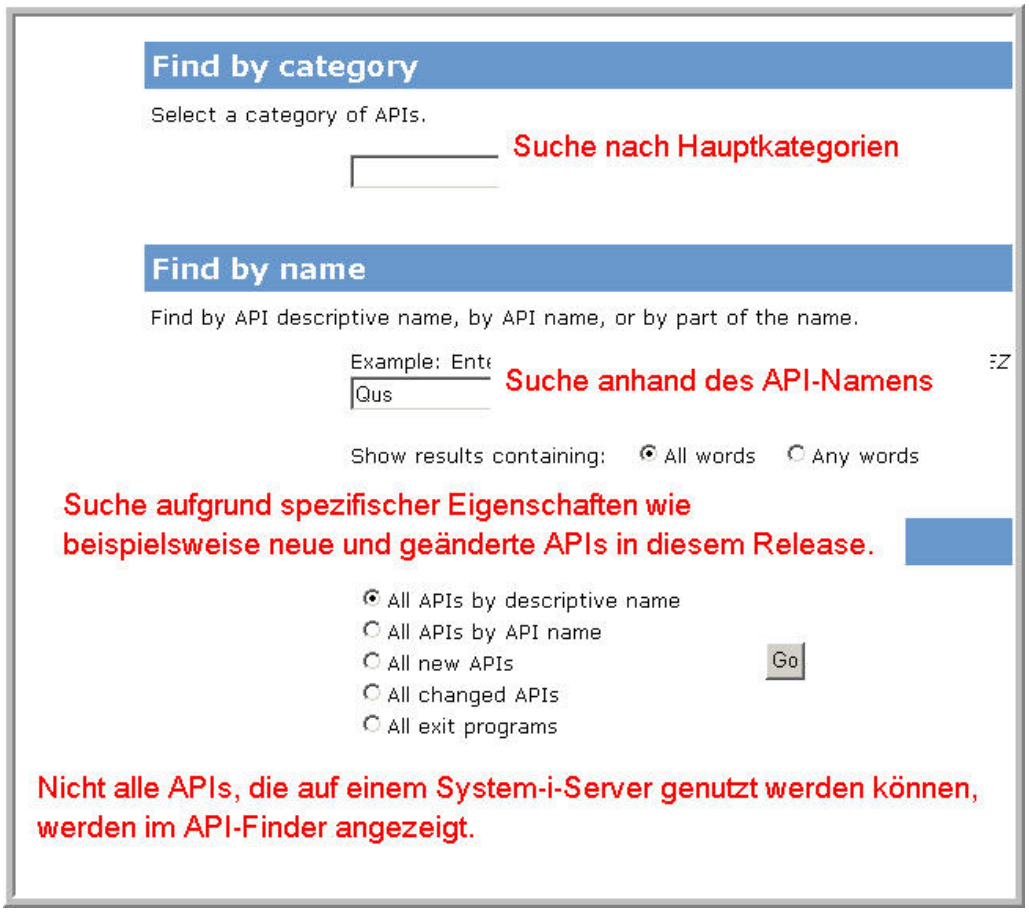
Bei einer derartigen Vielzahl von APIs stellt sich natürlich die Frage, wie Sie das richtige API zur Lösung Ihres Problems finden. Das IBM Information Center stellt die notwendigen Informationen zur Verfügung:



Informationen über verfügbare APIs – Release V6R1

Die verfügbaren APIs können entweder in alphabethischer Reihenfolge oder je Hauptkategorie angezeigt werden. Darüber hinaus enthält das Information Center einen API-Finder.

Der API-Finder ermöglicht eine Suche nach Kategorie, Name und anderen Eigenschaften.



Der API-Finder

Bei der Suche nach dem API-Namen können entweder der komplette Name, die ersten Zeichen des API, aber auch einige beschreibende Worte auf Englisch angegeben werden. Die letztgenannte Variante führt nach meiner Erfahrung nur selten zum Erfolg. Die Anzeige je Kategorie ist gut, erfordert aber ein wenig Zeit. Am meisten Erfolg haben Sie natürlich bei der Suche über den Namen. Dies setzt allerdings voraus, dass Sie zumindest „ungefähr“ den Namen des API kennen.

2.15
Seite 8

Allgemeine Informationen zur Parameterübergabe

Nachdem Sie „Ihr" API gefunden haben, müssen Sie den korrekten Aufruf in Ihr Programm integrieren und dabei dem API oftmals eine Reihe von Parametern übergeben. Grundsätzlich unterscheiden APIs hierbei drei unterschiedliche Parametertypen: erforderliche Parameter, optionale Parameter und auszulassende Parameter:

Erforderliche Parameter
Diese Parameter müssen im Aufruf verwendet werden. Die Reihenfolge der Parameter ist ebenso bindend wie der Datentyp und die Länge.

Optionale Parameter
Sie können alle optionalen Parameter weglassen oder Sie nutzen alle optionalen Parameter. Es ist niemals möglich nur einzelne optionale Parameter zu verwenden.

Auszulassende Parameter
Bei Serviceprogramm-APIs können Sie einige Parameter bei der Übergabe auslassen. Anstelle des Parameters müssen Sie dann aber einen Null-Pointer übergeben. Es versteht sich außerdem von selbst, dass Sie sich bei der Übergabe von Objektnamen stets an die gültigen Namenskonventionen für Objekte halten müssen. Das API nutzt den Objektnamen unverändert und führt auch sonst keine weiteren Objektprüfungen durch.

Input-Parameter
Sie können immer davon ausgehen, dass Input-Parameter durch das API nicht verändert werden, so dass der übergebene Wert in Ihrem Programm auch nach der API-Ausführung vorhanden ist.

Output-Parameter
Anders verhält es sich hingegen mit Output-Parametern. Deren Inhalte können durch das API verändert werden. Falls ein API I/O-Parameter anfordert, handelt es sich hierbei stets um Datenstrukturen. Innerhalb der Datenstruktur gibt es dann sowohl Input- als auch Outputfelder. Jedes Datenstrukturfeld bestimmt hierbei letztlich, ob es sich um einen Input- oder einen Output-Parameter handelt.

Verwendung von User Spaces
Wenn das aufgerufene API Informationen in einen User Space stellt, sollten Sie stets mit den Offset-Werten arbeiten, die der Header des User Space enthält, und nicht mit absoluten Werten. IBM gewährleistet nicht, dass sich die Offset-Positionen in künftigen Betriebssystemversionen nicht ändern.

Fehlercodestrukturen
Die meisten APIs nutzen Fehlercodestrukturen, um dem aufrufenden Programm Fehlersituationen zu melden. Hierbei werden grundsätzlich zwei Formate unterschieden: ERRC0100 und ERRC0200.

Format ERRC0100 kann verwendet werden, wenn keine Code-Page-Unterstützung nötig ist. Das Format ERRC0100 beinhaltet einen Input-Parameter, der die Länge der Datenstruktur beinhaltet. Nur wenn dieser Wert im Aufruf den Wert 8 oder einen größeren Wert enthält, werden die restlichen Felder im Fehlerfall mit den entsprechenden Inhalten versorgt. Falls dieses Feld aber mit dem Wert 0 übergeben wird, wird die Datenstruktur ignoriert und das aufrufende Programm erhält stattdessen eine ESCAPE-Nachricht.

Datenstrukturformat ERRC0100 (				
Offset				
DEC	HEX	Verwendung	Typ	Beschreibung
0	0	INPUT	BINARY(4)	Länge der Datenstruktur
4	4	OUTPUT	BINARY(4)	Länge der Rückgabe-Informationen
8	8	OUTPUT	CHAR(7)	Nachrichten-ID
15	F	OUTPUT	CHAR(1)	Reserviert
16	10	OUTPUT	CHAR(*)	Nachrichtenersatzwerte

Format ERRC0200 muss hingegen verwendet werden, wenn der Aufrufer Codepage-Unterstützung benötigt. Das Format ERRC0200 beinhaltet im Gegensatz zur Datenstruktur ERRC0100 zusätzlich zwei Input-Felder. Das erste Feld – der Schlüssel – muss den Wert 1 im Aufruf enthalten, wenn Codepage-Unterstützung für Zeichenfelder erforderlich ist. Wenn das zweite Input-Feld den Wert 12 oder einen größeren Wert enthält, werden die restlichen Datenstrukturfelder im Fehlerfall gefüllt. Falls dieses Feld aber mit dem Wert 0 übergeben wird, wird die Datenstruktur in Fehlersituationen ignoriert und das aufrufende Programm erhält stattdessen eine ESCAPE-Nachricht.

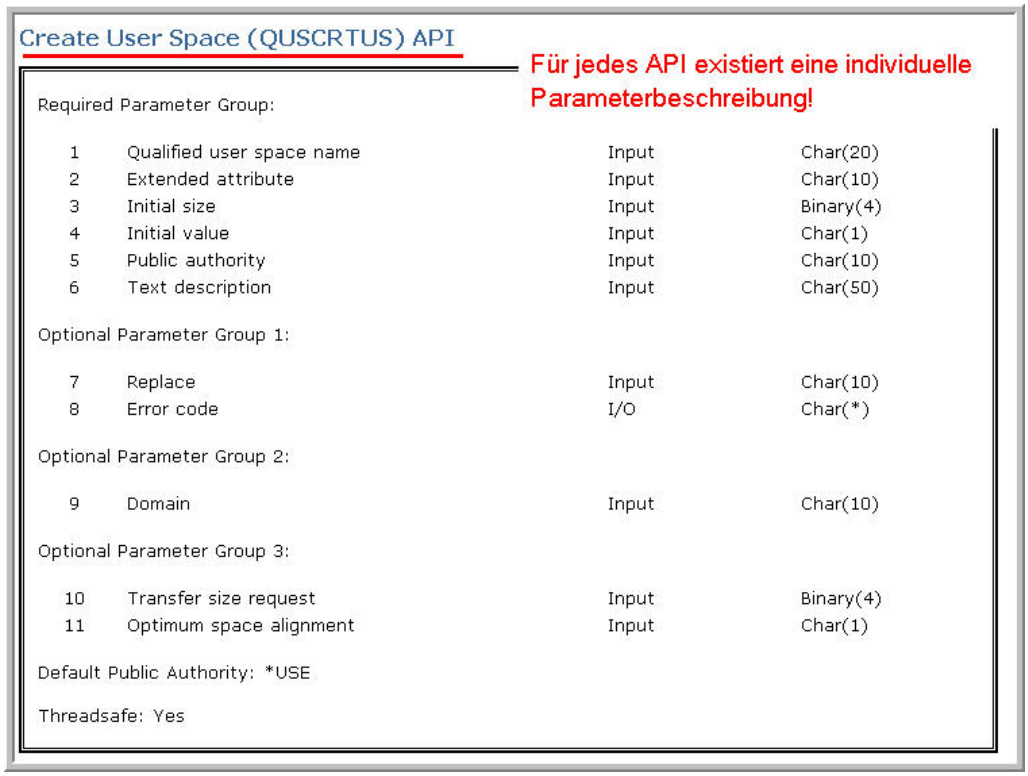
Datenstrukturformat ERRC0200 (				
Offset	Verwendung	Type	Beschreibung	
Dec	Hex			
0	0	INPUT	BINARY(4)	Schlüssel
4	4	INPUT	BINARY(4)	Länge der Datenstruktur
8	8	OUTPUT	BINARY(4)	Länge der Rückgabe-Informationen
12	C	OUTPUT	CHAR(7)	Nachrichten-ID
19	13	OUTPUT	CHAR(1)	Reserviert
20	14	OUTPUT	BINARY(4)	CCSID der CCHAR-Daten
24	18	OUTPUT	BINARY(4)	Offset-Position der Nachrich-ten
28	1C	OUTPUT	BINARY(4)	Länge der Nachrichten-daten
		OUTPUT	CHAR(*)	Nachrichtener-satzwerte

Der 6. Parameter enthält die CCSID der Zeichendaten. Ist dieses Feld 0, wird die CCSID des Jobs verwendet. Alternativ kann der Parameter einen gültigen CCSID-Wert im Wertebereich 1 bis 65535 enthalten – mit Ausnahme des Wertes 65534. Und noch eine Anmerkung zum letzten Parameter: Der Wert CHAR(*) kennzeichnet ein variabel langes Zeichenfeld. Ist dieses Feld zu kurz, um alle Informationen aufzunehmen, tritt keine Fehlersituation ein. Die Informationen werden einfach abgeschnitten.

Einige APIs überlassen es Ihnen, ob Sie eine Fehlercodestruktur übergeben oder nicht. Bei solchen APIs ist dieser Parameter ein optionaler Parameter. Wird keine Datenstruktur für Ausnahmesituationen an das API übergeben, erhält der Aufrufer lediglich ESCAPE-Nachrichten zugestellt. Ist die Datenstruktur hingegen vorhanden, werden auch Diagnosenachrichten gesendet.

28. Ergänzung 9/2011

Damit Sie wissen, welche Parameter ein API, das Sie aufrufen wollen, benötigt, müssen Sie sich stets die Beschreibung des API im Information Center ansehen:



Create User Space (QUSCRTUS) API

Für jedes API existiert eine individuelle Parameterbeschreibung!

Required Parameter Group:

1	Qualified user space name	Input	Char(20)
2	Extended attribute	Input	Char(10)
3	Initial size	Input	Binary(4)
4	Initial value	Input	Char(1)
5	Public authority	Input	Char(10)
6	Text description	Input	Char(50)

Optional Parameter Group 1:

7	Replace	Input	Char(10)
8	Error code	I/O	Char(*)

Optional Parameter Group 2:

9	Domain	Input	Char(10)

Optional Parameter Group 3:

10	Transfer size request	Input	Binary(4)
11	Optimum space alignment	Input	Char(1)

Default Public Authority: *USE

Threadsafe: Yes

API-Parameterbeschreibungen

Diese API-Beschreibungen enthalten zusätzliche Erläuterungen für jeden Parameter und geben Auskunft über mögliche Eingabe- bzw. Ausgabewerte und deren Bedeutung.

Wenn Sie heute APIs einsetzen, stellt sich natürlich die Frage: Wie sicher sind Ihre Applikationen in zukünftigen Betriebssystemversionen? Müssen womöglich alle Programme irgendwann einmal modifiziert werden, weil IBM Parameterlisten verändert? Derzeit gelten folgende Aussagen:

– Zusätzliche Parameter werden nur an das Ende und als optionale Parameter angefügt werden. Feldlängen, Datentypen oder die Reihenfolge der aktuellen Parameter werden auch in Zukunft nicht verändert.

– Innerhalb einer Datenstruktur werden zusätzliche Informationen nur an das Ende der Datenstruktur angefügt. Müssen neue Informationen ein- oder ausgegeben werden, die sich an keine der im API definierten Datenstrukturen anfügen lassen, werden neue Formate generiert. Über diese neuen Formate werden anschließend die neuen Datenstrukturen zugeordnet.

– In vielen APIs wird über Offset-Informationen der Start einer weiteren Datenstruktur oder des nächsten Feldes angegeben. In diesen Fällen sollte man die Startpositionen niemals „hart“, also mit absoluten Werten codieren, sondern immer mittels der Offset-Informationen des allgemeinen API-Headers berechnen und anschließend entweder über einen Pointer oder mit Hilfe anderer Funktionen auf die Informationen zugreifen. Die Kapitel 2.15.2.4.3 und 2.15.2.4.4 stellen beide Methoden exemplarisch vor. Nur wenn Sie wie in den Beispielen gezeigt mit APIs arbeiten, kann gewährleistet werden, dass der Aufruf eines API auch noch im nächsten Release problemlos funktioniert.

28. Ergänzung 9/2011

2.15.1
Seite 6

User Spaces (*USRSPC)

User Spaces haben eine besondere Bedeutung im API-Umfeld. Daher wollen wir diesen Objekten hier ein wenig mehr Aufmerksamkeit widmen. User Spaces sind Anwenderbereiche, die ebenso wie Datenwarteschlangen (*DTAQ) und Datenbereiche (*DTAARA) dazu genutzt werden können, Informationen abzulegen. Viele APIs nutzen diese Objekte für **die Datenausgaben$ oder?**), aber auch Java-Programmierer können mit entsprechenden Methoden der Java-Toolbox auf Anwenderbereiche zugreifen. Ein Anwenderbereich ist ein Objekt der Art *USRSPC, das erstellt, manipuliert und gelöscht werden kann. Da die User Spaces auf der Festplatte erstellt werden, können diese Objekte – wie alle anderen Objekte auch – gesichert und wiederhergestellt werden. Grundsätzlich lassen sich User Spaces mit Data Areas vergleichen. Es handelt sich auch bei einem User Space zunächst um einen unstrukturierten Datenbereich, in dem beliebige Informationen gespeichert werden können.

Für die Erstellung, aber auch für die Verwaltung eines User Space gibt es – anders als bei den Data Areas – keine entsprechenden CL-Befehle. Alle Zugriffe erfolgen ausschließlich über APIs und im Gegensatz zu den Data Areas, deren Maximalgröße auf 2 KB begrenzt ist, können User Spaces bis zu 16 Megabyte groß sein und ihre Größe dynamisch anpassen.

2.15.2
Seite 2

User Space erstellen und löschen

Da es keine CL-Befehle für die Arbeit mit User Spaces gibt, benötigen wir also auch dieses Mal – analog zur Verarbeitung von Datenwarteschlangen – APIs, die das Erstellen, Lesen und Manipulieren des Anwenderbereichs ermöglichen. Wenn häufig mit den Anwenderbereichen gearbeitet wird, sollten Sie sich ggf. eigene Systembefehle definieren, um diese Arbeit zu erleichtern. Wie eigene Commands definiert werden, lesen Sie in Kapitel 2.14.

User Spaces können mit Hilfe des APIs QUSCRTUS (Create User Space) erstellt werden. Schaut man im Information Center nach, erfährt man, dass das API die folgenden Parameter anfordert:

Create User Space (QUSCRTUS) API

Required Parameter Group:

1	Qualified user space name	Input	Char(20)
2	Extended attribute	Input	Char(10)
3	Initial size	Input	Binary(4)
4	Initial value	Input	Char(1)
5	Public authority	Input	Char(10)
6	Text description	Input	Char(50)

Optional Parameter Group 1:

7	Replace	Input	Char(10)
8	Error code	I/O	Char(*)

Optional Parameter Group 2:

9	Domain	Input	Char(10)

Optional Parameter Group 3:

10	Transfer size request	Input	Binary(4)
11	Optimum space alignment	Input	Char(1)

Default Public Authority: *USE

Threadsafe: Yes

Parameter des APIs QUSCRTUS

Das Programm ist schnell geschrieben:

Das API erwartet den qualifizierten Namen des User Space, wobei erst der Objektname und in den letzten 10 Byte der Bibliotheksname genannt werden muss. Diese Werte übergebe ich als Parameter dem CL-Programm CRTUSRSPC. Als Sonderwert für die Bibliothek könnte ich später den Wert *CURLIB verwenden. Es ist natürlich durchaus möglich, den User Space in der Bibliothek QTEMP anzulegen, so dass der Anwenderbereich automatisch gelöscht wird, wenn der Job endet.

2.15.2.1
Seite 2

```
PGM PARM(&OBJ &LIB)
       DCL VAR(&USRSPC) TYPE(*CHAR) LEN(20)
       DCL VAR(&OBJ) TYPE(*CHAR) LEN(10)
       DCL VAR(&LIB) TYPE(*CHAR) LEN(10)
```

Leider kann ich für den Objektnamen und die Bibliothek keine Datenstruktur einsetzen, da Datenstrukturfelder nicht als Parameter im Aufruf verwendet werden können.

Über den Parameter „Extended Attribute“ können Sie dem User Space ein Zusatzattribut zuweisen – ähnlich einer physischen Datei, die das Zusatzattribut PF trägt, oder wie bei einer Druckdatei, die anhand des Zusatzattributs *PRTF identifiziert werden kann. Ob Sie ein Zusatzattribut verwenden und wenn ja welches, bleibt Ihnen überlassen, solange Sie die gültigen Namenskonventionen des Betriebssystems beachten.

```
DCL VAR(&EXT) TYPE(*CHAR) LEN(10)
```

Ich verzichte auf diese Angabe und übergebe einen leeren Parameter &EXT an das API.

Nun muss noch die Anfangsgröße des Bereichs festgelegt werden. Werte zwischen 1 und 16.776.704 sind möglich. Ich entscheide mich zunächst einmal für 500 Byte.

```
DCL VAR(&SIZE) TYPE(*INT) LEN(4) VALUE(500)
```

Für den Parameter „Initial Value“ wird empfohlen den Wert X‘00‘ zu verwenden, um den Bereich entsprechend zu initialisieren. An diese Vorgabe halte ich mich zunächst einmal:

```
DCL VAR(&INIT) TYPE(*CHAR) LEN(1) VALUE(X'00')
```

Die beiden letzten erforderlichen Parameter bestimmen nun noch die Berechtigung der Benutzergruppe *PUBLIC. Hier gelten die allgemeinen Berechtigungsklassen: *ALL, *CHANGE, *USE, *EXCLUDE. Aber auch den Namen einer Berechtigungsliste können Sie einsetzen oder den Sonderwert *LIBCRTAUT verwenden. Ich entscheide, der „Öffentlichkeit“ keine Rechte an dem User Space zu erteilen:

```
DCL VAR(&AUT) TYPE(*CHAR) LEN(10) VALUE('*EXCLUDE')
```

Nun können wir noch einen beschreibenden Text für das Objekt hinzufügen und haben damit alle zwingend erforderlichen Parameter deklariert.

```
DCL VAR(&TEXT) TYPE(*CHAR) LEN(50)
```

Da es mir wenig sinnvoll erscheint, einen generellen, beschreibenden Text für alle User Spaces zu verwenden, lasse ich diesen Parameter wieder leer.

Um sicherzustellen, dass der Anwenderbereich auch dann neu erstellt wird, wenn er bereits existiert, nutze ich jetzt zusätzlich die optionale Parametergruppe 1. In der Parametergruppe 1 können Sie die Werte *YES oder *NO verwenden, um vorsorglich zu entscheiden, ob ein bestehender Anwenderbereich gleichen Namens ersetzt werden soll. *NO ist für diesen Parameter der Standardwert.

```
DCL VAR(&REPLACE) TYPE(*CHAR) LEN(10) VALUE('*YES')
```

Da ich einen Parameter der optionalen Parametergruppe 1 nutzen will, komme ich um eine Fehlerdatenstruktur nicht herum. Diese Fehlerdatenstruktur codiere ich jedoch nicht im CL-Programm, sondern hinterlege sie als COPY-Element:

```
DCL VAR(&ERR_DS) TYPE(*CHAR) STG(*AUTO) LEN(256)
DCL VAR(&LEN) TYPE(*INT) STG(*DEFINED) LEN(4) DEFVAR(&ERR_DS)
DCL VAR(&RTN_LEN) TYPE(*INT) STG(*DEFINED) LEN(4) DEFVAR(&ERR_DS 5)
DCL VAR(&MSGID) TYPE(*CHAR) STG(*DEFINED) LEN(7) DEFVAR(&ERR_DS 9)
DCL VAR(&RESERVED) TYPE(*CHAR) STG(*DEFINED) LEN(1) DEFVAR(&ERR_DS 16)
DCL VAR(&MSGDTA) TYPE(*CHAR) STG(*DEFINED) LEN(240) DEFVAR(&ERR_DS 17)

CHGVAR VAR(&LEN) VALUE(256) /* LÄNGE DER ERRORCODESTRUKTUR */
```

Die Copy-Strecke CP_ERR_DCL

Den Aufbau der Datenstruktur kennen Sie bereits aus dem vorangegangenen Kapitel. Und auch Datenstrukturen haben wir bereits in Kapitel 2.12.6.12.2 kennengelernt, so dass hier keine weiteren Erläuterungen notwendig sind. Ich habe für unser Beispiel übrigens die Struktur ERRC0100 verwendet. Diese Datenstruktur muss ich jetzt nur noch in das Programm einbinden:

```
INCLUDE SRCMBR(CP_ERR_DCL)
```

Als Nächstes müssen wir den qualifizierten Objektnamen verketten, da ich ja Bibliotheksname und Objektname als einzelne Parameterwerte an mein Programm übergebe. Achten Sie aber darauf, dass erst der Objektname und dann der Bibliotheksname genannt werden muss.

```
CHGVAR &USRSPC VALUE(&OBJ *CAT &LIB)
```

Jetzt kann der Aufruf des API QUSCRTUS erfolgen:

```
CALL PGM(QUSCRTUS) PARM(&USRSPC &EXT &SIZE &INIT +
                &AUT &TEXT &REPLACE &ERR_DS)
```

2.15.2.1
Seite 4

Damit ich erfahre, ob Fehler aufgetreten sind, prüfe ich das Feld der Fehlerstruktur &RTN_LEN. Enthält diese Variable den Wert 0, sende ich die Meldung, dass der User Space erzeugt wurde. Im Fehlerfall gebe ich die ESCAPE-Nachricht aus, die mir das API über die Fehlerstruktur zurückgibt.

```
IF COND(&RTN_LEN *NE 0) THEN(DO)
SNDPGMMSG MSGID(&MSGID) MSGF(QCPFMSG) + MSGDTA(&MSGDTA)
  ENDDO
ELSE CMD(DO)
SNDPGMMSG MSG('User Space' *BCAT &OBJ *BCAT 'in +
Bibliothek' *BCAT &LIB *BCAT 'wurde + erstellt')
  ENDDO
ENDPGM
```

Nun ist es an der Zeit unser CL-Programm zu testen:

```
CALL CRTUSRSPC PARM(,RASCHE‘ ,EPS002CR‘)
```

```
Parameter oder Befehl
===>
F3=Verlassen      F4=Bedienerführung    F5=Aktualisieren
F9=Auffinden      F10=Befehlseingabe    F23=Weitere Angabe
User Space RASCHE in Bibliothek EPS002CR wurde erstellt
```

Der User Space wurde erstellt

Wenn Sie der Nachricht nicht trauen, können Sie die Existenz des Objekts natürlich auch ganz einfach prüfen, indem Sie sich den Inhalt der entsprechenden Bibliothek anzeigen lassen:

```
                          Mit Objekten arbeiten

Auswahl eingeben und Eingabetaste drücken.
  2=Berechtigung editieren   3=Kopieren   4=Löschen   5=Berechtigung anz
  7=Umbenennen   8=Beschreibung anzeigen   13=Beschreibung ändern

Opt  Objekt      Art       Bibliothek  Attribut    Text
__   RASCHE      *USRSPC   EPS002CR                TEST ANWENDERBEREICH

                                                                   Ende
Parameter für Auswahlmöglichkeiten 5, 7 und 13 oder Befehl
===> WRKOBJ OBJ(EPS002CR/*ALL) OBJTYPE(*USRSPC)
```

*Erstellt wurde ein Objekt der Art *USRSPC*

Tatsächlich finden wir in der Bibliothek EPS002CR jetzt ein Objekt des Typs *USRSPC.

Eines sollten Sie jedoch beim Aufruf des API beachten. Alle Objektnamen müssen in Großbuchstaben übergeben werden, ansonsten erhalten Sie vom API die folgende Fehlermeldung:

```
Parameter oder Befehl
===> _____________________________________
F3=Verlassen      F4=Bedienerführung
F9=Auffinden      F10=Befehlseingabe
Objektname RasCHE ist nicht gültig.
```

USRSPC wurde nicht erstellt

Die Fehlersituation zeigt aber auch, dass unsere Fehlerstruktur korrekt ist und unsere Routine im CL-Programm die Nachrichten des API korrekt anzeigt.

Noch einfacher ist es, einen Anwenderbereich wieder zu löschen.

```
PGM PARM(&OBJ &LIB)
DCL VAR(&USRSPC) TYPE(*CHAR) LEN(20)
DCL VAR(&OBJ) TYPE(*CHAR) STG(*DEFINED) LEN(10) DEFVAR(&USRSPC 1)
DCL VAR(&LIB) TYPE(*CHAR) STG(*DEFINED) LEN(10) DEFVAR(&USRSPC 11)

INCLUDE SRCMBR(CP_ERR_DCL)   SRCFILE(EPS002CR/QCLSRC)
CHGVAR VAR(&USRSPC) VALUE(&OBJ *CAT &LIB)
CALL PGM(QUSDLTUS) PARM(&USRSPC &ERR_DS)

IF COND(&RTN_LEN *NE 0) THEN(DO)
   SNDPGMMSG  MSGID(&MSGID) MSGF(QCPFMSG) MSGDTA(&MSGDTA)
   ENDDO
ELSE CMD(DO)
   SNDPGMMSG  MSG('User Space' *BCAT &OBJ *BCAT 'in +
              Bibliothek' *BCAT &LIB *BCAT 'wurde +
              gelöscht!')
     ENDDO
ENDPGM_
```

Das CL-Programm DLTUSRSPC

Der Aufruf des API QUSDLTUS ist einfach, da als einziger Parameter der qualifizierte Name des User Space und unsere Fehlerstruktur erforderlich sind, damit wir wieder informiert werden, falls es unliebsame Überraschungen gibt.

2.15.2.1

Seite 6

User Space Attribute prüfen und ändern

Mit dem API QUSRUSAT können Sie die Eigenschaften eines bestehenden User Space ermitteln. Folgende Informationen können dabei abgerufen werden:

– Erweitert das Betriebssystem den Anwenderbereich dynamisch, wenn das Ende erreicht ist? Der Returnwert 0 zeigt an, dass der Bereich nicht dynamisch erweitert wird; ist der Returnwert 1, so erfolgt eine automatische Vergrößerung seitens des Betriebssystems.

– Wie groß ist der User Space?

– In welcher Bibliothek wurde der User Space gespeichert? Diese Information ist hilfreich, wenn der User Space über die Sonderwerte *LIBL oder *CURLIB adressiert wird.

Um ein entsprechendes CL-Programm zu codieren, benötigen wir zunächst wieder die Beschreibung des API:

Retrieve User Space (QUSRTVUS) API

Required Parameter Group:

1	Qualified user space name	Input	Char(20)
2	Starting position	Input	Binary(4)
3	Length of data	Input	Binary(4)
4	Receiver variable	Output	Char(*)

Optional Parameter Group:

5	Error code	I/O	Char(*)

Default Public Authority: *USE

Threadsafe: Yes

Parameter des API QUSRUSAT

Die Parameter sind überschaubar. Als Erstes benötigen wir eine entsprechende Variable, die den qualifizierten Objektnamen enthält:

```
DCL VAR(&USRSPC) TYPE(*CHAR) LEN(20)
```

Welche Informationen abgerufen werden, bestimmt der Inhalt meiner Variablen &FORMAT:

```
DCL VAR(&FORMAT) TYPE(*CHAR) LEN(8) VALUE(‚SPCA0100‘)
```

Viele APIs verwenden den Formatparameter. Der Parameter bestimmt letztendlich, wie die gewünschten Informationen zurückgegeben werden. Die unterstützten Formate sind in der spezifischen API-Dokumentation zu finden.

2.15.2.2
Seite 2

Ich nutze für mein Beispielprogramm das Format SPCA0100, das mir die folgenden Informationen liefert:

Offset		Type	Field
Dec	Hex		
0	0	BINARY(4)	Anzahl zurückgegebener Bytes
4	4	BINARY(4)	Verfügbare Bytes
8	8	BINARY(4)	Größe des User Space in Bytes
12	C	CHAR(1)	Dynamische Erweiterung (0) Nein (1) Ja
13	D	CHAR(1)	Initialisierungswert
14	E	CHAR(10)	Bibliotheksname

Zusätzlich muss eine Variable deklariert werden, in der die abgefragten Informationen gespeichert werden. Die Länge der Empfangsvariablen richtet sich nach dem verwendeten Format. Die von mir gewählte Formatstruktur ist 14 Bytes lang, entsprechend lang deklariere ich meine Empfangsvariable:

```
/* Variable für die Informationen */
DCL VAR(&RCVDTA) TYPE(*CHAR) LEN(14)
```

Die Empfangsvariable kann als Datenstruktur übergeben werden und variiert in der Länge aufgrund des verwendeten Formats. Diese Informationen können Sie der Dokumentation entnehmen, da dort der Datentyp CHAR(*) verwendet wird. Ich benötige im Programm lt. Formatbeschreibung die Offset-Position 12, wenn ich erfahren will, ob mein soeben erstellter User Space dynamisch erweitert wird. Aus diesem Grund deklariere ich das Datenstrukturfeld &DYN. Es wird den Wert 0 enthalten, wenn der Bereich nicht dynamisch wächst und den Wert 1, wenn die Größe automatisch angepasst wird.

```
DCL VAR(&DYN) TYPE(*CHAR) STG(*DEFINED) LEN(1) +
            DEFVAR(&RCVDTA 13)
```

Allerdings muss ich in meiner Programmdatenstruktur die Offset-Positionen noch um den Wert 1 erhöhen, d. h., die Offset-Positionen verschieben sich gegenüber der Dokumentation um den Wert 1, da wir im CL-Programm keine Offset-Position 0 kennen, sondern stets mit dem Wert 1 starten.

Wie so häufig benötigen wir noch eine weitere Variable, die die Länge der Variablen &RCVDTA – also die Anzahl Bytes – enthält, die das API an das CL-Programm zurückliefern soll.

```
DCL VAR(&LENGTH) TYPE(*INT) LEN(4) VALUE(14)
```

Nun fügen wir – wie in den anderen Beispielprogrammen – die Fehlerstruktur hinzu, rufen das API auf und geben entweder eine Fehlermeldung aus oder werten die Attributinformation aus:

```
INCLUDE SRCMBR(CP_ERR_DCL)
CALL PGM(QUSRUSAT) PARM(&RCVDTA &LENGTH &FORMAT +
              &USRSPC &ERR_DS)
IF COND(&RTN_LEN *NE 0) THEN(DO)
SNDPGMMSG MSGID(&MSGID) MSGF(QCPFMSG) +
           MSGDTA(&MSGDTA)
  ENDDO
ELSE CMD(DO)
SELECT
WHEN (&DYN = '1') THEN( +
SNDPGMMSG MSG('User Space wird erweitert.'))
WHEN (&DYN = '0') THEN( +
SNDPGMMSG MSG('User Space wird nicht erweitert.'))
   ENDSELECT
  ENDDO
ENDPGM
```

Prüfen wir auch dieses Programm:

```
Parameter oder Befehl
===> _____________________________
F3=Verlassen     F4=Bedienerführu
F9=Auffinden     F10=Befehlseinga
User Space wird nicht erweitert.
```

Ergebnis des Programmaufrufs RTVUSRSPCA

Das Ergebnis scheint in Ordnung zu sein, da zunächst alle User Spaces fixe Längen haben und wir explizit die dynamische Erweiterung anfordern müssen, indem wir die Attribute unseres User Space mit einem entsprechenden

2.15.2.2
Seite 4

API ändern.

Das API QUSCUSAT vergrößert oder verkleinert einen Anwenderbereich, ändert das Attribut bezüglich der dynamischen Erweiterung, ändert den Initialisierungswert und/oder die Seitenanzahl, die vom Hauptspeicher auf die Festplatte geschrieben werden. Dieses API erfordert bezüglich seiner Parameter ein wenig mehr Aufmerksamkeit:

Change User Space Attributes (QUSCUSAT) API

Required Parameter Group:

1	Returned library name	Output	Char(10)
2	Qualified user space name	Input	Char(20)
3	Attributes to change	Input	Char(*)
4	Error code	I/O	Char(*)

Default Public Authority: *USE

Threadsafe: Yes

Parameter des API QUSCUSAT

Der erste Parameter enthält den Namen der Bibliothek, in der der User Space geändert wurde. Dieser Parameter ist uns aufgrund der vorangegangenen Beispiele hinlänglich bekannt und im Grunde nur nützlich, wenn der qualifizierte Objektname die Sonderwerte *LIBL oder *CURLIB enthält. Dann können wir nach der API-Ausführung prüfen, in welcher Bibliothek der User Space geändert wurde.

Mit dem zweiten Parameter „Qualified User Space Name“ haben wir auch keine Probleme, und auch die Fehlerstruktur ist vorhanden.

Etwas genauer müssen wir uns um den dritten Parameter kümmern. Hierbei handelt es sich wiederum um eine Empfangsdatenstruktur, deren Aufbau in der Dokumentation folgendermaßen beschrieben ist:

Empfangsdatenstruktur			
Offset		Type	Field
Dec	Hex		
0	0	BINARY(4)	Anzahl der verwendeten Schlüssel bzw. Anzahl der übergebenen Array-Elemente
4	Array mit Angabe		
		BINARY(4)	Schlüssel
		BINARY(4)	Länge der Daten
		CHAR(*)	Daten

Das erste Element der Datenstruktur bestimmt, wie viele Attribute verändert werden sollen.

```
DCL VAR(&ATR_CHG) TYPE(*CHAR) LEN(13)
/* Anzahl der verwendeten Codes */
DCL &NBR_KEYS TYPE(*INT) LEN(4) +
          DEFVAR(&Atr_CHG 1) STG(*DEFINED)
```

Die Variable wird zum Zeitpunkt der Ausführung den Wert 1 enthalten, da ich nur die dynamische Größenanpassung aktivieren will.

Im zweiten Element muss ein Schlüssel hinterlegt werden, der dem API mitteilt, welche Attribute geändert werden sollen.

```
/* Der Codeschlüssel */
DCL VAR(&CODE) TYPE(*INT) STG(*DEFINED) LEN(4) +
            DEFVAR(&ATR_CHG 5)
```

2.15.2.2
Seite 6

Die jeweiligen Codeschlüssel können Sie immer der API-Beschreibung entnehmen:

Code	Type	Bedeutung
1	BINARY(4)	Größe des User Space ändern. Ist der neue Wert kleiner als die aktuelle Größe, wird der User Space verkleinert. Ist der Wert hingegen größer, findet eine Erweiterung statt.
2	CHAR(1)	Initialisierung ändern
3	CHAR(1)	Automatische Erweiterung aktivieren oder deaktivieren: (0) Deaktivieren (1) Aktivieren
4	BINARY(4)	Blockgrößen für das Schreiben auf die Festplatte verändern. Werte zwischen 0 und 32 Seiten sind möglich.

Ich werde also später im Programm die Variable &CODE mit dem Wert 1 initialisieren.

Das dritte Element der Datenstruktur legt fest, wie lang die Daten sind, die Sie für die Änderungen übergeben. Abschließend folgt dann der Wert, der die Änderung bestimmt; die neue Größe, der neue Initialisierungswert oder wie in unserem Beispielprogramm, der Wert ‚1', um die dynamische Erweiterung des User Space zu aktivieren:

```
DCL VAR(&LENGTH) TYPE(*INT) LEN(4) +
          DEFVAR(&ATR_CHG 9) STG(*DEFINED)
DCL VAR(&DATA) TYPE(*CHAR) LEN(1) +
          STG(*DEFINED) DEFVAR(&ATR_CHG 13)
```

Wir müssen also im Programm die Variable &LENGTH mit dem Wert 1 initialisieren, wenn wir den Wert 1 für die erforderliche Attributänderung übergeben.

Etwas schwieriger wird es im CL-Programm, wenn mehr als eine Eigenschaft des User Space verändert werden soll, da wir noch keine Arrays deklarieren können. Wir müssen daher einfach die Tabellenstruktur simulieren, indem wir die Felder &CODE, LENGHT und &WERT entsprechend oft wiederholen. Der Inhalt der Variable &NBR_KEYS erhöht sich dann natürlich entsprechend.

Die Initialisierung der Datenstrukturfelder kann entweder durch Parameterübergaben im Programmaufruf oder im Programm selbst realisiert werden. Ich habe mich fürs Erste entschieden, die Parameterwerte konstant im Programm zu hinterlegen:

```
CHGVAR &USRSPC VALUE(&OBJ *CAT &LIB)
/* CODE FÜR DYNAMISCHE ERWEITERUNG */
CHGVAR &KEY VALUE(3)
/* LÄNGE DER DATENÜBERGABE */
CHGVAR &LENGTH VALUE(1)
/* ANZAHL DER VERWENDETEN SCHLÜSSEL */
CHGVAR &NBR_KEYS VALUE(1)
/* NEUER WERT */
CHGVAR &DATA VALUE('1')
```

Der Aufruf des API und das Fehlerhandling sind identisch mit den zuvor gezeigten Beispielprogrammen, so dass wir darauf verzichten können, uns den Quellcode anzusehen.

Ob unsere Attribute erfolgreich geändert wurden, lässt sich nach Aufruf unseres Programms schnell feststellen, wenn wir wieder die User Space-Attribute mit unserem Programm RTVUSRSPCA abrufen:

```
Parameter oder Befehl
===>
F3=Verlassen      F4=Bedien
F9=Auffinden      F10=Befeh
User Space wird erweitert.
```

Die User Space-Attribute wurden erfolgreich verändert

2.15.2.2

Seite 8

User Spaces im Einsatz

Wie einfach die Arbeit mit den User Spaces wird, nachdem sie erstellt wurden, hängt ein wenig davon ab, wofür Sie genutzt werden sollen. Ich kann mit wenig Aufwand Daten in einen User Space schreiben. Hierfür verwenden wir das das API QUSCHGUS.

Change User Space (QUSCHGUS) API

Required Parameter Group:

1	Qualified user space name	Input	Char(20)
2	Starting position	Input	Binary(4)
3	Length of data	Input	Binary(4)
4	Input data	Input	Char(*)
5	Force changes to auxiliary storage	Input	Char(1)

Optional Parameter:

6	Error code	I/O	Char(*)

Default Public Authority: *USE

Threadsafe: Yes

Parameter des API QUSCHGUS

Die Parameter des API QUSCHGUS erinnern ein wenig an den CL-Befehl CHGDTAARA. Auch dafür haben wir eine Startposition und eine Längenangabe benötigt und mussten mit dynamischen Offset-Positionen arbeiten. Ähnlich arbeitet das API „Change User Space". Sie übergeben nur die Anfangsposition im User Space, die Länge des Eintrags sowie den Wert der geschrieben werden soll; und falls Sie möchten, können Sie auch wieder die Fehlerstruktur verwenden.

Sind die Daten erst einmal im User Space abgelegt, können Sie mit dem API QUSRTVUS jederzeit wieder ausgelesen werden.

Retrieve User Space Attributes (QUSRUSAT) API

Required Parameter Group:

1	Receiver variable	Output	Char(*)
2	Length of receiver variable	Input	Binary(4)
3	Format name	Input	Char(8)
4	Qualified user space name	Input	Char(20)
5	Error code	I/O	Char(*)

Default Public Authority: *USE

Threadsafe: Yes

Parameter des API QUSRTVUS

2.15.2.3
Seite 2

Die Verwendung und der Aufruf des API QUSRTVUS bergen ebenfalls keine Überraschungen mehr und müssen daher für den Moment nicht näher erklärt werden.

List-APIs und User Spaces

Etwas aufwendiger wird die Verwendung von User Spaces, wenn List-APIs eingesetzt werden. Viele der List-APIs liefern zwar Informationen, die auch mit entsprechenden CL-Befehlen erlangt werden können. Nur sind die APIs häufig schneller in der Ausführung und liefern – verglichen mit den entsprechenden CL-Befehlen – detailliertere Informationen, so z. B. das API QUSROBJD (Retrieve Object Description). Hinzu kommt, dass bei der Verwendung von CL-Befehlen und der Ausgabe in Dateien nur selten die gewünschten Informationen spezifiziert werden können. List-APIs kennen hingegen häufig unter-schiedliche Formate, über die Sie selbst den Umfang der Ergebnisinformationen bestimmen können.

Allerdings schreiben List-APIs die Ergebnisse immer in einen User Space. Die Ausgabe ist somit zwar schnell, aber die Verarbeitung der Informationen ist anschließend auch komplexer. Und das fördert nicht gerade die Motivation mit APIs oder List-APIs zu arbeiten! Hinzu kommt, dass die Parameterübergaben an das API bis vor kurzem zwar nicht unmöglich, aber doch sehr aufwendig waren, da APIs häufig binäre Parameter und Datenstrukturen erfordern. Auch die Arbeit mit Pointern war im CL-Programm lange Zeit nicht möglich. Nun sind diese Restriktionen aufgehoben, und hat man erst einmal ein exemplarisches Beispiel, ist die Arbeit mit List-APIs gar nicht so schwer, wie es anfänglich den Anschein haben mag.

2.15.2.4
Seite 2

Die Header-Informationen

Allen List-APIs ist gemeinsam, dass sie eine globale Datenstruktur für die Ausgabe im User Space nutzen. Diese Datenstruktur hat folgenden grundlegenden Aufbau:

Allgemeiner Aufbau eines User Space für List-APIs

Die eigentlichen Informationen, die das API in den User Space schreibt, werden im Datenbereich abgelegt. Stammen die Informationen von einem API, werden die Informationen in der Regel als Datenstrukturen zur Verfügung gestellt. Der Aufbau der Datenstrukturen ist dabei vom verwendeten API abhängig. Zusätzlich stehen im User Space noch allgemeingültige Header-Informationen. Sie enthalten neben diesen Informationen auch die Offset-Position des Datenbereichs. Schließlich wird noch eine Kopie der verwendeten Input-Parameter zur Verfügung gestellt. Die spezifischen Header-Informationen enthalten schließlich abhängig vom genutzten API weitere Informationen.

2.15.2.4.1
Seite 2

Interessant sind für unsere CL-Programme aber v. a. die Informationen im allgemeinen Header. Er enthält nicht nur die Offset-Position der Daten, sondern auch die Länge eines Eintrags und die Anzahl der Einträge. Unterschieden werden hierbei die Formate 100 und 300, wobei programmbasierende APIs – also APIs, die Sie mit der Anweisung CALL aufrufen – stets das Format 100 verwenden:

Allgemeines Header Format 100			
Offset		Type	Field
Dec	Hex		
0	0	CHAR(64)	Anwenderbereich
64	40	BINARY(4)	Größe des allgemeinen Header
68	44	CHAR(4)	Release-Information
72	48	CHAR(8)	Formatname
80	50	CHAR(10)	Verwendetes API
90	5A	CHAR(13)	Erstellungsinformation (Datum und Uhrzeit)
103	67	CHAR(1)	Informationsstatus
104	68	BINARY(4)	Größe des verwendeten USER Space-Bereichs
108	6C	BINARY(4)	Offset des Input-Parameter-Bereichs
112	70	BINARY(4)	Größe des Input-Parameter-Bereichs
116	74	BINARY(4)	Offset des spezifischen Header-Bereichs
120	78	BINARY(4)	Größe des Header-Bereichs
124	7C	BINARY(4)	Offset des Datenbereichs
128	80	BINARY(4)	Größe des Datenbereichs
132	84	BINARY(4)	Anzahl der Dateneinträge
136	88	BINARY(4)	Größe des einzelnen Eintrags
140	8C	BINARY(4)	CCSID der Dateneinträge
144	90	CHAR(2)	Landes-ID
146	92	CHAR(3)	Sprachen-ID
149	95	CHAR(1)	Subsetistindicator **(**
150	96	CHAR(42)	Reservierter Bereich

Einige Informationen der allgemeinen Header-Struktur müssen sicherlich ein wenig erläutert werden:

Erstellungsinformation (Datum und Uhrzeit)
werden im Format CYYMMDDHHMMSS dargestellt. Dabei kennzeichnet wie üblich das erste Byte das Jahrhundert. Der Wert 1 repräsentiert das 21. Jahrhundert und der Wert 0 das 20. Jahrhundert.

Informationsstatus
Anhand des Informationsstatus lässt sich schnell feststellen, ob die Informationen vollständig und korrekt abgelegt wurden. Die folgende Tabelle erläutert die verschiedenen Stati und ihre Bedeutung.

Informationsstatus	
C	Korrekt und vollständig
I	Unvollständig. Die abgerufenen Daten sind entweder nicht vollständig oder fehlerhaft.
P	Unvollständig aber korrekt. Die abgerufenen Daten sind korrekt, aber das API hat mehr Informationen als der User Space aufnehmen kann.

Auf Serviceprogrammen basierende List-APIs nutzen hingegen das Format 300. Dieses Format enthält alle Informationen des Formats 100 und zusätzlich die folgenden Informationen:

Header Format 300			
Offset		Type	Field
Dec	Hex		
192	C0	CHAR(256)	API Entry-Point-Name
448	1C0	CHAR(128)	Reserved

Hinzu kommt lediglich der API Entry-Point-Name und ein weiterer reservierter Bereich. Hierbei handelt es sich um den Prozedurnamen, den Sie benötigen, wenn Sie das API in eines Ihrer ILE-Programme einbinden wollen.

2.15.2.4.1
Seite 4

List-APIs ausführen

Es gibt verschiedene List-APIs, die unterschiedliche Informationen liefern:

API	Funktion
QUSLMBR	Informationen über die Member physischer Dateien
QUSLFLD	Informationen über die Felder physischer Dateien
QUSLJOB	Informationen über die Jobs auf dem Sytem-i-Server
QUSLOBJ	Informationen über die i5-Objekte
QUSLSPL	Informationen über vorhandene Spoolfiles
QUSLRCD	Informationen über verwendete Satzformate

Wir wollen das API QUSLSPL aufrufen, um detaillierte Informationen über unsere Spoolfiles zu erhalten. Schauen wir uns als Erstes wieder die Parameterbeschreibung an:

List Spooled Files (QUSLSPL) API

Required Parameter Group:

1	Qualified user space name	Input	Char(20)
2	Format name	Input	Char(8)
3	User name	Input	Char(10)
4	Qualified output queue name	Input	Char(20)
5	Form type	Input	Char(10)
6	User-specified data	Input	Char(10)

Optional Parameter Group 1:

7	Error Code	I/O	Char(*)

Optional Parameter Group 2:

8	Qualified job name	Input	Char(26)
9	Key for the fields to return	Input	Array(*) of Binary(4)
10	Number of fields to return	Input	Binary(4)

Optional Parameter Group 3:

11	Auxiliary storage pool	Input	Binary(4)

Optional Parameter Group 4:

12	Job system name	Input	Char(8)
13	Starting spooled file create date	Input	Char(7)
14	Starting spooled file create time	Input	Char(6)
15	Ending spooled file create date	Input	Char(7)
16	Ending spooled file create time	Input	Char(6)

Optional Parameter Group 5:

17	Auxiliary storage pool device name	Input	Char(10)

Default Public Authority: *USE

Threadsafe: No

Parameter des API QUSLSPL

2.15.2.4.2
Seite 2

Hier erkennt man bereits die Komplexität solcher List-APIs. Dieses API kennt insgesamt fünf optionale Parametergruppen. Leider stehen die Parametergruppen nicht wirklich optional zur Verfügung. Sie müssen nämlich die Parametergruppe 1 und 2 verwenden, wenn Sie beispielsweise die Gruppe 3 benötigen. Von einer freien Wahl der Parametergruppen kann somit nicht die Rede sein.

Im nächsten Programmbeispiel möchte ich beispielsweise die Ausgabe anhand eines Datums selektieren. Der ASP ist mir dabei egal; dennoch werde ich auch diesen Parameter an das API übergeben müssen.

Außerdem müssen Sie sich für eines der zur Verfügung gestellten Ausgabeformate entscheiden, weil einerseits die zu verwendenden Parameter häufig durch das Ausgabeformat festgelegt werden, und weil andererseits die Ausgabeformate auch die Initialisierungen der Parameter bestimmen.

Das API QUSLSPL kennt insgesamt vier verschiedene Ausgabeformate:

Format	Kurzbeschreibung
SPLF0100	Das Format enthält neben dem Namen der Ausgabewarteschlange Informationen über den Job, der das Spoolfile erstellt hat. Zusätzlich wird die interne Spoolfile-ID ausgegeben, die allerdings nur von APIs für den Zugriff auf das Spoolfile verwendet werden kann.
SPLF0200	Das Format 200 enthält alle Informationen, die eine WRKSPLF-Ausgabe liefert. Sie können allerdings bei diesem Format durch Angabe von Schlüsselwerten explizit entscheiden, welche Informationen im User Space abgelegt werden sollen.
SPLF0300	Das Format 300 enthält ebenfalls alle Informationen, die der WRKSPLF-Befehl liefert. Im Unterschied zu den Formaten 100 und 200 werden die Informationen aber anhand eines internen Index ermittelt, so dass die Existenz des Spoolfiles nicht unmittelbar geprüft wird. Dafür ist die Ausgabe aber schneller als eine vergleichbare Ausgabe der Formate PLF0100 und PLF0200.
SPLF0400	Auch das Format SPL0400 enthält wieder die Basisinformationen und wird wie Format SPL0300 aufgrund eines internen Index erstellt. Im Unterschied zum Format SPL0300 unterstützt dieses Format UTC-Formate für Datum und Uhrzeit und verwendet nicht die lokale Zeitangabe des Jobs.

Die API-Dokumentation enthält neben dieser Kurzbeschreibung selbstverständlich noch eine ausführliche Beschreibung aller Ausgabeinformationen. Darauf sollten wir an dieser Stelle aus Platzgründen aber besser verzichten. Ich habe mich für das Format SPL0300 entschieden; es enthält alle Informationen, die ich benötige, um die selektierten Spoolfiles anschließend zu löschen; und ein Abgleich mit den existierenden Spoolfiles ist nicht notwendig, da ich diese Fehlersituation später im CL-Programm abfangen werde.

Es folgt die Parameterdefinition. Hierbei handelt es sich um viel Fleißarbeit, die ärgerlich ist, aber getan werden muss. Außerdem müssen Sie anhand der API-Beschreibung sehr genau prüfen, welche Initialisierungswerte für die Parameter möglich oder gar erforderlich sind, und zwar in Abhängigkeit ihrer Verwendung und des gewählten Formats. Auch das ist ein beschwerlicher Weg, aber leider nicht zu umgehen:

```
PGM PARM(&USRSPC)
/* erforderliche Parameter */
DCL VAR(&USRSPC) TYPE(*CHAR) LEN(20)
/* Formatname */
DCL VAR(&FORMAT) TYPE(*CHAR) LEN(8) +
              VALUE('SPLF0300')
/* für alle USER und für alle OUTQ */
DCL VAR(&USER) TYPE(*CHAR) LEN(10) VALUE('*ALL')
DCL VAR(&OUTQ) TYPE(*CHAR) LEN(20) VALUE('*ALL')
/* Formularart beispielsweise *STD oder auch *ALL */
DCL VAR(&FORM_TYP) TYPE(*CHAR) LEN(10) VALUE('*ALL')
/* spezielle Benutzerinformationen */
DCL &USRSPECDTA *CHAR 10 VALUE('*ALL')
/* Optionale Gruppe 1 */
INCLUDE SRCMBR(CP_ERR_DCL)
/* Optional Gruppe 2 */
DCL &QUALJOB *CHAR 26 VALUE(' ')
/* Array mit Keyfeldern nur für Format 200, muss sonst leer +
übergeben werden*/
DCL &ARRAY *CHAR 20
DCL VAR(&O_SPLNAME) TYPE(*INT) STG(*DEFINED) LEN(4) +
              DEFVAR(&ARRAY 1)
```

2.15.2.4.2
Seite 4

```
DCL VAR(&O_JOBNAME) TYPE(*INT) STG(*DEFINED) LEN(4) +
              DEFVAR(&ARRAY 5)
DCL VAR(&O_USERNAME) TYPE(*INT) STG(*DEFINED) LEN(4)+
              DEFVAR(&ARRAY 9)
DCL VAR(&O_JOBNBR) TYPE(*INT) STG(*DEFINED) LEN(4) +
              DEFVAR(&ARRAY 13)
DCL VAR(&O_SPLNBR) TYPE(*INT) STG(*DEFINED) LEN(4) +
              DEFVAR(&ARRAY 17)
DCL VAR(&NBR_KEYS) TYPE(*INT) STG(*AUTO) LEN(4) +
              VALUE(0)
/* optionale Gruppe 3, weil Gruppe 4 benötigt wird */
DCL VAR(&ASP) TYPE(*INT) STG(*AUTO) LEN(4) VALUE(-1)
/* optionale Gruppe 4 für Datum */
DCL &JOBSYSNAM *CHAR 8 value(*CURRENT)
DCL &CRT_DATE *CHAR 7 VALUE('1100707')
DCL &CRT_TIME *CHAR 6 VALUE('235959')
DCL &END_DATE *CHAR 7 VALUE('*LAST')
DCL &END_TIME *CHAR 6 VALUE(' ')
```

Ist diese Aufgabe erledigt, folgen im eigentlichen Programmteil nur noch der API-Aufruf und die aus den vorangegangenen Beispielen bekannte Fehlerprüfung:

```
CALL PGM(QUSLSPL) PARM(&USRSPC &FORMAT &USER +
&OUTQ &FORM_TYP &USRSPECDTA &ERR_DS +
       &QUALJOB &ARRAY &NBR_KEYS &ASP +
&JOBSYSNAM &CRT_DATE &CRT_TIME &END_DATE +
&END_TIME)
IF COND(&RTN_LEN *NE 0) THEN(DO)
SNDPGMMSG MSGID(&MSGID) MSGF(QCPFMSG) +
          MSGDTA(&MSGDTA)
ENDDO
ELSE CMD(SNDMSG MSG('Informationen wurden erfolgreich +
erstellt')
```

```
Parameter oder Befehl
===> _____________________________________________________
F3=Verlassen      F4=Bedienerführung    F5=Aktu
F9=Auffinden      F10=Befehlseingabe    F23=Wei
INFORMATIONEN WURDEN ERFOLGREICH AUSGEGEBEN!
```

Informationen wurden erfolgreich ausgegeben

Wir haben jetzt keine Möglichkeit, diese Aussage zu prüfen, etwa indem wir uns den User Space-Bereich anschauen. Aber das API müsste den allgemeinen Header im User Space erstellt haben und dort gibt es ein Statusfeld, das wir prüfen können:

```
IF COND(&RTN_LEN *NE 0) THEN(DO)
SNDPGMMSG MSGID(&MSGID) MSGF(QCPFMSG) +
          MSGDTA(&MSGDTA)
ENDDO
ELSE CMD(DO)
```

Aufrufen müssen wir hierfür das API QUSRTVUS. Die Information steht an der Offset-Position 104 und ist ein Byte lang. Mit diesen Werten habe ich meine Variablen bereits initialisiert.

```
CALL PGM(QUSRTVUS) PARM(&USRSPC &START &LENGTH +
              &STATUS &ERR_DS)
IF COND(&STATUS = 'C') THEN( +
SNDPGMMSG MSG('Informationen wurden erfolgreich +
           ausgegeben!'))
ENDDO
```

Die Abfrage des Statuscodes verleiht ein wenig zusätzliche Sicherheit und ist mit wenig Aufwand umzusetzen.

2.15.2.4.2

Seite 6

List-API-Informationen verarbeiten – ohne Pointer

Die Verarbeitung der Daten ist nicht wirklich schwierig. Wir müssen lediglich eine Datenstruktur definieren, um die Informationen aufzunehmen:

```
* DS für Infos Format 300 */
DCL VAR(&DATEN) TYPE(*CHAR) LEN(133)
DCL VAR(&JOB) TYPE(*CHAR) STG(*DEFINED) LEN(10) +
          DEFVAR(&DATEN 1)
DCL VAR(&USERNAME) TYPE(*CHAR) STG(*DEFINED) LEN(10) +
          DEFVAR(&DATEN 11)
DCL VAR(&JOBNBR) TYPE(*CHAR) STG(*DEFINED) LEN(6) +
          DEFVAR(&DATEN 21)
DCL VAR(&SPLNAME) TYPE(*CHAR) STG(*DEFINED) LEN(10) +
          DEFVAR(&DATEN 27)
DCL VAR(&SPLFNBR) TYPE(*INT) STG(*DEFINED) LEN(4) +
          DEFVAR(&DATEN 36)
```

Die notwendigen Offset-Positionen entnehme ich der Formatbeschreibung SPL300 meines List-API:

Offset		Type	Field
Dec	Hex		
0	0	CHAR(10)	Jobname
10	A	CHAR(10)	Benutzername
20	14	CHAR(6)	Jobnummer
26	1A	CHAR(10)	Spoolfile-Name
36	24	BINARY(4)	Spoolfile-Nummer
...	...	...	...
132	84	BINARY(4)	Internet print protocol job identifier

Beachten müssen Sie lediglich, dass die Datenstruktur insgesamt 132 Byte umfassen muss, auch wenn sie nur einzelne Informationen benötigen. Außerdem müssen Sie wieder darauf achten, dass sich die Offset-Position entsprechend um den Wert 1 erhöht.

Im nächsten Schritt deklarieren Sie noch die Input-Variablen für das API QUSRTVUS, das wir für den Datenabruf einsetzen müssen:

2.15.2.4.3
Seite 2

```
/*qualifizierter User Space Name */
DCL VAR(&USRSPC) TYPE(*CHAR) LEN(20)
DCL VAR(&START) TYPE(*INT) LEN(4)
DCL VAR(&LENGTH) TYPE(*INT) LEN(4)
INCLUDE SRCMBR(CP_ERR_DCL)
```

Und wir benötigen noch einige Hilfsvariablen für die Schleifendurchläufe:

```
DCL VAR(&NR_ENTRYS) TYPE(*INT) LEN(4)
DCL VAR(&I) TYPE(*INT) LEN(4)
DCL VAR(&LEN_DTA) TYPE(*INT) LEN(4)
```

In einem ersten Schritt ermitteln wir die Anzahl der Einträge und die Länge eines Dateneintrags anhand der allgemeinen Header-Struktur:

```
/* Anzahl Einträge ermitteln */
CHGVAR &LENGTH VALUE(4)
CHGVAR &START VALUE(133)
CALL PGM(QUSRTVUS) PARM(&USRSPC &START &LENGTH +
              &NR_ENTRYS &ERR_DS)
/* Länge eines Eintrags ermitteln */
CHGVAR &START VALUE(137)
CHGVAR &LENGTH VALUE(4)
CALL PGM(QUSRTVUS) PARM(&USRSPC &START &LENGTH +
              &LEN_DTA &ERR_DS)
```

Auch die Startposition des ersten Dateneintrags können wir so ermitteln:

```
/* Start erster Eintrag */
CHGVAR &START VALUE(125)
CHGVAR &LENGTH VALUE(4)
CALL PGM(QUSRTVUS) PARM(&USRSPC &START &LENGTH
              &START &ERR_DS)
CHGVAR &START VALUE(&START + 1)
```

Sie dürfen nur nicht vergessen, die so ermittelte Offset-Position des ersten Dateneintrags um den Wert 1 zu erhöhen.

Nun können die Daten bequem in einer Schleife abgerufen werden:


```
* Einträgeauslesen */
DOFOR VAR(&I) FROM(1) TO(&NR_ENTRYS)
        CALL PGM(QUSRTVUS) PARM(&USRSPC &START +
                  &LEN_DTA &DATEN )
/* Verarbeitung der Daten */
SNDPGMMSG MSG('Folgende Daten wurden gelesen:' +
              *CAT &SPLNAME *CAT &JOB) +
           TOUSR(RASCHE)
       CHGVAR &START VALUE(&START + &LEN_DTA)
ENDDO
```

Wir dürfen nur nicht vergessen, innerhalb der Schleife die Offset-Position um die Länge eines Eintrags zu erhöhen.

2.15.2.4.3

Seite 4

List-API-Informationen verarbeiten – mit Pointer

Etwas einfacher ist der Zugriff auf die Informationen, wenn wir Pointer einsetzen:

Damit wir auf die Informationen des allgemeinen Headers zugreifen können, definieren wir dieses Mal einfach eine Datenstruktur mit den Offset-Positionen, die wir der Header-Beschreibung entnehmen können. Diese Datenstruktur &GENHEADER erhält allerdings keinen eigenen Speicherbereich zugewiesen, sondern basiert auf einer Pointer-Adresse. Dies ist auch der Grund, warum der Pointer &HDRPTR vor der Datenstruktur deklariert wurde. Falls Sie mit den Deklarationen nicht vertraut sind, das Kapitel 2.10 enthält die notwendigen Basisinformationen hierzu.

```
DCL &HDRPTRTYPE(*PTR)
DCL VAR(&GENHEADER) TYPE(*CHAR) STG(*BASED) +
LEN(250) BASPTR(&HDRPTR)
DCL VAR(&OFFDTA) TYPE(*INT) STG(*DEFINED) LEN(4) +
DEFVAR(&GENHEADER 125)
DCL VAR(&NBRENTRYS) TYPE(*INT) STG(*DEFINED) LEN(4) +
DEFVAR(&GENHEADER 133)
DCL VAR(&LENENTRYS) TYPE(*INT) STG(*DEFINED) LEN(4) +
DEFVAR(&GENHEADER 137)
```

Eine zweite Datenstruktur im Programm ist wieder für die Daten zuständig. Das Format und die Feldbeschreibungen entsprechen dem vorangegangenen Beispiel und basieren auf der Formatbeschreibung SPL0300 des entsprechenden API. Nur verwende ich auch für diese Datenstruktur nun einen Pointer. Die Adresse des Pointers bestimmt auch hier später die Adresse der Datenstruktur:

2.15.2.4.4
Seite 2

```
/* DS für Infos Format 300 */
DCL VAR(&DTAPTR) TYPE(*PTR)
DCL VAR(&DATEN) TYPE(*CHAR) LEN(133) STG(*BASED) +
          BASPTR(&DTAPTR)
DCL VAR(&JOB) TYPE(*CHAR) STG(*DEFINED) LEN(10) +
          DEFVAR(&DATEN 1)
DCL VAR(&USERNAME) TYPE(*CHAR) STG(*DEFINED) LEN(10) +
             DEFVAR(&DATEN 11)
DCL VAR(&JOBNBR) TYPE(*CHAR) STG(*DEFINED) LEN(6) +
             DEFVAR(&DATEN 21)
DCL VAR(&SPLNAME) TYPE(*CHAR) STG(*DEFINED) LEN(10) +
             DEFVAR(&DATEN 27)
DCL VAR(&SPLFNBR) TYPE(*INT) STG(*DEFINED) LEN(4) +
             DEFVAR(&DATEN 36)
```

Unsere Pointer haben allerdings bislang keine Adressen, auf die sie zeigen. Diese Adresszuweisung übernimmt das API QUSPTRUS für unseren Pointer &HDRPTR.

```
CALL QUSPTRUS PARM(&USRSPACE &HDRPTR)
```

Nach Ausführung des API hat der Pointer die Adresse des User Space erhalten. Damit überlagert die Datenstruktur &GENHEADER jetzt unseren User Space. Alle Variablen der Datenstruktur sind nun gefüllt und können ohne weitere Initialisierungen verwendet werden, solange der &HDR-Pointer auf die Anfangsadresse des User Space zeigt.

Nun müssen wir noch den zweiten Zeiger initialisieren, indem die Startadresse des User Space um die Offset-Position des Datenbereichs erhöht wird:

```
CHGVAR &DTAPTR VALUE(&HDRPTR)
CHGVAR%OFS(&DTAPTR) VALUE(%OFS(&DTAPTR) &OFFDTA)
```

Mit Hilfe des Funktion %OFS kann ich zu einer Speicheradresse jeden beliebigen Integer-Wert addieren. Die Variable &OFFDTA der Datenstruktur &GENHEADER enthält den hierfür notwendigen Integer-Wert.

Nun durchlaufen wir wieder unsere Schleife, wobei wir die Anzahl der Einträge auch wieder sehr bequem in der Header-Struktur finden.

```
DOFOR VAR(&I) FROM(1) TO(&NBRENTRYS)
/* Die Verarbeitung der Daten */
SNDPGMMSG MSG('Folgende Daten wurden gelesen:' +
*CAT &SPLNAME *CAT &JOB) +
          TOUSR(RASCHE)
```

Natürlich müssen wir auch dieses Mal die Offset-Position für den nächsten Datenzugriff innerhalb der Schleife um die Länge eines Eintrags erhöhen. Auch hierfür verwenden wir ein Datenstrukturfeld. Die Variable &LENENTRYS enthält die Länge eines Eintrags als Integer-Wert und ist in Kombination mit der Funktion %OFS wieder sehr gut geeignet, die neue Offset-Position zu bestimmen:

```
CHGVAR%OFS(&DTAPTR) VALUE(%OFS(&DTAPTR) +
       &LENENTRYS)
ENDDO
```

2.15.2.4.4

Seite 4

3 Kapitel

Datenhaltung / Datenbank

3

Seite 2

System i DB2 Universal Database (UDB)

Eigentlich hatte die Datenbank der iSeries bis Mitte 1994 niemals einen offiziellen Namen und der Ausdruck „AS/400 Datenbank“ entwickelte sich somit auf natürliche Weise im Laufe der Zeit. Für diejenigen, die ständig mit der iSeries zu tun hatten, war dies zu keiner Zeit ein Problem. Wenn ich mich aber an meine Anfangszeit erinnere, dann erntete ich häufig merkwürdige – ja manchmal gar mitleidige Blicke – von meinen Windowskollegen, wenn ich auf die Frage nach dem Namen der Datenbank keine Antwort geben konnte. Auch die Erläuterung, dass die Datenbank keinen Namen benötigt, weil sie in der Architektur des Servers integriert sei, löste eher Verwirrung aus anstatt anerkennende Blicke.

Wer allerdings mit der DB2 for iSeries vertraut ist, weiß, dass gerade dies eines der herausragenden Merkmale der Datenbank ist. Das Problem der Namensgebung hat IBM bereits Mitte 1994 behoben, indem die „AS/400 Datenbank“ zum offiziellen Teil der relationalen Datenbankprodukte der DB2-Familie erklärt wurde. Die Integration in den LIC Bereich des Servers wurde Gott sei Dank nicht verändert. Heute ist die „DB2 UDB for iSeries“ eine der leistungsfähigsten Datenbanken am Markt. Sie unterstützt den SQL2003-Standard, Trigger, Stored Procedures und alle gängigen Interfaces, so dass die unterschiedlichsten Applikationen denkbar sind. Angefangen von traditionellen Host basierenden Anwendungen über Client/Server Lösungen bis hin zu intelligenten Business Applikationen ist alles denkbar. Die Datenbank verfügt hierfür über zwei unabhängige Schnittstellen:

Die native Schnittstelle stammt noch vom Erbe des Systems/38. Sie enthält die Data Description Specifications (DDS), einige Command Language Befehle, um Datenbankobjekte zu erstellen und zu verwalten und Abfragetools (Query/400). Sie arbeitet allerdings nicht SQL-orientiert. Die native Schnittstelle wird seitens IBM nicht mehr weiterentwickelt. Sie gilt als stabilisiert. Neue Datenbankfunktionalitäten implementiert IBM heute nur noch SQL basierend – entsprechende DDS Syntax existiert nicht.

Bereits zur Markteinführung der AS/400 erhielt der Server neben der nativen Schnittstelle eine unabhängige SQL Schnittstelle. Damals war diese Schnittstelle eher unbedeutend. Heute allerdings ist SQL sowohl die von IBM als auch von der Industrie anerkannte relationale Datenbanksprache. Mit SQL ist es möglich, Daten aufzufinden, ihre Inhalte zu verändern, neue Daten einzufügen und natürlich auch zu löschen. Auch Zugriffsberechtigungen können erteilt und entzogen werden. SQL kann direkt in HLL-Programmen genutzt werden. Man spricht dann von statischen oder dynamischem embedded SQL. Vergessen sollte man auch nicht, dass SQL eine Programmiersprache der vierten Generation ist. So können Sie mit SQL auch eigenständige Programme schreiben. Alle Sprachelemente wie z. B. Schleifenanweisungen, Bedingungen

und dergleichen stehen hierfür zur Verfügung. Der Anwender/Programmierer muss bei Verwendung von SQL nicht mehr angeben „WIE“ die Daten beschafft werden sollen, sondern er definiert lediglich, welche Daten benötigt werden und welche Bedingungen die Daten zu erfüllen haben. Die Zugriffsmethode wird vom SQL Optimizer ermittelt. Den eigentlichen Datenzugriff führt das Data Base Management System oder kurz DBMS durch.

Datenbanklimits DB2 UDB Release V5R4
Um ein Maximum an Portabilität und Flexibilität zu erreichen, wurden im Release V5R4 einige Datenbanklimits erneut erweitert. Nachfolgend ein Überblick über die aktuellen Datenbanklimits V5R4:

Beschreibung	Maximumwert
Anzahl Bytes je Zeile	32.766 (32.740 bei Verwendung variabler Spalten)
Anzahl Zeilen je Member	4 294 967 294 Zeilen
Anzahl Member je physischer Datei	32.767
Anzahl Bytes je Member	1 869 162 846 624 Bytes (ca. 1,7 TB)
Anzahl Spalten je Zeile	8.000 Spalten
Anzahl Spalten je Schlüsseldefinition	120 Felder
Anzahl logischer Dateien je physischer Datei	3686 logische Dateien
Anzahl Bytes je Zugriffspfad	1 099 511 627 776 Bytes (1 TB)
Anzahl verknüpfter Tabellen je SQL Statement	1000
Verknüpfte Tabellen in Views/MQTs	256
Anzahl gleichzeitig geöffneter Tabellen je Zugriff	Ca. 500.000
Anzahl Constraints je Tabelle	300
Anzahl Trigger je Tabelle	300
Anzahl rekursiver Update- und Insert-Trigger	200

Der Begriff relationale Datenbank ist auf einem iSeries Server manchmal irreführend, da das gesamte System eine einzige Datenbank mit separaten Bibliotheken darstellt. Die meisten Menschen, die von einer iSeries Datenbank sprechen, meinen damit jedoch einzelne Bibliotheken, die physische und logische Dateien enthalten.

Die Verwirrung wird noch verstärkt durch die SQL-Anweisung CREATE DATABASE. Tatsächlich erstellt die Anweisung keine Datenbank, sondern ebenfalls nur eine einzelne Bibliothek. Das gleiche Ergebnis erzielen Sie aber auch mit der SQL Anweisung CREATE COLLECTION oder dem Befehl CREATE SCHEMA. Welcher Befehl ist denn nun der richtige? Und wie versteht Ihre iSeries den Begriff der relationalen Datenbank?

Um dies zu beantworten, müssen wir erneut die Frage aufgreifen: Was ist eine relationale Datenbank? Eine relationale Datenbank ist eine Sammlung verschiedener Tabellen und anderer Objekte, deren Daten miteinander logisch in Beziehung stehen. Nirgendwo steht geschrieben, dass die Daten sich in einer einzigen Bibliothek befinden müssen!

IBM unterscheidet auf einem iSeries System drei verschiedene relationale Datenbanktypen:

Systemdatenbank, Userdatenbank, Remote Datenbank
Auf jedem iSeries System finden Sie eine Default Datenbank. Diese *relationale Systemdatenbank (System relational database)* ist stets identisch mit Ihrem lokalen System. Sie enthält alle Datenbankobjekte, die auf Ihrer Festplatte existieren. Geben Sie einmal auf einer 5250-Befehlszeile folgenden Befehl ein:

```
WRKRDBDIRE
```

Anschließend erscheint der folgende Dialog:

```
                  Mit Einträgen im RDB-Verzeichnis arbeiten

Listenanfang bei . . . .   ____________________

Auswahl eingeben und Eingabetaste drücken.
  1=Hinzufügen   2=Ändern   4=Entfernen   5=Details anzeigen
  6=Details drucken

Aus-                       Ferner
wahl  Eintrag              Standort                  Text

 _    ____________________
 _    ITPSERVER            *LOCAL
 =    SQLSERVER            172.16.110.22
 _
```

Der Befehl WRKRDBDIRE

Die Anzeige „Mit Einträgen im RDB-Verzeichnis arbeiten“ enthält alle Einträge im Verzeichnis der relationalen Datenbanken. Der Eintrag *LOCAL bezeichnet die Systemdatenbank. *LOCAL kann nur für einen Eintrag im RDB-Verzeichnis angegeben werden. Häufig ist der Systemdatenbankname identisch mit dem Systemnamen Ihres Servers. Er kann aber auch abweichen. Zusätzlich können Sie Benutzerdatenbanken (User relational database) erstellen, wenn Sie mit unabhängigen ASPs arbeiten. Jeder ASP wird als eine eigenständige relationale Benutzerdatenbank verstanden. Der Name der benutzerdefinierten Datenbank entspricht dem Namen des ASPs. Es sei denn, Sie geben der Datenbank einen eigenständigen Namen. Eine solche benutzerdefinierte Datenbank enthält alle Datenbankobjekte, die sich im entsprechenden ASP befinden. Zusätzlich besteht eine logische Verbindung zu abhängigen

Objekten der Systemdatenbank. Doch trotz dieser logischen Verbindung zwischen der System- und Benutzerdatenbank gibt es ein paar Restriktionen, die Sie beachten müssen, wenn Sie unabhängige ASPs einsetzen:

Erstellen Sie Views und Indizes stets in dem Schema (=Bibliothek), in dem sich die Referenzobjekte befinden.

Datenbanktrigger und Datenbankbedingungen (=Constraint) müssen Sie im selben ASP erstellen, wobei unterschiedliche Bibliotheken verwendet werden können.

Falls Sie mit benutzerdefinierten Datentypen arbeiten (=distinct types), müssen die erstellten Datentypen und die Tabellen, die diese Datentypen verwenden, im selben ASP liegen.

Bei referentiellen Integritätsbedingungen müssen die Parent- und Childtabellen im selben ASP gespeichert sein.

Alle anderen Querverbindungen zwischen der Systemdatenbank und der Benutzerdatenbank sind erlaubt. So kann eine Prozedur, die in der Systemdatenbank liegt, durchaus Objekte der Benutzerdatenbank verwenden und umgekehrt. Mit relationalen Datenbanken (Remote relational database) anderer iSeries Systeme oder fremder Systeme können Remoteverbindungen hergestellt werden. Dafür muss die Remote Datenbank mit dem 5250-Befehl ADDRDBDIRE (add relational database directory entry) im Datenbankverzeichnis registriert sein.

Benutzerschema
Ein Schema dient innerhalb einer Datenbank der Objektorganisation, d. h. man verwendet eine unterschiedliche Anzahl Schemata, um die Datenbankobjekte logisch zu strukturieren. Sie kennen dieses Vorgehen bereits, denn auch im 5250-Umfeld verwenden Sie Bibliotheken, um ihre Objekte zu strukturieren. Schema ist lediglich der SQL Name für eine i5 Bibliothek. Ein Schema wird auch Collection oder Library genannt. Außerdem wird das Schema zur Qualifizierung der Objekte verwendet. Mittels des Schemanamens können Sie die Datenbankobjekte adressieren. Auch die unqualifizierte Objektadressierung ist möglich. Sie sollte Ihnen bekannt sein! Im Hintergrund der unqualifizierten Adressierung sucht das System über die Bibliotheksliste die entsprechenden Objekte. Sollten Sie mit der Objektadressierung im 5250-Umfeld bislang nicht vertraut sein, finden Sie im Kapitel 2.4.2.2 detaillierte Informationen. SQL kann zwar auch mittels der Suchliste auf die Tabellen zugreifen, doch hängt die Verwendung der Suchliste von unterschiedlichen Faktoren ab: Statisches SQL und dynamisches SQL qualifizieren

unterschiedlich. Auch die verwendete Namenskonvention *SYS oder *SQL spielt eine entscheidende Rolle und nicht zuletzt die verwendeten Front-End-Tools bzw. die Konfiguration der Schnittstellen. Wenn Sie nicht Gefahr laufen wollen, versehentlich falsche Tabellen zu manipulieren, rate ich Ihnen, in den SQL-Statements Ihre Objekte zu qualifizieren. Andernfalls können schnell Probleme auftreten und Sie machen sich vielleicht sehr unbeliebt!

Namenskonvention
Damit kommen wir zu einem weiteren Punkt. Sollten wir die SQL-Namenskonvention oder Systemnamenskonvention verwenden?

Gemäß der Systemnamenskonvention dient ein Slash (/) als Trennzeichen zwischen zwei Bezeichnern. Auf einer 5250-Befehlszeile werden Sie die Datenbankobjekte also stets folgendermaßen qualifizieren:

```
DSPFD ITPSQL/DETAIL
```

Der i5 Server unterstützt aber im SQL Umfeld auch den Punkt als Trennzeichen. Hier ist aber vor allem die Einstellung des verwendeten Front-End-Tools maßgeblich. Schauen wir uns hierfür einige unterschiedliche Front-End-Tools an:

Für die interaktive Ausführung Ihrer SQL Abfragen stehen verschiedene Oberflächen zur Verfügung. Eine Möglichkeit bietet hier eine 5250-Bildschirmsitzung. Für den Aufruf der Oberfläche geben Sie auf einer i5-Befehlszeile den folgenden Befehl ein:

```
STRSQL
```

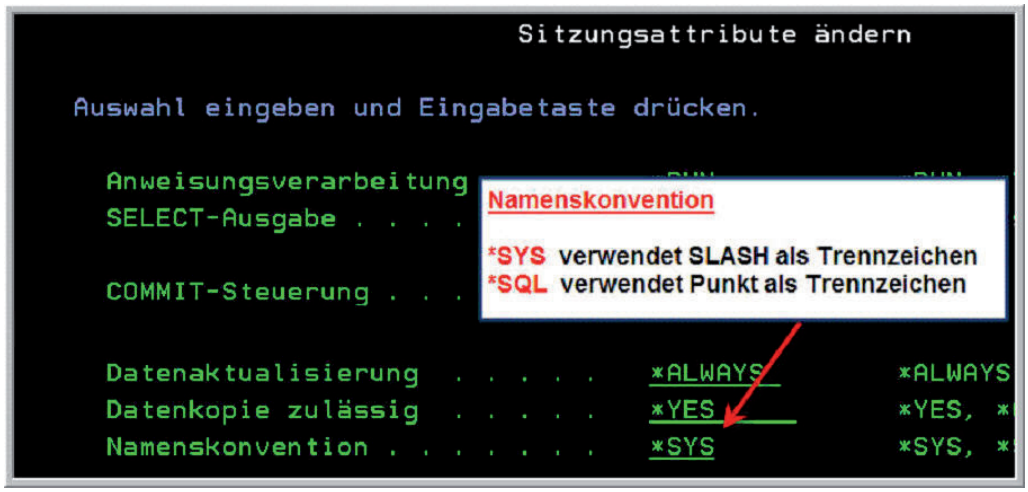
Anschließend sollten Sie folgenden Bildschirm erhalten:

```
                          SQL-Anweisungen eingeben

SQL-Anweisung eingeben und Eingabetaste drücken.
     Aktuelle Verbindung besteht zur relationalen Datenbank EPOSI501.
===>
```

Interaktive 5250-Sitzung

3.2
Seite 6

Die Anzeige „SQL-Anweisungen eingeben“ ist die Hauptanzeige für interaktives SQL. Von dieser Anzeige aus können SQL-Anweisungen eingegeben und verarbeitet werden. Die Funktionstaste F13 „Service“ öffnet den Dialog „Sitzungsattribute ändern“, der es ihnen u. a. erlaubt, die verwendete Namenskonvention für diese Sitzung zu definieren.



Sitzungsattribute einer interaktiven 5250-Sitzung ändern

Aber diese 5250-Oberfläche ist alt und seitens IBM kein zeitgemäßes Tool mehr für die Entwicklung und Ausführung von SQL Abfragen. IBM entwickelt die Oberfläche nicht mehr weiter. Sie ist stabilisiert, was soviel bedeutet wie „Neue Funktionen werden nicht oder nur begrenzt integriert.“ Strategisches SQL Produkt ist iSeries Navigator. iSeries Navigator ist eine leistungsstarke grafische Oberfläche, mit der Sie ohne großen Aufwand mit dem iSeries-Server arbeiten können und problemlos Informationen über Ihre bestehenden Tabellen erhalten, aber auch neue Datenbanken erstellen oder adhoc-Abfragen schreiben können. Des Weiteren bietet die Oberfläche verschiedene Möglichkeiten die Performance Ihrer Datenbankzugriffe zu optimieren. Mit jedem neuen Release kommen weitere Komponenten hinzu, so dass die grafische Oberfläche ständig erweitert und verbessert wird. Dieses Tool erhält zwar im Release V6R1 einen neuen Namen. Er heißt jetzt System i Navigator. Doch auch System i Navigator verliert seine strategische Bedeutung nicht, da es IBM bis zu diesem Zeitpunkt nicht gelungen ist, die Datenbankfunktionalität in den IBM Director zu integrieren. Dies ist zwar geplant, konnte aber bis dato nicht realisiert werden. Wenn Sie also SQL einsetzen, führt kein Weg an System i Navigator vorbei. System i Navigator stellt eine separate Prozedurumgebung zur Verfügung. Grundsätzlich kann jedes SQL Statement hier ausgeführt werden. Die Prozedurumgebung rufen Sie auf, indem Sie eine Verbindung zu Ihrem Server herstellen und anschließend den Eintrag „Datenbanken“ erweitern. Im Taskpad des System i Navigators rufen Sie dann, wie in der folgenden Abbildung dargestellt, die Prozedurumgebung auf:

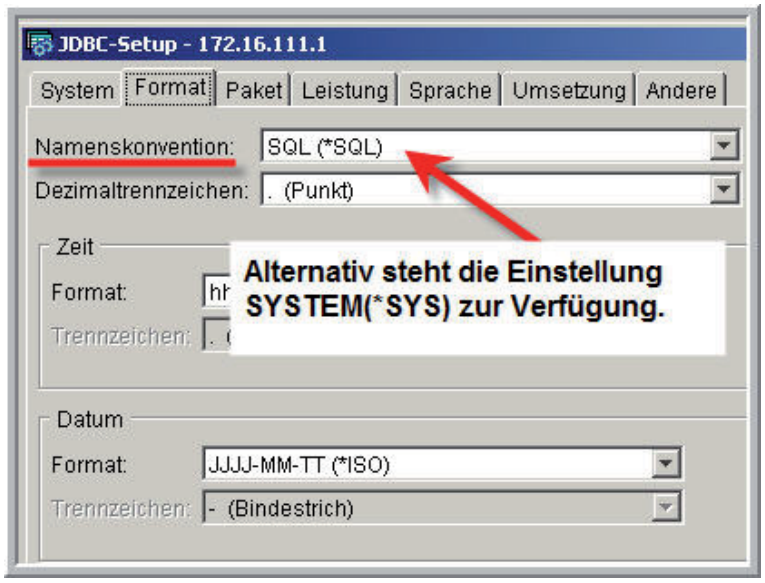
Aufruf der Prozedurumgebung

Das System i Scripting-Tool ist in Java geschrieben und stellt die Verbindung zum i5 Server über einen JDBC-Treiber her. Diese Verbindung sollten Sie zunächst über das JDBC Setup Fenster konfigurieren. Öffnen Sie dieses Fenster, indem Sie im Verbindungs-Menü „JDBC Setup" wählen.

Konfiguration der JDBC-Schnittstelle

Falls Sie andere Menüpunkte sehen als in der Abbildung gezeigt, sollten Sie sich nicht wundern. Ich setze bereits die Version 6 dieses Produktes ein, werde aber immer darauf hinweisen, wenn ein Feature Versionsabhängig ist. Wobei ich davon ausgehe, dass sie mindestens Release V5R3 auf ihrem Server installiert haben.

Das Setup-Menü selbst besteht aus verschiedenen Registern. Es reicht, wenn wir uns im Moment auf die Einstellung der Namenskonvention beschränken. Dafür öffne ich das Register „FORMAT":



Namenskonvention einstellen

Ich ziehe häufig den Punkt als Qualifizierungszeichen vor und verwende daher in vielen Beispielen die Namenskonvention *SQL. Sie sehen aber auch, dass Sie die Namenskonventionen individuell in den Prozeduren oder entsprechenden Front-End-Tools festlegen. Sie sollten sich jedoch für eine Namenskonvention entscheiden und auf keinen Fall mal mit der Systemnamenskonvention und mal mit der *SQL-Namenskonvention arbeiten. Anderenfalls müssten Sie die Syntax Ihres SQL Codes ständig anpassen. Entscheiden Sie sich für eine Methodik und bleiben Sie anschließend dabei!

Systemschema/Katalog
Jede relationale Datenbank enthält Systemschemata. Manchmal spricht man in diesem Zusammenhang auch von Katalogen. Das Schema SESSION ist ein solches Systemschema und wird stets für die Deklaration temporärer SQL Tabellen verwendet. Zusätzlich verwendet Ihr System eine Reihe weiterer Systemschemata, die alle mit dem Buchstaben „Q“ oder der Zeichenfolge „SYS“ beginnen. Sie selbst sollten also besser eine andere Namenskonvention verwenden.

Systemschemata enthalten Tabellen, die nützliche Informationen über die Datenbank liefern. Schauen Sie sich bei Gelegenheit einmal das Schema QSYS2 oder SYSIBM an. Es enthält viele Systemtabellen und Katalogsichten, die wertvolle Informationen liefern. Ich weiß nicht mehr, wann IBM diese Systemkataloge eingeführt hat, aber das spielt auch keine Rolle. Viel wichtiger ist, dass man weiß, dass diese Informationen sich nicht nur auf SQL Tabellen beziehen sondern auch alle physischen und logischen DDS-Dateien verzeichnet sind. So gibt es z. B. die Tabelle SYSCOLUMNS. Sie enthält alle Datenbankfelder aller physischen Dateien/Tabellen. Und auch hier spielt es überhaupt keine Rolle, wie die Tabelle erzeugt wurde. Mussten Sie auch schon einmal herausfinden, in welchen Tabellen (=Dateien) eine bestimmte Spalte (Feld) verwendet wird? Fragen Sie doch zukünftig einfach die entsprechenden Systemtabellen ab, um derartige Informationen zu erhalten:

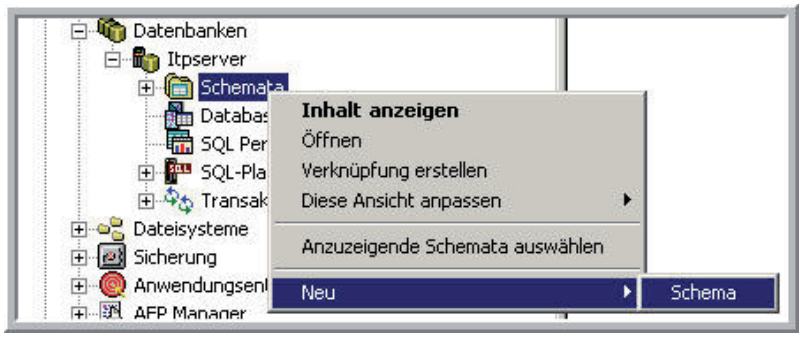
```
SELECT COLUMN_NAME, TABLE_NAME, ORDINAL_POSITION
FROM QSYS2.SYSCOLUMNS
WHERE COLUMN_NAME like 'KDKDNR';
```

COLUMN_NAME	TABLE_NAME	ORDINAL_POSITION
KDKDNR	EML200001	2
KDKDNR	EPSKDPLZ0	2
KDKDNR	EPSKDPLZ0A	3
KDKDNR	EPSKDPLZ1A	2
KDKDNR	EPSKDPLZ2A	2
KDKDNR	EPSKDPLZ3A	3
KDKDNR	EPSKDPLZ4A	2
KDKDNR	EPSKDPLZ5A	2
KDKDNR	EPSKDPLZ6A	2
KDKDNR	EPSKDPLZ7A	2
KDKDNR	EPSKDPLZ8A	2
KDKDNR	EPSKDPLZ9A	2

Das Abfrageergebnis

Ein solches Abfrageergebnis nennt man im Übrigen Result Set. Das Result Set existiert als Tabelle temporär im Arbeitsspeicher. Sie sehen, die Crossreferenztabellen der IBM enthalten nützliche Informationen, die ich im weiteren Verlauf sicherlich noch das ein oder andere Mal verwende und zu einem späteren Zeitpunkt auch detailliert vorstellen werde.

Kommen wir zurück zum Begriff „Schema“. Ich möchte ein eigenes Schema erstellen und setze einmal voraus, dass Sie sich ein klein wenig mit dem System i Navigator auskennen. Um ein neues Datenbankschema zu erzeugen, wählen Sie im Kontextmenü des Menüpunktes „Schemata“ den Eintrag „Neu“ ➜ „Schema“ wie in der Abbildung gezeigt.



Neues Benutzerschema mit System i Navigator

Anschließend erscheint der folgende Dialog:

Ein neues Schema erstellen

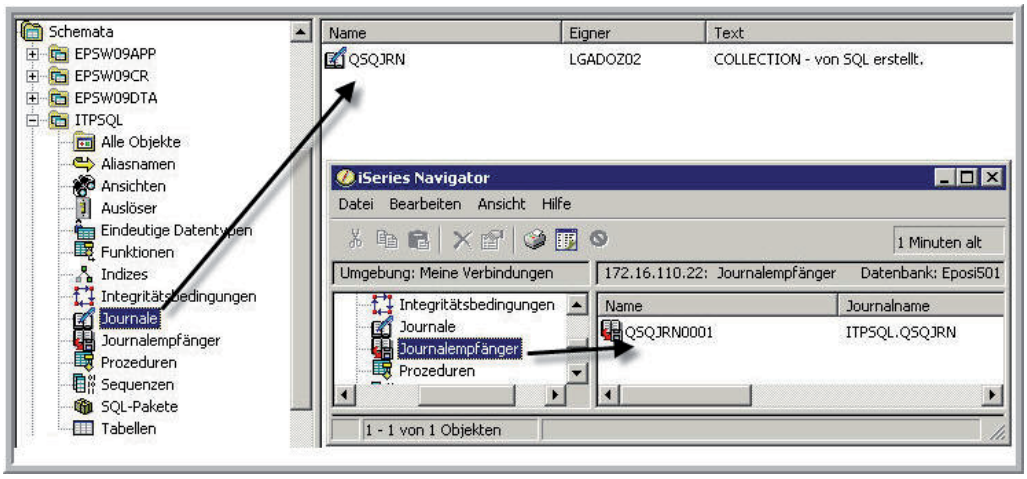
Ich finde der Dialog ist selbsterklärend. Sie geben den Schemanamen an, bestimmen den zu verwendenden ASP und können einen erklärenden Text hinzufügen. Das Gute an allen System i Navigator Komponenten im Datenbankumfeld ist, dass Sie sich immer das zugehörige SQL Statement anzeigen lassen können. So werden Sie feststellen, dass Sie anstelle der grafischen Oberfläche auch den folgenden SQL Befehl verwenden könnten:

```
CREATE SCHEMA schemaname
```

Alternativ können Sie auch im SQL Statement anstelle des Wortes SCHEMA den Begriff COLLECTION oder DATABASE verwenden. COLLECTION ist ebenso ein Synonym für den Begriff Schema wie das Wort DATABASE. Allerdings entsprechen die Schlüsselworte COLLECTION und DATABASE nicht dem ANSI-Standard, sondern stammen aus früheren Zeiten.

Wenn Sie mittels SQL ein Schema erzeugen, haben Sie ein *LIB-Objekt erstellt. Sie können die SQL Anweisung also mit dem i5/OS-Befehl „Create Library“(CRTLIB) gleichsetzen. Nichts wird darauf hinweisen, dass die Bibliothek mit einem SQL Statement erzeugt wurde.

Allerdings gibt es Unterschiede, wenn Sie sich den Inhalt des Schemas anschauen. Das SQL Schema ist nicht leer, wie Sie es gewohnt sind, sondern enthält Objekte.



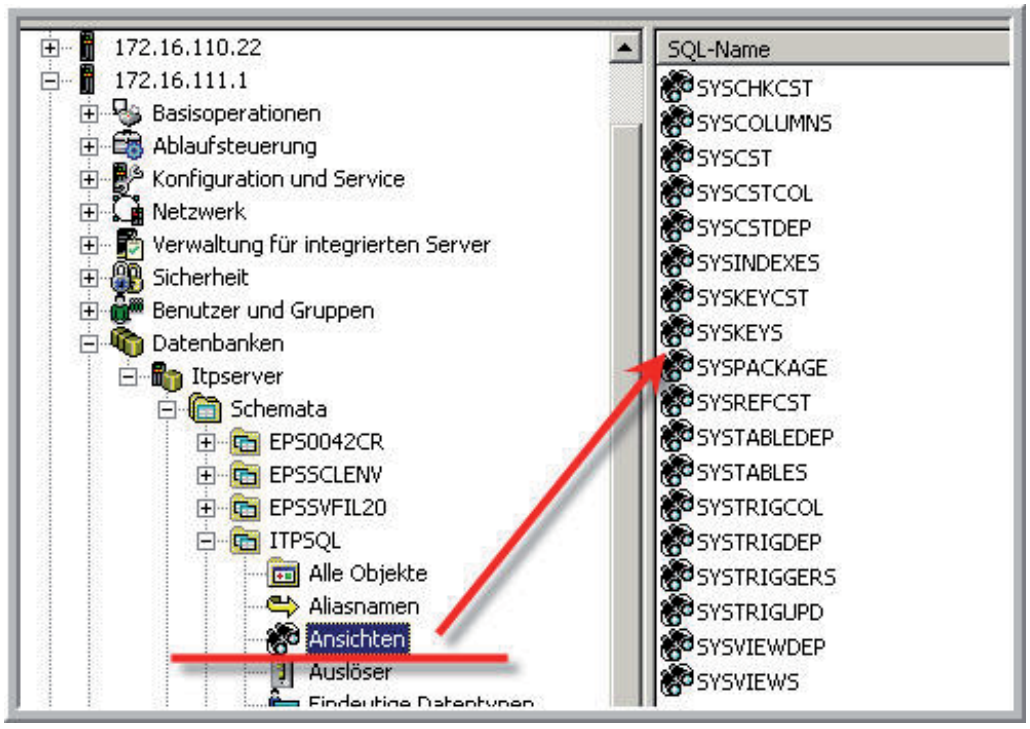
Jedes SQL Schema enthält ein Journal

In einem Schema existiert das Journal QSQJRN und der zugehörige Empfänger QSQJRN001. Alle Tabellen, die wir später in dem Schema erstellen, werden automatisch aufgezeichnet. Dabei ist es egal, ob Sie die Tabellen mit DDS oder SQL erstellen. Die Empfänger werden bei jeder Datensicherung automatisch gewechselt und gesichert. Sie müssen dann nur in regelmäßigen Abständen alte Empfänger bereinigen. Falls Sie jetzt glauben, diese Tatsache sei SQL spezifisch, so irren Sie sich! Erweitern Sie einmal die Parameter des Befehls CRTLIB, dann werden Sie feststellen, dass auch der CL-Befehl diese Möglichkeit vorsieht! Lediglich die Defaulteinstellungen sind unterschiedlich.

Falls Sie keine Aufzeichnungen Ihrer Tabellen wünschen, müssen Sie die Journalisierung manuell stoppen oder eine Standardbibliothek erstellen, die kein Journal enthält. Doch bedenken Sie, dass viele SQL Features die Journalisierung voraussetzen und beachten Sie, dass Sie zukünftig mit einem SQL-Statement eine Vielzahl an Zeilen manipulieren. Häufig lassen sich fehlerhafte Manipulationen nur korrigieren, wenn die Tabellen aufgezeichnet wurden. Weiterführende Informationen zum Thema „Journalisierung“ enthält das Kapitel 3.5.6.

3.2
Seite 12

Schauen Sie sich noch ein wenig genauer den Inhalt eines Schemas an. Ich öffne zu diesem Zweck den Eintrag „Ansichten“ des soeben erstellten Schemas:



Katalogsichten des SQL Schemas

Jedes SQL-Schema enthält logische Sichten. Sie beziehen sich nur auf das jeweilige Schema, zeigen Ihnen somit einen Ausschnitt der bereits besprochenen Systemkataloge.

Fassen wir kurz zusammen:

SQL		i5/OS
	Synonym	
SCHEMA	COLLECTION oder DATABASE	Bibliothek
CREATE SCHEMA (es wird ein Objekt *LIB mit Journal und Katalogsichten erstellt)	CREATE COLLECTION (es wird ein Objekt *LIB mit Journal und Katalogsichten erstellt)	CRTLIB (es wird ein leeres Objekt *LIB erstellt)

Tabellen

Die Tabelle ist die Grundlage des relationalen Datenbanksystems. Die Tabelle selbst ist eine Sammlung von Spalten, die Sie normalerweise als Felder bezeichnen. Inhalt der Tabelle sind viele Zeilen, die Sie als Datensätze kennen. Die Begriffe Datensatz und Zeile bzw. die Begriffe Spalte und Feld sind beliebig austauschbar.

Im 5250-Umfeld sprechen wir eher von physischen Dateien und weniger von Tabellen. Physische Dateien werden in der Regel mit DDS erzeugt. DDS ist eine mächtige Sprache, die sowohl für die Beschreibung physischer als auch logischer Dateien verwendet wird. Nur entwickelt IBM diese Sprache nicht weiter! Dateien können wir aber auch mit SQL erstellen:

```
CREATE TABLE ITPSQL.TEST
               (
                  Spalte1 CHAR(10),
                  Spalte2 DEC(10, 2),
                  Spalte3 CHAR(8)
               )
```

Die Anweisung CREATE TABLE erlaubt es uns, Tabellen zu erzeugen. Im Beispiel wird die Tabelle Test im Schema ItpSQL erstellt. Die Tabelle enthält drei Spalten – zwei alphanumerische und eine numerische Spalte. Vergleichbar ist die Anweisung mit dem i5/OS Befehl „Create physical file“ (CRTPF). Zugegeben, die Anweisung ist einfach, aber an dieser Stelle soll es auch noch nicht um die Erstellung von Tabellen gehen, sondern um eine erste Begriffsdefinition.

Wie verhält es sich nun aber mit den Begriffen physische Datei und Tabelle? Sind diese Begriffe wirklich beliebig austauschbar? Aus Sicht des Betriebssystems i5/OS können wir die Frage mit Ja beantworten. Sie werden keine Unterschiede in der Handhabung der Objekte feststellen, denn alle SQL Tabellen sind *FILE-Objekte. Ob ein Dateiobjekt mittels DDS oder SQL erzeugt wurde, lässt sich lediglich anhand der Dateibeschreibung feststellen.

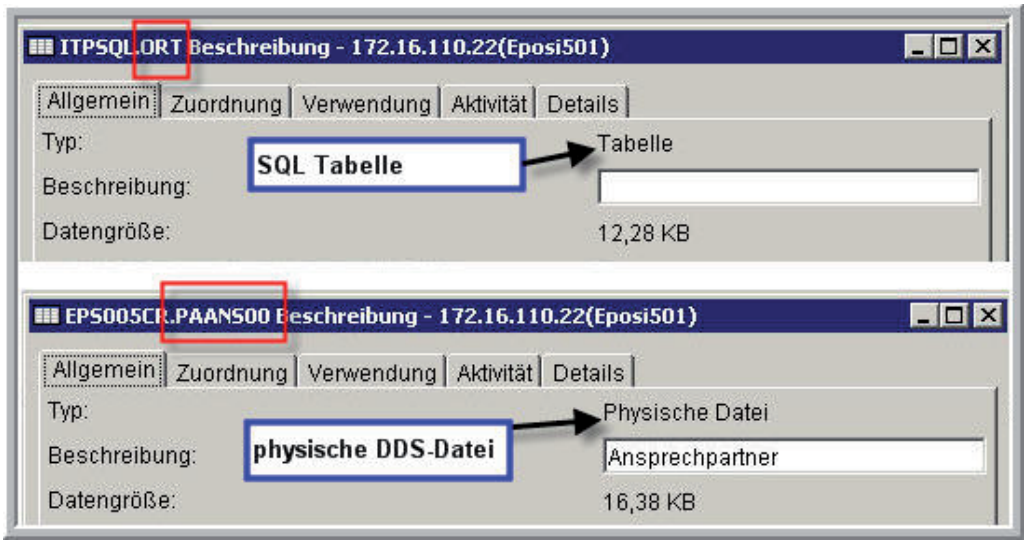
Geben Sie hierfür folgenden Befehl auf einer 5250-Befehlszeile ein:

```
DSPFD ITPSQL/TEST
```

Anschließend erhalten Sie einen 5250-Bildschirm mit einer Vielzahl von Informationen. Nur anhand der Dateiattribute können Sie feststellen, ob die Datei mit einem SQL Befehl erstellt wurde:

Dateibeschreibung anzeigen (DSPFD)

Alle SQL Tabellen erhalten die SQL-Dateiart „TABLE“. Auch der Navigator unterscheidet in diesem Punkt:



Unterscheidung zwischen physischer Datei und SQL Tabelle

Weitere Unterschiede gibt es nicht. Auch ein Hochsprachenprogramm wird keine Unterschiede zwischen einer physischen DDS-Datei und einer SQL-Tabelle feststellen.

Falls Sie einmal ermitteln müssen, welche Ihrer Tabellen mit SQL und welche mittels DDS erzeugt wurden, können Sie hierfür die Systemtabelle SYSTABLES im Schema QSYS2 nutzen. Sie enthält alle Tabellen Ihres Systems. Dabei unterscheidet der Eintrag in der Spalte TABLE_TYPE zwischen logischen DDS-Dateien (L), physischen DDS-Dateien (P) und SQL-Tabellen (T).

Fassen wir auch hier kurz zusammen:

SQL		i5/OS
Deutsch	Englisch	
Zeile	Row	Datensatz
Spalte	Column	Feld
Tabelle	Table	Physische Datei

View/Sicht

Neben den Tabellen existieren weitere Datenbankobjekte, so genannte Sichten oder Views. Der Sinn von Sichten liegt darin, den Benutzern andere Betrachtungsweisen der Daten zu ermöglichen, ohne dass die Tabellen real existieren. Mit der SQL Anweisung CREATE VIEW erstellen Sie ein Datenbankobjekt, das keine Daten enthält, sondern nur die Regeln für den Abruf der Daten. Nachdem die View erstellt ist, kann Sie – mit einigen Ausnahmen – genutzt werden wie jede physische Tabelle. Vieles an einer SQL View erinnert an eine logische DDS-Datei. Tatsächlich erstellen Sie auf dem i5-Server auch eine logische Datei – ähnlich den logischen DDS-Dateien. Dennoch gibt es einen entscheidenden Unterschied zur logischen DDS-Datei. Sie werden in einer SQL View niemals eine Sortierfolge finden – die Klausel ORDER BY ist verboten. Versuchen Sie es dennoch, erhalten Sie entsprechende Fehlermeldungen. Aus diesem Grund erzeugt eine SQL-Sicht aber auch keinerlei System-Overhead, denn der Zugriffspfad einer SQL View muss seitens des Betriebssystems nicht gepflegt werden. Daher können Sie auch eine beliebige Anzahl SQL-Sichten erzeugen, ohne dass die System-Performance beeinträchtigt wird. Auf den Begriff der View werde ich zu einem späteren Zeitpunkt noch zurückkommen.

Index

Werden SQL Abfragen zu langsam ausgeführt, so kann es sinnvoll sein, über Sortierspalten oder Selektionsspalten einen Index zu legen. Ein Index hilft dem DBMS die benötigten Daten schneller zur Verfügung zu stellen. Aber Vorsicht! SQL kennt verschiedene Indextypen. SQL unterscheidet zwischen Binärindizes und Encoded Vector Indizes (EVI). Es gibt abgeleitete Binärindizes und temporäre Indizes und auch einige logische DDS-Dateien können als Index verwendet werden – selbst dann, wenn System i Navigator dies nicht zeigt. Hinzu kommt, dass der SQL Index nicht direkt in einer SQL Abfrage verwendet wird. Er arbeitet im Hintergrund und Sie haben nur indirekten Einfluss auf seine Verwendung. Schauen Sie sich dazu die folgenden Beispiele an:

Zunächst erstelle ich zwei Binärindizes:

```
CREATE INDEX I_PLZ ON ITPSQL.ORT(PLZ);
CREATE INDEX I_LKZ ON ITPSQL.ORT(LKZ);
```

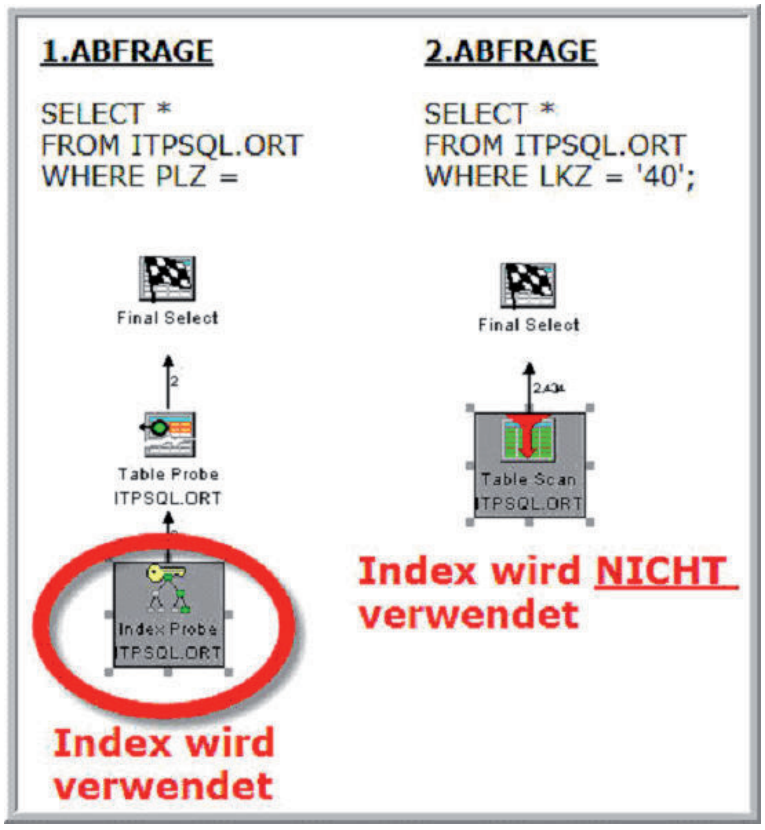
Sie können diese Binärindizes mit einfachen logischen DDS-Dateien vergleichen. Der Index „I_PLZ“ ist nach der Postleitzahl (PLZ) sortiert. Der zweite Index enthält die Spalte Länderkennzeichen (LKZ). Für eindeutige Indizes müsste ich das Statement lediglich ein wenig erweitern: CREATE UNIQUE INDEX

3.2
Seite 16

Jetzt führe ich die folgenden Abfragen aus:

```
select * from itpsql.ort where plz = ‚83646‘;
select * from itpsql.ort where LKZ = ‚40‘;
```

Die erste Abfrage liefert als Result Set alle Orte mit der Postleitzahl ‘83646’. Die zweite zeigt alle Zeilen mit dem Länderkennzeichen ‚40’. Für beide Abfragen steht ein entsprechender Index bereit. In der Abfrage selbst verwende ich aber die physische Datei oder um es mit den SQL Begriffen zu sagen: Ich verwende in den Abfragen immer die Tabellen. Falls Sie versuchen den SQL Index in der FROM-Klausel zu verwenden, erhalten Sie eine Fehlermeldung. Schauen wir uns jetzt an, was im Hintergrund passiert:



Indexverwendung

Sie sehen, obwohl die Abfragen fast identisch sind und für beide Abfragen ein Index zur Verfügung steht, fällt die Ausführung unterschiedlich aus!

Ein Index wirkt sich aber nicht nur positiv aus. Er kann Ihr System auch negativ beeinflussen, denn wenn Zeilen in einer indizierten Tabelle geändert, gelöscht oder hinzugefügt werden, so benötigt die gleichzeitig erforderliche Pflege der Indextabellen natürlich Zeit. Die Folge: Ihre INSERT, UPDATE und DELETE Anweisungen werden langsamer. Wenn Sie aber nicht wissen, welche Indizes erforderlich sind und welche nicht, beeinflussen Sie die Performance Ihres Servers negativ, ohne es zu wollen. Wenn Sie also bislang wenig oder keine SQL Erfahrungen haben, sollten Sie zunächst auf die Erstellung der Indizes verzichten. Bevor Sie sich dann entschließen geeignete Indizes aufzubauen, sollten Sie sehr sorgfältig das Kapitel 3.4 lesen und sich einen genauen Überblick über die verschiedenen Indextypen und deren Wirkungsweise verschaffen.

Abschließend können wir Folgendes feststellen:

SQL	i5/OS
Schema, Library, Schema, Collection	Bibliothek
Tabelle, Table	Datei
--	Satzformatname
Zeile, Row	Datensatz
Spalte, Column	Feld
Sicht, View	Logische Datei
Index	

3.2
Seite 18

Daten auswerten

Das SELECT-Statement

Eine der wichtigsten Aufgaben in einem Datenbanksystem ist das Wiederauffinden von gespeicherten Daten, das so genannte „Datenretrieval“ oder „Selektieren von Informationen“. Die SELECT-Anweisung übernimmt diese Aufgabe in ihren mannigfaltigen Formen. Zu betonen ist bei diesem Statement, dass durch den Anwender definiert wird, welche Daten in welcher Zusammenstellung gesucht werden sollen und nicht wie auf die Informationen zuzugreifen ist. Durch dieses Vorgehen wird sichergestellt, das eine Prämisse von SQL, nämlich seine plattformübergreifenden Fähigkeiten, erreicht werden können – und zwar unabhängig von den unterschiedlichsten Speichertechnologien. Das SELECT-Statement erschöpft seine Fähigkeiten nun nicht darin, ein direktes Beschaffen von Daten durchzuführen, es findet seinen Platz auch in der Erstellung von Ansichten oder auch virtuellen Tabellen. Eine sehr einfache Ausprägung des SELECT könnte so aussehen:

```
SELECT * FROM epswdsc/lager
```

Diese Syntax liefert den Inhalt aller Spalten der Tabelle „Lager“ und zudem, ohne eine bestimmte Auswahl zu treffen, alle Zeilen der Tabelle. Besteht nur ein Interesse an bestimmten Spalten, so werden diese als kommagetrennte Liste nach dem Schlüsselwort SELECT wie folgt angegeben:

```
SELECT REGAL, FACH, ANZAHL FROM epswdsc/lager
```

In diesem Fall werden lediglich die Inhalte der Spalten REGAL, FACH und ANZAHL zurückgegeben. Die Angabe, aus welcher Tabelle Zeilen zu lesen sind, erfolgt nach dem Schlüsselwort FROM durch die Angabe des Tabellennamens.

Die Angabe des * in der Spaltenliste liefert alle Spalten in der Reihenfolge, wie sie in der Tabellendefinition angegeben sind. Wird eine Spaltenliste definiert, so kann eine beliebige Reihenfolge vorgegeben werden. Nun kann es sein, dass der Anwender sich nicht einfach alle Zeilen zurückgeben lassen möchte, sondern ihn interessiert vielleicht nur, welche Materialien im Regal mit der Nummer 4711 eingelagert sind. Für einen solchen Zweck setzt man eine WHERE-Bedingung ein, die die entsprechende Selektion vornimmt.

```
SELECT REGAL, FACH, ANZAHL
     FROM epswdsc/lager
     WHERE REGAL = 4711
```

Jede SELECT-Anweisung liefert die Zeilen aus der gewünschten Tabelle in Form eines Result Sets beziehungsweise einer Ergebnistabelle zurück, immer vorausgesetzt, dass Zeilen gefunden werden, die mit den angegebenen Krite-

rien übereinstimmen. Der Spaltenaufbau dieser Ergebnistabelle ergibt sich aus der Spaltenliste; das gilt natürlich auch für den Datentyp. Auf die verschiedenen Datentypen, die in SQL bekannt sind, wird an späterer Stelle noch eingegangen. Das System untersucht für jede Zeile der Quelltabelle, ob die in der WHERE-Bedingung angegebenen Kriterien den logischen Ausdruck WAHR (TRUE) ergeben und übernimmt die betreffende Zeile dann in die Ergebnistabelle. Ergibt die logische Auswertung FALSCH (FALSE), so lehnt das System die betreffende Zeile ab, ohne sie in die Ergebnismenge zu übernehmen. Es stehen diverse Funktionen und Vergleichsoperatoren und logische Operatoren bereit, um auch umfangreiche Bedingungen aufzubauen.

NOT	Stellt eine logische Negation dar.
AND	Verbindet zwei Operanden durch ein logisches „Und".
OR	Verbindet zwei Operanden durch ein logisches „Oder".
BETWEEN	Untersucht, ob ein Inhalt zwischen zwei Grenzwerten zu finden ist.

Als vergleichende Operatoren können <, >, <=, >=, <> verwendet werden.

EXISTS	Kann überprüfen, ob in einer Ergebnistabelle zumindest eine Zeile vorhanden ist oder nicht. Dieses Schlüsselwort findet oft seine Verwendung in so genannten Unterabfragen (SubQueries).
IN	Stellt fest, ob ein Wert in einer bestimmten Menge oder Liste von Vergleichswerten enthalten ist.
LIKE	Führt einen Vergleich einer Zeichenfolge mit bestimmten Zeichenmustern durch.
NULL	Dient zur Überprüfung, ob in einer Spalte der Wert NULL vorliegt.

Eine vollständige Aufzählung der möglichen Schlüsselworte befindet sich in der SQL-Referenz.

Die SELECT-Anweisung kann weiter durch arithmetische Konstrukte erweitert werden. Diese einfachen Berechnungen können sowohl bei der Definition der zu wählenden Spalten vorgenommen als auch in der WHERE-Bedingung eingesetzt werden.

Nehmen wir einmal an, in der Spalte GEWICHT unserer Tabelle LAGER wird angegeben, welches maximale Gewicht in einem Fach nicht überschritten werden darf. Es soll ein bestimmtes Werkstück mit der Teilenummer TNUMMER ‚A0110' und einem Gewicht von 6,00 Einheiten existieren. Bestimmt werden soll das noch belegbare Gewicht in den Fächern, in denen dieses Werkstück eingelagert ist. Damit kann das folgende Statement aufgebaut werden:

```
SELECT REGAL, FACH, ANZAHL, GEWICHT - ANZAHL * 6.00
    FROM epswdsc/lager
    WHERE TNUMMER = ‚A0110‘
```

Aktueller Bestand in der Tabelle LAGER:

```
Anfang auf Zeile . . . . . .
....+....1....+....2....+....3....+....4....+....5....+....6....+....7
      REGAL          FACH   TNUMMER       ANZAHL       GEWICHT
          1             1   A0110              2      1.000,00
          1             2   A0110             15      1.000,00
********   Datenende   ********
```

Ergebnis der Abfrage

Wenn obige Abbildung die Zeilen darstellt, die mit dem Werkstück belegt sind, so liefert unser SELECT-Statement dann folgendes Resultat. Da hier keine Angabe gemacht wurde, unter welchem Spaltennamen der berechnete Ausdruck in der Ergebnistabelle ausgegeben werden soll, trägt das System eine Standardüberschrift ein.

```
Anfang auf Zeile . . . . . .
....+....1....+....2....+....3....+....4....+....5....+....6....+....
      REGAL          FACH        ANZAHL   Numerischer Ausdruck
          1             1             2                 988,00
          1             2            15                 910,00
********   Datenende   ********
```

Ergebnis mit Standardüberschrift

Ein weiteres Beispiel für eine einfache Berechnung haben wir, wenn wir wissen wollen, in welchen Fächern des Regals 4 das lagernde Gewicht 100 kg überschreitet. Es soll hier ein Gewicht von 6 kg pro eingelagerten Teil angenommen werden. Die Datenbasis hat in diesem Fall das folgende Bild:

```
Anfang auf Zeile . . . . . .
....+....1....+....2....+....3....+....4....+....5....+....6....+....7
      REGAL          FACH   TNUMMER       ANZAHL       GEWICHT
          1             1   A0110              2      1.000,00
          1             2   A0110             15      1.000,00
          4             1   A0120              7      1.000,00
          4             1   A0120            120      1.000,00
          4             1   A0120            310      1.000,00
```

Datenbasis

3.2.1
Seite 4

```
SELECT REGAL, FACH
    FROM epswdsc/lager
    WHERE ANZAHL * 6 > 100
    AND REGAL = 4
```

Die Ausführung dieses Statements liefert dann folgende Ergebnistabelle:

Ergebnis der Abfrage

Aus Performancegründen wird hier ein ResultSet mit mehreren identischen Zeilen geliefert, wobei ich dieses Ergebnis durch die bestimmte Wahl der auszugebenden Spalten provoziert habe. Eigentlich entspricht dieses Ergebnis nicht dem grundlegenden Prinzip relationaler Datenbanken, Daten nicht mehrfach (redundant) zu speichern. Möchte man nun dafür sorgen, dass diese doppelten Zeilen nicht angezeigt werden, sondern immer nur eine Zeile von Mehrfachvorkommen, so kann das durch die Verwendung des Schlüsselwortes DISTINCT erzwungen werden.

```
SELECT DISTINCT REGAL,FACH
FROM epswdsc/lager
WHERE ANZAHL * 6 >100
AND REGAL = 4
```

Dadurch ergibt sich dann folgendes Bild:

Ergebnis nach der Abfrage mit DISTINCT

Nehmen wir an, wir suchen nun nach Stellplätzen, auf denen bisher keine Werkstücke eingelagert wurden. In diesem Fall können wir nach Zeilen in der Tabelle „Lager“ suchen, bei denen die Spalte TNUMMER den Inhalt NULL hat. Achtung, genau genommen darf bei NULL nicht von einem Wert in dem Sinne gesprochen werden, da NULL gerade die Definition – nicht spezifiziert – nicht definiert hat!

Solche Zeilen lassen sich finden, indem die in Frage kommende Spalte auf NULL abgefragt wird. Zu Beginn liegt wieder eine leicht geänderte Datenbasis zugrunde. Beachten Sie bitte die letzten beiden Sätze:

```
Anfang auf Zeile . . . . . .
....+....1....+....2....+....3....+....4....+....5....+....6....+....7
      REGAL          FACH  TNUMMER        ANZAHL       GEWICHT
          1             1  A0110               2      1.000,00
          1             2  A0110              15      1.000,00
          4             1  A0120               7      1.000,00
          4             1  A0120             120      1.000,00
          4             1  A0120             310      1.000,00
          5             1  -                   0        300,00
          5             2  -                   0        300,00
```

Datenbasis

```
SELECT * FROM EPSWDSC/LAGER WHERE TNUMMER IS NULL
```

Das Ergebnis nach Durchführung dieser Abfrage lautet dann:

```
Anfang auf Zeile . . . . . .
....+....1....+....2....+....3....+....4....+....5....+....6....+....7
      REGAL          FACH  TNUMMER        ANZAHL       GEWICHT
          5             1  -                   0        300,00
          5             2  -                   0        300,00
```

Ergebnis der Abfrage

Verknüpfen von mehreren Tabellen

Die bisherigen Beispiele lieferten ihre Ergebnisse aus einer einzelnen Tabelle heraus. Gerade bei modernen relationalen Datenbanken werden die benötigten Informationen aber aus zwei oder mehr Quelltabellen beschafft werden müssen. Diese Verteilung der Daten über eine unter Umständen recht hohe Anzahl von Tabellen ergibt aus Design-Grundsätzen eine relationale Datenbank. Hier können Sie sich die Stichworte Normalisierung und Vermeidung redundanter Datenhaltung schon einmal vormerken. Fürs Erste reicht aber die Vorstellung eines relativ einfachen Gebildes von Kopf- und Positionssätzen aus.

Entwickeln wir ein weiteres kleines Beispiel: In der Datenbank gibt es nun eine weitere Tabelle TEILE, die einem Werkstück mit der Teilenummer TNUMMER ein bestimmtes Gewicht zuordnet. Nehmen wir an, wir wollen wissen, bei welchen Stellplätzen sich das Lagergewicht um weniger als 5 kg vom höchstens zulässigen Gewicht unterscheidet, das auf einem Stellplatz

vorhanden sein darf. Aus der Tabelle LAGER kann die Anzahl der gelagerten Werkstücke und das maximale Gewicht entnommen werden.

Aus der neuen Tabelle TEILE beschaffen wir das Gewicht eines einzelnen Werkstücks.

Die folgende Abbildung zeigt den Inhalt der neuen physischen Tabelle TEILE. Bitte beachten Sie auch den geänderten Inhalt der Tabelle LAGER in der darauffolgenden Abbildung.

Anfang auf Zeile . . . . . .

....+....1....+....2..

TNUMMER	GEWICHT
A0110	10,20
A0120	20,40

Inhalt der Tabelle TEILE

Anfang auf Zeile . . . . . .

....+....1....+....2....+....3....+....4....+....5....+....6....+....7

REGAL	FACH	TNUMMER	ANZAHL	GEWICHT
1	1	A0110	2	1.000,00
1	2	A0110	15	1.000,00
4	1	A0120	98	1.000,00
4	2	A0120	49	1.000,00
5	1	-	0	300,00
5	2	-	0	300,00

Inhalt der Tabelle LAGER

Damit kann nun ein neues SELECT-Statement erstellt werden.

```
SELECT REGAL, FACH
    FROM EPSWDSC/LAGER, EPSWDSC/TEILE
    WHERE (ANZAHL * TEILE.GEWICHT) > (LAGER.GEWICHT - 5)
          AND LAGER.TNUMMER = TEILE.TNUMMER
```

Hier ist zu beachten, dass in den beiden verwendeten Tabellen Spalten mit identischem Namen auftauchen. In solchen Fällen muss dem Spaltennamen durch einen Punkt getrennt der Tabellenname vorangestellt werden. Durch diesen Vorgang wird eine Vollqualifizierung durchgeführt; die Spaltennamen in der Ergebnistabelle sind nun eindeutig.

Die logische Verbindung zwischen beiden Tabellen wird durch die WHERE-Bedingung LAGER.TNUMMER = TEILE.TNUMMER hergestellt. Im Ergebnis werden nur solche Zeilen zurückgeliefert, bei denen dieser logische Vergleich WAHR ergibt, alle anderen werden nicht berücksichtigt.

Wie kann nun die Erzeugung der Ergebnistabelle verstanden werden? Im ersten Schritt werden sämtliche Kombinationen aller Zeilen der zugrunde liegenden Tabellen gebildet. Jede Zeile der Ergebnistabelle ist damit eine Verknüpfung einer Zeile aus LAGER mit einer Zeile aus TEILE (Kreuzprodukt). Letztendlich sichtbar sind dann nur die Zeilen aus diesem Kreuzprodukt, die den angegebenen Bedingungen genügen. Das Ergebnis wird in der nächsten Abbildung verdeutlicht:

Ergebnis der Abfrage (Kreuzprodukt)

Für Statements in der eben angeführten Form gibt es eine weitere Möglichkeit über das Schlüsselwort JOIN, mit dem ebenfalls mehrere Tabellen miteinander verknüpft werden können. Die Spalten, über die eine Verknüpfung der Tabellen erfolgen soll, werden hier in einer ON-Bedingung angegeben. Die Tabellen selbst stehen links und rechts neben dem Schlüsselwort JOIN. Die Ergebnistabelle nimmt bei diesem Aufbau die gleiche Form an und auch die von der Abfrage gelieferten Zeilen entsprechen dem Ergebnis der vorhergehenden Abfrage.

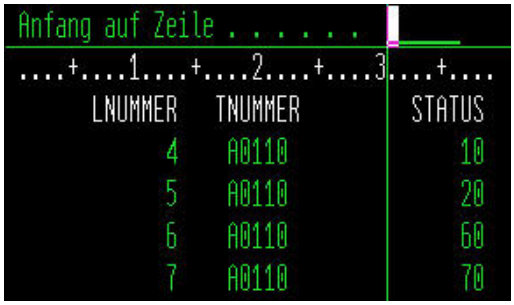
```
SELECT REGAL, FACH
    FROM EPSWDSC/LAGER JOIN EPSWDSC/TEILE
        ON LAGER.TNUMMER = TEILE.TNUMMER
    WHERE (ANZAHL * TEILE.GEWICHT) > (LAGER.GEWICHT - 5)
```

Nun kann ein weiterer Typ einer Verbindung auftreten. Beispielsweise soll das Werkstück mit der Teilenummer A0110 nachbestellt werden. Nehmen wir an, die Daten, welcher Lieferant dieses Teil liefern kann, sind in einer Tabelle LIEFERBAR zu finden. Eine weitere Forderung soll sein, auch in einer weiteren Tabelle LIEFERANT nachzuschauen, ob es dort Lieferanten gibt, die von einer Lieferantenbewertung her mit einem positiven Status versehen sind, der über einem (in diesem Falle willkürlichen) Wert von 50 liegt. Damit Sie sich den Sachverhalt besser verdeutlichen können, sehen Sie im Folgenden zwei Abbildungen aus der Datenbasis mit den neuen Tabellen LIEFERBAR und LIEFERANT.

LIEFERBAR:

Datenbasis LIEFERBAR

LIEFERANT:



Datenbasis LIEFERANT

Jetzt ist es natürlich denkbar, hier mehrere Abfragen nacheinander durchzuführen und dann die einzelnen Ergebnisse mehr oder weniger von Hand zusammenzufügen, aber es ist auch eine andere Variante denkbar. Die Lösung besteht in der Verwendung des Schlüsselwortes UNION. Damit kann aus Tabellen mit identischer Spaltenstruktur eine Ergebnistabelle erzeugt werden, die sämtliche Zeilen aus den beteiligten Tabellen eben nur ein einziges Mal enthält. Für den Fall, dass nur die Lieferantennummer ausgegeben werden soll, lautet die Abfrage wie folgt:

```
SELECT LNUMMER
      FROM EPSWDSC/LIEFERBAR
      WHERE TNUMMER = 'A0110'
UNION
SELECT LNUMMER FROM EPSWDSC/LIEFERANT
      WHERE STATUS > 50
```

Schaut man sich die Inhalte der Tabellen in obigen Abbildungen an, so ist leicht zu erkennen, dass die Abfrage die Lieferantennummern „3“ aus der Tabelle LIEFERBAR und die Nummern „6“ und „7“ aus der Tabelle LIEFERANT zurückgeben sollte. Schauen wir uns das Ergebnis an:

Ergebnis der Abfrage

Spaltenfunktionen

In SQL existiert eine ganze Reihe von Spaltenfunktionen, die auf einzelne Spalten ihren Einfluss haben. Auch hier soll nicht die ganze Reihe vorhandener Funktionen aufgezählt, sondern nur eine kleine Auswahl getroffen werden. So wird man folgende Funktionen beobachten können:

AVG	Diese Funktion bildet den Mittelwert über eine Spalte.
MAX	Berechnet den maximal vorhandenen Wert.
MIN	Bildet analog dazu den minimalen Wert.
SUM	Bildet eine Aufsummierung.
COUNT	Gibt die Anzahl der gewählten Zeilen zurück.

Hier ist zu beachten, dass AVG und SUM lediglich für numerische Spalten angewendet werden können. MIN und MAX können auch auf alphanumerische Spalten bezogen werden. Die COUNT-Funktion erfordert keinen speziellen Datentyp der betroffenen Spalte. So könnte die Frage, wie viele noch nicht durch Werkstücke belegte Stellplätze es noch im Lager gibt, durch die einfache Anfrage

```
SELECT COUNT(*) FROM EPSWDSC/LAGER
      WHERE TNUMMER IS NULL
```

beantwortet werden. Vielleicht soll auch das Gesamtgewicht der in einem Regal eingelagerten Teile ermittelt werden. Zu diesem Zweck dient eine Abfrage der Form:

```
SELECT sum(anzahl * teile.gewicht)
      FROM EPSWDSC/LAGER, EPSWDSC/TEILE
      WHERE LAGER.REGAL = 4 AND
      LAGER.TNUMMER = TEILE.TNUMMER
```

3.2.1
Seite 10

Unterabfragen

Unterabfragen dienen dazu, eine Vorselektion aufgrund bestimmter Kriterien zu betreiben und dann anschließend auf der Ergebnistabelle dieser Grundabfrage aufsetzend ein weiteres SELECT-Statement durchzuführen. Sollen zum Beispiel Stellplätze ermittelt werden, auf denen Werkstücke mit einem Gewicht von mehr als 15 kg eingelagert sind, so muss in einem ersten Schritt bestimmt werden, welche Teile überhaupt schwerer als dieses Grenzgewicht sind. Die Ergebnistabelle einer solchen Abfrage dient dann als Grundlage für die eigentliche Auswahl der Daten in einem zweiten Schritt. Ein kurzer Blick in die Tabelle TEILE liefert die Information, dass nur das Werkstück mit der Teilenummer A0120 schwerer als der Grenzwert ist. Die Tabelle LAGER besagt, das dieses Teil im Regal 4 auf den Stellplätzen 1 und 2 eingelagert ist.

```
SELECT REGAL, FACH
    FROM EPSWDSC/LAGER
        WHERE TNUMMER IN
            (SELECT TNUMMER
                FROM EPSWDSC/TEILE
                    WHERE GEWICHT > 15.0)
```

Das Ergebnis der Abfrage stellt sich folgendermaßen dar:

Ergebnis der Abfrage

An dieser Stelle soll erwähnt sein, wie diese Abfrage von der Systemseite her durchgeführt wird. Im ersten Schritt wird die innere (Unter-)Abfrage ausgeführt, die als Ergebnistabelle erzeugt wird und die lediglich aus einer einzelnen Spalte aufgebaut ist. Zu beachten ist, dass der Datentyp dieser Spalte mit dem der Spalte TNUMMER in der Tabelle LAGER identisch ist. Über den Operator IN wird nun für jede Zeile der Tabelle LAGER geprüft, ob das dort angegebene Werkstück in der bereits generierten Ergebnistabelle der Unterabfrage auftaucht. Trifft diese Bedingung zu, so wird die entsprechende Zeile in die Ergebnismenge der kompletten Abfrage übernommen.

Soll nun ermittelt werden, welche Artikel einen derart geringen Lagerbestand (in Summe) haben, dass sie nachbestellt werden müssen (das heißt, die Bestelluntergrenze ist bei diesen Artikeln unterschritten), so werfen sich neue Fragen

auf. So muss in diesem Fall für jedes Teil der aktuelle Bestand berechnet werden UND geprüft werden, ob die Bestellgrenze unterschritten ist. Das läuft darauf hinaus, dass die entsprechende Unterabfrage nicht nur einmal laufen, sondern für jeden Artikel neu ausgeführt werden muss, der gerade in der übergeordneten SELECT-Anweisung im Zugriff ist. Es muss also ein Verweis oder eine Referenz hergestellt werden. Erreichen kann man das durch die Vergabe eines Labels in der übergeordneten Abfrage. Um die Situation etwas zu vereinfachen, wird hier einfach ein Minimalbestand von 30 Stück für jeden Artikel im Lager angesetzt. Ein kurzer Blick zurück auf die vorhergehende Abbildung „Inhalt der Tabelle Lager" lässt erkennen, dass A0110 mit einem Gesamtbestand von 17 Stück unter der Bestellgrenze liegt. Die folgende Abfrage sollte also diese Teilenummer als nachzubestellenden Artikel liefern.

```
SELECT TNUMMER
       FROM EPSWDSC/TEILE TEMP
       WHERE 30 >
             (SELECT SUM(ANZAHL)
                   FROM EPSWDSC/LAGER
                   WHERE TNUMMER = TEMP.TNUMMER)
```

Ergebnis der Abfrage

Das in der übergeordneten Abfrage eingeführte Label TEMP sorgt nun dafür, dass die Summenberechnung in der Unterabfrage nur für das bestimmte Werkstück durchgeführt wird, welches in der Hauptabfrage zurzeit untersucht wird. Das Ergebnis der Unterabfrage ist eine Ergebnistabelle mit einer numerisch definierten Spalte, die ihrerseits wieder mit dem Minimalbestand 30 verglichen werden kann. Achtung, eine weitere Vereinfachung ist die Annahme, dass jeder Artikel mindestens einmal im Lager eingelagert wurde. In der Realität könnte es aber durchaus sein, dass das nicht bei jedem Teil der Fall ist.

Gruppierungen von Tabellen

Die Aufgabe einer Gruppierung taucht dann auf, wenn eine Berechnung für eben eine bestimmte Gruppe von Zeilen durchgeführt werden soll. Das kann der Fall sein, wenn für jedes Werkstück ermittelt werden soll, welche Anzahl im Lager vorhanden ist. Ein Ansatz zur Lösung ist die GROUP-BY-Klausel, die innerhalb eines SELECT-Statements eingesetzt werden kann. Dabei gibt

es jedoch Einschränkungen in Bezug auf die Definition der Spaltenliste in der Anweisung. So sind in der Ergebnistabelle nur Spalten zugelassen, die auch zur Gruppierung selbst verwendet werden sowie Spalten, die durch den Einsatz von Spaltenfunktionen erzeugt wurden. Bauen wir nun ein Statement auf, das die Anzahl der eingelagerten Artikel ermittelt – bezogen auf jeden einzelnen Artikel. In diesem Fall muss eine Gruppierung über die Teilenummer durchgeführt werden und wir können dafür sorgen, dass Stellplätze ohne eingelagerte Artikel nicht in die Abfrage mit aufgenommen werden. Die Teilenummer auf solchen Stellplätzen ist nämlich NULL! Über eine HAVING-Klausel im Gruppierungsabschnitt der Anweisung kann eine solche Bedingung definiert werden (analog zur WHERE-Bedingung). Es stellt sich dann natürlich auch die Frage, ob es nicht doch interessant ist, von diesen nicht belegten Stellplätzen zu erfahren.

```
SELECT TNUMMER, SUM(ANZAHL)
       FROM EPSWDSC/LAGER
       GROUP BY TNUMMER
       HAVING TNUMMER IS NOT NULL
```

Auch hier sei durch eine Abbildung das Ergebnis verdeutlicht:

Ergebnis der Abfrage

Sortierungen

Die Daten in Form einer Ergebnistabelle, die von einer SELECT-Anweisung zurückgegeben werden, sind im Allgemeinen ungeordnet. Eine Sortierung lässt sich durch eine ORDER-BY-Klausel einfach erreichen.

```
SELECT * FROM LIEFERANT
    ORDER BY LNUMMER
```

Ohne weitere Angaben ist die Sortierung aufsteigend. Wird DESC hinzugefügt, so erfolgt die Sortierung absteigend.

Ansichten (Views)

Eine View enthält – wie schon weiter oben erwähnt – keine physisch gespeicherten Daten, sondern sie erstellt lediglich eine logische Sicht auf vorhandene Informationen in physischen Tabellen. Es kann auch eine Sicht erstellt werden, die auf einer anderen View beruht – zumindest unter der Voraussetzung, dass die anderen Sichten unabhängig von dieser Definition generiert werden können.

Eine solche virtuelle Tabelle enthält nur Informationen darüber, wie die zu liefernden Daten aus den vorhandenen physischen Tabellen oder anderen Sichten heraus zusammengesetzt werden sollen. Im Unterschied zur bisher besprochenen SELECT-Anweisung, die ja interaktiv definiert und ausgeführt wurde, wird eine View jedoch dauerhaft in der Datenbank selbst abgelegt. Allerdings liegt einer Ansicht eine SELECT-Anweisung zugrunde, um die Daten zu beschaffen. Sobald ein Anwender oder ein Prozess auf die View zugreift, wird die Ergebnistabelle in diesem Moment aufgrund der hinterlegten Definition zusammengestellt. Mit Hilfe von fest abgelegten Ansichten kann nun eine spezielle Sicht der Dinge erstellt werden, die auf einzelne Benutzer, Benutzergruppen und deren spezielle Interessen zugeschnitten ist. Hier findet sich also schon ein erster Ansatz in Richtung auf eine Unterscheidung zwischen physischen und logischen Betrachtungsweisen eines in einer relationalen Datenbank abzubildenden Systems. Hier ein Beispiel, wie eine View erstellt werden kann, die einem Lagermitarbeiter Informationen zu einem Stellplatz liefern kann:

```
CREATE VIEW EPSWDSC/LAGERORT
      AS SELECT TNUMMER, REGAL, FACH
            FROM EPSWDSC/LAGER
                  WHERE ANZAHL > 0
```

Soll eine View wieder gelöscht werden, so kann die Anweisung DROP wie folgt aufgerufen werden. Hier wird natürlich nur die eigentliche View gelöscht, aber nicht die zugrunde liegenden Daten.

```
DROP VIEW EPSWDSC/LAGERORT
```

3.2.1
Seite 14

Das SELECT-Statement im Detail

Wenn Sie sich bis zu dieser Stelle der Ausführungen über das SELECT-Statement gearbeitet haben, so haben Sie sich einen groben Überblick über die Möglichkeiten dieser Anweisung verschaffen können. Damit sind die Anwendungsmöglichkeiten aber bei weitem noch nicht erschöpft. Wir wollen nicht bis in die letzten kleinen Details gehen, aber eine weitergehende Untersuchung sollte doch erfolgen. Wenn wir die Behandlung des Statements an dieser Stelle abschließen würden, so käme ein hoher Anteil nicht zum Tragen.

Folgender grundsätzlicher Aufbau des SELECT ist zu beachten:

1) Der SELECT-Abschnitt

2) Der FROM-Abschnitt des Statements

3) Die WHERE-Bedingung

4) Die GROUP BY Klausel

5) Die HAVING-Klausel

Beginnen wir mit dem SELECT-Abschnitt.

SELECT ALL… ist die Standardangabe für die anschließend folgende Definition der zu selektierenden Spalten einer Ergebnistabelle. Sie hat zur Folge, dass alle Sätze, die mit einer in der WHERE-Klausel angegebenen Bedingung übereinstimmen, als Ergebnis zurückgegeben werden. Die Anzeige von doppelten Zeilen kann durch DISTINCT vermieden werden. Von jeder auftretenden Zeile wird jeweils nur eine einzige geliefert, alle anderen doppelt auftretenden Sätze werden dadurch ausgeblendet. Obwohl ein NULL-Wert eigentlich mit keinem anderen Wert sinnvoll verglichen werden kann, filtert DISTINCT auch solche Sätze. Betrachten wir dazu wieder ein Beispiel aus der Datenbank. Folgendes SELECT-Statement wird in der interaktiven Umgebung eingegeben.

```
SELECT TNUMMER FROM EPSWDSC/LAGER
```

SELECT-Statement

3.2.1.1
Seite 2

Es liefert folgende Datensätze zurück.

Ergebnis der Abfrage

Beachten Sie bitte die letzten beiden Zeilen, die in der Spalte TNUMMER jeweils NULL-Werte beinhalten. Nun wird die SELECT-Anweisung durch DISTINCT abgewandelt.

```
SELECT DISTINCT TNUMMER FROM EPSWDSC/LAGER
```

Das abgewandelte Statement

Nach der Ausführung erhält man folgendes Bild.

Ergebnis mit abgewandeltem SELECT

Ergebnis der Abfrage

Durch DISTINCT wurden also die Zeilen mit NULL-Werten in den Teilenummern auf einen einzigen Satz reduziert, ebenso die Sätze mit der Artikelnummer A0120 und A0110.

Zu bedenken ist bei diesem Schlüsselwort, dass es nicht verwendet werden darf, wenn eine der zurückzugebenden Spalten den Datentyp LOB oder DATALINK besitzt.

In manchen Situationen ist es eventuell notwendig oder wünschenswert, den Namen einer Spalte zu ändern, also eine Umbenennung vorzunehmen. Dies wird durch die Zuteilung der gewünschten neuen Bezeichnung mit AS ermög-

licht. Beispielsweise soll die Spalte TNUMMER als Teilenummer zurückgegeben werden.

```
SELECT TNUMMER AS TEILENUMMER FROM EPSWDSC/LAGER
```

Umbenennung einer Spalte

Folgendes Resultat wird dadurch erzielt.

Ergebnis nach Verwendung von AS

Ergebnis der Abfrage

Der neu vergebene Spaltenname muss nun weder eindeutig sein noch ist es notwendig, ihn zu qualifizieren. Es muss also kein Tabellenname mit einem Punkt vorangestellt werden. Zur Verdeutlichung vergebe ich für die Spalte REGAL und für die Spalte FACH aus der Tabelle LAGER einfach einmal identische Namen durch die Verwendung von AS. Dazu wird das folgende Statement eingegeben.

Doppelte Spaltennamen

```
SELECT FACH AS NEUESPALTE,
       REGAL AS NEUESPALTE
FROM EPSWDSC/LAGER
```

Eingabe doppelter Spaltennamen

Das entsprechende Resultat sieht dann wie folgt aus.

Grundwerk 10/2004

3.2.1.1
Seite 4

Ergebnistabelle mit doppelten Spaltennamen

Ergebnis der Abfrage

Die Anwendung der Spaltenliste im SELECT ist davon abhängig, ob anschließend eine GROUP BY-Klausel oder eine HAVING-Klausel verwendet wird.

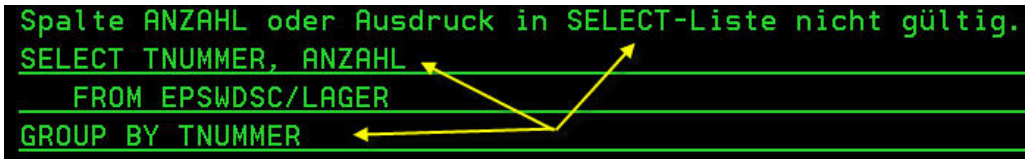
Wird GROUP BY/HAVING verwendet, so muss jeder Spaltenname in der Feldliste einem Ausdruck in der Gruppierung entsprechen. Auf eine solche Spalte in der Ergebnistabelle kann ein Operator angewendet werden.

```
SELECT TNUMMER, ANZAHL * 10
  FROM EPSWDSC/LAGER
GROUP BY TNUMMER, ANZAHL
```

Beispiel einer Gruppierung mit Operation auf einer Spalte

Die folgende SELECT-Anweisung wird eine Fehlermeldung zur Folge haben, da die Spalte ANZAHL zwar in der Feldliste angegeben wird, jedoch nicht in der Gruppierung auftaucht.

Ungültiges Statement



Die Fehlermeldung ungültiges Statement wird ausgegeben

Ist die eigentliche Gruppierungsangabe kein Spaltenname, sonder eventuell das Ergebnis einer Berechnung, so ist es nicht erlaubt, einen Operator auf die Ergebnisspalte anzuwenden, wie im anschließenden Beispiel deutlich wird. Im ersten Fall erstelle ich eine mögliche SELECT-Anweisung mit einer Gruppierung.

```
SELECT TNUMMER, ANZAHL + 1 AS ANZNEU
   FROM EPSWDSC/LAGER
GROUP BY TNUMMER, ANZAHL + 1
```

Eine erlaubte Gruppierung

Es ist durchaus zulässig, der Ergebnisspalte ANZAHL + 1 mit Hilfe von AS einen neuen Namen zuzuordnen. Wenn nun aber eine Operation auf ANZAHL + 1 vorgenommen werden soll, wird auch in diesem Fall ein Fehler gemeldet.

Ungültiges Statement.

```
Spalte ANZAHL oder Ausdruck in SELECT-Liste nicht gültig.
SELECT TNUMMER, (ANZAHL + 1) * 2 AS ANZNEU
   FROM EPSWDSC/LAGER
GROUP BY TNUMMER, ANZAHL + 1
```

Die Fehlermeldung ungültiges Statement wird ausgegeben

Die angegebene Spaltenliste wird für jede Gruppe innerhalb des Result Tables angewendet und die zurückgegebene Ergebnistabelle enthält dann genau so viele Zeilen, wie es Gruppen gibt, die aufgrund der Gruppierungskriterien ermittelt wurden.

Wenn Operationen auf die Spalten angewendet werden, so dienen nur die Daten der aktuellen Gruppe als Argumentmenge.

Wird keine GROUP BY / HAVING-Klausel verwendet und enthält die Spaltenliste auch keine Spaltenfunktionen, so sind in der Ergebnistabelle so viele Sätze enthalten, wie durch eine WHERE-Bedingung festgelegt wurde oder eben alle Sätze.

Sind jedoch solche Funktionen verwendet worden, so dient hier die gelieferte Satzauswahl als Argumentmenge und in der Ergebnistabelle wird lediglich ein einzelner Satz geliefert.

Wenn Spaltenfunktionen in der Feldliste benutzt werden sollen, so ist dafür Sorge zu tragen, dass jeder Ausdruck für eine Ergebnis-Spalte als Funktion definiert wird, oder es können nur normale Spaltenbezeichnungen (ohne funktionale Ausdrücke) angegeben werden.

Zur Veranschaulichung folgen einige Abbildungen.

```
SELECT SUM(ANZAHL)
   FROM EPSWDSC/LAGER
```

Gültige Anweisung mit einer Funktion

```
SELECT AVG(GEWICHT), SUM(ANZAHL)
   FROM EPSWDSC/LAGER
```

Gültige Anweisung mit zwei Funktionen

Die Ergebnistabellen der beiden vorhergehenden Statements liefern dabei genau einen Satz. Im Gegensatz dazu ist die anschließende Anweisung ungültig und kann nicht vom System ausgewertet werden.

```
Spalte TNUMMER oder Ausdruck in SELECT-Liste nicht gültig.
SELECT TNUMMER, SUM(ANZAHL) AS X
   FROM EPSWDSC/LAGER
```

Die Fehlermeldung ungültiges Statement wird ausgegeben

In diesem Fall muss also entweder die Summenfunktion wieder entfernt werden oder auf die Spalte TNUMMER wird ebenfalls eine Funktion angewendet. In der dritten Variante kann auch die Ergebnisspalte TNUMMER entfallen.

Im Zusammenhang mit NULL-Inhalten sind ebenfalls einige Dinge zu beachten. So kann jede Ergebnisspalte NULL-Werte erhalten, wenn ihr Inhalt von einer Spaltenfunktion ermittelt wurde - abgesehen von COUNT oder COUNT_BIG.

Beruht eine Ergebnisspalte auf dem Inhalt einer Tabellenspalte, die von ihrer Definition her NULL-Inhalt annehmen kann, so ist sie natürlich ebenfalls in der Lage, diesen Wert anzunehmen.

Der FROM-Abschnitt des SELECT-Statements

Zuerst einige allgemeine Anmerkungen zu der kommagetrennten Auflistung der Tabellen, aus denen SELECT seine Informationen und Ergebnisse zusammenstellen soll.

Der einfachste Fall liegt natürlich dann vor, wenn lediglich eine einzige Tabelle in dieser Liste angegeben wird. Dieser Fall wurde schon besprochen und ist ja auch nicht allzu schwer verständlich. Welches Verhalten zeigt sich aber, wenn mehrere Tabellen definiert werden? Hier wurde weiter oben schon einiges dahingehend angemerkt, dass nun alle möglichen Kombinationen von Zeilen gebildet werden. Die Anzahl der zurückgegebenen Zeilen entspricht dabei dem Produkt der Zeilenzahl in den einzelnen Tabellen der Liste. Die Angabe eines Tabellennamens im FROM-Teil wird auch oft als Tabellenreferenz bezeichnet. Als Referenz muss nicht zwangsweise eine reale Tabelle verwendet werden, auch die Angabe des Namens einer Ansicht ist hier zulässig.

Soll nun weder eine View noch eine reale Tabelle in der Liste auftauchen, so kann auch ein weiteres SELECT-Statement als Tabellenreferenz dienen. Im folgenden Beispiel wird der Ergebnismenge des geklammerten Statements der Tabellenname TABLE_A zugewiesen.

```
SELECT * FROM
   (SELECT * FROM EPSWDSC/LIEFERBAR) AS TABLE_A
```

SELECT-Anweisung als Referenz

Auch hier wieder eine Abbildung des Ergebnisses.

Ergebnistabelle

```
Anfang auf Zeile . . . . . .
....+....1....+....2...
     LNUMMER   TNUMMER
           1   A0120
           2   A0130
           3   A0110
```

Ergebnis der Abfrage

Auch bei diesen Konstrukten sind zum Beispiel die verschiedenen Arten der Umbenennungen, die schon weiter oben besprochen wurden, anwendbar. So ist erkennbar, dass eine solche Ergebnistabelle als Resultat der geklammerten Anweisung eine vollwertige Tabellenreferenz darstellt.

3.2.1.1
Seite 8

Daten neu anlegen mit dem INSERT-Statement

1. Ergänzung 12/2004

3.2.2
Seite 2

Allgemeine Informationen

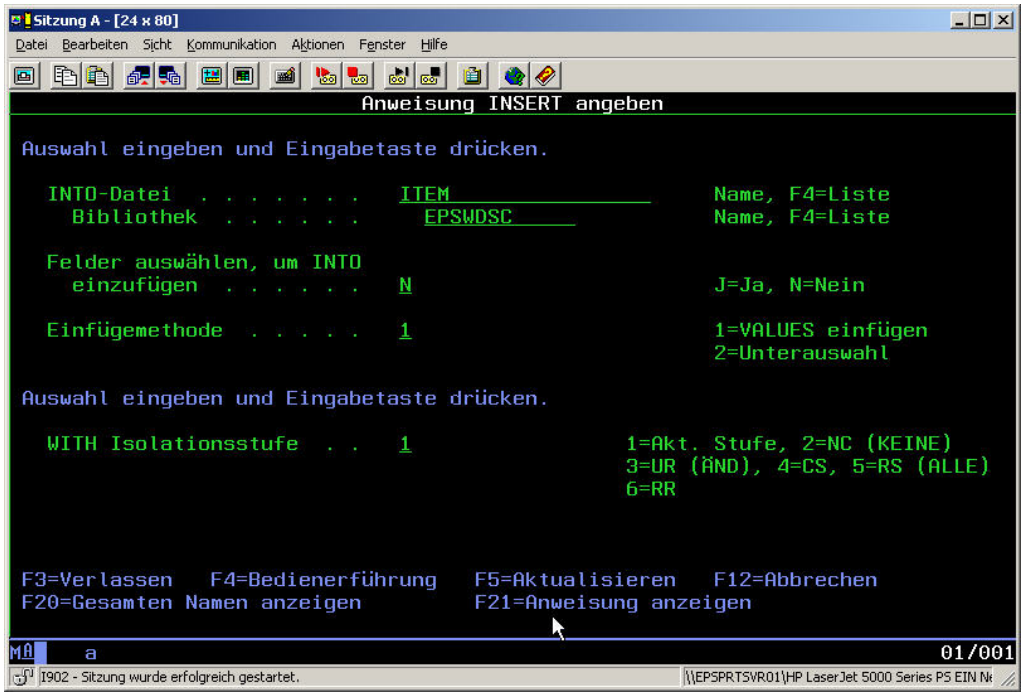
Mit INSERT können in eine Tabelle und/oder in eine View neue Datensätze geschrieben werden. Werden einer Ansicht (View) Sätze hinzugefügt, werden diese Informationen natürlich auch in die der View zugrunde liegende physische Tabelle geschrieben.

Dabei kann das INSERT-Statement auf drei unterschiedliche Arten eingesetzt werden. Möchte man einzelne Zeilen in eine Tabelle oder View schreiben, so können den einzelnen Spalten bestimmte Werte zugewiesen werden, oder es wird ein SELECT-Statement verwendet, das die Zeilen in einem Result Table für INSERT liefert.

Die dritte Variante beinhaltet die Daten für die neuen Zeilen in einem Array von Host-Datenstrukturen. Host-Variablen werden in Hochsprachen (RPG, C usw.) im Zusammenhang mit eingebettetem SQL verwendet. Sie empfangen Daten, die von den eingebetteten Anweisungen geliefert werden oder dienen ihnen als Datenquellen. Das soll zur Erläuterung in Bezug auf den Begriff Host-Variable in diesem Zusammenhang genügen.

Ein INSERT-Statement kann sowohl interaktiv aufgerufen werden als auch in eine Hochsprache eingebettet sein. Bei der Einbettung kann sowohl statisches als auch dynamisches SQL verwendet werden. Sollen aber Datenstruktur-Arrays (ROWS) verwendet werden, ist dynamisches SQL nicht zulässig. Auch in REXX kann diese Variante des INSERT nicht verwendet werden.

Betrachten wir einmal, wie ein INSERT-Statement über die interaktive Umgebung eingegeben werden kann.



INSERT über die interaktive Umgebung eingeben 1

3.2.2.1
Seite 2

Geben Sie ihre Ziel-Collection und Tabelle an oder wählen Sie sie über die Bedienerführung mit F4 aus. Wollen Sie alle Spalten mit Werten belegen, so bleibt die Einstellung „N“ im Parameter für die Feldauswahlmethode stehen. Tragen sie dort ein „J“ ein, dann müssen Sie im Folgebildschirm die gewünschten Spalten auswählen.

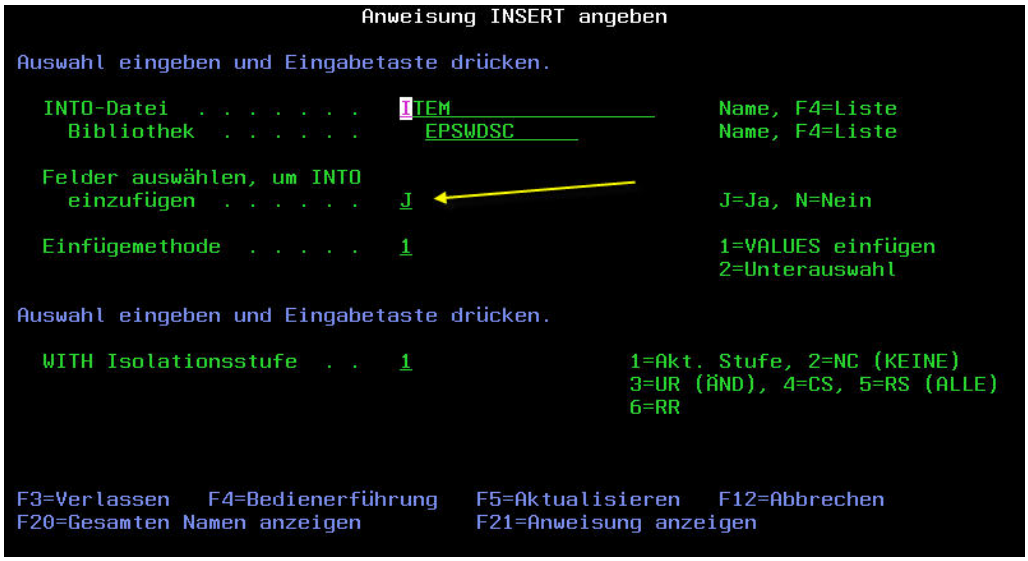
Über den Parameter „Einfügemethode“ legen Sie fest, ob eine Unterabfrage die Daten liefern soll oder ob Sie die Werte selbst eintragen wollen. Was sich hinter der „Isolationsstufe“ verbirgt, wird weiter unten erläutert. Nach der Datenfreigabe erhalten Sie eine Maske zur Dateneingabe in die vorhandenen Spalten. Dort tragen Sie nun die gewünschten Werte ein.

Mit Werten zu belegende Spalten

Sie können sich über die Funktionstaste F21 das SQL-Statement anzeigen lassen, das aus den bisherigen Informationen erzeugt wird.

Das dazu gehörende INSERT-Statement

Wenn Sie die Zielspalten selbst bestimmen wollen, vielleicht weil es nicht notwendig ist, alle Felder zu belegen, so tragen Sie im Parameter für die Feldauswahl ein „J“ ein.

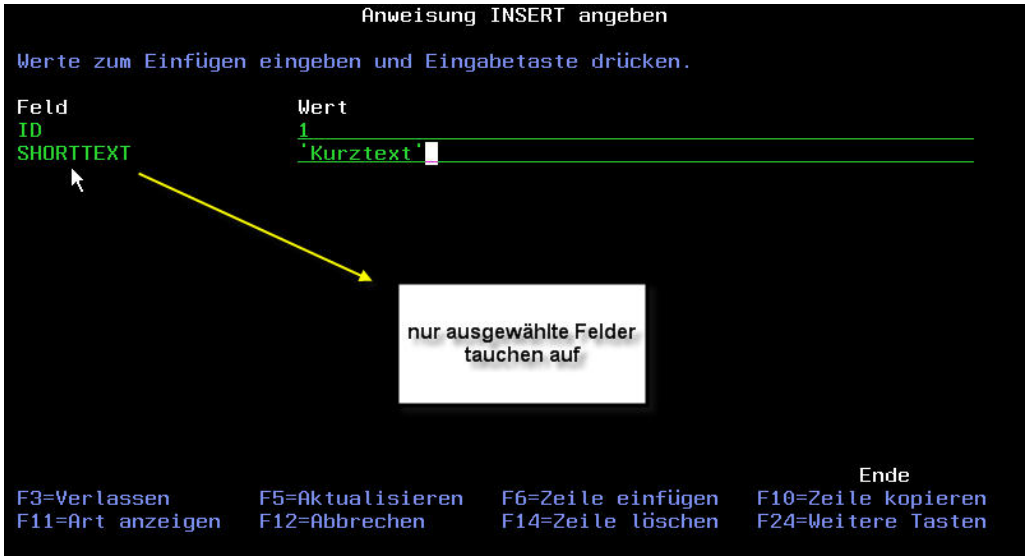


Eigene Auswahl der Zielspalten

In der darauf folgenden Maske wählen Sie die Spalten aus.

Auswahl der Zielspalten

3.2.2.1
Seite 4



Maske mit selektierten Spalten

Lassen Sie sich auch hier das Statement über F21 anzeigen. Es ergibt sich folgendes Bild:

INSERT-Statement mit einer Spaltenauswahl

Halten wir einmal die beiden bisher erzeugten Statements im Vergleich gegeneinander.

```
INSERT INTO EPSWDSC/ITEM VALUES(1, 'Kurztext', 'Beschreibung', 2)
INSERT INTO EPSWDSC/ITEM (ID, SHORTTEXT) VALUES(1, 'Kurztext')
```

Statements im Vergleich

Im ersten Fall wird keine besondere Liste von Zielspalten angegeben. Für diesen Fall ist für jede Spalte der Zieltabelle oder Ansicht ein Wert anzugeben. Nur wenn eine Spalte NULL-Inhalte zulässt, brauchen Sie keine Daten eingeben. Das System ersetzt fehlende Daten durch NULL und gibt eine entsprechende Meldung zurück, wenn die betreffende Spalte keine NULL-Werte enthalten darf.

Das zweite Statement besitzt eine Zielspaltenauswahl. Hier müssen Anzahl und Typ der in VALUES definierten Werte mit der Anzahl und den Typen der Zielspalten übereinstimmen. Für die Angabe von NULL-Inhalten gilt das gleiche, wie im ersten Statement.

3.2.2.1
Seite 6

Einzelne Abschnitte des Statements

INTO-Abschnitt:

In dieser Klausel legen Sie fest, in welche Zieltabelle oder View Sie neue Datensätze schreiben wollen. Hier ist eine bestehende Tabelle auf dem aktuellen Server anzugeben. Einfüge-Operationen für das Ziel müssen zulässig sein; Sie dürfen sich nicht auf Katalogtabellen oder Sichten beziehen.

Liste der Zielspalten:

Diese Liste gibt die Spalten an, deren Inhalt im Statement mit Werten beschickt werden soll. Jede Spalte muss unqualifiziert angegeben werden und darf auch nur einmal in der Liste auftauchen. Spalten, die kein Einfügen zulassen, dürfen auch in der Liste nicht angesprochen werden.

VALUES-Klausel:

Durch diesen Abschnitt wird ein Datensatz von den Inhalten her beschrieben. Der erste Wert wird in das erste Feld der (Ziel)Spaltenliste geschrieben, der zweite Wert in das zweite Feld und so fort. Wenn keine Spaltenliste angegeben ist, erwartet das System Werte für jede in der Tabelle oder Ansicht vorhandenen Spalte. Auf welche Art die VALUE-Liste auch angegeben wird, die Anzahl der Zielspalten muss mit der Anzahl der einzusetzenden Werte übereinstimmen. Sollten für eine Zielspalte NULL-Werte zulässig sein, so kann NULL auch als Angabe in der Werteliste verwendet werden. Auch die Angabe von DEFAULT darf dort eingesetzt werden. Das Ergebnis ist davon abhängig, wie die Spalte definiert wurde.

Wenn bei der Erstellung die WITH DEFAULT-Klausel angegeben wurde, schreibt das System den dort definierten Standardwert in die Zielspalte. Wurde WITH DEFAULT oder NOT NULL nicht angegeben, so wird NULL geschrieben.

Wurde WITH DEFAULT nicht verwendet, aber NOT NULL angegeben, darf DEFAULT für diese Spalte nicht verwendet werden. Auch wenn bei der Erstellung DEFAULT NULL verwendet wurde, ist DEFAULT bei der INSERT-Anweisung nicht zulässig.

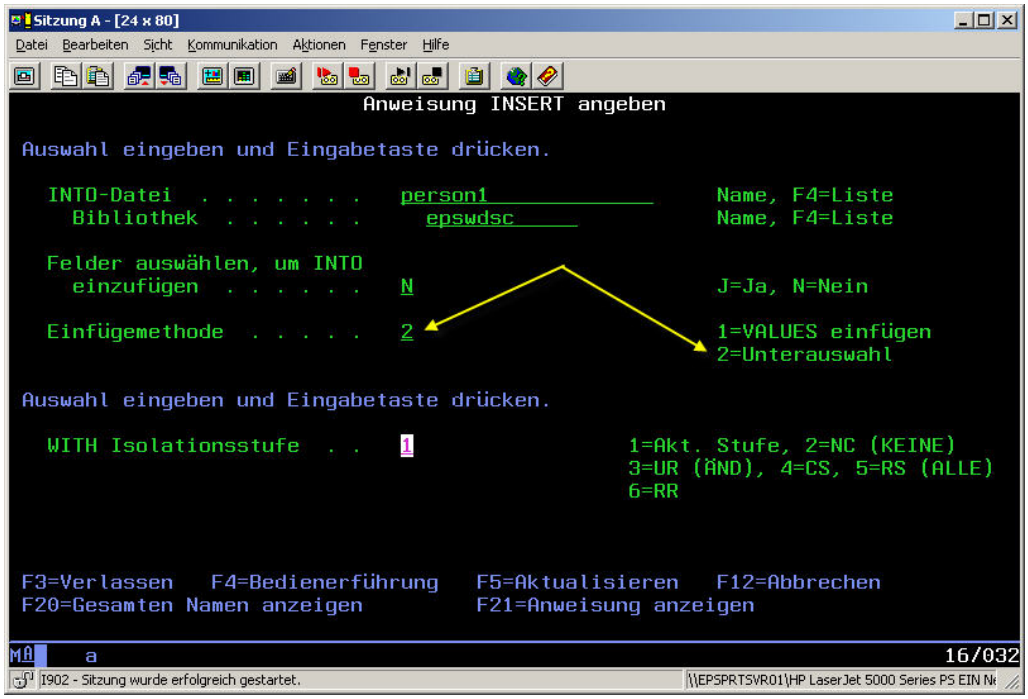
Sollen mehrere Sätze in einer Tabelle neu angelegt werden, so kann es recht zeitaufwendig werden, die diversen INSERT-Anweisungen interaktiv einzugeben. Für den Fall, das eine andere Tabelle vorhanden ist, deren Dateninhalte als Quelle für die neuen Sätze dienen können, kann man sich die Arbeit etwas erleichtern.

INSERT verlangt nicht unbedingt eine VALUES-Klausel, es kann auch das Ergebnis eines SELECT-Statements verwendet werden, um die Daten zu liefern.

1. Ergänzung 12/2004

3.2.2.2
Seite 2

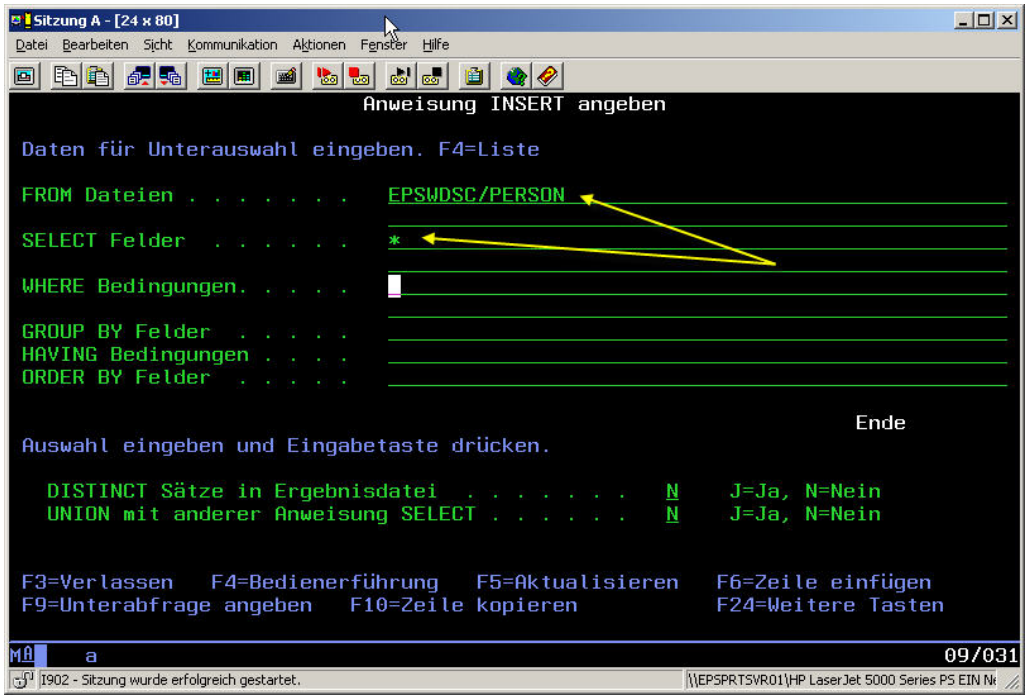
Geben Sie in der interaktiven SQL-Umgebung den Befehl INSERT ein und rufen Sie die Bedienerführung mit F4 auf. Den Parameter „Einfügemethode" überschreiben Sie mit „2". Dadurch teilen Sie dem System mit, dass Sie ein SELECT-Statement (Unterabfrage) als Datenquelle verwenden wollen.



INSERT mit Bedienerführung

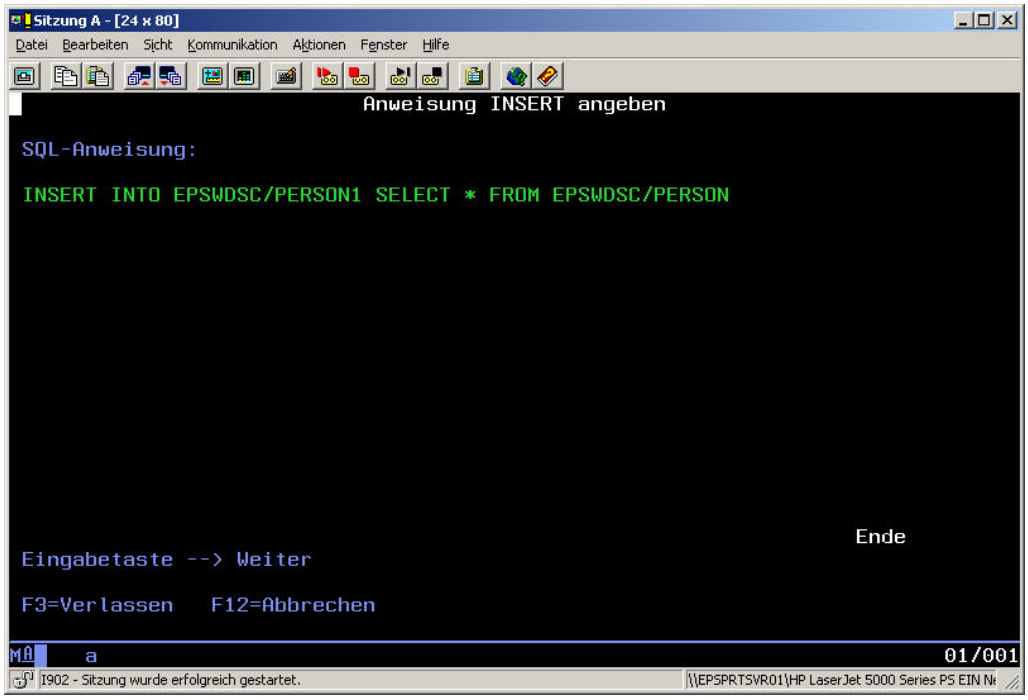
In der bestehenden Datenbank habe ich einfach die Struktur der Tabelle PERSON verwendet und die Tabelle PERSON1 aufgebaut. In diese neue Tabelle sollen nun die Daten aus der Datei PERSON übertragen werden.

Anschließend drücken Sie Datenfreigabe und eine weitere Maske wird angezeigt. Hier werden Sie Ihre Unterabfrage definieren. Über den Aufbau eines SELECT-Statements ist Ihnen ja schon einiges aus den ersten Kapiteln bekannt. Die Tabelle PERSON dient hier nun als Quelle. Um das kleine Beispiel einfach zu halten, soll jede Spalte übertragen werden.



Angeben der Unterabfrage

Auch hier kann man sich über die Funktionstaste F21 das entstandene INSERT-Statement anschauen.

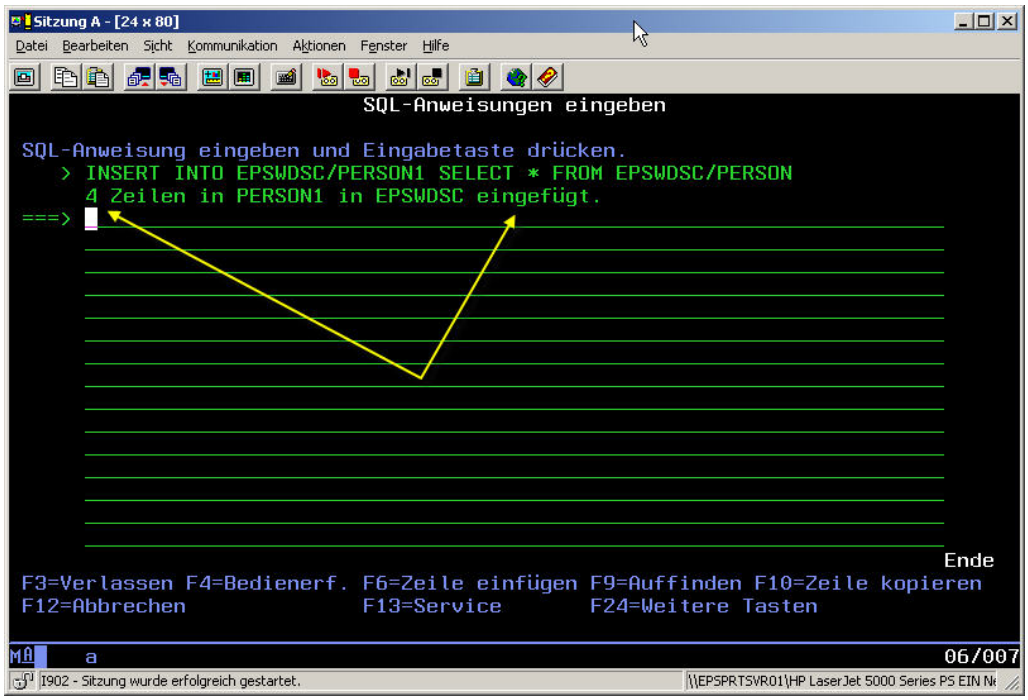


Generiertes INSERT-Statement

3.2.2.2
Seite 4

Das Schlüsselwort VALUES entfällt also und an seine Stelle tritt das SELECT-Statement.

Wenn dieses Statement nun an das System zur Ausführung übergeben wurde, erhalten wir anschließend die folgende Ansicht.



Die Sätze wurden eingefügt

Die vier Datensätze, die schon in der PERSON-Tabelle eingepflegt waren, wurden von der Unterabfrage bereitgestellt und in die Zieltabelle PERSON1 eingefügt.

Über diese Variante des INSERT steht uns also eine recht gute und flexible Möglichkeit zum Kopieren von Sätzen zur Verfügung.

Anmerkungen

Abschließend sollen noch einige Dinge erläutert werden, die bei INSERT-Statements zu beachten sind.

DEFAULT-Werte:

Jede Spalte, die nicht in der Zielspaltenliste aufgeführt ist, erhält beim Einfügen ihren DEFAULT-Wert. Die Spalten, für die kein Standardwert definiert ist, müssen in der Spaltenauflistung angegeben werden. Sollten Sie Daten in einer View (Ansicht) einfügen, so wird der DEFAULT-Wert in jede Spalte der zugrunde liegenden Tabelle geschrieben, die nicht in der View verwendet wird. Das bedeutet natürlich, dass alle in der Ansicht nicht verwendeten Spalten mit DEFAULT-Werten definiert sein müssen.

Gültigkeit:

Wenn zu der Zieltabelle oder der Basistabelle einer Ansicht eindeutige Indizes vorliegen, so muss jede einzufügende Zeile diesen Bedingungen genügen. Außer wenn COMMIT(*NONE) angegeben wurde, erfolgt die Prüfung auf Gültigkeit am Ende des Statements.

Weiter muss auch jede Prüfbedingung (CHECK CONSTRAINT) einer Zieltabelle durch jeden Satz erfüllt sein. Auch die CHECK CONSTRAINTS werden am Ende des Statements untersucht.

Natürlich müssen beim Einfügen auch sämtliche Kriterien der referentiellen Integrität eingehalten werden.

3.2.2.3

Seite 2

Daten ändern mit dem UPDATE-Statement

1. Ergänzung 12/2004

3.2.3 Seite 2

Allgemeine Informationen

Mit einem UPDATE-Statement ändern Sie Daten in einer Tabelle direkt oder über eine Ansicht. Dabei kann dieses Statement in zwei Varianten verwendet werden.

Zum einen können die zu ändernden Sätze aufgesucht werden (searched UPDATE form). Hier können auch mehrere Zeilen geändert werden. Die zweite Variante bestimmt die zu ändernde Zeile durch Positionierung (positioned UPDATE form). In diesem Fall kann lediglich der einzelne angezogene Satz fortgeschrieben werden.

Beide Arten des Statements können in Hochsprachen eingebettet werden. Die erste Variante (searched UPDATE form) kann zusätzlich auch aus der interaktiven Umgebung heraus durchgeführt werden, was bei der positionierten UPDATE-Form nicht möglich ist. Eine Dynamisierung (dynamic embedded SQL) ist bei der Einbettung zulässig.

Zur Ausführung eines UPDATE-Statements wird eine Änderungsberechtigung (UPDATE) für die fortzuschreibende Tabelle oder für jede zu ändernde Spalte benötigt. Oder man ist Besitzer der Tabelle und hat *EXECUTE-Berechtigung für die Bibliothek, in der sich die Zieltabelle befindet.

Die UPDATE-Berechtigung eines Profils für eine Tabelle liegt vor, wenn es Eigner ist, wenn die Berechtigung gesondert erteilt wurde oder wenn die Systemberechtigungen *OBJOPR und *UPD für die Tabelle erteilt wurden.

Handelt es sich bei dem Ziel um eine Ansicht, so muss ebenfalls die UPDATE-Berechtigung vorliegen. Wenn *OBJOPR und *UPD für die View erteilt wurden, sollte auch die Berechtigung *UPD für die erste Tabelle oder Ansicht der ersten FROM-Klausel der View-Spezifikation erteilt worden sein.

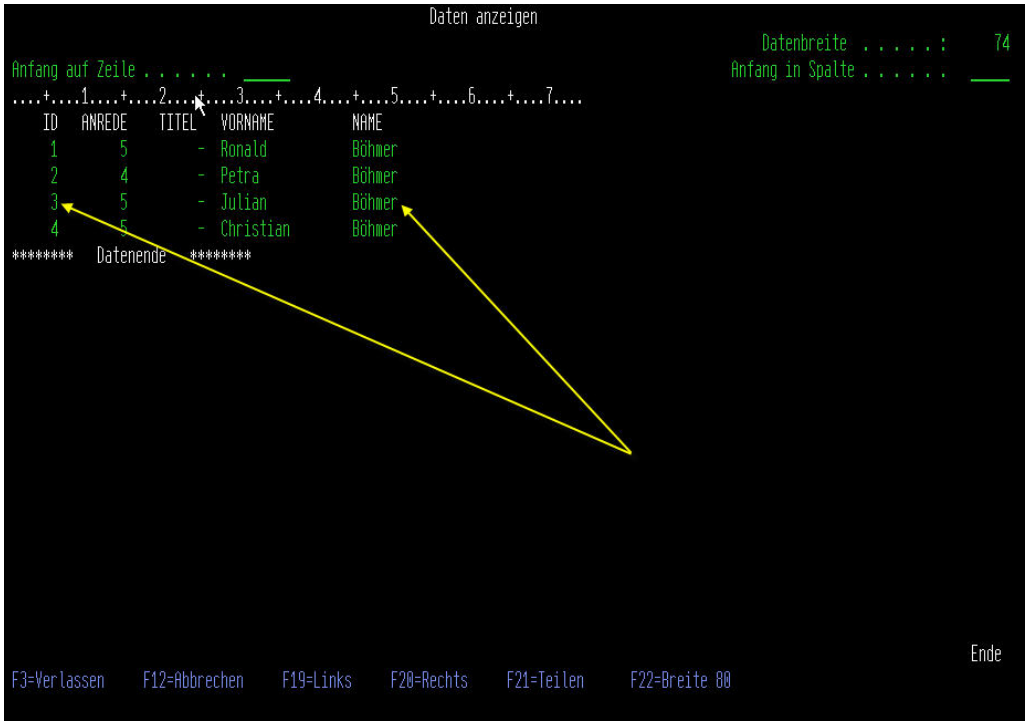
Findet man dort wieder eine Ansicht vor, dann ist diese Regel auch auf die View anzuwenden. Die Struktur der Berechtigungen kann also durchaus etwas verwickelt erscheinen.

3.2.3.1

Seite 2

Einige praktische Beispiele

Nehmen wir an, wir wollen den Vornamen einer einzelnen Person in der Tabelle PERSON unserer Datenbank ändern.



Vorhandene Personen

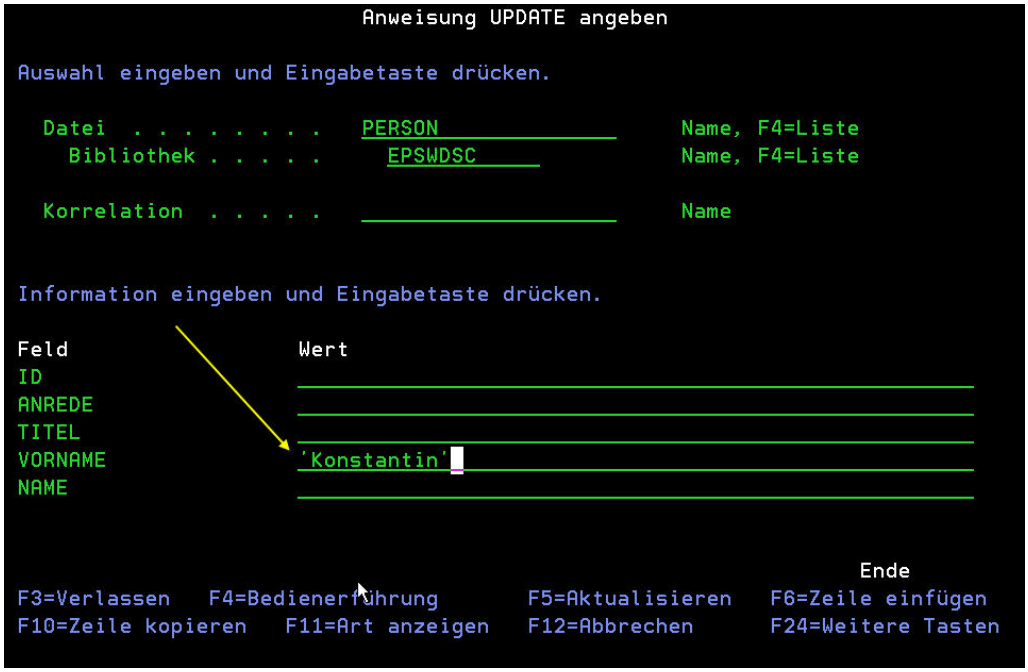
Der zu ändernde Satz hat die ID „3“ und der Vorname soll von „Julian“ auf „Konstantin“ geändert werden. Bauen wir unser UPDATE-Statement über die bekannte interaktive Umgebung auf.

In der Befehlszeile geben wir UPDATE ein, rufen die Bedienerführung auf und tragen die Tabelleninformationen ein.

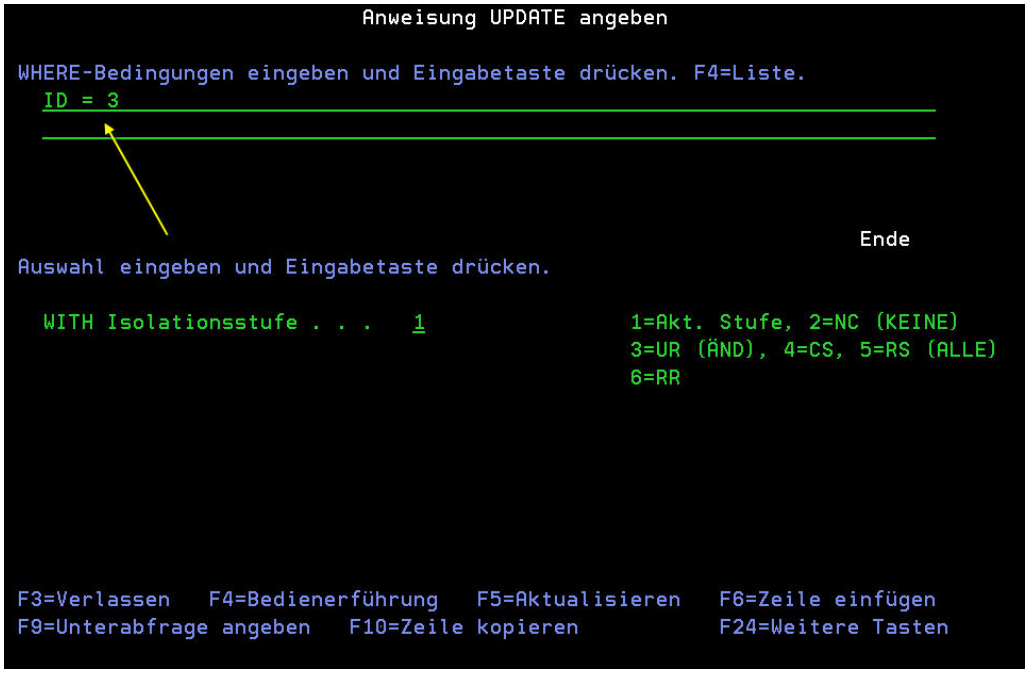
Fortzuschreibende Tabelle bei UPDATE angeben

3.2.3.2
Seite 2

Nach der Datenfreigabe tragen wir in der Spalte „Vorname“ den neuen Inhalt ein und wechseln zur nächsten Maske.

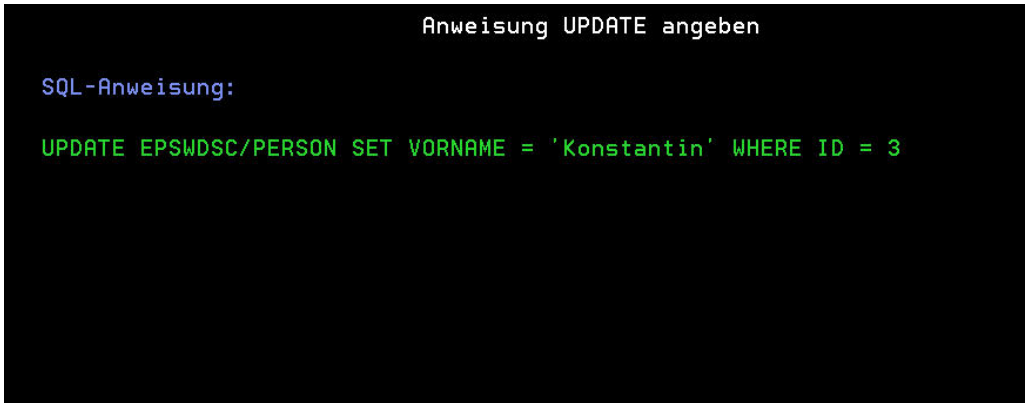


Wert für neuen Vornamen eintragen



Zu ändernden Satz identifizieren

Auch hier lassen wir uns mit F21 das bisherige Statement in Textform anzeigen.



UPDATE für einzelnen Satz

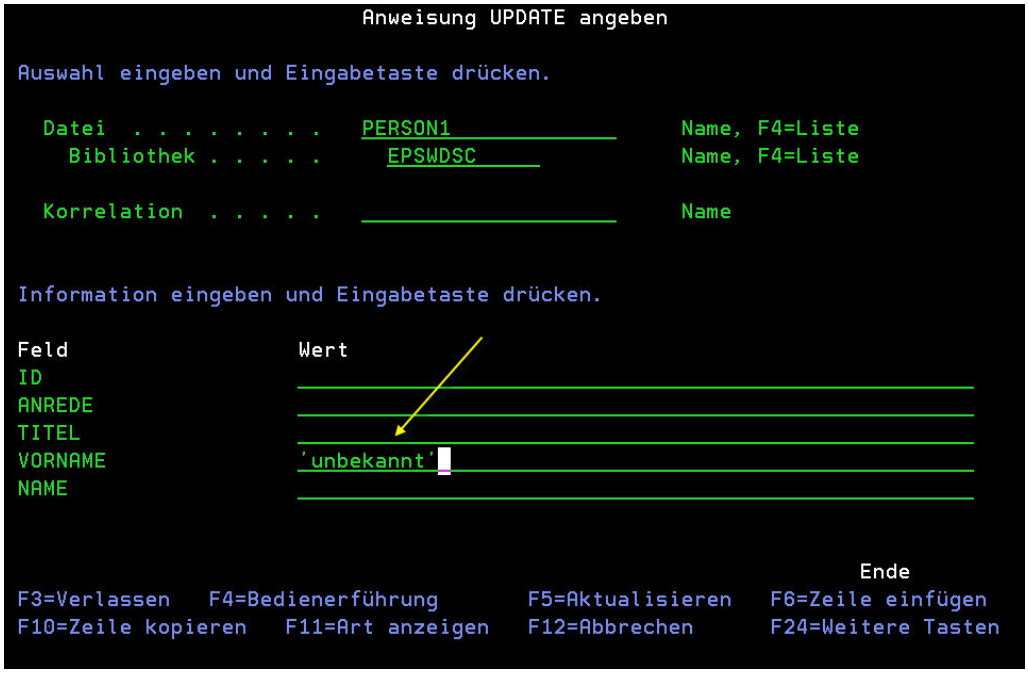
Wenn das Statement via Datenfreigabe an das System übergeben wird, sollte folgende Information angezeigt werden.

Bestätigte Änderung in der Datenbank

Mit UPDATE können auch gleich mehrere Sätze geändert werden, wenn die WHERE-Klausel entsprechend aufgebaut wird. Achtung: Sollte keine WHERE-Bedingung vorhanden sein, werden alle in der Tabelle vorhandenen Sätze entsprechend geändert.

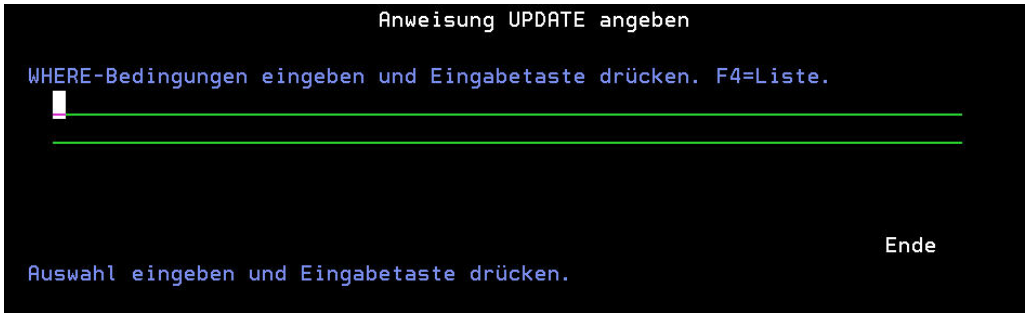
Bauen wir einmal ein solches Statement auf. In diesem Beispiel sollen die Vornamen in allen Sätzen, deren ID zwischen 2 und 4 liegt auf „unbekannt“ gesetzt werden.

3.2.3.2
Seite 4

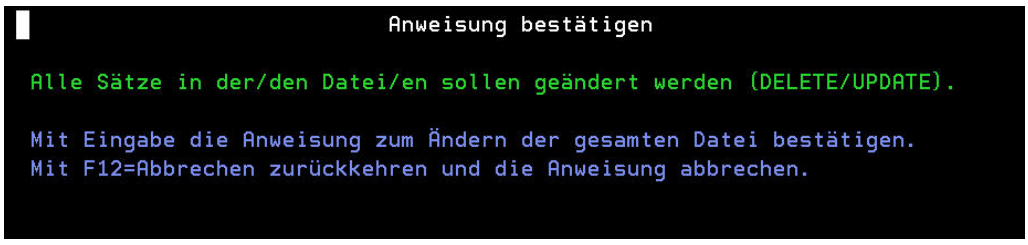


Vornamen auf „unbekannt“ setzen

Wie oben erwähnt, definieren wir hier keine WHERE-Bedingung. Anschließend fragt das System nach, ob wirklich alle Sätze der Tabelle auf die neuen Werte gesetzt werden sollen.

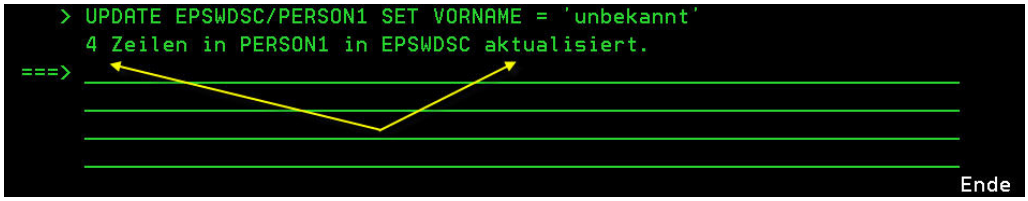


Keine WHERE-Klausel im Statement



Abfrage zur Bestätigung

Schaut man sich die Informationen des Systems an, nachdem das Statement ausgeführt wurde, stellt man fest, dass alle (4) Sätze der Tabelle geändert wurden.



Informationen nach der Änderung aller Sätze

Zur Kontrolle lassen wir uns alle Sätze der Tabelle PERSON1 auflisten und prüfen, ob wirklich „unbekannt“ in der Spalte VORNAME eingetragen wurde.

Die geänderten Sätze

Das UPDATE-Statement erschöpft sich nun nicht nur darin, einzelne Spalten auf neue Werte zu setzen. Der neue Inhalt einer Spalte kann auch durch eine Berechnung ermittelt werden. Auch die WHERE-Klausel ist erweiterbar. Es können unter anderem auch logische Operatoren in ihr verwendet werden.

Beachten Sie die Syntaxbeispiele in den beiden folgenden Abbildungen.

3.2.3.2
Seite 6

```
UPDATE PROJECT
  SET PRSTAFF = PRSTAFF + 1.5
  WHERE DEPTNO = 'D21'
```

UPDATE mit Berechnung

```
UPDATE EMPLOYEE
  SET JOB=NULL, SALARY=0, BONUS=0, COMM=0
  WHERE WORKDEPT = 'E21' AND JOB <> 'MANAGER'
```

UPDATE mit logischen Operatoren

Daten löschen mit dem DELETE-Statement

1. Ergänzung 12/2004

3.2.4
Seite 2

Allgemeine Informationen

Mit der Anweisung DELETE können Sie einen oder mehrere Sätze aus einer Datentabelle löschen. Wird DELETE im Zusammenhang mit einer Ansicht verwendet, so werden die angezogenen Zeilen aus der Basistabelle der View gelöscht. Ähnlich wie bei UPDATE gibt es hier zwei Formen: zum einen die Verwendung mit einer Suchbedingung, um eine bestimmte Auswahl der zu entfernenden Zeilen festzulegen, und zum anderen die positionierte Variante. In diesem Fall wird der Satz gelöscht, der durch die aktuelle Position in einer Tabelle definiert ist. DELETE kann auch in Hochsprachen eingebettet und in dynamischen SQL-Anweisungen eingesetzt werden. Auch bei dieser Anweisung müssen ein paar Dinge zu den benötigten Berechtigungen beachtet werden, um die Anweisung erfolgreich ausführen zu können.

Für die Tabelle oder Ansicht ist die Lösch-Berechtigung erforderlich und für die Bibliothek die Systemberechtigung *EXECUTE, in der die Tabelle (oder View) abgelegt ist.

Diese Berechtigungen sind vorhanden, wenn Besitzrechte bei der Tabelle bestehen oder wenn die DELETE-Berechtigung erteilt wurde (granted). Auch wenn die Systemberechtigung *OBJOPR und *DLT erteilt wurde, ist das ausreichend.

Sollte in der Suchbedingung, die zu entfernende Sätze festlegt, eine Unterabfrage definiert sein, muss die SELECT-Berechtigung für die Tabelle und die Berechtigung *EXECUTE für die Bibliothek vorliegen. Die SELECT-Berechtigung wurde entweder erteilt (granted) oder die Rechte *OBJOPR und *READ liegen vor.

3.2.4.1

Seite 2

Beispiele zum DELETE-Statement

Mit dem folgenden Statement, das über die interaktive Schnittstelle eingegeben wird, entfernen wir einen genau bestimmten Satz aus der Tabelle PERSON1 der Bibliothek EPSWDSC.

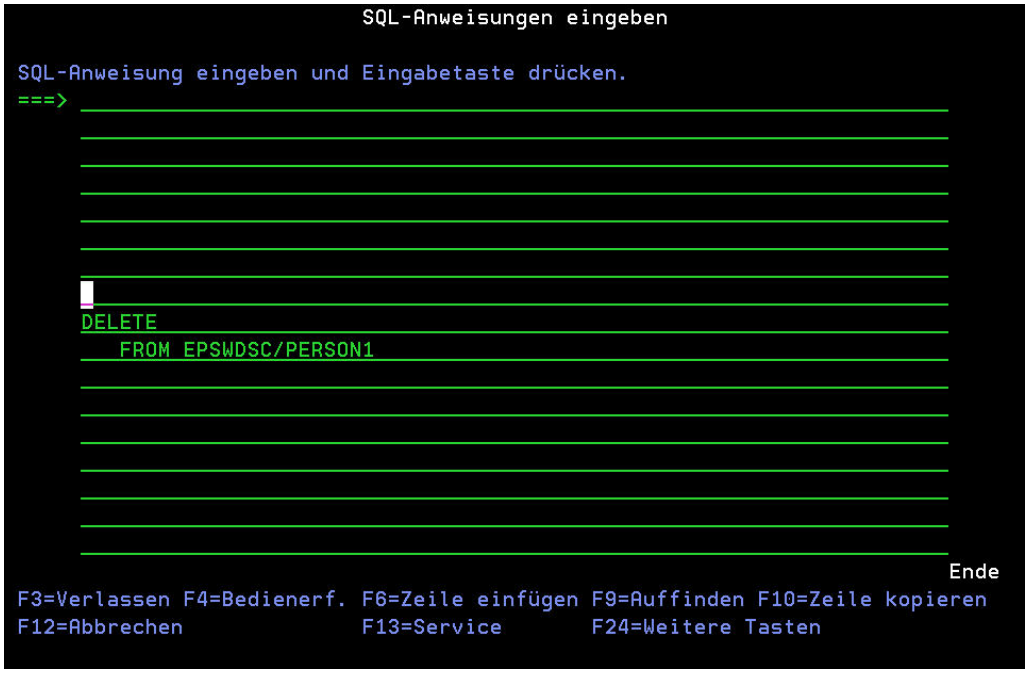
```
                          SQL-Anweisungen eingeben

SQL-Anweisung eingeben und Eingabetaste drücken.
===>
     DELETE
        FROM EPSWDSC/PERSON1
        WHERE ID = 1
                                                                        Ende
F3=Verlassen F4=Bedienerf.  F6=Zeile einfügen F9=Auffinden F10=Zeile kopieren
F12=Abbrechen               F13=Service       F24=Weitere Tasten
```

DELETE-Anweisung mit einer Suchbedingung

```
                          SQL-Anweisungen eingeben

SQL-Anweisung eingeben und Eingabetaste drücken.
===>
     DELETE
        FROM EPSWDSC/PERSON1
        WHERE NAME LIKE 'Böh%'
                                                                        Ende
F3=Verlassen F4=Bedienerf.  F6=Zeile einfügen F9=Auffinden F10=Zeile kopieren
F12=Abbrechen               F13=Service       F24=Weitere Tasten
```

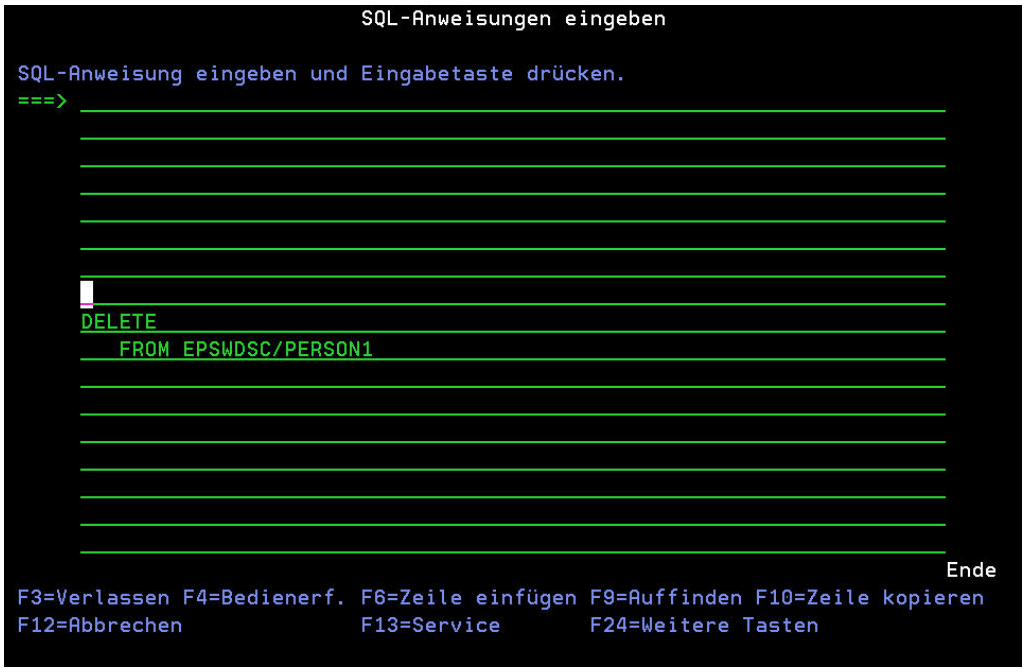
Im nächsten Fall soll eine Gruppe von Zeilen entfernt werden, deren Inhalt in der Spalte NAME mit den Zeichen „Böh“ beginnt.

3.2.4.2
Seite 2



Suchbedingung mit LIKE

Die folgende Abbildung zeigt die wahrscheinlich einfachste und radikalste Form des DELETE. Hier werden alle Sätze einer Tabelle gelöscht. Das System wird zur Sicherheit noch einmal gefragt, ob diese Aktion beabsichtigt ist, aber nur, wenn Sie die Anweisung im Dialog eingeben. Also Vorsicht bei eingebetteten Anweisungen!



Alle Zeilen einer Tabelle löschen

Joins

Tabellenverknüpfungen durch JOIN

In vielen Fällen wird es notwendig sein, logische Verbindungen zwischen mehreren Tabellen herzustellen, um eine Ergebnistabelle der gewünschten Struktur einem Anwender zur Verfügung zu stellen. Es ist sogar ein ausgesprochen typisches Merkmal moderner relationaler Datenbanken, dass Informationen über viele Tabellen verteilt gespeichert werden. Diese Strukturen ergeben sich bei der Modellierung dieser Systeme aus der Notwendigkeit, eine redundante Datenhaltung zu vermeiden, möglichst flexible und leicht erweiterbare Datenstrukturen zu erhalten, und vielem mehr. Als Stichwort sei hier nur noch die Normalisierung in ihren unterschiedlichen Stufen genannt. Einfache Beispiele für die logische Verknüpfung haben wir auch eingangs gesehen. Vereinfacht ausgedrückt findet man diese Notwendigkeit schon bei der Zusammenstellung von so genannten Kopf- und Positionssätzen. Auf dem bisherigen Weg wurde die Verbindung über eine entsprechend konfigurierte WHERE-Klausel hergestellt. Aber auch in diesem Zusammenhang werden von SQL wesentlich weitergehende Möglichkeiten zu Verfügung gestellt. Und damit kommen wir zu den Verknüpfungen von Tabellen, die über JOIN hergestellt werden.

Das Schlüsselwort JOIN wird dabei zwischen die Tabellenreferenzen gestellt und kann mit verschiedenen vorangestellten Operatoren versehen werden. Zu diesen Operatoren zählt man INNER, LEFT OUTER, RIGHT OUTER, LEFT EXCEPTION, RIGHT EXCEPTION oder CROSS.

Hat die Verbindung keine spezielle Angabe eines dieser Operatoren, so wird INNER als Standard angenommen. Bei Verbindungen über JOIN ist zu beachten, dass die Reihenfolge, in der die einzelnen Operationen vorgenommen werden, Einfluss auf das gelieferte Ergebnis hat. Allgemein gilt, dass JOINs von links nach rechts abgearbeitet werden. Um die Lesbarkeit von vielfach verschachtelten Verbindungen zu verbessern, sollten Klammern verwendet werden.

Eine Verbindung über LEFT JOIN

```
TABLE1 LEFT JOIN TABLE2 ON TABLE1.C1=TABLE2.C1
    LEFT JOIN TABLE3 LEFT JOIN TABLE4 ON TABLE3.C1=TABLE4.C1
            ON TABLE1.C1=TABLE3.C1
```

Verschachtelte Verbindungen ohne Klammern

3.2.5
Seite 2

Die gleiche Verbindung, jedoch mit Klammern übersichtlicher

```
(TABLE1 LEFT JOIN TABLE2 ON TABLE1.C1=TABLE2.C1)
      LEFT JOIN (TABLE3 LEFT JOIN TABLE4 ON TABLE3.C1=TABLE4.C1)
            ON TABLE1.C1=TABLE3.C1
```

Verschachtelte Verbindung mit Klammern

Der INNER JOIN kombiniert nun alle Zeilen der auf der linken Seite des Schlüsselwortes stehenden Tabellenreferenz mit den Zeilen der rechten Tabelle. In die Ergebnistabelle werden aber nur die Sätze übernommen, die den Bedingungen in der JOIN-Klausel entsprechen. Die Prüfung dieser Bedingung muss wie bei der WHERE-Klausel einen logischen Wert (WAHR, FALSCH) ergeben.

Es ist bei diesen Arten der Verknüpfung möglich, dass aus einer oder aus beiden Tabellen Sätze nicht in der Ergebnistabelle auftauchen.

Möchte man diese vom INNER JOIN nicht berücksichtigten Zeilen mit einschließen, so kann man den OUTER Operator verwenden. Durch diesen werden sowohl die vom inneren JOIN erzeugten Zeilen als auch die dabei übergangenen zurückgegeben.

Für den Fall, dass man sich lediglich die Zeilen zurückgeben lassen möchte, die bei einem INNER JOIN verloren gehen, setzt man den EXCEPTION JOIN ein. Diese Art der Verbindung kann zusammen mit einem LEFT- oder RIGHT JOIN eingesetzt werden.

Zur Verdeutlichung wird nun ein INNER JOIN aufgebaut, der diese Gestalt annimmt.

INNER JOIN

```
SELECT * FROM EPSWDSC/LIEFERBAR A
         JOIN EPSWDSC/LAGER B
         ON A.TNUMMER = B.TNUMMER
```

Anweisung mit INNER JOIN

Diese Anweisung liefert dann folgende Ergebnistabelle.

Ergebnistabelle des INNER JOIN

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9....+
     LNUMMER   TNUMMER        REGAL          FACH   TNUMMER        ANZAHL       GEWICHT
           1   A0120             4             1   A0120             98      1.000,00
           1   A0120             4             2   A0120             49      1.000,00
           3   A0110             1             1   A0110              2      1.000,00
           3   A0110             1             2   A0110             15      1.000,00
```

Ergebnis der Abfrage

Anschließend wird zum Vergleich ein Statement eingegeben, über das eine LEFT JOIN Verknüpfung der beiden Tabellen hergestellt wird.

LEFT JOIN

```
SELECT * FROM EPSWDSC/LIEFERBAR A
         LEFT JOIN EPSWDSC/LAGER B
         ON A.TNUMMER = B.TNUMMER
```

Anweisung mit LEFT JOIN

Entsprechend sieht die Ergebnistabelle aus.

Result Set des LEFT JOIN

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9....+
     LNUMMER   TNUMMER        REGAL          FACH   TNUMMER        ANZAHL       GEWICHT
           1   A0120             4             1   A0120             98      1.000,00
           1   A0120             4             2   A0120             49      1.000,00
           2   A0130             -             -   -                  -             -
           3   A0110             1             1   A0110              2      1.000,00
           3   A0110             1             2   A0110             15      1.000,00
           3   B1000             -             -   -                  -             -
           4   B2000             -             -   -                  -             -
```

Ergebnis der Abfrage

Damit dieses Ergebnis besser verstanden werden kann, sei hier eine weitere Abbildung eingefügt, die die Inhalte der Tabelle LIEFERBAR darstellt.

3.2.5
Seite 4

Sätze in der Tabelle LIEFERBAR

```
....+....1....+....2...
     LNUMMER   TNUMMER
           1    A0120
           2    A0130
           3    A0110
           3    B1000
           4    B2000
```

Inhalt der Tabelle LIEFERBAR

Die beiden letzten Zeilen der Tabelle verdeutlichen, dass die Lieferanten mit der Nummer 3 und 4 die Artikel B1000 beziehungsweise B2000 liefern können.

Jetzt sollte gegen diesen Datenbestand der Inhalt der Tabelle LAGER gehalten werden.

Bestand in der Tabelle LAGER

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7
       REGAL          FACH   TNUMMER       ANZAHL       GEWICHT
           1             1   A0110              2      1.000,00
           1             2   A0110             15      1.000,00
           8             -   -                  0      1.000,00
           4             1   A0120             98      1.000,00
           4             2   A0120             49      1.000,00
           5             1   -                  0       300,00
           5             2   -                  0       300,00
```

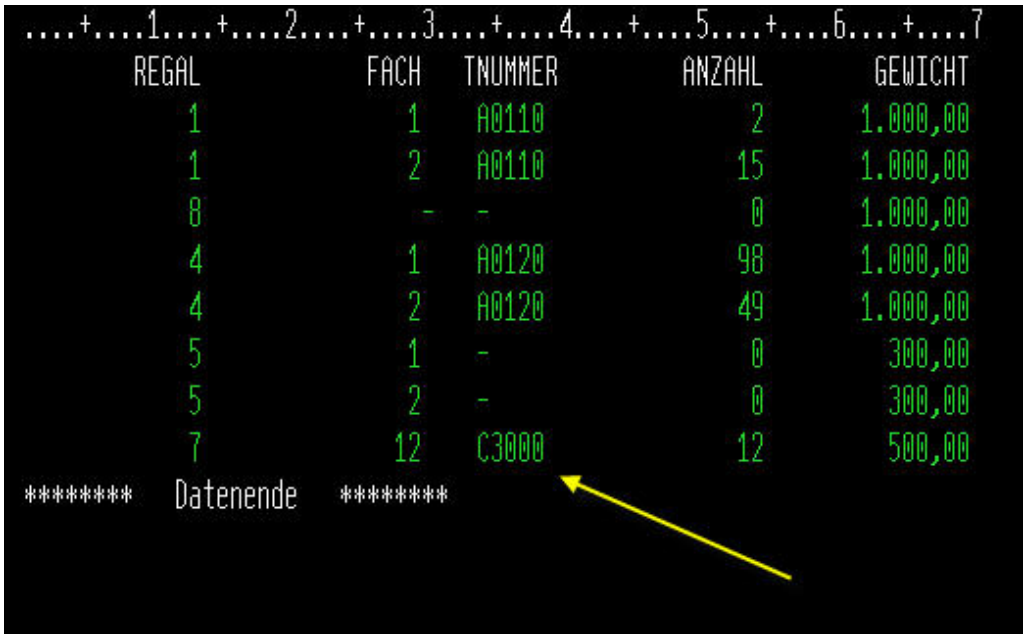
Inhalt der Tabelle LAGER

Bei etwas genauerer Betrachtung ist leicht festzustellen, dass im Lager keine Artikel mit den Teilenummern B1000 beziehungsweise B2000 eingelagert sind. Damit liegt eine Erklärung vor, warum in den Spalten der aus der JOIN-Verknüpfung hervorgegangenen Ergebnistabelle NULL-Inhalte auftauchen. Diese erscheinen nun gerade in Spalten, die ursprünglich aus der Tabelle LAGER (auf der rechten Seite des LEFT JOIN) stammen.

Wir wollen nun überlegen, welches Bild sich ergeben sollte, wenn statt des LEFT JOIN nun ein analog aufgebautes Statement mit einem RIGHT JOIN verwendet wird.

Die Tabelle rechts des RIGHT JOIN ist LAGER und die dort gespeicherten Zeilen werden in der Ergebnistabelle geliefert.

Bevor dieser Versuch unternommen wird, wird jedoch noch ein Artikel in LAGER eingefügt, der in keiner Zeile der Tabelle LIEFERBAR auftaucht.



REGAL	FACH	TNUMMER	ANZAHL	GEWICHT
1	1	A0110	2	1.000,00
1	2	A0110	15	1.000,00
8	-	-	0	1.000,00
4	1	A0120	98	1.000,00
4	2	A0120	49	1.000,00
5	1	-	0	300,00
5	2	-	0	300,00
7	12	C3000	12	500,00

******** Datenende ********

Neuer Inhalt der Tabelle LAGER

Die Zeilen, die nun aufgrund der ON-Klausel ohne eine Entsprechung in der linksseitigen Tabelle (also in LIEFERBAR) sind, werden in den dortigen Spalten dann einen NULL-Inhalt aufweisen. Soweit die Vermutung. Geben wir in der Umgebung folgendes Statement ein.

Ein RIGHT JOIN

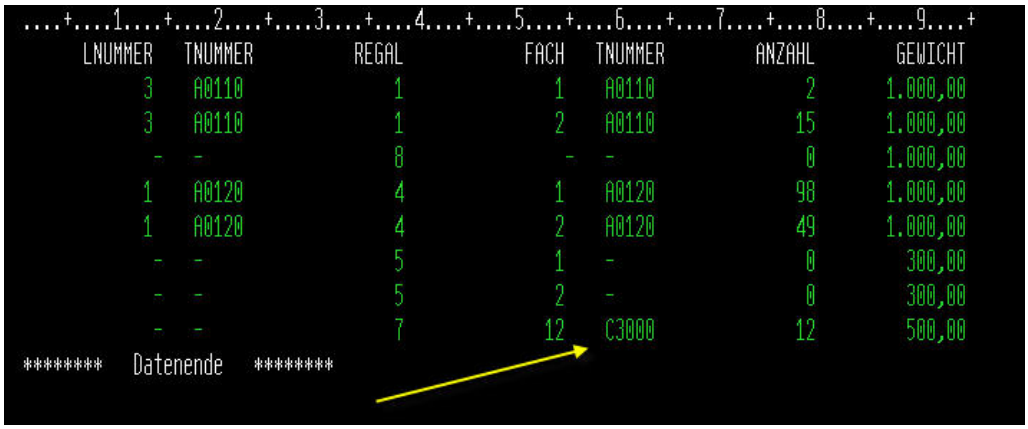
```
SELECT * FROM EPSWDSC/LIEFERBAR A
       RIGHT JOIN EPSWDSC/LAGER B
       ON A.TNUMMER = B.TNUMMER
```

Anweisung mit RIGHT JOIN

Das System führt die Anweisung korrekt aus und liefert folgende Ergebnisse.

3.2.5
Seite 6

Ergebnistabelle des RIGHT JOIN



LNUMMER	TNUMMER	REGAL	FACH	TNUMMER	ANZAHL	GEWICHT
3	A0110	1	1	A0110	2	1.000,00
3	A0110	1	2	A0110	15	1.000,00
-	-	8	-	-	0	1.000,00
1	A0120	4	1	A0120	98	1.000,00
1	A0120	4	2	A0120	49	1.000,00
-	-	5	1	-	0	300,00
-	-	5	2	-	0	300,00
-	-	7	12	C3000	12	500,00

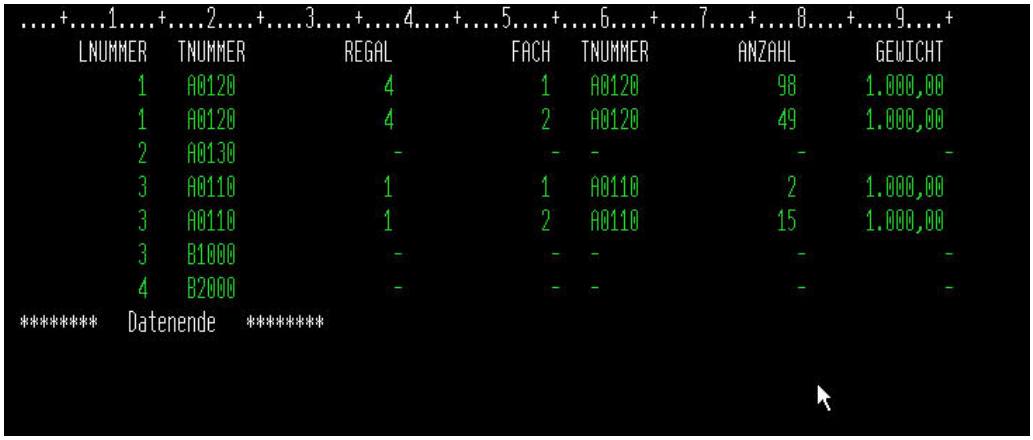
******** Datenende ********

Ergebnis der Abfrage

Auf der rechten Seite, also im Bereich der Tabelle LAGER, befinden sich Stellplätze, auf denen kein Werkstück eingelagert ist. Diese Zeilen liefern in der Spalte TNUMMER dann den Inhalt NULL. Zu diesen Artikeln wird man selbstverständlich auch keinen Lieferanten finden, also tauchen in den entsprechenden Zeilen der Tabelle LIEFERBAR NULL-Werte auf. Aber die Vermutung in Bezug auf den neu eingelagerten Artikel C3000 bestätigt sich hier auch. Zu dieser TNUMMER wird ebenfalls kein Lieferant gefunden, die Folge davon: NULL-Inhalte in den der Tabelle LIEFERBAR zugeordneten Spalten.

Nun ist es noch von Interesse, welche Resultate zu erwarten sind, wenn die verbleibenden Varianten eingesetzt werden. Also bauen wir einmal ein Statement auf, das eine LEFT EXCEPTION JOIN verwendet. Als zugrunde liegende Tabellenreferenzen bleiben wir bei LIEFERBAR und LAGER und bilden noch einmal der Übersicht halber das Ergebnis des LEFT JOIN ab.

LEFT JOIN



LNUMMER	TNUMMER	REGAL	FACH	TNUMMER	ANZAHL	GEWICHT
1	A0120	4	1	A0120	98	1.000,00
1	A0120	4	2	A0120	49	1.000,00
2	A0130	-	-	-	-	-
3	A0110	1	1	A0110	2	1.000,00
3	A0110	1	2	A0110	15	1.000,00
3	B1000	-	-	-	-	-
4	B2000	-	-	-	-	-

******** Datenende ********

Ergebnis der Abfrage

Anschließend wird folgendes Statement aufgerufen.

LEFT EXCEPTION JOIN

```
SELECT * FROM EPSWDSC/LIEFERBAR A
         LEFT EXCEPTION JOIN EPSWDSC/LAGER B
         ON A.TNUMMER = B.TNUMMER
```

Anweisung mit LEFT EXCEPTION JOIN

Nach der Ausführung findet man die Ergebnistabelle.

Ergebnistabelle LEFT EXCEPTION JOIN

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9....+
     LNUMMER   TNUMMER        REGAL          FACH   TNUMMER        ANZAHL       GEWICHT
           2   A0130              -             -   -                  -             -
           3   B1000              -             -   -                  -             -
           4   B2000              -             -   -                  -             -
********   Datenende   ********
```

Ergebnis der Abfrage

Wenn man jetzt noch einmal einen Blick auf das Resultat der LEFT JOIN Anweisung wirft, so erkennt man sofort, dass die Werkstücke mit den Teilenummern A0130, B1000 und B2000 in Bereich der Spalten der Tabelle LAGER nur NULL-Inhalte aufweisen. Mit anderen Worten, es gibt im Lager keine Daten–Zeilen, die den eben genannten Artikelnummern entsprechen. Dieser Sachverhalt hat nun zur Folge, dass diese Sätze in einem INNER JOIN nicht geliefert werden.

```
....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9....+
     LNUMMER   TNUMMER        REGAL          FACH   TNUMMER        ANZAHL       GEWICHT
           1   A0120              4             1   A0120              98      1.000,00
           1   A0120              4             2   A0120              49      1.000,00
           3   A0110              1             1   A0110               2      1.000,00
           3   A0110              1             2   A0110              15      1.000,00
```

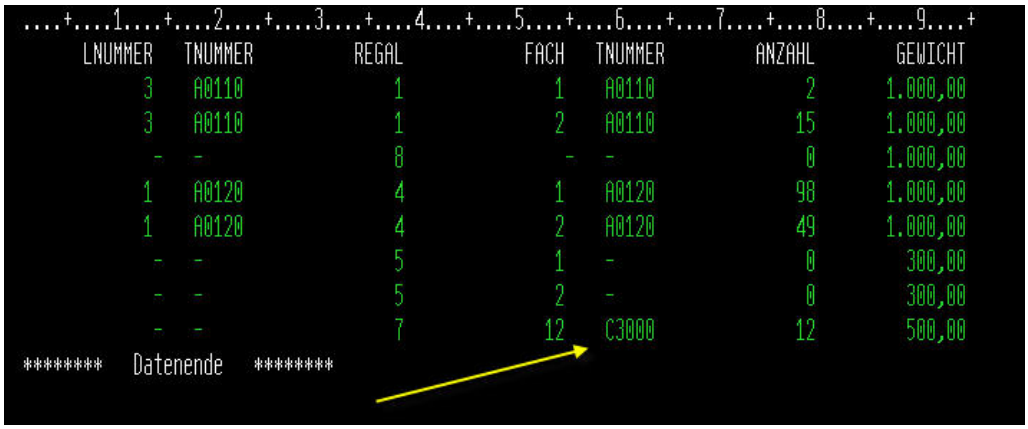
Der entsprechende INNER JOIN

Ein Vergleich der Ergebnistabellen zeigt nun, dass der EXCEPTION JOIN tatsächlich die fortgelassenen Sätze liefert.

Spielen wir nun diesen Sachverhalt noch mit der gespiegelten Variante durch. Für diese Situation müssen wir dann folgende Vergleiche durchführen.

3.2.5
Seite 8

Noch einmal das Resultat des RIGHT JOIN



LNUMMER	TNUMMER	REGAL	FACH	TNUMMER	ANZAHL	GEWICHT
3	A0110	1	1	A0110	2	1.000,00
3	A0110	1	2	A0110	15	1.000,00
-	-	8	-	-	0	1.000,00
1	A0120	4	1	A0120	98	1.000,00
1	A0120	4	2	A0120	49	1.000,00
-	-	5	1	-	0	300,00
-	-	5	2	-	0	300,00
-	-	7	12	C3000	12	500,00

******** Datenende ********

Ergebnis der Abfrage

Aufbauen eines RIGHT EXCEPTION JOIN

```
SELECT * FROM EPSWDSC/LIEFERBAR A
         RIGHT EXCEPTION JOIN EPSWDSC/LAGER B
         ON A.TNUMMER = B.TNUMMER
```

Anweisung mit RIGHT EXCEPTION JOIN

Ergebnistabelle des RIGHT EXCEPTION JOIN

LNUMMER	TNUMMER	REGAL	FACH	TNUMMER	ANZAHL	GEWICHT
-	-	8	-	-	0	1.000,00
-	-	5	1	-	0	300,00
-	-	5	2	-	0	300,00
-	-	7	12	C3000	12	500,00

******** Datenende ********

Ergebnis der Abfrage

Die Überlegung zur Kontrolle:

Der INNER JOIN lässt auf der linken Seite alle Zeilen aus, die durchgängig NULL-Inhalte in ihren Spalten haben. Ein Vergleich der Abbildungen zeigt, dass auch hier vom RIGHT EXCEPTION JOIN diese Sätze zurückgegeben werden.

Vereinfachung der ON-Bedingung

Die bisher gestalteten logischen Verknüpfungen sind recht einfach zu überblicken, was den Aufbau der ON-Bedingung angeht. Wenn aber die durchzuführenden Prüfungen komplexer werden, ist es leicht einzusehen, dass der Aufbau der Kriterien schnell unübersichtlich werden kann. Diese Strukturen können nun aber das Schlüsselwort USING vereinfacht werden.

Denken wir uns einmal folgenden Aufbau. Hier kommt also eine ON-Klausel mit einer ganzen Reihe logisch AND-verknüpfter Vergleiche zum Einsatz.

```
TB1 INNER JOIN TB2
    ON TB1.C1 = TB2.C1 AND
       TB1.C2 = TB2.C2 AND
       ...
       TB1.Cn = TB2.Cn
```

Komplexere ON-Bedingungen

Unter der Voraussetzung, dass die Spalten, über die eine Verknüpfung hergestellt werden soll, in beiden Tabellenreferenzen die gleichen Bezeichnungen haben, kann die ON-Bedingung durch das Schlüsselwort USING in der folgenden Art und Weise ersetzt werden.

```
TB1 INNER JOIN TB2
    USING (C1, C2, ... Cn)
```

USING ersetzt ON in bestimmten Fällen

Übertragen auf unsere obigen einfachen Beispiele kann eine Vereinfachung wie folgt erreicht werden.

Herkömmliche Variante des JOIN mit ON

```
SELECT * FROM EPSWDSC/LIEFERBAR A
         JOIN EPSWDSC/LAGER B
         ON A.TNUMMER = B.TNUMMER
```

Anweisung des JOIN mit ON

Beispielvariante mit USING

```
SELECT * FROM EPSWDSC/LIEFERBAR A
         INNER JOIN EPSWDSC/LAGER B
         USING (TNUMMER)
```

Anweisung mit des INNER JOIN mit USING

Fassen wir nun einmal zusammen, was im Zusammenhang zu den verschiedenen Ausprägungen der logischen Tabellenverbindung über JOIN zu sagen ist.

JOIN oder implizit INNER JOIN

Eine Verbindung der Art Tabelle 1 (INNER) JOIN Tabelle 2 liefert nur zusammengehörige Sätze aus den beteiligten Tabellenreferenzen. Zeilen einer Tabelle, die keine Entsprechungen der anderen Tabelle besitzen, werden nicht zurückgeliefert.

Verwendet man beim INNER JOIN eine ON–Klausel oder auch eine USING-Anweisung, so erhält man das gleiche Resultat, wie beim Einsatz einer entsprechend formulierten WHERE-Bedingung in Verbindung mit einer kommagetrennten Tabellenliste nach dem FROM.

LEFT JOIN beziehungsweise LEFT OUTER JOIN

Alle Zeilen aus der links stehenden Tabelle werden in der Ergebnistabelle geliefert, zusammen mit den entsprechenden Sätzen der rechten Tabelle. Wird zu einem Satz der linken Tabelle kein passender Satz aus der rechten Tabelle gefunden, so werden die dortigen Spalten mit NULL-Werten beschickt.

RIGHT JOIN beziehungsweise RIGHT OUTER JOIN

Diese Verbindung liefert alle Zeilen aus der Tabelle rechts vom JOIN und die passenden Zeilen aus der linken Tabelle. Sätze aus der rechten Tabelle ohne Entsprechung in der linken Tabelle füllen die Ergebnisspalten der linken Tabelle mit NULL-Werten auf.

LEFT EXCEPTION JOIN

Diese Verbindung liefert die unpaarigen Sätze der links stehenden Tabelle und die mit NULL aufgefüllten Spalten der rechten Tabelle.

RIGHT EXCEPTION JOIN

Eine Verknüpfung in dieser Form stellt das Spiegelbild des LEFT EXCEPTION JOIN dar, liefert also alle unpaarigen Zeilen der rechten Tabelle und die NULL-Inhalte der linken Tabelle.

CROSS JOIN

Diese bisher nicht behandelte Spielart des JOIN liefert als Ergebnistabelle eine Verbindung von jedem Satz aus der links stehenden Tabelle mit jedem Satz aus der rechts stehenden. Die erste Zeile (linke Tabelle) wird mit der ersten, zweiten, dritten… Zeile der anderen Tabelle (rechts) verbunden. Der gleiche Vorgang erfolgt dann weiter für jede Zeile der ersten Tabelle. Beachten Sie bitte die folgenden Abbildungen zum CROSS JOIN. In der Grafik der Ergebnistabelle sind die Wiederholungen der Zeilen der Tabelle LIEFERBAR markiert.

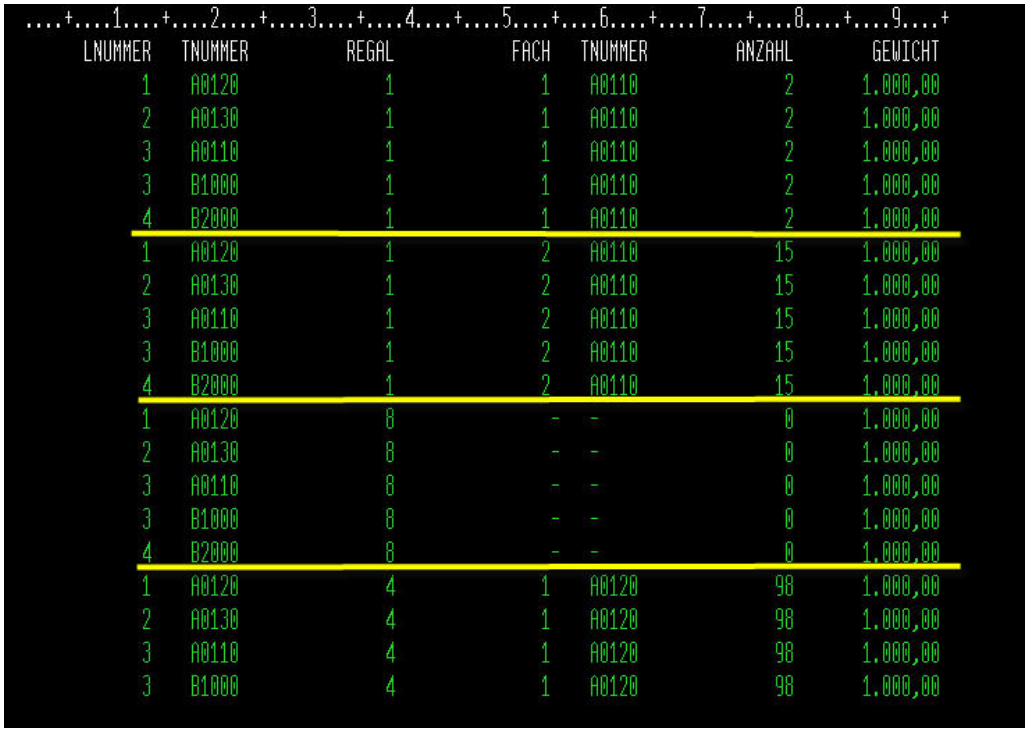
CROSS JOIN Statement

```
SELECT * FROM EPSWDSC/LIEFERBAR A
         CROSS JOIN EPSWDSC/LAGER B
```

Anweisung mit CROSS JOIN

3.2.5
Seite 12

Ergebnistabelle des CROSS JOIN



LNUMMER	TNUMMER	REGAL	FACH	TNUMMER	ANZAHL	GEWICHT
1	A0120	1	1	A0110	2	1.000,00
2	A0130	1	1	A0110	2	1.000,00
3	A0110	1	1	A0110	2	1.000,00
3	B1000	1	1	A0110	2	1.000,00
4	B2000	1	1	A0110	2	1.000,00
1	A0120	1	2	A0110	15	1.000,00
2	A0130	1	2	A0110	15	1.000,00
3	A0110	1	2	A0110	15	1.000,00
3	B1000	1	2	A0110	15	1.000,00
4	B2000	1	2	A0110	15	1.000,00
1	A0120	8	-	-	0	1.000,00
2	A0130	8	-	-	0	1.000,00
3	A0110	8	-	-	0	1.000,00
3	B1000	8	-	-	0	1.000,00
4	B2000	8	-	-	0	1.000,00
1	A0120	4	1	A0120	98	1.000,00
2	A0130	4	1	A0120	98	1.000,00
3	A0110	4	1	A0120	98	1.000,00
3	B1000	4	1	A0120	98	1.000,00

Ergebnis der Abfrage

Unions

Weiterführendes zum SELECT–Statement

Die bis zu dieser Stelle besprochenen Syntaxen und Auswertungsmöglichkeiten sind dem so genannten SubSelect zuzuordnen. Dieser SubSelect ist nun ein Bestandteil des FullSelect. Und hier kommen nun weitere Schlüsselworte zum Tragen. Auch der FullSelect repräsentiert eine Ergebnistabelle.

Werden in einem solchen Statement weder die Schlüsselworte wie UNION, EXCEPT oder INTERSECT verwendet, so entspricht die Ergebnistabelle dem Ergebnis des SubSelects.

Mit UNION DISTINCT oder UNION ALL können nun zwei Ergebnistabellen miteinander vereinigt werden. Kommt UNION ALL zum Einsatz, so wird das Resultat aus allen Sätzen beider Result Tables bestehen. Ohne die Angabe ALL werden doppelte Sätze eliminiert.

Verbindet man die Ergebnistabellen T1 und T2 mit einem EXCEPT DISTINCT, so zeigt sich das Ergebnis in Form aller Zeilen, die nur in T1 zu finden sind, wobei eventuell doppelte Zeilen auch hier nicht aufgeführt werden.

Die Anwendung eines INTERSECT DISTINCT resultiert in einer Ergebnistabelle, die nur die Zeilen enthält, die in beiden Tabellen enthalten sind. Es wird also eine Schnittmenge gebildet. Mehrfach auftretende Sätze werden nicht geliefert.

Es sollen noch einige Bedingungen für Spalten aufgeführt werden, die in diesem Zusammenhang beachtet werden müssen. So müssen die miteinander verbundenen Ergebnistabellen eine identische Anzahl von Spalten haben und der Datentyp der x-ten Spalte der ersten Ergebnistabelle muss mit dem Datentyp der x-ten Spalte des zweiten Result Sets übereinstimmen. Erwähnt sei hier noch, dass Character-Typen mit Zeit – Datum – Typen vereinbar sind.

Bei der Verwendung von INTERSECT, UNION und EXCEPT dürfen wie an anderer Stelle auch keine LOB und DATALINK–Typen verwendet werden.

Beispiel für UNION

```
SELECT T1 FROM
   (SELECT TNUMMER AS T1 FROM EPSWDSC/LIEFERANT) AS A
      UNION
   (SELECT TNUMMER AS T2 FROM EPSWDSC/LIEFERBAR)
```

Anweisung mit UNION

3.2.6
Seite 2

Die Ergebnistabelle des vorhergehenden Statements

```
T1
A0110
A0120
A0130
B1000
B2000
********   Datenende   ********
```

Ergebnis der Abfrage

Beispiel für UNION ALL

```
SELECT T1 FROM
   (SELECT TNUMMER AS T1 FROM EPSWDSC/LIEFERANT) AS A
      UNION ALL
   (SELECT TNUMMER AS T2 FROM EPSWDSC/LIEFERBAR)
```

Anweisung mit UNION ALL

Die Ergebnistabelle des UNION ALL

```
T1
A0110
A0110
A0110
A0110
A0120
A0130
A0110
B1000
B2000
********   Datenende   ********
```

Ergebnis der Abfrage

Beispiel-Statement für EXCEPT DISTINCT (leere Ergebnistabelle!)

```
SELECT T1 FROM
   (SELECT TNUMMER AS T1 FROM EPSWDSC/LIEFERANT) AS A
      EXCEPT DISTINCT
   (SELECT TNUMMER AS T2 FROM EPSWDSC/LIEFERBAR)
```

Anweisung mit EXCEPT DISTINCT

Beispiel mit vertauschten SELECTs

```
SELECT T1 FROM
   (SELECT TNUMMER AS T1 FROM EPSWDSC/LIEFERBAR) AS A
      EXCEPT DISTINCT
   (SELECT TNUMMER AS T2 FROM EPSWDSC/LIEFERANT)
```

Anweisung mit vertauschten SELECTs

Die resultierende Ergebnistabelle liefert nur Sätze, die in LIEFERBAR vorliegen.

EXCEPT DISTINCT für LIEFERBAR

```
T1
A0120
A0130
B1000
B2000
********   Datenende   ********
```

Ergebnis der Abfrage

Die Artikelnummer A0110 taucht hier nicht auf. Zur Kontrolle selektieren wir einmal die Tabellen LIEFERBAR und LIEFERANT auf folgende Weise.

3.2.6
Seite 4

Vergleichende SELECTs

```
SELECT * FROM EPSWDSC/LIEFERBAR WHERE TNUMMER = 'A0110'
SELECT * FROM EPSWDSC/LIEFERANT WHERE TNUMMER = 'A0110'
```

Anweisung mit vergleichenden SELECTs

Wir erhalten dabei folgende Resultate.

LIEFERBAR

```
     LNUMMER   TNUMMER
           3    A0110
********   Datenende   ********
```

Ergebnis der Abfrage

LIEFERANT

```
     LNUMMER   TNUMMER        STATUS
           4    A0110             10
           5    A0110             20
           6    A0110             60
           7    A0110             70
********   Datenende   ********
```

Ergebnis der Abfrage

Da A0110 nun in beiden Tabellen zu sehen ist, darf diese Artikelnummer von EXCEPT DISTINCT auch nicht zurückgegeben werden. Das Resultat ist also korrekt.

Beispiel für INTERSECT DISTINCT

```
   (SELECT TNUMMER AS T1 FROM EPSWDSC/LIEFERBAR)
      INTERSECT DISTINCT
   (SELECT TNUMMER AS T2 FROM EPSWDSC/LIEFERANT)
```

Anweisung mit INTERSECT DISTINCT

Wie erläutert, bildet INTERSECT eine Schnittmenge und die Ergebnistabelle stellt sich wie folgt dar.

Schnittmenge

Ergebnis der Abfrage

Ein Wort noch zu den notwendigen Berechtigungen. Bei jeder Form einer Abfrage muss zumindest die Lese-Berechtigung für jede angegebene Tabellenreferenz und jeden View vorliegen sowie die Ausführungsberechtigung für die Bibliothek, die die Tabelle oder den View enthält.

Wenn ein Ausdruck die Durchführung einer Funktion erfordert, soll ebenfalls die Ausführungsberechtigung erteilt sein.

Die Vervollständigung des SELECT–Statements

Nachdem nun die Begriffe SubSelect und auch FullSelect besprochen wurden, können wir das SELECT–Statement vervollständigen.

Im folgenden Statement wird Ihnen ein kleiner Ausblick auf die Einbettung von SQL-Anweisungen in eine Hochsprache gegeben, in diesem Fall in C. Eine nähere Besprechung der Syntaxen zur Einbettung ist hier nicht notwendig. Aber was geschieht hier?

Es wird ein so genannter Cursor deklariert, der es dem Programm erlaubt, Satz für Satz aus der Ergebnistabelle des SELECT-Statements auszulesen und weiter zu verarbeiten. Die Spalten PRSTDATE und PRENDDATE sind hier für ein Fortschreiben an anderer Stelle des Programms deklariert. Um Einfluss auf die Ausführungsgeschwindigkeit zu nehmen, wird das Statement für die Beschaffung von zwei Datensätzen voreingestellt. Der Zugriffspfad liefert so eine hohe Performance für die angegebene Anzahl von Sätzen.

Natürlich ist damit keinesfalls gesagt, dass nur die zwei Zeilen gelesen werden können – die Rückgabe von weiteren Sätzen ist zulässig. In diesem Fall kann die Geschwindigkeit der Ausführung aber unter Umständen Einbußen erleiden.

3.2.6
Seite 6

```
EXEC SQL  DECLARE UP_CUR CURSOR FOR
            SELECT PROJNO, PRSTDATE, PRENDATE
              FROM PROJECT
              FOR UPDATE OF PRSTDATE, PRENDATE
              OPTIMIZE FOR 2 ROWS ;
```

Anweisung

Über das Schlüsselwort kann das Isolation Level eingestellt werden. Im folgenden Beispiel auf einen Wert „wiederholtes Lesen (RS)“. Was sich hinter diesen Begriffen verbirgt wird in anderen Abschnitten noch weitergehend erläutert werden.

Isolation Level beeinflussen

```
SELECT * FROM EPSWDSC/LIEFERBAR
   WHERE TNUMMER = 'A0110'
   WITH RS
```

Anweisung mit RS

SQL-Performance-Betrachtungen

Möglicherweise haben Sie sich bisher beim Einsatz von SQL-Abfragen eher dafür interessiert, wie Ihr Select-Statement aufgebaut sein muss, damit Sie die benötigten Daten erhalten. Sie haben Abfragen geschrieben, ausgeführt und sich anschließend unter Umständen gefragt, warum eine so einfache Anfrage „soooo“ lange braucht. Mit dem guten alten „Read“ wäre das alles viel schneller gegangen, mag so mancher sattelfester und gestandener RPG-Spezialist dann sagen.

Ganz so einfach ist dieser Vergleich jedoch nicht: Mit einer RPG-Anweisung greifen Sie stets auf einen Satz bzw. eine Zeile der Tabelle zu. Mit SQL bearbeiten Sie Datenmengen. Nicht selten werden mehr als 100.000 Zeilen verarbeitet, um das gewünschte Ergebnis zu erzielen. Außerdem müssen Sie in Ihrem RPG-Programm die Logik kodieren – d. h.: Sie bestimmen im RPG-Programm nicht nur, welche Daten benötigt werden, sondern kodieren auch den Algorithmus, der diese Daten zur Verfügung stellt. Das ist der entscheidende Unterschied: Wenn Sie eine SQL-Abfrage erstellen, definieren Sie das Abfrageergebnis nicht aber den Ausführungsweg. Über den geeigneten Ausführungsplan entscheidet das System – genauer gesagt der Teil des Systems, der Query Engine beziehungsweise Optimizer genannt wird. Der Optimizer schätzt unterschiedliche Ausführungskosten für unterschiedliche Zugriffswege und wird sich letztendlich für den Zugriffsweg mit den geringsten Kosten entscheiden. Als Kosten werden hier sowohl der Aufwand im Eingabe-/Ausgabe-Bereich (I/O-Kosten) als auch der Kostenfaktor im Arbeitsspeicher (CPU Kosten) für die Datenverarbeitung berücksichtigt. Aufgabe des Optimizers ist es nun, bei der Zugriffgestaltung zwischen diesen Kosten abzuwägen und die rationellste Variante einzusetzen. Um diesen Optimierungsprozess zu verstehen, müssen Sie sich zunächst ein wenig mit dem Optimizer auseinandersetzen. Wie ist der Optimizer aufgebaut – und vor allem: anhand welcher Gesichtspunkte entscheidet der Optimizer?

3.4
Seite 2

Überblick SQL Query Optimizer

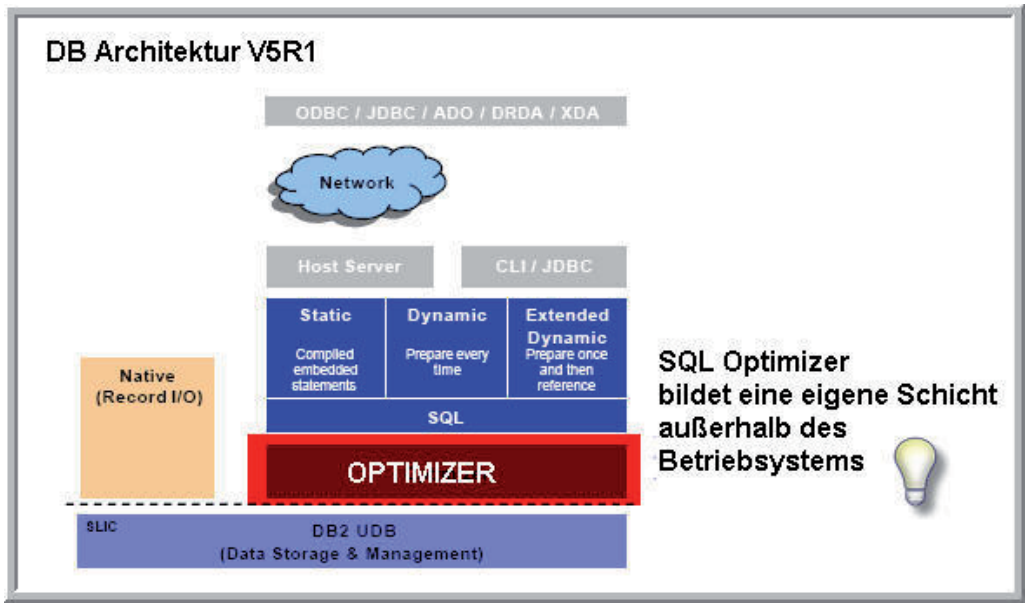
Der Query Optimizer ist eine der wichtigsten Komponenten der DB2 for i5/OS. Er

- entscheidet, welche indizierten Zugriffspfade für die Abfrage in Frage kommen.
- identifiziert die infrage kommenden Techniken.
- wählt die effizienteste Zugriffsmethode für die Abfrage aus.

Der Query Optimizer versucht für Ihre SQL-Abfrage stets den optimalen Zugriff auf die Daten zu ermitteln. Hierbei arbeitet er kostenbasierend – d. h.: Er berechnet für unterschiedliche Implementierungen die Ausführungskosten und entscheidet sich aufgrund dieser Berechnungen für die Ausführungsstrategie mit dem vermutlich geringsten Kostenfaktor. Für diesen Entscheidungsprozess spielen einerseits die I/O-Kosten und andererseits die CPU-Kosten eine entscheidende Rolle. Um die Berechnungen durchführen zu können, sammelt der Query Optimizer zunächst Informationen über die Tabelle und die Query-Umgebung. Die Anzahl möglicher Tabellensätze, die Größe der Tabelle und vorhandene Indizes spielen hierbei eine entscheidende Rolle. Es wird aber auch die Anzahl der CPUs und die Größe des Arbeitsspeichers ermittelt. Auf Basis dieser Informationen bestimmt der Query Optimizer letztendlich den auszuführenden Zugriffsplan.

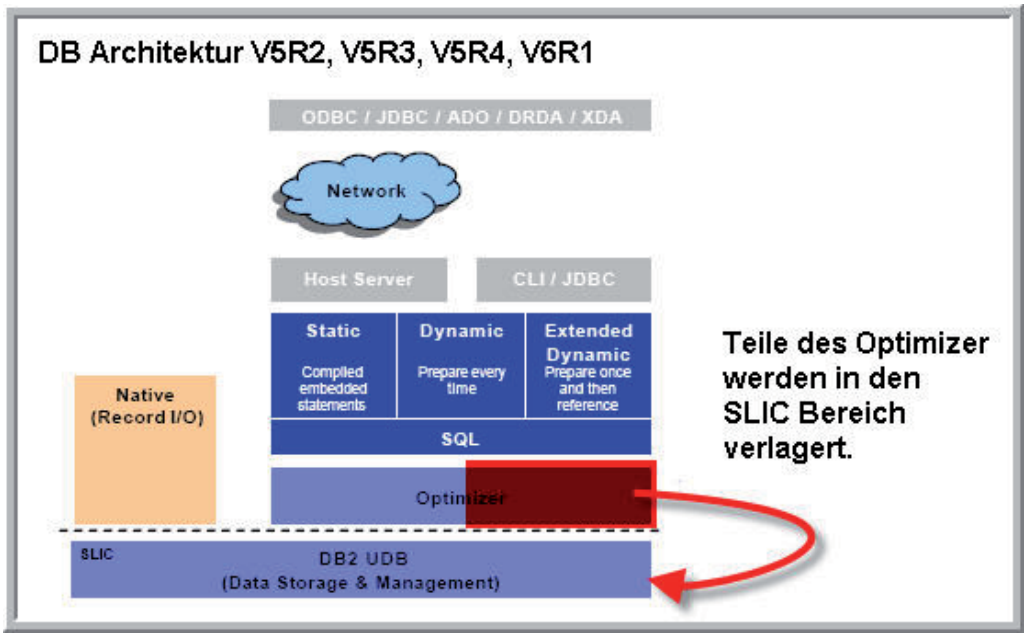
Die Wurzeln der DB2 for i5/OS finden wir auf dem IBM System /38. Bereits damals war die Datenbank eine in das Betriebssystem integrierte relationale Datenbank. Die AS/400 übernahm 1988 diese Datenbanktechnologie und unterscheidet sich hierdurch bis heute von allen Konkurrenzprodukten. Neben dem traditionellen nativen Interface implementierte IBM aber auf der AS/400 bereits 1988 ein eigenständiges SQL Interface. Beide Schnittstellen – native Zugriffsmethoden und SQL-Zugriffe – existierten nebeneinander und erfüllten hervorragend die Kundenanforderungen der letzten 20 Jahre. Allerdings bildeten der Query Optimizer und die SQL Engine bis zum Release V5R2 eine eigene Schicht. Das SQL Interface war somit nicht integraler Bestandteil des Betriebssystems.

3.4.1
Seite 2



Datenbankarchitektur vor V5R2

Doch gerade in den letzten Jahren wandelten sich die Anforderungen an moderne Datenbanken. Immer mehr Applikationen nutzten aufgrund neuer Technologien SQL für den Datenbankzugriff. Immer schneller musste die Datenbank neue Anforderungen erfüllen und neue Features enthalten, um am Markt bestehen zu können. Und schließlich kam der Punkt an dem IBM sich langfristig für eine Strategie entscheiden musste, um die DB2 for i5/OS zukunftsfähig zu machen. IBM entwickelte große Teile des Query Optimizers und Teile der Runtime Engine neu, entschied sich aber, den neuen Query Optimizer parallel zum bisherigen Optimizer zu installieren und garantierte dadurch, dass bestehende Applikationen weiterhin problemlos arbeiteten. Gleichzeitig konnten alte und neue Applikationen von der neuen Technologie profitieren. Diese neue Engine ist unter dem Namen SQL Query Engine (SQE) bekannt. Der bestehende Query Optimizer erhielt den Namen Classic Query Engine (CQE). Die SQE-Neuentwicklung veränderte aber auch die DB-Architektur als Ganzes. IBM verlagerte nämlich große Teile der SQL-Optimierung in das Betriebssystem, so dass jetzt auch die SQL Engine integraler Bestandteil des Betriebssystems ist.

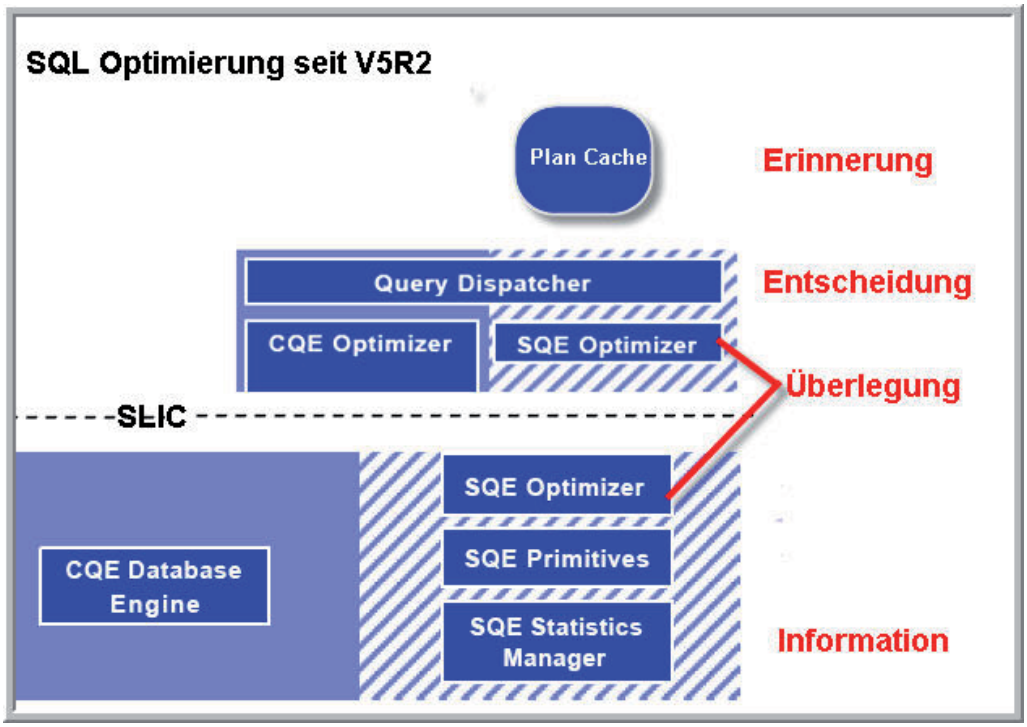


DB-Architektur seit V5R2

Im Release V5R2 wurden zunächst nur lesende Abfragen von der SQE unterstützt. Nach und nach erhöhte sich die Menge der ausgeführten SQL Statements auf Seiten der SQE. Dieser Prozess wird fortgesetzt, bis alle SQL Statements, die über ein SQL Interface (interaktives SQL, embedded SQL, ODBC, JDBC) ausgeführt werden, von der SQE übernommen werden. Das Release V6R1 beendet diesen Prozess vorerst.

Aber es wurde nicht nur der Optimierungsprozess in den LIC verlagert, der neue Query Optimizer erhielt auch viele neue, eigenständige Komponenten, wie nachfolgende Grafik zeigt:

3.4.1
Seite 4



SQE-Komponenten Release V5R2

Jede der Komponenten nimmt unterschiedliche Aufgaben wahr:

Dem Optimierungsprozess vorgeschaltet ist der **Query Dispatcher**. Er hat die Aufgabe, jedes SQL Statement zu analysieren; er entscheidet anhand der verwendeten Syntax, ob die CQE oder die SQE die Verarbeitung übernehmen soll. Diese „Abfertigungsstelle" können Sie nicht umgehen; Sie haben auch keinen direkten Einfluss auf die Entscheidung des Dispatchers. Welcher Optimizer die Verarbeitung des SQL Statements übernimmt, ist abhängig von der verwendeten Syntax im SQL Statement. Je nach Release bestimmen die verwendeten Klauseln und SQL-Funktionen den Ausführungsort. Das Release V6R1 wird den Übernahmeprozess auf die SQE weitestgehend abschließen. Bis auf wenige Ausnahmen können alle Abfragen auf der SQE ausgeführt werden. Allerdings werden auch weiterhin SQL-Abfragen mit logischen DDS-Dateien nicht durch die SQE unterstützt.

Für den SQE-Optimierungsprozess liefert der **Statistik-Manager** anschließend die benötigten Informationen. Vor Release V5R2 war der Optimizer selbst für das Sammeln von Informationen zuständig – d. h.: Der Optimizer prüfte während des Optimierungsprozesses die Tabelleneigenschaften und deren Indizes, um anschließend anhand der ermittelten Daten den Zugriffsplan zu erstellen. Die CQE arbeitet noch heute nach diesem Prinzip. Für die SQE gibt es allerdings seit Release V5R2 eine eigenständige Komponente: den Statistik Manager, der diese Aufgabe erfüllt. Diese Komponente ist nicht für die Optimierung oder Ausführung des SQL Statements verantwortlich, son-

dern hat die Kontrolle auf den Zugriff der Metadaten einer Tabelle. Er verfügt über weitere Informationen, die für den Optimierungsprozess notwendig sind. Man kann sagen, der Statistik-Manager hat zwei grundlegende Aufgaben zu erfüllen:

– Einerseits erstellt und pflegt er Statikinformationen für einzelne Spalten einer Tabelle.
– Andererseits beantwortet der Statistik-Manager Fragen der SQE, die für die Ermittlung des optimalen Zugriffplanes relevant sind.

Der System i Navigator enthält daher seit Release V5R2 einige Funktionen, um mit Statistikdaten zu arbeiten. Ausführliche Informationen über die Arbeit mit Statistikdaten enthält das Kapitel 3.5.7.

SQE Data Access Primitives ist eine weitere Komponente der SQE. Diese Komponente ist für die eigentliche Ausführung des SQL Statements verantwortlich. Sie nutzt viele Zugriffsverfahren, die bereits für die CQE implementiert wurden, verfügt aber auch über eine Vielzahl neuer Möglichkeiten. Sie werden eventuell feststellen, dass Hauptspeicher und Subsystemressourcen offensiver von der SQE genutzt werden als durch die CQE. Dies ist die Folge der verbesserten Integration in den LIC. SQE Data Access Primitives lagern geschickt die benötigten Daten seitenweise in den Arbeitsspeicher und können im weiteren Verlauf die Daten der eingelagerten Seiten direkt verarbeiten, indem Hash-Tabellen oder Bitmaps aufgebaut werden. Dies ist auch der Grund, warum die SQE sehr viel häufiger als die CQE „Table Scans“ durchführt. Es ist sicherlich nicht notwendig, alle Ausführungskomponenten und Implementierungsmethoden des Optimizers detailliert zu besprechen, aber für die spätere Arbeit ist es hilfreich, sich ein wenig mit den zur Verfügung stehenden Implementierungsmethoden auseinanderzusetzen:

Für den Datenzugriff stehen zwei grundlegende Methoden zur Auswahl:

– SCAN
  Wenn die SQL Engine Objekte während der Verarbeitung scannt, werden die Sätze sequentiell und seitenweise verarbeitet, ohne dass Auswahlkriterien für die benötigten Daten zuvor überprüft werden.
– PROBE
  Das Ausführungsverfahren PROBE dagegen entspricht einem wahlfreien Zugriff auf die Zeilen einer Tabelle. Die Selektionskriterien werden geprüft, bevor die Sätze ausgewählt werden.

Beide Zugriffsverfahren können auf permanente und temporäre Objekte angewandt werden. Die Datenbank kennt zwei permanente Objekte: Tabellen und Indizes, so dass die eben beschriebenen Zugriffsverfahren sowohl für Indizes als auch für Tabellen genutzt werden können.

17. Ergänzung 12/2008

Neben den permanenten Objekten stehen aber auch zahlreiche temporäre Objekte zur Verfügung, die zur Laufzeit im Arbeitsspeicher aufgebaut werden, aber deshalb nicht weniger effektiv sein müssen.

– Temporäre Hash-Tabellen
  Es können temporäre Hash-Tabellen aufgebaut werden. Eine Hash-Tabelle ist eine Datenstruktur, in der die Tabellendaten anhand einer oder mehrerer Tabellenspalten sortiert werden. Die Hash-Tabelle kann, nachdem sie gefüllt wurde, verwendet werden, um alle Tabellendaten, die einem bestimmten Hash-Wert entsprechen, abzurufen.

– Temporäre Zeilennummernlisten
  Eine Zeilennummernliste ist ebenfalls eine temporäre Datenstruktur, die die Zeilen der Tabelle repräsentiert, die den lokalen Auswahlkriterien der SQL-Abfrage entsprechen. Da die ausgewählten Zeilen durch die relative Satznummer abgebildet werden, können mehrere Listen gemischt und kombiniert werden, um komplexe Selektionskriterien zu verarbeiten, ohne dass die Tabelle selbst in den Arbeitsspeicher verlagert werden muss. Zeilennummernlisten ermöglichen somit ein Zusammenführen mehrerer Teilergebnisse. Erst im Anschluss daran müssen die Sätze in den Arbeitsspeicher verlagert werden. Hierdurch reduziert sich die Anzahl erforderlicher I/O-Operationen.

– Temporäres Zeilennummern-Bitmap
  Ein Zeilennummern-Bitmap ist vergleichbar mit der eben beschriebenen Zeilennummernliste. Allerdings wird im Zeilennummernbitmap der zugehörige Tabellensatz durch ein einzelnes Bit und nicht durch die relative Satznummer abgebildet.

– Temporäre Indizes
  Der Vollständigkeit halber sollen die temporären Indizes an dieser Stelle erwähnt werden. Temporäre Indizes können erstellt werden, weil die Abfrage einen Index erfordert, aber kein geeigneter Index vorhanden ist, oder weil ein Index mit zusätzlichen Schlüsselspalten das Laufzeitverhalten der Abfrage deutlich verbessern kann. Weiterführende Informationen finden Sie im Kapitel 3.4.2.

Das Ergebnis der Optimierung – der Ausführungsplan – wird letztendlich im **Plan Cache** abgelegt. Der Plan Cache ist keine unmittelbare Komponente der SQE; er ist das „Lager“ für bestehende Ausführungspläne. Die Wirkungsweise des Plan Cache ist einfach aber effektiv. Im Plan Cache werden alle optimierten Zugriffspläne der SQE gespeichert. Dadurch ist die Wiederverwendung garantiert. Wird dieselbe oder eine ähnliche Abfrage erneut ausgeführt, kann der gespeicherte Plan für die Ausführung der Abfrage verwendet werden. Die Optimierungszeit wird so eingespart. Die gespeicherten Pläne sind für alle SQL-Abfragen unabhängig vom genutzten Interface oder Benutzer zugänglich. Neben der Originalabfrage und dem Zugriffsplan enthält der

Ausführungsplan auch das generierte ausführbare Query-Runtime-Objekte und Informationen über die Laufzeitumgebung. Da das SQL Statement selbst als Primary Hash Key verwendet wird, ist es möglich, für identische SQL Statements derzeit maximal drei unterschiedliche Ausführungspläne im Plan Cache zu hinterlegen. So können unterschiedliche Laufzeitumgebungen abgebildet werden:

– unterschiedliche SMP-Einstellungen,

– unterschiedliche Bibliothekslisten,

– unterschiedliche ALWCPYDTA Einstellungen,

– unterschiedliche FOR OPTIMIZE Klauseln.

Unabhängig vom Maschinentyp und der Hardware Ihres Servers beansprucht der Plan Cache maximal 512 MB Hauptspeicher (ab V5R4). Wie viele Pläne in diesem Bereich gespeichert sind, ist nicht allgemein zu beantworten, da die Komplexität der jeweiligen SQL Statements hierbei eine entscheidende Rolle spielt. IBM spricht von ca. 6.000 Plänen. Wenn der Cache-Bereich die Maximalgröße erreicht, dann bereinigt ein Hintergrundjob automatisch die Einträge. Hierbei spielt das Datum der Erstellung und die Anzahl der Zugriffe auf den jeweiligen Plan eine entscheidende Rolle. Bis zum Release V5R4 haben Sie nur indirekt Einfluss auf die Neuerstellung der Pläne. Wenn Sie

– die Attributoption bzw. Compiler Directive REOPTIMIZE_ACCESS_PLAN(*YES) oder (*FORCE) verwenden,

– eine Tabelle oder einen Index löschen, der im gespeicherten Zugriffsplan verwendet wird,

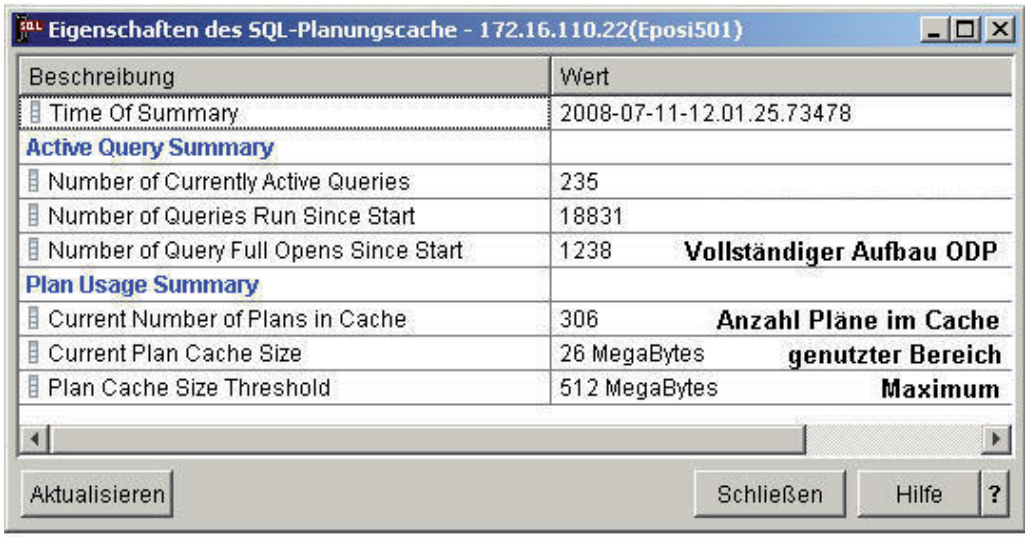
– Sie einen neuen Index erstellen, der als möglicher Kandidat für die Abfrage zur Verfügung steht,

wird der gespeicherte Ausführungsplan verworfen. Seit Release V5R4 können Sie sich die Eigenschaften Ihres Plan Cache übrigens ansehen. System i Navigator stellt dafür ein geeignetes Interface zur Verfügung. Öffnen Sie – wie in unten stehender Abbildung gezeigt – Ihre Datenbank und wählen Sie anschließend den Eintrag „Momentaufnahmen für SQL-Planungscache“ → „SQL-Planungscache“ → „Eigenschaften“.

3.4.1
Seite 8

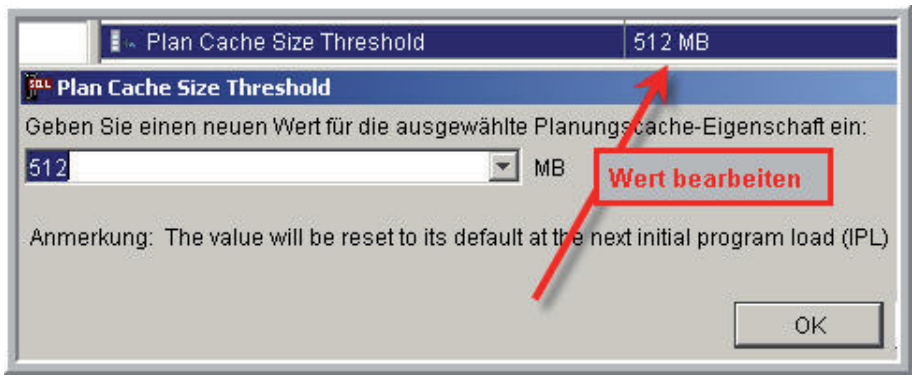
Plan-Cache-Eigenschaften anzeigen V5R4

Daraufhin erscheint folgendes Fenster:



Eigenschaften des Plan Cache

Seit Release V6R1 können Sie die Maximalgröße des Cache-Bereiches manuell beeinflussen. Hierfür klicken Sie mit der rechten Maustaste auf den Eintrag „Plan Cache Size Threshold“ und wählen „Wert bearbeiten“. Im daraufhin folgenden Dialog können Sie die Maximalgröße individuell bestimmen.

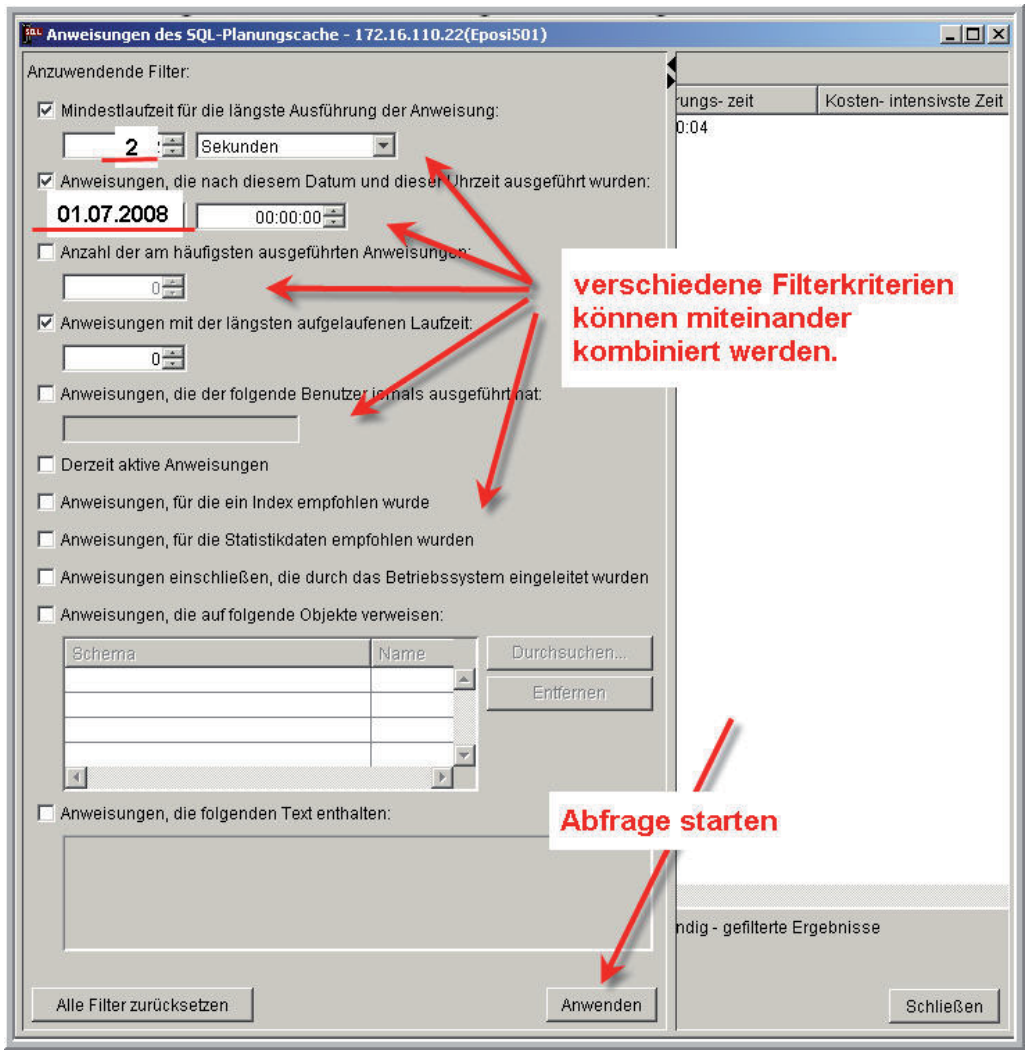


Eigenschaften des Plan Cache ändern V6R1

Seien Sie aber vorsichtig bei der Veränderung dieser Einstellungen und beobachten Sie genau, welche Auswirkungen sie auf die Performance Ihrer SQL-Abfragen haben. Wenn Sie ein IPL durchführen, wird dieser Wert auf den Defaultwert 512 MB zurückgesetzt.

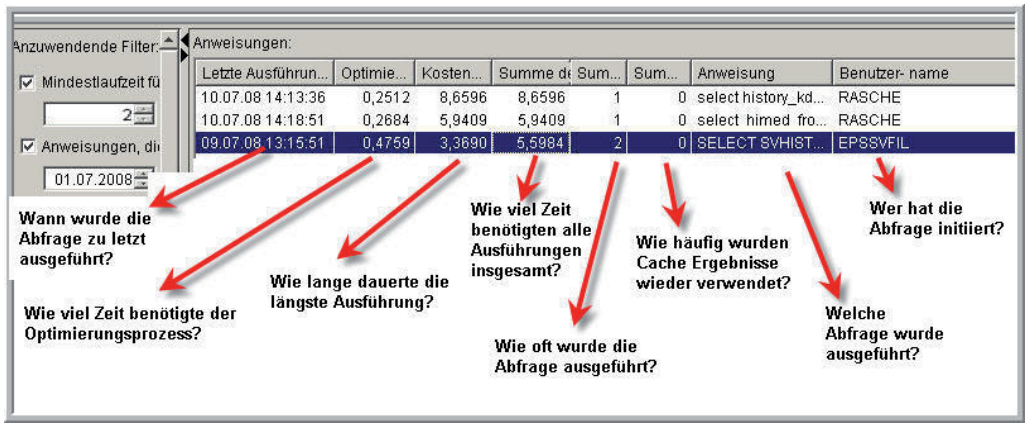
Vielleicht interessiert es Sie auch, welche Pläne der Plan Cache enthält? Seit Release V5R4 ist es möglich, sich den Inhalt anzeigen zu lassen. Klicken Sie hierfür auf den Eintrag „Anweisungen anzeigen…“. Daraufhin öffnet sich ein neuer Dialog. Das Fenster „Anweisungen des SQL-Planungscache“ besteht aus zwei Bereichen: der linke Bereich dient zur Selektion der SQL-Anweisungen. Hierbei können Sie zwischen verschiedenen Kriterien wählen, auch eine Kombination ist möglich.

3.4.1
Seite 10



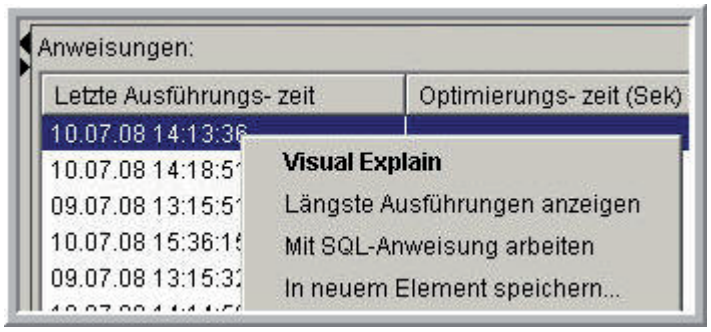
Ausführungspläne selektieren

Im Beispiel möchte ich alle Ausführungspläne selektieren, die nach dem 01.07.2008 ausgeführt wurden und für die Ausführung mindestens zwei Sekunden benötigten. Hierfür trage ich zunächst im Selektionskriterium „Anweisungen, die nach diesem Datum und dieser Uhrzeit ausgeführt wurden“ das Datum „01.07.2008“ ein und setze im Kriterium „Mindestlaufzeit für die längste Ausführung der Anweisung“ den Wert „2“ ein. Anschließend klicke ich auf den Button „Anwenden“. Nach einer mehr oder wenig langen Wartezeit erscheinen die Ergebnisse im rechten Fensterbereich:



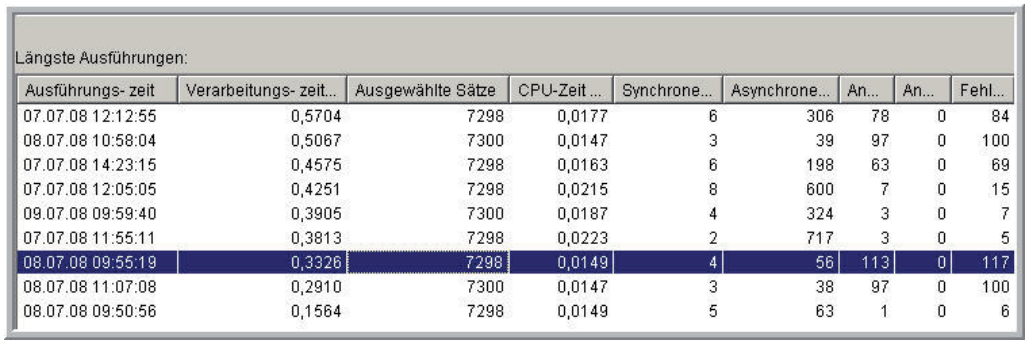
Selektierte Ausführungspläne

Sie erhalten eine Vielzahl nützlicher Informationen. Über ein Kontextmenü stehen weitere Funktionen zur Verfügung:



Kontextmenü

Interessant in diesem Zusammenhang ist die Option „Längste Ausführungen von SQL-Anweisungen anzeigen“. Wenn Sie diese Funktion wählen, werden anschließend in einem neuen Fenster die zehn längsten Ausführungen dieses Statements dargestellt.

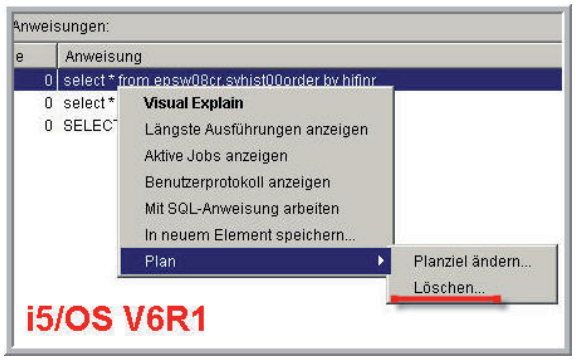


Längste Ausführungen:

Ausführungs- zeit	Verarbeitungs- zeit...	Ausgewählte Sätze	CPU-Zeit ...	Synchrone...	Asynchrone...	An...	An...	Fehl...
07.07.08 12:12:55	0,5704	7298	0,0177	6	306	78	0	84
08.07.08 10:58:04	0,5067	7300	0,0147	3	39	97	0	100
07.07.08 14:23:15	0,4575	7298	0,0163	6	198	63	0	69
07.07.08 12:05:05	0,4251	7298	0,0215	8	600	7	0	15
09.07.08 09:59:40	0,3905	7300	0,0187	4	324	3	0	7
07.07.08 11:55:11	0,3813	7298	0,0223	2	717	3	0	5
08.07.08 09:55:19	0,3326	7298	0,0149	4	56	113	0	117
08.07.08 11:07:08	0,2910	7300	0,0147	3	38	97	0	100
08.07.08 09:50:56	0,1564	7298	0,0149	5	63	1	0	6

Längste Ausführungen anzeigen

Hier können Sie u. a. die benötigte CPU-Zeit, die Anzahl synchroner und asynchroner Lesezugriffe u. v. m vergleichen. Seit Release V6R1 hat sich das eben gezeigte Kontextmenü ein wenig verändert:



Kontextmenü V6R1

Erwähnenswert in diesem Zusammenhang ist insbesondere der Punkt „Planziel ändern“ und „Löschen“. Die Option „Löschen“ ermöglicht es jetzt, gezielt einzelne Ausführungspläne aus dem Plan Cache zu entfernen. Mit der Funktion „Planziel ändern“ haben Sie Einfluss auf die automatische Bereinigung des Planungscache. Je höher der hier hinterlegte Wert ist, desto unempfindlicher sind die Pläne bei der automatischen Bereinigung.

Die Analysemöglichkeiten über den Plan Cache erleichtern die Untersuchung einzelner SQL-Abfragen ungemein. Zuvor mussten Sie einerseits das SQL Statement kennen, das Probleme bereitet, andererseits mussten Sie das Statement ausführen, um es zu analysieren. Beides entfällt seitdem Sie über den Plan Cache auf die Ausführungspläne zugreifen können. Abzuwarten bleibt die Entwicklung in diesem Bereich. Derzeit ist die SQE leider noch nicht in der Lage, die Informationen im Plan Cache mit der realen Ausführung zu vergleichen. Dies könnte eine Vision sein: ein selbstlernender SQE Query Optimizer. Dieser wäre dann in der Lage festzustellen, dass die tatsächliche Ausführungszeit länger als erwartet ist. Er würde daraufhin selbstständig versuchen, eine bessere Implementierungsstrategie zu finden.

Sie sehen, es hat sich viel getan seit Release V5R2. Viele neue Features sind hinzugekommen, die eigenständige Aufgaben im Gesamtoptimierungsprozess übernehmen. Von den neuen Komponenten profitiert allerdings nur die SQE. Sie verfügt aufgrund der Weiterentwicklung über wesentlich mehr Möglichkeiten im Optimierungsprozess und kann auch bei der Ausführung der SQL Statements sehr viel flexibler reagieren. Aber auch die CQE ist weiterhin vorhanden. Sie wird allerdings nicht weiterentwickelt, sondern verbleibt lediglich aus Kompatibilitätsgründen auf dem System. Von den technologischen Entwicklungen profitieren daher hauptsächlich SQL Statements, die von der SQE ausgeführt werden.

SQL-Indizes

Was ist ein SQL-Index? Ein Index ist eine Sammlung von Pointern, die auf die Sätze einer Tabelle zeigen. Sobald Sie den Index erstellen, erzeugt der Datenbankmanager die Struktur und verwaltet ihn als eigenständiges Objekt. Die Pflege der Indizes kostet Systemressourcen, denn schließlich muss jeder Satz, den Sie in einer physischen Tabelle einfügen, auch in den abhängigen Indizes eingefügt werden. Auch wenn Sie Sätze löschen oder ändern müssen, muss das System die Indizes entsprechend modifizieren. Aber dies kennen Sie ja bereits von den logischen DDS-Dateien. In einem Punkt unterscheidet sich der SQL-Index allerdings deutlich von den bekannten DDS-Dateien. Den SQL-Index können Sie nicht direkt in einer SQL-Abfrage nutzen! Eine logische DDS-Datei kann problemlos in der FROM-Klausel des SELECT Statements verwendet werden. Nutzen Sie allerdings logische DDS-Dateien in den SQL-Abfragen, dann wird Ihr SQL Statement immer von der CQE ausgeführt werden; Sie profitieren **nicht** von neuen Technologien!

Wofür brauchen wir dann überhaupt Indizes? Der Optimizer erhält während des Optimierungsprozesses Informationen über die Tabelle vom Statistik-Manager. Mit Hilfe der gewonnenen Informationen wird der Optimizer jetzt die Kosten für unterschiedliche Ausführungspläne abschätzen. Einerseits werden hierbei die Kosten für die Plattenzugriffe (I/O Kosten) geschätzt, andererseits die CPU-Kosten berücksichtigt, die natürlich auch nur eine Schätzung sind. Der anschließende Datenzugriff kann realisiert werden, indem die Tabelle untersucht und gelesen wird – oder indem ein Index verwendet wird, um die gesuchten Sätze zu identifizieren. Es könnten aber auch temporäre Objekte aufgebaut werden, um die benötigten Sätze zu selektieren. Es stehen viele Möglichkeiten für die Ausführung der SQL-Abfrage zur Verfügung. Ob ein Index also letztendlich für die Ausführung Ihrer SQL-Abfrage verwendet wird, ist nur vom Optimizer bzw. den geschätzten Implementierungskosten abhängig. Wenn Sie aber keine geeigneten Indizes bereitstellen, unterbinden Sie – ohne es zu wollen – von vornherein mögliche Ausführungsstrategien. Sie sehen, es ist wichtig die geeigneten Indizes zur Verfügung zu stellen. Sie liefern damit das Werkzeug für die Ausführung. Fehlt das geeignete Werkzeug, könnte die Ausführung der Arbeit gegebenenfalls mangelhaft ausfallen. Andererseits können Sie nicht jeden erdenklichen Index erstellen, denn schließlich müssen Sie auch die Systemressourcen im Auge behalten und sollten die Datenbank nicht unnötig belasten. Eine sinnvolle Indexstrategie muss also die Balance finden zwischen Systembelastung und SQL Tuning.

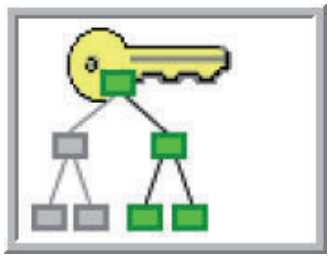
3.4.2
Seite 2

Die Entscheidung zu treffen, welche Indizes erstellt werden müssen, fällt häufig schwer. Zu allem Überfluss gibt es aber nicht nur einen SQL-Index, sondern man unterscheidet zwei grundlegende Indextypen: Binary Radix Tree Index und Encoded Vector Index. Es gibt auch nicht nur einen Befehl, um Indizes zu erstellen. Sie können mittels SQL mit der Anweisung „CREATE INDEX", aber auch mit dem i5/OS-Befehl „Create logical File" (CRTLF) Indizes erzeugen. Aber auch wenn Sie referentielle Integritätsbedingungen definieren, erzeugen Sie Indizes. Bevor wir uns also mit der Frage beschäftigen, welche Indizes nützlich für die SQL-Abfragen sind, sollten wir uns zunächst mit den Unterschieden der einzelnen Indextypen vertraut machen.

Der Binary Radix Tree Index

In den meisten Datenbanken ist der Standardindex ein Binary Radix Tree Index (Binärbaum). Die DB2 for i5/OS nutzt seit 1988 die Binary Radix Tree Indizes. Hierbei handelt es sich um einen binären Suchbaum. Der binäre Suchbaum ist eine spezielle Baumstruktur, bei der jeder Knoten höchstens zwei Unterbäume – einen linken und einen rechten – besitzt. Ein binärer Suchbaum ist eigentlich nichts anderes als ein sortierter Binärbaum. Er hat eine Wurzel sowie innere und äußere Knoten. Die Wurzel ist der Grundknoten des Baumes, da sie keinen Vorgänger besitzt. Ist dieser Knoten leer, so ist der gesamte Baum leer. Äußere Knoten nennt man jene, die keinen Nachfolger haben. Sie enthalten die Adresse des physischen Satzes aus der zugrunde liegenden Tabelle. Diese Knoten befinden sich zumeist auf der untersten Ebene eines Baums. Alle anderen Knoten, denen ein Suchbegriff zugewiesen ist, heißen innere Knoten. Jeder innere Knoten hat genau zwei Nachfolger und enthält einen Teil des Schlüsselwerts. Außerdem gilt, dass alle Knoten im linken Teilbaum kleiner und alle im rechten Teilbaum größer sind. In dieser Baumstruktur können große Mengen unterschiedlicher Schlüsselwerte effizient gespeichert werden; beim Sortieren und Selektieren von Daten kann sich ein Binärindex als ausgesprochen nützlich erweisen.

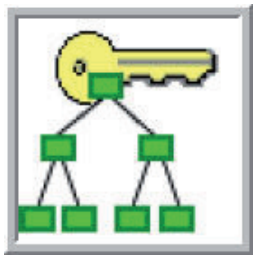
Die SQE kann die Binärindizes für den Datenzugriff unterschiedlich nutzen:



Index Radix Probe

Wenn der Optimizer die Methode „Radix Index Probe“ verwendet, dann stehen die benötigten Sätze im Mittelpunkt des Zugriffes. Sie können mit wenigen Zugriffen auf das Indexobjekt geprüft und selektiert werden. Vor dem Lesen des jeweiligen Schlüsseleintrags erfolgt in diesem Fall die Überprüfung der Selektionskriterien. Dies reduziert deutlich die Anzahl der zu verarbeiteten Datensätze, denn nur Sätze, deren Spaltenwerte den lokalen Selektionskriterien entsprechen, werden in den Arbeitsspeicher geladen und können in einem zweiten Schritt gezielt anhand der relativen Satznummern aus der Tabelle abgerufen werden. Die Sätze stehen in Schlüsselfolge sortiert zur Verfügung.

3.4.2.1
Seite 2



Index Radix Scan

Nutzt der Optimizer hingegen die Methode „Index Radix Scan“, entfällt die vorzeitige Prüfung der Selektionskriterien. Alle Einträge des Index werden in den Arbeitsspeicher geladen. Dies ist häufig der Fall, wenn die lokalen Selektionskriterien nicht den Schlüsselwerten des Index entsprechen. Dennoch bringt auch diese Methode Vorteile, da die geladene Seite eines Index wesentlich mehr Sätze abbildet als die vergleichbare Seite eines Table Scans; die I/O-Kosten reduzieren sich deutlich. Außerdem stehen die Daten auch hier anschließend sortiert zur Verfügung.

Tatsächlich haben Sie schon häufig Binärindizes erstellt, ohne darüber nachzudenken, denn auch unsere mit DDS erstellten logischen Dateien sind binäre Suchbäume. Hier ein Beispiel:

Logische DDS-Datei „D_KdOrt“

Das Objekt wird anschließend mit dem Befehl „Create Logical File“ (CRTLF) erstellt. Wenn ich mit SQL einen Index erstellen möchte, dann nutze ich dafür das SQL Statement „CREATE INDEX“. Das Statement ist einfach zu handhaben, denn es enthält nur wenig unterschiedliche Klauseln:

```
CREATE INDEX I_KdOrt ON KdSt00 (KDORT);
Label on Index I_KdOrt is ‚Schlüssel: Wohnort des
Kunden NICHT eindeutig‘;
```

Mit dem zweiten SQL Statement hinterlege ich noch einen beschreibenden Text – vergleichbar mit dem DDS-Schlüsselwort „TEXT“. Der eben erstellte Index ist kein UNIQUE-Index. Doch auch eindeutige Indizes lassen sich einfach erstellen:

```
CREATE UNIQUE WHERE NOT NULL INDEX I_KdKdnr ON KdSt00(KdKdnr);
LABEL ON INDEX I_KdKdnr IS ‚Schlüssel: Kundennummer eindeutig‘;
```

Die Klausel „WHERE NOT NULL“ schließt NULL-Werte für die Spalte „KDKDNR“ aus. Falls einige Ihrer SQL Statements noch auf der CQE ausgeführt werden, sollten Sie darauf achten, dass die Schlüsselspalten Ihrer Indizes keine NULL-Werte enthalten. Die CQE kann die Indizes ansonsten nicht in den Abfragen verwenden. Diese Einschränkung entfällt bei der SQE.

Den physischen Dateien des System i Servers können Sie mit Hilfe einer DDS-Beschreibung eine Eingangsreihenfolge zuweisen. Mit SQL gibt es diese Möglichkeit nicht. Sie können eine Datei nicht als Datei mit Schlüsselfolge deklarieren. Eine SQL-Tabelle kann jedoch als Tabelle mit UNIQUE- oder Primärschlüsselbedingungen deklariert werden. Der Primär- oder UNIQUE-Schlüssel besteht aus einer oder mehrerer Spalten, die eine einzelne Zeile in der SQL-Tabelle eindeutig definieren. Schlüsselbedingungen können während der Erstellung der Tabelle deklariert werden oder später mit Hilfe der Anweisung ALTER TABLE hinzugefügt werden.

Schauen Sie sich die nachfolgenden Beispiele an:

```
CREATE TABLE FIRMA (
FIKURZ  INTEGER                 NOT NULL UNIQUE,
PEKURZ  INTEGER                 NOT NULL DEFAULT 0
…
```

Die Anweisung erstellt die Tabelle „FIRMA“. Die Tabelle besitzt einen eindeutigen Zugriffspfad, der aus der Spalte „FiKURZ“ besteht. Das bedeutet, dass keine zwei Einträge in der Tabelle einen identischen „FiKURZ“-Wert haben können. Das gleiche Ergebnis erziele ich aber auch mit folgender Definition:

```
CREATE TABLE ITPSQL.FIRMA (
FIKURZ INTEGER                          PRIMARY KEY,
PEKURZ INTEGER                          NOT NULL DEFAULT 0 …
```

Technisch gesehen ist ein Primärschlüssel (Primary Key) also einfach eine Kombination aus UNIQUE-Bedingung und NOT-NULL-Bedingung.

Eine Fremdschlüsselbedingung (Foreign Key) gibt an, dass die Werte in einer Spalte oder einer Gruppe von Spalten mit den Werten einer anderen Tabelle übereinstimmen müssen. Man sagt, dass dadurch die referentielle Datenintegrität zwischen zwei zusammenhängenden Tabellen gesichert wird.

```
CREATE TABLE….
:
:
CONSTRAINT ITPCR.FK_HIKDNR_KDKDNR
FOREIGN KEY (HiKdnr)
REFERENCES ITPCR.KDST00 (KdKdnr)
);
```

3.4.2.1
Seite 4

Weiterführende Informationen zum Thema „Integritätsbedingungen“ enthält das Kapitel 3.5.2. Für uns ist im Moment nur interessant, dass in den gezeigten Beispielen immer Binärindizes erstellt werden, die die SQL Engine für den Datenzugriff verwenden kann.

Fassen wir zusammen: Sie erstellen Binary Radix Tree Indizes, wenn Sie:

– physische DDS-Objekte erstellen, die Schlüsseldefinitionen enthalten,

– logische DDS-Dateien erstellen, die Schlüsselbedingungen enthalten,

– Objekte erstellen, die mit dem SQL-Befehl „CREATE INDEX“ erstellt wurden,

– Primär- oder Fremdschlüsselbedingungen definieren.

Alle Objekte kann der SQL Optimizer im Optimierungsprozess und für die Ausführung der SQL-Abfrage nutzen.

SQL-Index oder logische DDS-Datei?

Sie haben gesehen, dass es möglich ist, einerseits Indizes mit dem SQL-Befehl „CREATE INDEX“ zu erstellen andererseits werden aber auch logische DDS-Dateien als Index vom SQL Optimizer berücksichtigt. Da taucht schnell die Frage nach den Unterschieden auf. Am einfachsten ist es, sich die Unterschiede an einem konkreten Beispiel zu verdeutlichen. Erinnern Sie sich noch an die Erstellung der logischen DDS-Datei „D_KDORT“? Sie wurde mit dem i5/OS-Befehl „Create logical File“ aufgrund einer DDS-Quelle erstellt. Außerdem hatte ich den SQL-Index „I_KDORT“ erstellt:

```
CREATE INDEX I_KdOrt ON KdSt00 (KDORT);
LABEL ON INDEX I_KdOrt IS ‚Schlüssel:
Wohnort des Kunden NICHT eindeutig‘;
```

Um die Unterschiede zwischen beiden Objekten zu ermitteln, gebe ich jetzt auf einer 5250-Befehlszeile den nachstehenden Befehl ein:

```
WRKF FILE(ITPCR/*ALL) FILEATR(LF)
```

Daraufhin erscheint folgende Anzeige:

```
D_KDORT      ITPCR        LF
I_KDKDNR     ITPCR        LF
I_KDORT      ITPCR        LF
```

Ergebnis des Befehls WRKF

Auf den ersten Blick lassen sich keine Unterschiede zwischen den SQL-Indizes und der logischen DDS-Datei feststellen. Tatsächlich können Sie die Objekte unabhängig von Ihrem Erstellungsbefehl identisch administrieren – und auch in Ihren HLL-Programmen wie gewohnt einsetzen. Dennoch gibt es Unterschiede. Schauen Sie sich die Dateiattribute an. Nutzen Sie hierfür einfach – wie gewohnt – den i5/OS-Befehl „Display File Describtion“ (DSPFD):

```
Datenbankdateiattribute
  Extern beschriebene Datei . . . . . :          Ja
  SQL-Dateiart  . . . . . :          INDEX
  Zugriffspfad, logische Seitengröße  . . . . : PAGESIZE      64
  Maximale Schlüssellänge . . . . . :          4
                  Zum Vergleich:
                  Mit DDS erstellte logische Dateien haben 8K Seiten (V5R4)
```

Dateiattribute eines SQL-Index

3.4.2.2
Seite 2

Der Parameter „PAGESIZE“ bestimmt die logische Seitengröße des Zugriffspfads dieser Datei. Er beträgt für logische DDS-Dateien in der Regel 8 KB. Ein SQL-Index hat hingegen häufig 64 KB große Seiten. Die Seitengröße des Zugriffspfades bestimmt letztendlich die Anzahl Bytes, die aus dem Zusatzspeicher in den Speicher-Pool des Jobs für eine Fehlseite verschoben werden. Diesen Parameter finden Sie im Befehl „Create Logical File“(CRTLF) in den zusätzlichen Parametern. Der Unterlassungswert *KEYLEN sorgt dafür, dass die Seitengröße anhand der Schlüssellänge ermittelt wird. Die logische Seitengröße für den Zugriffspfad eines SQL-Index definieren Sie im Statement „CREATE INDEX“ mit der Klausel „PAGESIZE“:

```
CREATE UNIQUE WHERE NOT NULL INDEX I_CusKdnr
ON KdSt00(Customer_Number) PAGESIZE 64;
```

Es steht der Wertebereich 64K bis 512K zur Verfügung. Der Unterlassungswert für einen SQL-Index beträgt 64 KB und ändert sich auch nicht im Release V6R1.

Aber es gibt einen weiteren wichtigen Unterschied, den Sie kennen sollten, bevor Sie beginnen, SQL-Indizes zu erstellen. SQL-Indizes können sich die Zugriffspfade in der Regel nicht mit anderen Binärindizes teilen. Schauen Sie sich dazu folgende Beispiele an:

Ich erstelle zunächst die logische DDS-Datei „D_MED_KDNR“ mit den Schlüsselfeldern „HIMED“ und „HIKDNR“. Anschließend wird eine weitere logische DDS-Datei erzeugt, die nur das Schlüsselfeld „HIMED“ enthält. Schauen wir uns die Dateibeschreibung der logischen Datei „D_MED“ genauer an:

Zugriffspfadnutzung der logischen DDS-Datei „D_MED“

Sie sehen, die Datei „D_MED“ nutzt den bestehenden Zugriffspfad der Datei „D_MED_KDNR“. Es entsteht somit auch kein System-Overhead, da weiterhin nur ein Zugriffspfad vom Datenbankmanager gepflegt werden muss.

Das gleiche Beispiel führe ich jetzt mit SQL aus:

```
CREATE INDEX I_MED_KD ON HIST00(Himed, HiKdnr);
CREATE INDEX I_MED ON HIST00(Himed);
```

Schauen wir uns auch hier die Eigenschaften der erstellten Objekte an. Als Erstes die Datei „I_MED_KD“:

```
Impl. gemeinsame Ben. des Zugriffspfads :          Nein      Datei I_MED_KD
Zugriffspfad aufgezeichnet  . . . . . :            Nein
Anzahl eindeutiger Teilschlüsselwerte . :
  Schlüsselfeld 1 . . . . . :                                        6
  Schlüsselfelder 1 - 2 . . . . . :                               8273
```

Zugriffspfadnutzung SQL-Index „I_MED_KD“

Obwohl bereits aufgrund der logischen DDS-Datei „D_MED_KDNR“ ein Zugriffspfad mit diesen Schlüsselfeldern existiert, wird ein eigenständiger Zugriffspfad für den SQL-Index aufgebaut. Aufgrund der unterschiedlichen logischen Seitengröße können die erstellten SQL-Indizes nicht die Zugriffspfade der bestehenden logischen DDS-Dateien nutzen. Und wie sieht es mit dem zweiten SQL-Index aus?

```
Impl. gemeinsame Ben. des Zugriffspfads :          Nein      Datei I_MED
Zugriffspfad aufgezeichnet  . . . . . :            Nein
Anzahl eindeutiger Teilschlüsselwerte . :
  Schlüsselfeld 1 . . . . . :                                        6
```

Zugriffspfadnutzung SQL-Index „I_MED“

Auch für diesen SQL-Index wird ein eigenständiger Zugriffspfad erstellt. Anders als logische DDS-Dateien nutzt der SQL-Index nur bestehende Zugriffspfade, wenn sie zu 100 Prozent deckungsgleich sind. Wir haben also zwei SQL-Zugriffspfade, die beide gepflegt werden müssen, und zusätzlich den Zugriffspfad der logischen DDS-Dateien. Ich denke, das Beispiel zeigt deutlich, wie schnell Sie Ihre Datenbank belasten können und unnötigen Overhead produzieren können. Es existieren jetzt drei Zugriffspfade auf dem System, obwohl tatsächlich nur einer notwendig wäre: Ausreichend ist es, den SQL-Index „I_MED_KD“ mit den Schlüsselfeldern „HIMED“ und „HIKDNR“ zu erstellen.

3.4.2.2
Seite 4

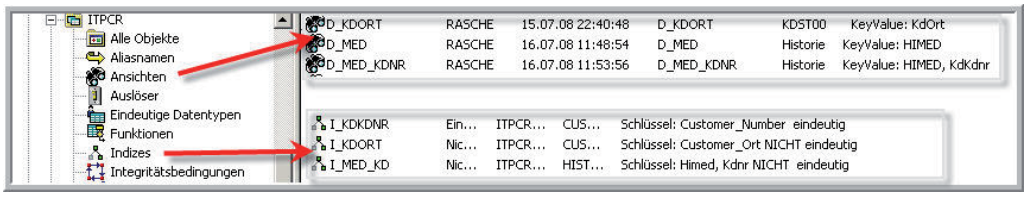
Wenn es aber – aus welchen Gründen auch immer – erforderlich sein sollte, weiterhin mit logischen DDS-Dateien zu arbeiten, sollten Sie die logischen DDS-Dateien trotz alledem löschen und neu erstellen. Passen Sie auf, was passiert, wenn ich die logische DDS-Datei „D_MED“ neu umwandle und mir anschließend die Eigenschaften des neu erstellten Objekts anschaue:

```
Impl. gemeinsame Ben. des Zugriffspfads :          Ja        Datei D_MED
  Zugriffspfad aufgezeichnet  . . . . . :          Nein
Anzahl eindeutiger Teilschlüsselwerte . :
  Schlüsselfeld 1 . . . . . :                                        6
  Schlüsselfelder 1 - 2 . . . . . :                               8273
Datei, der der Zugriffspfad gehört  . . :          ITPCR/I_MED_KD
Teildatei . . . . . :                              I_MED_KD
```

Dateiattribute der logischen DDS-Datei „D_MED“

Sie sehen, logische DDS-Dateien können die Zugriffspfade bestehender SQL-Indizes nutzen.

Erschwerend kommt leider hinzu, dass es bislang keine gute Darstellung aller logischen Dateien gibt. Auch System i Navigator ist hier nicht besonders hilfreich! Öffnen Sie Ihre Datenbank und das Schema, das die Indizes enthält. Klicken Sie auf den Eintrag „INDEX“. Es werden nur die SQL-Indizes angezeigt!



System i Navigator V5R4 DDS-Dateien und Indizes

Unsere logischen DDS-Dateien finden wir im Eintrag „Ansichten“. SQL versteht logische DDS-Dateien in erster Linie als SQL View – daher die Darstellung in der Rubrik „Ansichten“.

Warum? Logische DDS-Dateien können eingeschränkte Satzformate darstellen, Selektionskriterien enthalten, Spalten umbenennen u. v. m. Das sind Funktionen, die ein SQL-Index nicht abbilden kann. Bis Release V5R4 bietet System i Navigator keine Möglichkeit sich die gemeinsame Zugriffspfadnutzung anzuzeigen. Hier bleibt momentan nur die Rückkehr zur 5250-Befehlszeile. Etwas besser wird die Darstellung im Release V6R1 für SQL-Indizes:

SQL-Name	Schlüsselspalten	Typ des Eignerindex	Name des Eignerindex
I_CUSKDNR	KDKDNR (Auf)	Integritätsbedingung über Prim...	P_KDKDNR
I_CUSORT	KDORT (Auf)	Index	I_KDORT
I_HIANL	HIANL (Auf)	Index	I_HIANL
I_KDKDNR	KDKDNR (Auf)	Integritätsbedingung über Prim...	P_KDKDNR
I_KDORT	KDORT (Auf)	Index	I_KDORT
I_LOWER_HIANL	LOWER (HIANL) (Auf)	Index	I_LOWER_HIANL
I_MED_KD	HIMED (Auf), HIKDN...	Index	D_MED_KDNR
I_PLUS_BSTQTY...	PRDQTY + PRDBSTQ...	Index	I_PLUS_BSTQTY_QTY
S_KDORT	KDORT (Auf)	Index	S_KDORT

V6R1

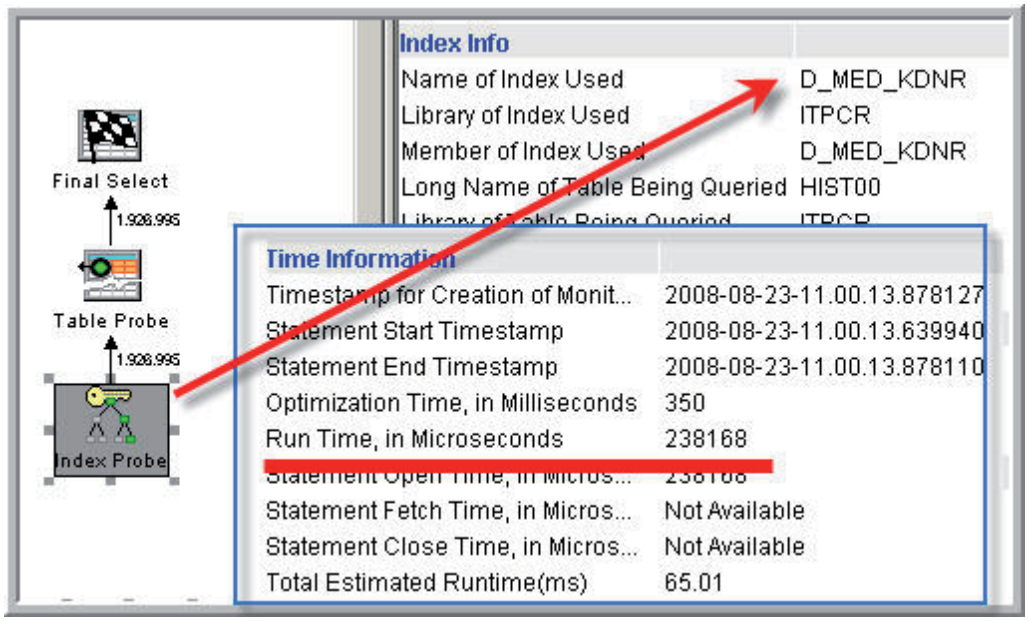
System i Navigator-Darstellung gemeinsam genutzter Zugriffspfade

System i Navigator stellt für SQL-Indizes ab V6R1 die gemeinsam genutzten Zugriffspfade dar. Der Zugriffspfad, der genutzt wird, heißt Eignerindex.

Aufgrund der Darstellung im System i Navigator könnte man auch vermuten, dass der SQL Optimizer bestehende Zugriffspfade logischer DDS-Dateien für die SQL-Abfragen nicht nutzt. So ist es allerdings nicht: Sowohl in der Optimierungsphase zur Informationsbeschaffung als auch zur Ausführungszeit für den Datenzugriff können logische DDS-Dateien verwendet werden. Schauen wir uns auch hierzu ein einfaches Beispiel an. Zunächst lösche ich aber alle SQL-Indizes und führe anschließend folgendes SQL Statement aus:

```
SELECT *
FROM Hist00
WHERE Himed = 'E-Mail' OR Himed = 'FAX';
```

Den Ausführungsplan schaue ich mir anschließend mit Visual Explain an:

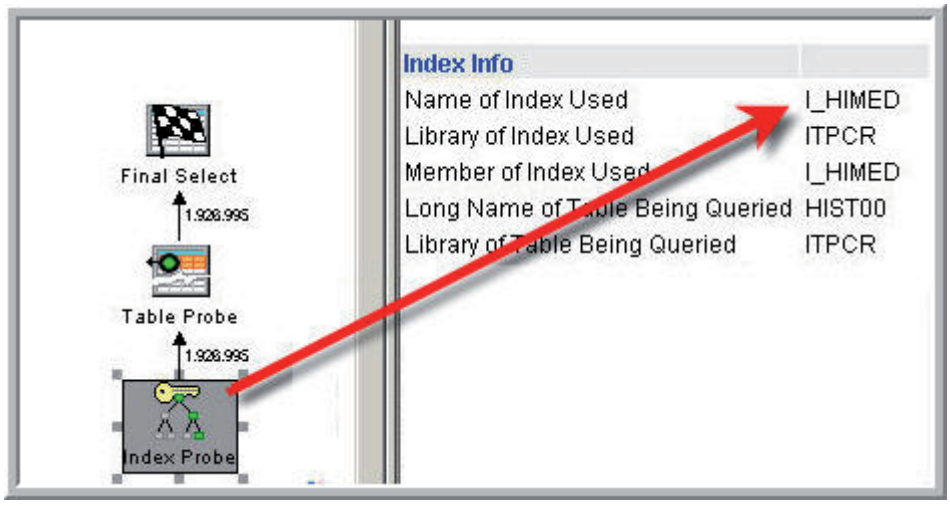


SQL-Abfrage mit logischer DDS-Datei

3.4.2.2
Seite 6

Der Optimizer nutzt für die Ausführung der Abfrage einen Zugriff über Index. Verwendet wird die logische DDS-Datei „D_Med_Kdnr“. Anhand der logischen DDS-Datei kann die Adresse der benötigten Sätze ermittelt werden. Der Datenabruf erfolgt anschließend wahlfrei.

Was passiert, wenn ich alternativ wieder den SQL-Index „I_HIMED“ erstelle? Für welches Objekt wird sich der Optimizer entscheiden?



Ausführung mit SQL-Index

Der Optimizer entscheidet sich tatsächlich für den SQL-Index und gegen unsere logische DDS-Datei. Der Grund hierfür ist die logische Seitengröße. Nur wenn Sie gezielt einzelne oder wenige Sätze benötigen, sind kleine logische Seiten vorteilhaft. Wenn aber Schlüsselwerte kombiniert, gruppiert, große Datenmengen abgerufen werden oder der Zugriffspfad seitenweise gelesen wird, sind größere logische Seiten vorteilhafter.

Was ist die Konsequenz? Mittelfristig sollten Sie versuchen, auf logische DDS-Dateien zu verzichten! Ersetzen Sie die DDS-Objekte – wie gezeigt – durch vergleichbare Indizes oder erstellen Sie die Objekte neu. Vielleicht lässt es sich nicht vermeiden, dass Sie Ihre HLL-Programme neu kompilieren müssen, aber wir werden nicht dauerhaft die alten und neuen Techniken parallel nutzen können. Wir müssen versuchen, alte Programme und neue Datenbanktechniken zu integrieren. Glauben Sie nicht, dass Sie alle SQL-Indizes zusätzlich erstellen können. „Ein paar Zugriffspfade mehr oder weniger können doch nicht stören!“ Ein Irrtum! Das Problem der Zugriffspfadwartung wird häufig unterschätzt.

Der Encoded Vector Index (EVI)

Um die Verwirrung zu vervollständigen, hat IBM mit dem Encoded Vector Index (EVI) einen weiteren, effizienten Indextypen geschaffen, der aufgrund seiner geringen Größe und des einfachen Aufbaus gute Möglichkeiten bietet, effizient große Datenbanktabellen abzufragen. Für dieses Objekt wird kein binärer Suchbaum erstellt, sondern es wird das Konzept der Bitmapindizes zugrunde gelegt. Es macht demzufolge Sinn, sich zunächst dieses Konzept anzuschauen. Ausgangspunkt ist die Tabelle „KDST00“:

RSN	CUSTOMER_NUMBER	PLZ	CUSTOMER_ORT
10	1001200	...	Düsseldorf
11	1001500	...	Düsseldorf
13	1001700	...	Düsseldorf
8	2008001	...	Leverkusen
9	2093001	...	Leverkusen
12	1001600	...	Leverkusen
14	1001900	...	Tübingen

Ausgangssituation

Wenn Sie einen Bitmapindex für die Spalte „Customer_Ort“ erstellen würden, sähe das Ergebnis für die dargestellte Ausgangssituation folgendermaßen aus:

Bitmap	Wert
0001 0110 0000 0000	Düsseldorf
0000 1001 1000 0000	Leverkusen
0010 0000 0000 0000	Tübingen

Die erste Zeile des Bitmap repräsentiert das Vorkommen des Werts „Düsseldorf“ in der Tabelle „KDST00“. Das erste Bit dieser Zeile ist „0“. Der Wert „0“ bedeutet in diesem Zusammenhang, dass der Wert „Düsseldorf“ nicht im ersten Satz der Tabelle vorhanden ist. Das 10. Bit hat dagegen den Wert „1“. Der 10. Satz der Tabelle enthält den Wert „Düsseldorf“. Wenn Sie in einer SQL-Abfrage alle Kunden aus Düsseldorf benötigen, dann kennt die SQL Engine aufgrund des Bitmap die relativen Satznummern des gesuchten Werts.

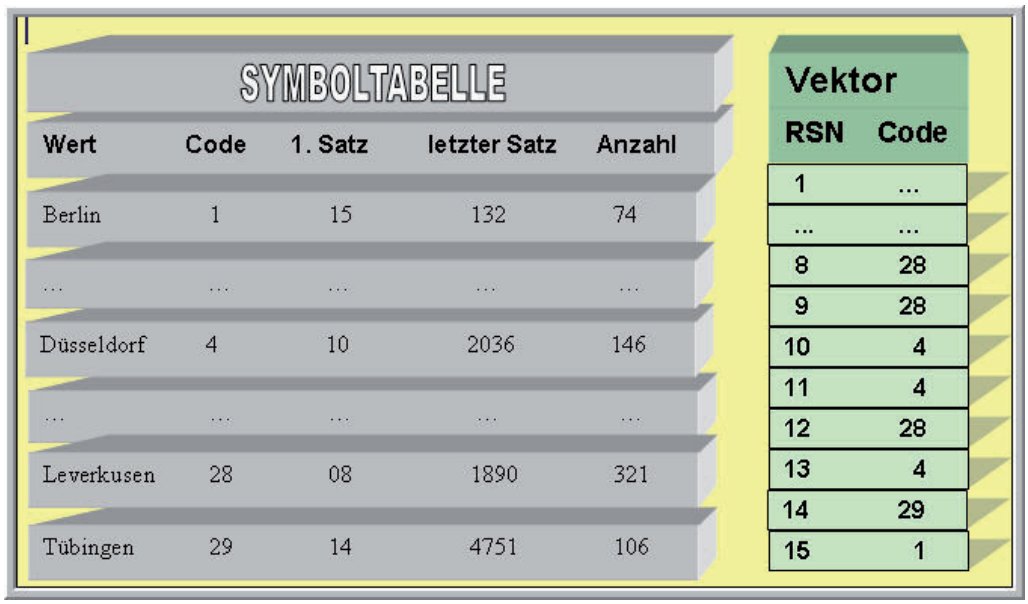
Außerdem lassen sich die Bitmaps durch logisches AND oder OR einfach kombinieren. Stellen Sie sich vor, Sie suchen die Kunden aus „Düsseldorf“ oder aus „Leverkusen“:

Bitmap	Wert
0001 0110 0000 0000	Düsseldorf
0000 1001 1000 0000	Leverkusen
0001 1111 1000 0000	Ergebnis

Es werden die Sätze 8, 9, 10, 11, 12 und 13 abgerufen.

3.4.2.3
Seite 2

Das Problem mit den ursprünglichen Bitmap-Strukturen ist nur: Sie benötigen zu viel Platz. Nehmen wir an, die Tabelle verfügt über 20.000 Datensätze mit 50 unterschiedlichen Ortsangaben. Sie würden 20.000 * 50 – also 1.000.000 Bits – für den Bitmapindex benötigen. Dies entspräche rund 122 KB. Aus diesem Grund hat IBM die Bitmap-Technologie weiterentwickelt. Die weiterentwickelte Index-Technologie besteht aus einer Symboltabelle und einem Vektor. Beide Komponenten werden gemeinsam als Datenstruktur gespeichert. In der Symboltabelle wird zunächst jedem eindeutigen Schlüsselwert ein binärer Code zugeordnet. So kann jeder Schlüsselwert unabhängig von Typ und Länge abgebildet werden. Zusätzlich werden in der Symboltabelle die erste und letzte relative Satznummer und die Anzahl der Vorkommen (=Kardinalität) hinterlegt. Im Vektor ist wiederum jeder relativen Satznummer der zugehörige Binärcode entsprechend der Symboltabelle zugeordnet. Die nachfolgende Grafik verdeutlicht diese Strukturen:



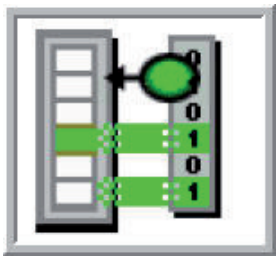
Encoded Vector Index

Die benötigte Byteanzahl zur Kodierung der vorhandenen Schlüsselwerte steht hierbei in direktem Bezug zur Anzahl der unterschiedlichen Vorkommen. Für die Codierung von bis zu 255 verschiedenen Vorkommen wird lediglich ein Byte benötigt. Wenn 256 bis 65535 Werte kodiert werden müssen, sind zwei Byte notwendig. Falls die Spalte mehr als 65536 unterschiedliche Werte enthält, wird eine Vektorgröße von 4 Byte gewählt.

Von	Bis	Vektorgröße
1	255	1
256	65535.	2
65536	4,2 Billionen	4

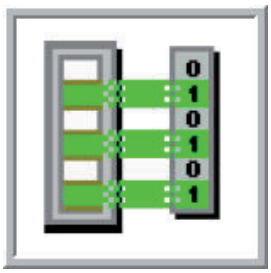
Der SQE Optimizer nutzt die Symboltabelle zur Informationsgewinnung während der Optimierungsphase.

Für die Ausführung der Abfrage kann der EVI anschließend unterschiedlich eingesetzt werden:



EVI Index-Probe

EVIs können für den Tabellenzugriff nicht direkt genutzt werden. Vielmehr müssen mit Hilfe des Vektors zur Laufzeit temporäre Listen oder Bitmaps aufgebaut werden, die für den wahlfreien Tabellenzugriff (Encoded Vector Probe) genutzt werden. Der Vorteil im Gegensatz zum Einsatz eines Binär-Indizes liegt im asynchronen Datenzugriff. I/O-Operationen können auf diese Weise sehr viel effektiver ausgeführt werden, da große Teile der Tabelle übersprungen werden, wenn Sätze nicht den Suchkriterien entsprechen. Auch für JOIN-Abfragen können EVIs mittlerweile in Kombination mit Hash-Tabellen eingesetzt werden.



EVI Table Scan

Entscheidet sich der Optimizer für die Methode „EVI Table Scan“, werden nur die Einträge der Symboltabelle sequentiell gelesen, ohne den Vektor oder die zugrunde liegende Tabelle zu nutzen. Dieses Verfahren kann Abfragen, die die Schlüsselwörter GROUP, DISTINCT, COUNT oder COUNT DISTINCT enthalten, extrem beschleunigen.

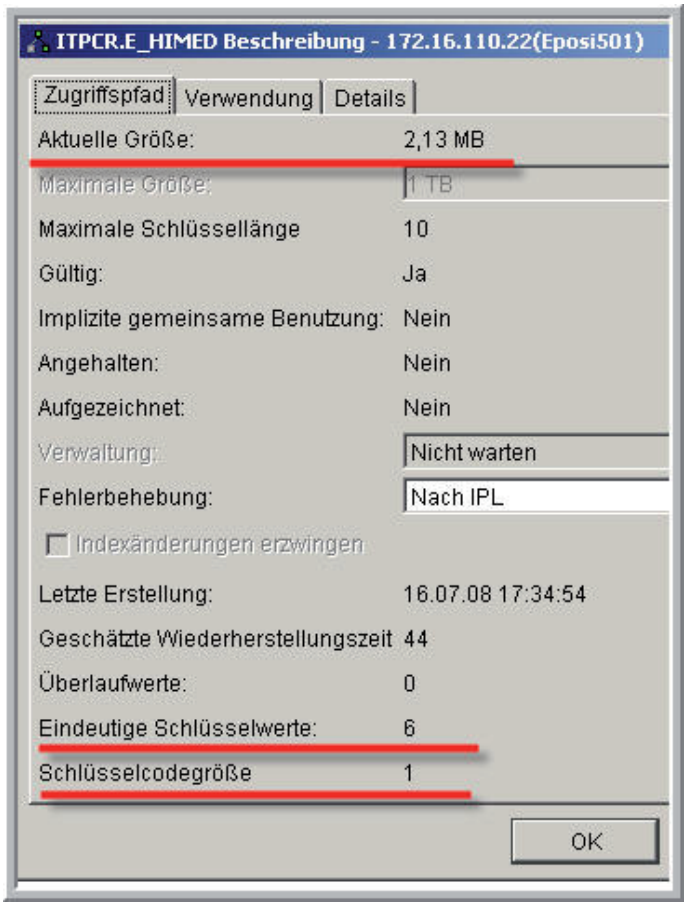
Um EVIs zu erstellen, ist die Klausel „ENCODED VECTOR“ im Statement „CREATE INDEX“ ausreichend.

```
CREATE ENCODED VECTOR INDEX E_Himed
ON HiSt00(Himed) WITH 255 DISTINCT VALUES;
```

3.4.2.3
Seite 4

Der Zusatz „WITH 255 DISTINCT VALUES“ bestimmt die Größe des Vektors. In meinem Beispiel ist ein Byte für die Kodierung der unterschiedlichen Schlüsselwerte ausreichend, da die Tabelle „HIST00“ Werbemedien enthält. Aktuell gibt es sechs verschiedene Medien: FAX, E-Mail, BRF, TEL, MESSE und UNKNOWN. Es ist nicht anzunehmen, dass irgendwann einmal mehr als 255 unterschiedliche Werte abgebildet werden, so dass ein Byte für die Kodierung der Schlüsselwerte ausreichend sein dürfte.

Schauen wir uns anschließend die Eigenschaften des EVIs an. Hierzu öffnen Sie im System i Navigator das Schema und anschließend den Eintrag „Indizes“. Auf der rechten Bildschirmseite finden Sie alle SQL-Indizes. Ich markiere jetzt mit der rechten Maus den erstellten EVI und öffne im Kontextmenü den Dialog „Beschreibung“. Daraufhin erscheint nachfolgende Abbildung:



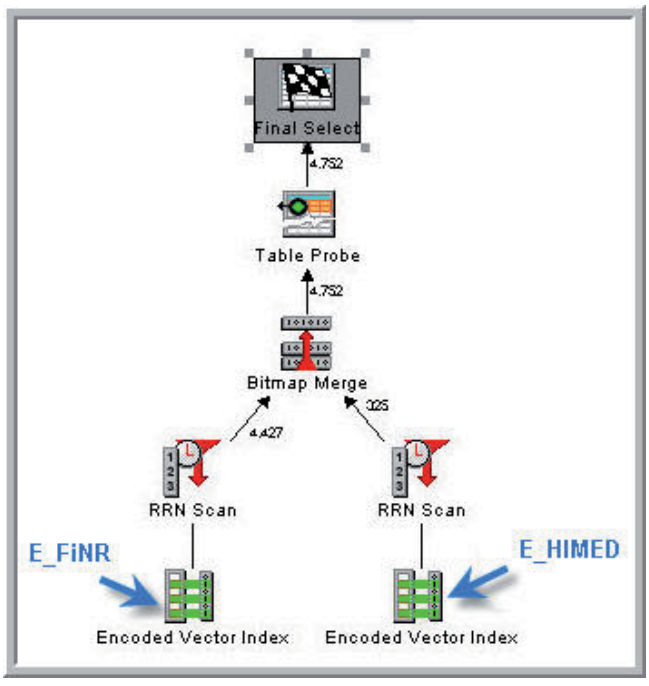
Encoded Vector Index-Eigenschaften

Die Schlüsselcodegröße beträgt aufgrund der Klausel „WITH n DISTINCT VALUES“ wie erwartet 1 Byte. Im Beispiel hätte ich auf die Angabe der WITH-Klausel sogar verzichten können, da die Tabelle „HIST00“ bereits alle Daten enthält und die Vectorgröße während der Indexerstellung automatisch anhand der Datenlage ermittelt wird. Ein Vorteil der EVIs ist die geringe Größe. Der eben erstellte Index ist nur 2,13 MB groß. Zum Vergleich: die aktuelle Größe des entsprechenden Binärindex beträgt 30,26 MB. Encoded Vector Indizes können häufig schneller erstellt werden als Binärindizes. Insbesondere wenn die Anzahl eindeutiger Schlüsselwerte gering ist. Außerdem liefern EVIs aufgrund der Symboltabelle exaktere Statistikdaten als Binärindizes. Sie können im Gegensatz zu Binärindizes über logisch UND bzw. ODER verknüpft werden. Um ein entsprechendes Beispiel auszuführen, erstelle ich zunächst einen zweiten EVI:

```
CREATE ENCODED VECTOR INDEX E_FINR
ON HIST00(HIFINR) WITH 10 DISTINCT VALUES;
```

Anschließend führe ich die folgende SQL-Abfrage aus:

```
SELECT *
FROM HIST00
WHERE HIMED = 'FAX' OR HIFINR = '10';
```



EVI-Logische Verknüpfung

3.4.2.3
Seite 6

Die SQE nutzt beide EVI-Objekte für die Ausführung, indem zunächst für jedes lokale Kriterium ein temporäres Bitmap aufgebaut wird. Die Teilergebnisse werden zu einem Bitmap vereint. Über das neue temporäre Objekt erfolgt der wahlfreie Zugriff auf die Tabelle „HIST00". Aufgrund dieser Eigenschaft sollten Sie Encoded Vector Indizes vorrangig für einzelne Spalten oder zusammengesetzte Spalten erstellen, die häufig in unterschiedlichen Kombinationen in lokalen Selektionskriterien verwendet werden. Achten Sie bei der Spaltenauswahl zusätzlich darauf, dass es sich um Spalten mit einer relativ geringen und konstanten Kardinalität handelt.

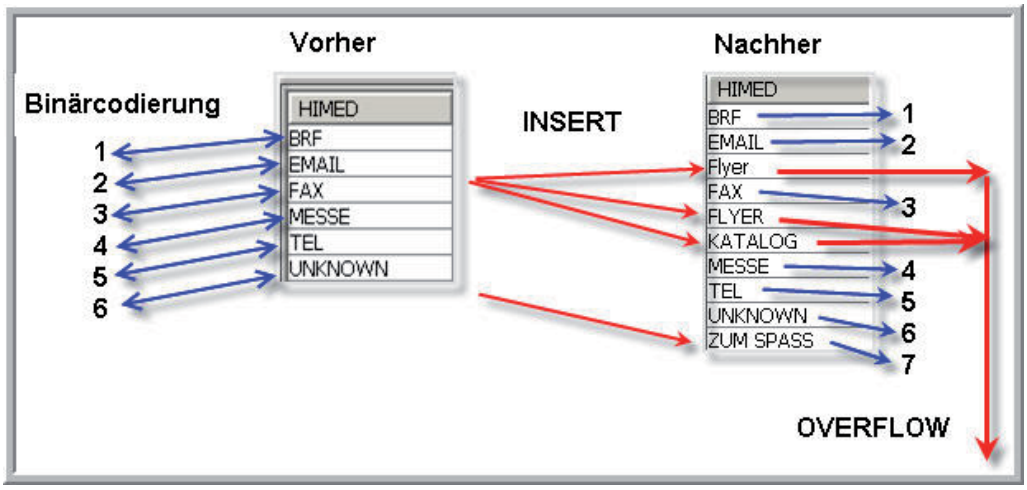
Der Optimizer könnte natürlich auch einen Binärindex für die Ausführung nutzen. Der Ausführungsplan ist letztendlich von der selektierten Datenmenge abhängig. Werden wenig Datensätze (ca. bis zu 20 Prozent) benötigt, ist der Binärindex effektiver als der Einsatz der EVI-Indizes. Sobald aber größere Datenmengen (ca. 20 bis 70 Prozent) abgefragt werden, wird der Optimizer möglicherweise die EVIs verwenden, da durch die asynchronen Lesevorgänge die I/O-Kosten in der Regel deutlich reduziert werden. Die Reduzierung der Plattenzugriffe rechtfertigt häufig die zusätzlichen Kosten für den Aufbau temporärer Objekte.

Encoded Vector Indizes haben aber nicht nur Vorteile. Insbesondere in der Pflege gibt es einige beachtenswerte Eigenheiten. Wenn Sie häufig Inhalte der Schlüsselspalte verändern, neue Zeilen mit neuen Schlüsselwerten hinzufügen oder bestehende Sätze löschen, könnte es sein, dass Sie die Symboltabelle gegebenenfalls von Zeit zu Zeit neu dimensionieren müssen, damit Schlüsselwerte nicht im Overflow-Bereich des Index abgelegt werden. Der Overflow-Bereich wird u. a. genutzt, wenn die gewählte Schlüsselcodegröße für die Darstellung der Schlüsselwerte nicht ausreichend ist. Wenn Ihre Tabelle zum Zeitpunkt der Indexerstellung leer oder nicht vollständig gefüllt ist, dann sollten Sie den Zusatz „WITH n DISTINCT VALUES" keinesfalls vergessen, um eine entsprechende Anzahl Bytes für die Schlüsselcodierung zu reservieren. Falls Sie sich nicht sicher sind, wie hoch die Anzahl eindeutiger Werte sein wird, verwenden Sie einfach den Maximalwert. Bei den Spalten „Werbemedium" und „Filialnummer" habe ich dieses Problem nicht. Wenn ich aber für die Spalte „Kundennummer" einen EVI erstelle, kann ich heute noch nicht sagen, wie sich der Inhalt dieser Spalte entwickelt. Den Index würde ich folgendermaßen erstellen:

```
CREATE ENCODED VECTOR INDEX E_KdNr
ON HiSt00(HiKdnr) WITH 100000 DISTINCT VALUES;
```

Aber der Overflow-Bereich wird auch eingesetzt, wenn nachträglich in unsortierter Reihenfolge Schlüsselwerte eingefügt werden. Die Spalte „Himed“ enthielt – wie erwähnt – zum Zeitpunkt der Indexerstellung die Werte „BRF“, „E-Mail“, „FAX“, „MESSE“, „TEL“ und „UNKNOWN“. In aufsteigender Sortierfolge wurde dem Wert „BRF“ der Binärcode 1, dem Wert „E-Mail“ der Binärcode 2 usw. zugeordnet. Schauen Sie, was passiert, wenn ich jetzt folgende Sätze in die Tabelle „HiSt00“ einfüge:

```
INSERT INTO HiSt00 VALUES(1209600,10, 49, 'Versand 2008/II',
'KATALOG', ' ', 'RASCHE', '2008-07-15', '10:42:12');

INSERT INTO HiSt00 VALUES(1209600,10, 49, 'Sommeraktion',
'FLYER', ' ', 'RASCHE', '2008-07-15', '10:42:12');

INSERT INTO HiSt00 VALUES(1209600,10, 49, 'Sommeraktion',
'Flyer', ' ', 'RASCHE', '2008-07-15', '10:42:12');

INSERT INTO HiSt00 VALUES(1209600,10, 49, ,ohne Angabe',
,ZUM SPASS', , ,, ,RASCHE', ,2008-07-15', ,10:42:12');
```

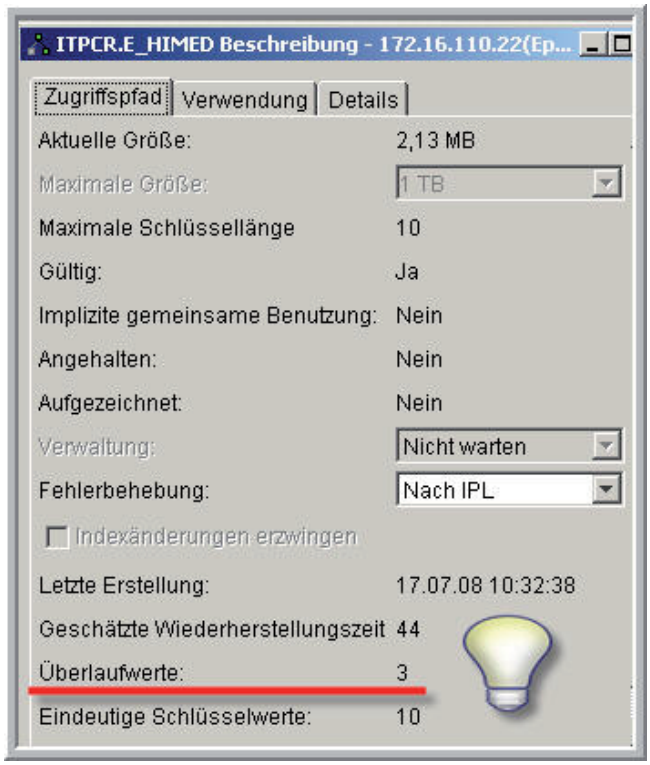


Auswirkung der INSERT-Anweisungen

Die neuen Schlüsselwerte ‚KATALOG’ und ‚FLYER’ müssten innerhalb der Symboltabelle eingefügt werden. Dafür wären aber ein Neuaufbau der Symboltabelle, eine neue Binärkodierung und der anschließende Neuaufbau des Vektors erforderlich. Stattdessen werden diese Werte in den Overflow-Bereich verschoben. Hierbei spielt selbstverständlich auch die Groß- und Kleinschreibung eine entscheidende Rolle. Der neue Wert ‚ZUM SPASS’ kann hingegen an die bestehende Symboltabelle angefügt werden.

3.4.2.3
Seite 8

Je mehr Werte in den Overflow-Bereich verschoben werden, je ineffektiver ist der Einsatz dieses Objekts. Ob im Overflow-Bereich des Index derzeit tatsächlich drei Einträge vorhanden sind, lässt sich schnell feststellen. Klicken Sie in System i Navigator mit der rechten Maus auf den entsprechenden Index und öffnen Sie im Kontextmenü den Eintrag „Beschreibung“:



Overflow-Area prüfen

Alternativ steht Ihnen für die Prüfung der i5/OS-Befehl „Display File Description“ (DSPFD) zur Verfügung.

Die Bereinigung der Overflow-Werte erfolgt, wenn Sie den Index neu erstellen. Sie können auch den Parameter „Force Rebuild Access Path“ (FRCRBDAP) des Befehls „Change Logical File“ (CHGLF) auf „*YES“ setzen. Diese Alternative sollten Sie in Betracht ziehen, wenn die Möglichkeiten und Informationen zur Indexneuerstellung fehlen.

Abschließend eine kurze Zusammenfassung der beschriebenen Eigenheiten:

	Situation	Wirkung
Sehr Effektiv – ohne Zusatz-Pflege	1. Schlüsselwert existiert. INSERT INTO HIST00 ... (... '**FAX**'...);	Statistiken der Symboltabelle werden modifiziert. Vektor erhält neues Element mit neuer RSN und existierenden Schlüsselcode.
	2. Schlüsselwert ist neu. Vektorgröße ist für die Kodierung ausreichend. Wert kann angefügt werden. INSERT INTO HIST00 ... (... '**ZUM SPASS**');	Symboltabelle erhält neue Zeile mit Schlüsselcodeeintrag und Statistiken. Vektor erhält neues Element mit neuer RSN und neuem Schlüsselcode.
	3. Schlüsselwert ist neu. Vektorgröße ist für die Codierung ausreichend. Außerhalb der bestehender Sortierfolge. INSERT INTO HIST00 ... (... 'FLYER'...);	Symboltabelle erhält neue Zeile für Schlüsselcodeeintrag und die zugehörigen Statistiken im Overflow-Bereich. Vektor erhält neues Element mit neuer RSN und neuem Schlüsselcode. **Überlegung:** Neuerstellung des Index, damit die Overflow-Werte integriert werden!
Mit Zusatz-Pflege/ wenig effektiv	3. Schlüsselwert ist neu. Vektorgröße ist nicht für die Codierung ausreichend. INSERT INTO HIST00 ... VALUES(... 'PERSON'...);	**Überlegung:** Neuerstellung des Index, damit die neuen Werte integriert werden!

Wenn Sie bereits EVIs erstellt haben, sollten Sie daher unbedingt prüfen, ob sich derzeit Werte im Overflow-Bereich befinden. Wenn ja, erstellen Sie den Index neu. Aufgrund dieser Eigenschaften erklärt sich auch, warum Sie Encoded Vector Indizes vorwiegend für

– Read-Only-Tabellen oder

– Spalten mit einem relativ statischen Wertebereich

definieren sollten.

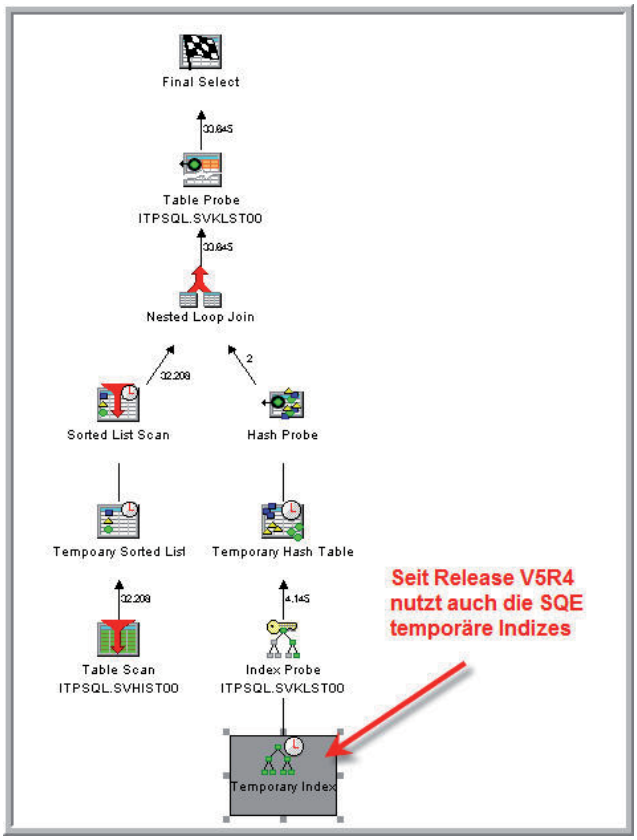
3.4.2.3
Seite 10

Maintained Temporary Index (MTI)

Vor Release V5R4 wurden temporäre Indizes ausschließlich von der CQE erstellt. Die SQE nutzte bis Release V5R4 keine temporären Indizes; sie bevorzugte die Methode „Table Scan“. Erst mit Release V5R4 wurden temporäre Indizes als neues Feature eingeführt. Der SQE Optimizer beobachtet – vor der Erstellung des temporären Indexes – die Ausführung mehrerer SQL-Abfragen und berücksichtigt die im Index Advisor hinterlegten Informationen, um zu entscheiden, ob die Erstellung eines temporären Index sinnvoll ist. Durch das Konzept der temporären Indizes kann sich die SQE ein Stück weit selbst tunen. Schauen wir uns die Verwendung an einem einfachen Beispiel an:

```
SELECT HIKDNR, KLNAM1, HIMED
FROM SVHIST00 JOIN SVKLST00 ON HIKDNR = KLIDNR
ORDER BY HIMED;
```

Selektiert werden alle Kundenkontakte. Es existiert für die Spalte „HIMED“ der Tabelle „SVHIST00“ kein permanenter Index, der verwendet werden könnte. Ebenso wenig existiert ein Index für die zweite JOIN-Spalte „HIKDNR“. Daher entscheidet sich der Optimizer einen temporären Index zu erstellen.



Erstellung eines temporären Indexes

3.4.2.4
Seite 2

Für die Erstellung und Nutzung des temporären Indexes benötigt die SQE zusätzliche Verarbeitungszeit. Dieser Zeitfaktor ist jedoch geringer als jede alternative Zugriffsmethode.

Das Konzept der temporären SQE-Indizes wurde allerdings vollständig überarbeitet.

Die CQE nutzt die temporären Indizes innerhalb eines Jobs für eine einzige SQL-Abfrage. Die Mehrfachverwendung eines solchen Index – auch innerhalb des gleichen Jobs – ist für die CQE ausgeschlossen. Für die SQE wurde die Erstellung und Wiederverwendung von temporären Indizes optimiert, indem die Informationen über die temporären Objekte im Plan Cache hinterlegt werden. Der Plan Cache enthält also nicht nur Ausführungspläne, sondern auch temporäre SQE-Indizes. Dadurch ist es möglich, einen temporären Index einmalig zu erstellen und anschließend beliebig vielen SQL-Abfragen in verschiedenen Jobs zur Verfügung zu stellen. Die temporären SQE-Indizes bleiben solange im Plan Cache existent, bis alle Zugriffspläne aus dem Plan Cache entfernt sind, die das temporäre Objekt nutzen. Die neuen temporären Indizes, die ausschließlich von der SQE erstellt werden, nennt man übrigens „Autonomic Index“ (AI) oder „Maintained Temporary Index“ (MTI).

So erklären sich auch die unterschiedlichen Verarbeitungszeiten einiger meiner SQL-Abfragen, die ich mir im Plan Cache anzeigen lasse. Ich öffne hierfür die Datenbank ➜ Rechtsklick auf „Momentaufnahmen für SQL Planungscache“ ➜ „SQL Planungscache“ ➜ „Anweisungen anzeigen…“. Daraufhin öffnet sich das Fenster „Anweisungen des SQL-Planungscache“. Jetzt selektiere ich die Ausführungspläne und wähle über das Kontextmenü des entsprechenden Ausführungsplanes den Eintrag „Längste Ausführungen“:

Längste Ausführungen:

Ausführungs- zeit	Verarbeitungs- zeit (Sek)	Ausgewählte Sätze	Benutzer- name
06.11.08 17:26:36	0,0140	181	RASCHE
06.11.08 17:26:42	0,0098	181	RASCHE
06.11.08 17:32:56	0,0076	181	RASCHE
06.11.08 17:32:58	0,0007	181	RASCHE
06.11.08 17:34:03	0,0007	181	RASCHE
06.11.08 17:34:06	0,0007	181	RASCHE

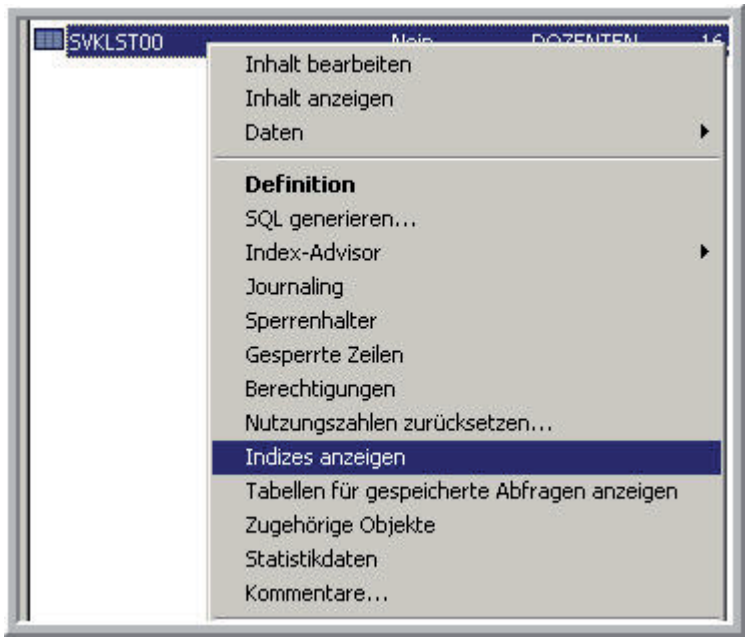
MTI wird erstellt.

Unterschiedliche Ausführungszeiten mit MTI

In der Abbildung ist deutlich zu erkennen, dass die erste Verarbeitungszeit um ein Vielfaches höher ist als die nachfolgenden. Den Grund kennen wir jetzt; der temporäre Index musste erstellt werden. Da der Plan Cache aber beim IPL geleert wird, sind diese quasi-permanenten Indizes nach einem IPL nicht mehr existent und müssen bei der nächsten Ausführung einer SQL-Abfrage, die einen solchen Index erfordert, neu erstellt werden. Es kann daher sinnvoll sein, die temporären Objekte permanent zu erstellen.

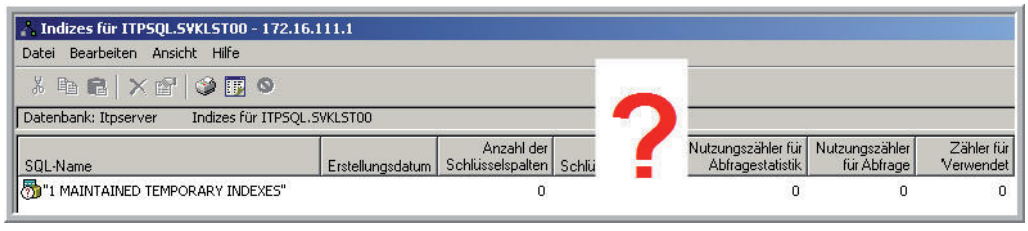
Doch da stellt sich die Frage „Wie spüren wir die temporär erstellten Indizes auf?“ System i Navigator bietet gleich mehrere Möglichkeiten, diese Objekte zu finden:

Die Funktion „Indizes anzeigen für…“
Sie können sich mit System i Navigator die Indizes anzeigen lassen, die für eine Tabelle erstellt wurden. Ich öffne hierfür zunächst die Datenbank und das Schema, das die Tabelle „DETAIL“ enthält. Anschließend wähle ich - wie in der folgenden Abbildung dargestellt - im Kontextmenü der Tabelle den Eintrag „Indizes anzeigen“.



System i Navigator - Indizes anzeigen

Für die folgende Bildschirmausgabe verändere ich zunächst die Reihenfolge der Spalten, weil die benötigten Informationen ansonsten sehr weit rechts stehen. Hierfür können Sie das Kontextmenü oder die Funktionstaste „F12“ nutzen.

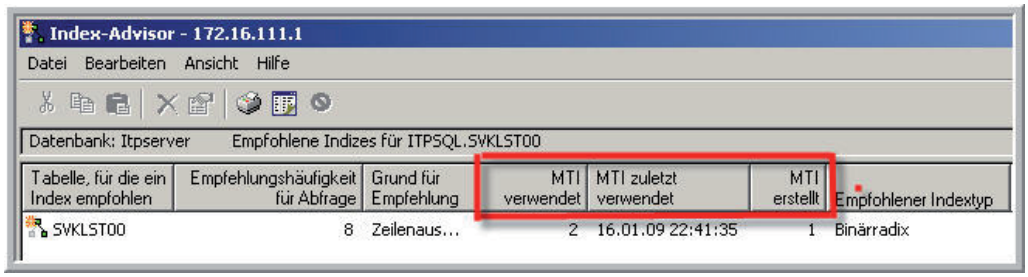


Indizes anzeigen - MTI

3.4.2.4
Seite 4

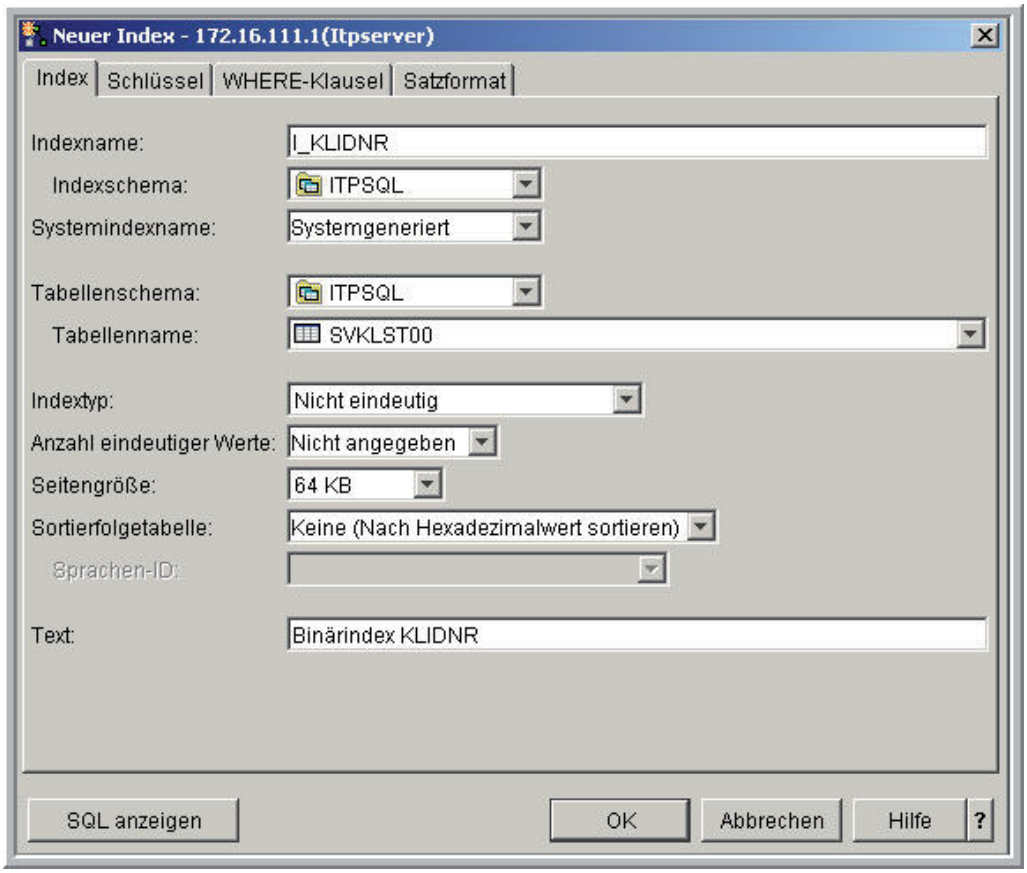
Tatsächlich wird der temporäre Index für die Tabelle „SVKLST00“ angezeigt. Allerdings sind die Informationen eher dürftig. Während für permanente Indizes Nutzungszähler existieren und die Schlüsselspalten der Indizes angezeigt werden, erhalten Sie in diesem Fall keine Informationen. Man kann nicht einmal erkennen, wann der Index erstellt und wie häufig er genutzt wurde. Für mich ist diese Anzeige enttäuschend. Vielleicht ändert sich an dieser Darstellung ja noch etwas im Laufe der Zeit.

Index-Advisor
Etwas besser sind die Informationen im Index Advisor. Hierfür wählen Sie im Kontextmenü der Tabelle den Eintrag „Index Advisor“. Ich werde zu einem späteren Zeitpunkt auf den Index Advisor zurückkommen. Jetzt beschränke ich mich lediglich auf die Darstellung der MTI-Informationen. Wir suchen alle Indexvorschläge mit entsprechenden Einträgen in den Spalten „MTI erstellt“ und „MTI verwendet“. Die benötigten Informationen stehen weit rechts, wenn Sie nicht die Spaltenreihenfolge der Bildschirmausgabe ändern – wie ich es getan habe:



Index Advisor – MTI

Die Informationen sind deutlich besser als in der vorangegangenen Bildschirmausgabe. Die Empfehlung für den Binärindex mit dem Schlüsselfeld „KLIDNR“ erfolgte insgesamt acht Mal aufgrund lokaler Selektionskriterien. Ein Mal wurde der temporäre Index mit dieser Spalte bisher aufgebaut und insgesamt zweimal genutzt. Zuletzt wurde der MTI am „16.01.2009“ verwendet. Wir hätten jetzt – aufgrund dieser Informationen – die Möglichkeit zu entscheiden, den Index permanent zu erstellen. Sie müssen hierfür den Dialog nicht verlassen, sondern könnten im Kontextmenü des Indexvorschlages den Eintrag „Index erstellen“ wählen. Daraufhin erscheint ein neuer Dialog:



Index Advisor – MTI erstellen

Alle benötigten Parameter sind voreingestellt – entsprechend der Indexempfehlung. Es fehlt nur noch ein Indexname und eventuell ein beschreibender Text. Im Release V6R1 wurden die Register „WHERE-Klausel“ und „Satzformat“ hinzugefügt. Auf diese Register kommen wir etwas später noch einmal zurück. Durch diesen Assistenten lassen sich sehr komfortabel Indizes erstellen. Die Anzeige könnte allerdings dazu verleiten, wahllos Indizes aufgrund der Indexempfehlungen zu erstellen. Sie sollten stets sorgfältig prüfen, ob der Index wirklich ein guter Indexkandidat ist oder ob es nicht wesentlich bessere Spaltenkombinationen gibt, und denken Sie daran, Indexempfehlungen nach der Bearbeitung aus der Liste zu entfernen, damit nicht zu einem späteren Zeitpunkt der Eindruck entsteht, es sei ein weiterer temporärer Index erstellt worden.

3.4.2.4
Seite 6

Die Nutzung des Index Advisors ist für die Untersuchung der MTIs besser geeignet als die zuvor vorgestellte Möglichkeit. Ein Nachteil bleibt allerdings bestehen: Sie müssen wissen, für welche Tabelle temporäre Indizes erstellt wurden. Natürlich könnten Sie den systemweiten Index Advisor nutzen und die MTIs darüber ermitteln. Allerdings stehen dort nicht selten mehr als 10.000 Indexempfehlungen – alte und aktuelle. Hinzu kommt, dass ich die Handhabung der grafischen Darstellung sehr umständlich finde. Mir sind daher kompaktere Informationen lieber.

Die Katalogtabelle: System Wide Index Advised (SYSIXADV)
Alle Informationen der Indexratgeber werden in der Systemtabelle QSYS2. SYSIXADV hinterlegt. Sie können mit Hilfe von SQL auf die Informationen zugreifen und dabei beliebige Filterkriterien verwenden. Diese Metatabelle benutze ich jetzt, um mir die notwendigen Informationen über die MTIs zu beschaffen:

```
SELECT TABLE_NAME AS TABLE, KEY_COLUMNS_ADVISED AS KEYS,
CHAR(DATE( LAST_ADVISED) , EUR), TIMES_ADVISED AS ADVICED,
MTI_USED AS USED, MTI_CREATED AS CREATED,
CHAR(DATE( LAST_ADVISED) , EUR) AS „LAST USED“
FROM QSYS2.SYSIXADV
WHERE TABLE_SCHEMA = ‘ITPSQL’ AND MTI_USED > 0
ORDER BY MTI_USED DESC;
```

Im Result Set erscheinen anschließend – kompakt dargestellt – die gewünschten Informationen:

```
SELECT TABLE_NAME as TAble, KEY_COLUMNS_ADVISED AS KEYS,
char(date( LAST_ADVISED) , EUR), TIMES_ADVISED as Adviced,
MTI_USED AS USED, MTI_CREATED AS CREATED,
char(date( LAST_ADVISED) , EUR) AS "LAST USED"

FROM QSYS2.SYSIXADV

WHERE TABLE_SCHEMA = 'ITPSQL' AND MTI_USED > 0
ORDER BY MTI_USED DESC;
```

TABLE	KEYS	00003	ADVICED	USED	CREATED	LAST USED
DETAIL	DEKURZ	06.11.2008	15	2	1	06.11.2008
ORT	LKZ	06.11.2008	12	2	1	06.11.2008

Index Advisor – SYSIXADV

Haben Sie Bedenken, dass die temporären Indizes Ihr System unnötig belasten? Ihre Sorge ist grundlos, da der SQE Optimizer – anders als die CQE - nicht für einfache lokale Selektionskriterien temporäre Indizes erstellt. Außerdem werden mögliche Ausführungsalternativen sorgfältig geprüft. Wenn Sie dennoch die Erstellung temporärer Indizes einschränken wollen, können Sie das Attribut „ALLOW_TEMPORARY_INDEXES“ in der Attributdatei QAQQINI ändern. Mit dieser Option legen Sie fest, inwieweit temporäre Indizes vom Optimierungsprogramm für die Ausführung der Abfrage berücksichtigt werden. Die Einstellung „*YES“ wird als Unterlassungswert verwendet; sie erlaubt dem Optimizer temporäre Indizes einzusetzen. Wenn Sie den Wert „*ONLY_REQUIRED“ angeben, werden temporäre Indizes für den Zugriffsplan nicht berücksichtigt. Der Optimizer wählt eine andere Implementierung unabhängig von den Kosten, um keinen temporären Index zu erstellen. Nur wenn kein funktionsfähiger Plan gefunden wird, wird ein temporärer Index benutzt. Wenn Sie hingegen verhindern wollen, dass die MTIs im Plan Cache gespeichert werden, weil Sie den Cache Bereich für die Ausführungspläne benötigen, bewirkt die Änderung der Option „CACHE_RESULTS“ in der Attributdatei „QAQQINI“, dass die MTIs gelöscht werden, wenn der Datenpfad der Abfrage geschlossen wird. Dies entspricht dem Verhalten der CQE.

3.4.2.4

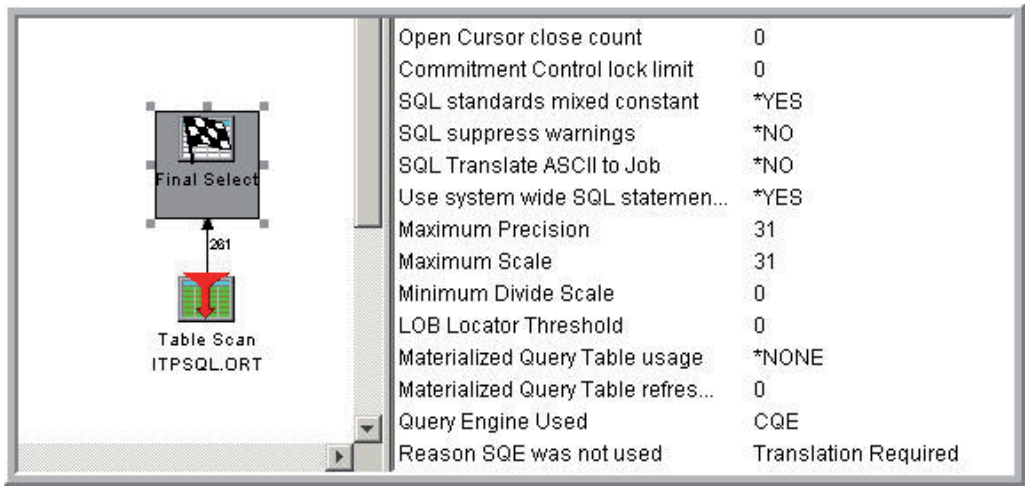
Seite 8

Abgeleitete Indizes Release V6R1

Traditionell konnten in SQL-Indizes nur einzelne Spalten oder Spaltenkombinationen verwendet werden. Seit Release V6R1 ist es auch möglich, Ausdrücke oder SQL-Funktionen im SQL-Index zu verwenden. Schauen Sie sich die folgenden Beispiele an, die in keinem Zusammenhang stehen, sondern einzig und allein gewählt wurden, um die jeweilige Problematik darstellen zu können:

```
SELECT *
FROM ORT
WHERE UPPER(ORTSNAME) = ‚BRÜSSEL‘;
```

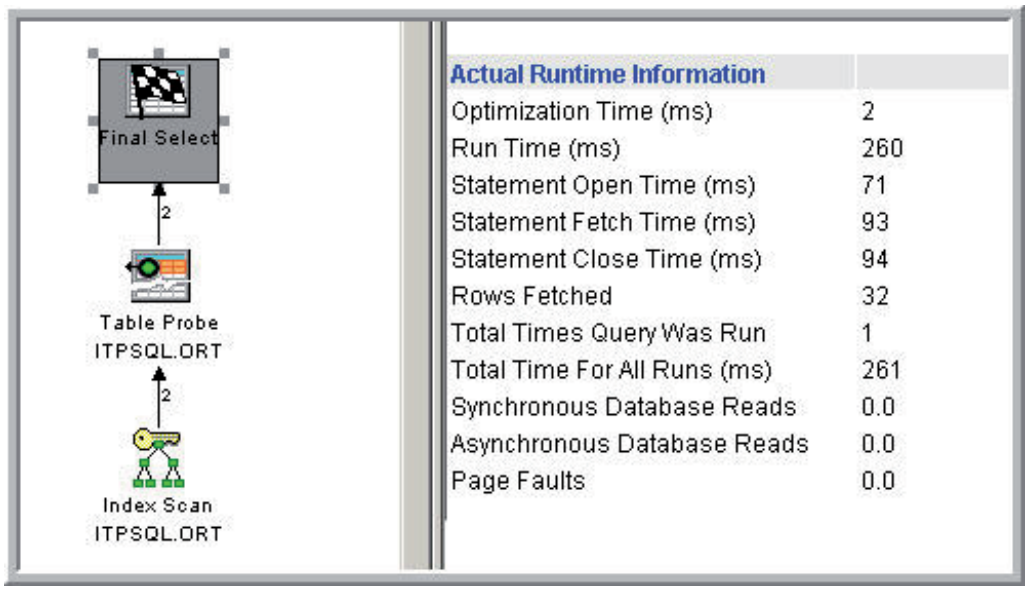
Die Funktionen „UPPER“, „LOWER“ und „TRANSLATE“ werden im Release V5R4 von der CQE ausgeführt. Als Grund finden Sie stets den Hinweis „Translation required“. Doch obwohl ein Binärindex für die Spalte „Ortsname“ existiert, kann die CQE den Index für den Datenzugriff nicht nutzen.



NLSS Konvertierung V5R4

3.4.2.5
Seite 2

Seit Release V6R1 kann die SQE derartige Abfragen ausführen und nutzt hierfür auch die bestehenden Binärindizes – wie folgende Abbildung beweist:



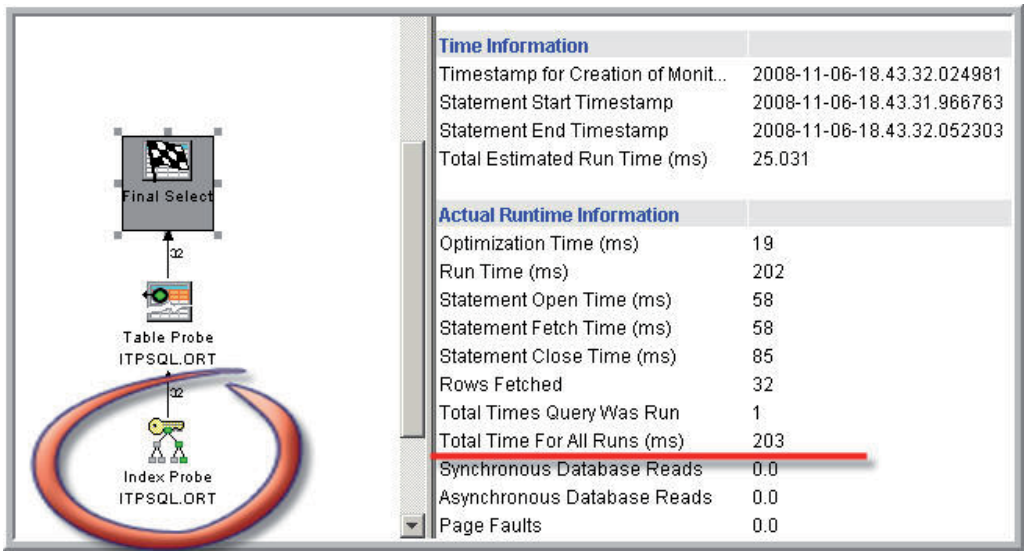
NLSS Konvertierung V6R1

Allerdings muss der Index noch seitenweise gelesen werden, da aufgrund der Funktion „UPPER“ ein direkter Vergleich der Schlüsselwerte nicht möglich ist. Der Vorteil der Methode „Index Scan“ liegt darin, dass über den Index sehr viel mehr Sätze mit einer einzigen I/O-Operation gelesen werden können. Die gelesene Seite enthält lediglich die relative Satznummer und den Schlüsselwert, während die I/O-Seite beim „Table Scan“ alle Spalten der Tabelle beinhaltet. So erklärt sich die Zeitersparnis, wenn die SQE die Abfrage ausführt.

Seit Release V6R1 geht es aber noch besser. Erstellen Sie einfach einen abgeleiteten Index, damit der direkte Zugriff über Schlüsselwerte möglich wird:

```
CREATE INDEX I_UPPER_ORTSNAME
ON  ITPSQL.ORT(UPPER(ORTSNAME))
```

Die Syntax ist einfach. Sie können Funktionen oder Ausdrücke in der Spaltenbezeichnung verwenden. Nach der Erstellung des abgeleiteten Index ändert sich der Ausführungsplan erneut.



Einsatz abgeleiteter Indizes V6R1

Die SQE wählt jetzt die Methode „Table Probe“. Während des Zugriffs auf die Daten können die Schlüsselwerte bereits geprüft werden, so dass nur noch die tatsächlich benötigten Sätze gelesen werden. Die Datenmenge und die benötigte CPU-Zeit reduzieren sich erneut. Natürlich können Sie auch andere skalare SQL Funktionen nutzen:

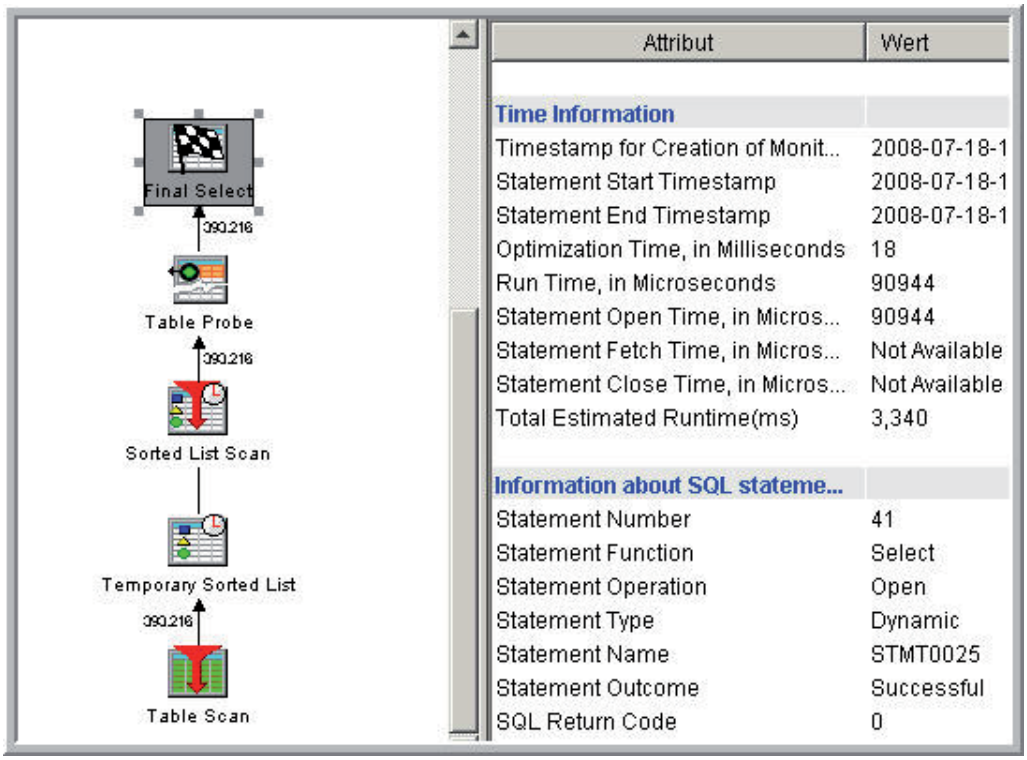
Eine ähnlich gute Wirkung lässt sich mit diesen Indizes erzielen, wenn Sie mathematische Ausdrücke z. B. in der „ORDER-BY“-Klausel nutzen.

Im nächsten Beispiel benötige ich eine Liste, die nach Bestandswerten (PRDQTY) sortiert ist. Hierbei soll aber auch die aktuelle Bestellmenge (PRDBSTQTY) berücksichtigt werden.

```
SELECT *
FROM PRDKT00
ORDER BY (PRDQTY+ PRDBSTQTY);
```

3.4.2.5
Seite 4

Die folgende Abbildung zeigt, dass die Daten zunächst vollständig gelesen werden. Erst im Arbeitsspeicher wird der Ausdruck aufgelöst, um die Sortierfolge zu bestimmen.

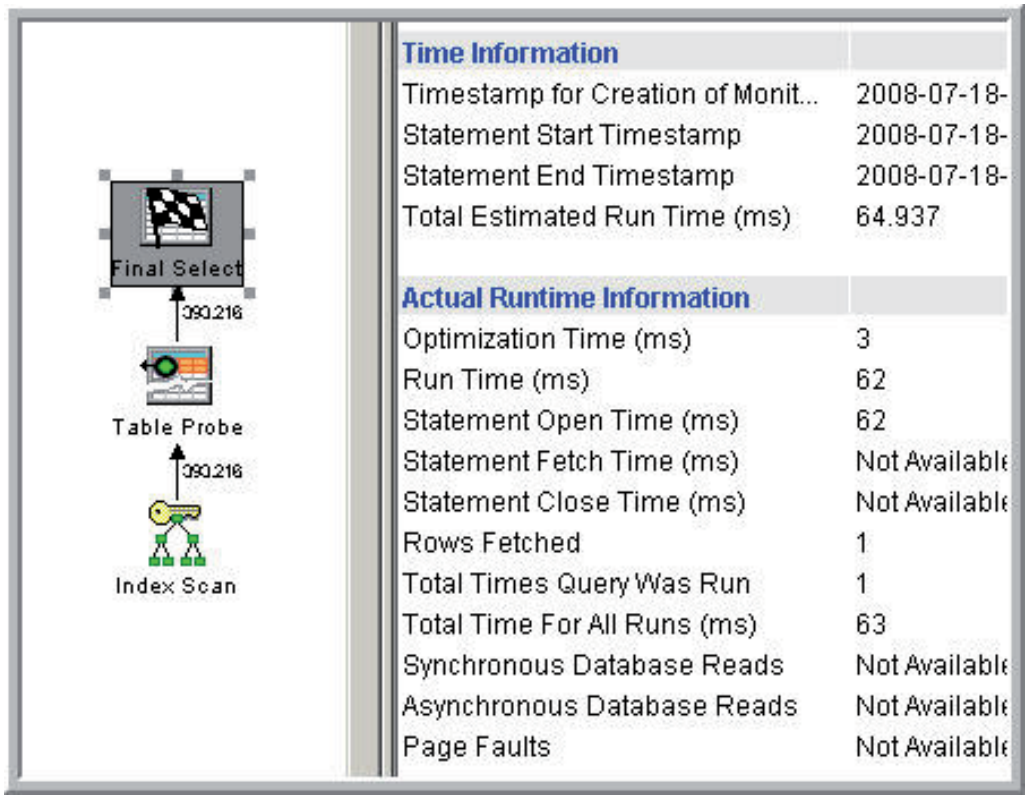


Mathematische Ausdrücke „ORDER BY“ V5R4

Angesichts der hohen Laufzeiten kann man schnell auf die Idee kommen, Summenspalten in den Tabellen zu nutzen, um derartige Ausführungspläne zu umgehen. V6R1 löst das Problem durch abgeleitete Indizes.

```
CREATE INDEX I_PLUS_BSTQTY_QTY
ON   PRDKT00(PRDBSTQTY+PRDQTY);
```

Die SQE liest die Daten jetzt anhand des abgeleiteten Index „I_PLUS_ BST-QTY_QTY“ in sortierter Folge.



Mathematische Ausdrücke „ORDER BY“ V5R4

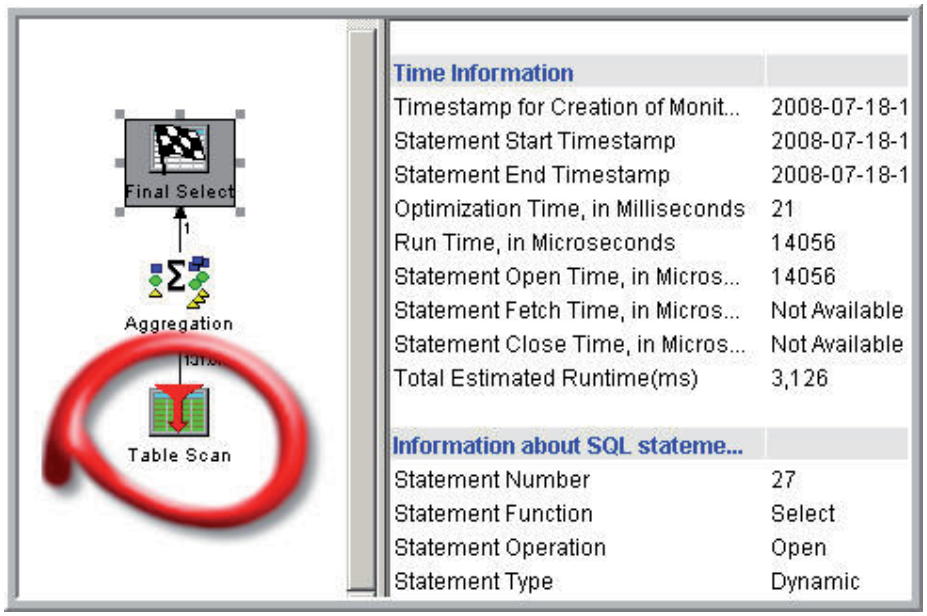
Alle temporären Sortieralgorithmen entfallen. Die Abfrage benötigt daraufhin anstatt 90,944 ms nur noch 0,0063 ms.

Aber auch, wenn Sie Aggregatfunktionen einsetzen, bringen die neuen Indizes Vorteile: Nehmen wir an, wir möchten wissen, wie viele Artikel mehr als 100.000,00 Euro Bestandswert haben. Hierbei sollen wieder die aktuellen Bestellmengen berücksichtigt werden. Also müssten wir folgende SQL-Abfrage schreiben:

```
Select COUNT(*) AS “Anzahl”
from PRDKT00
where (PRDQTY + PRDBSTQTY) > 100000;
```

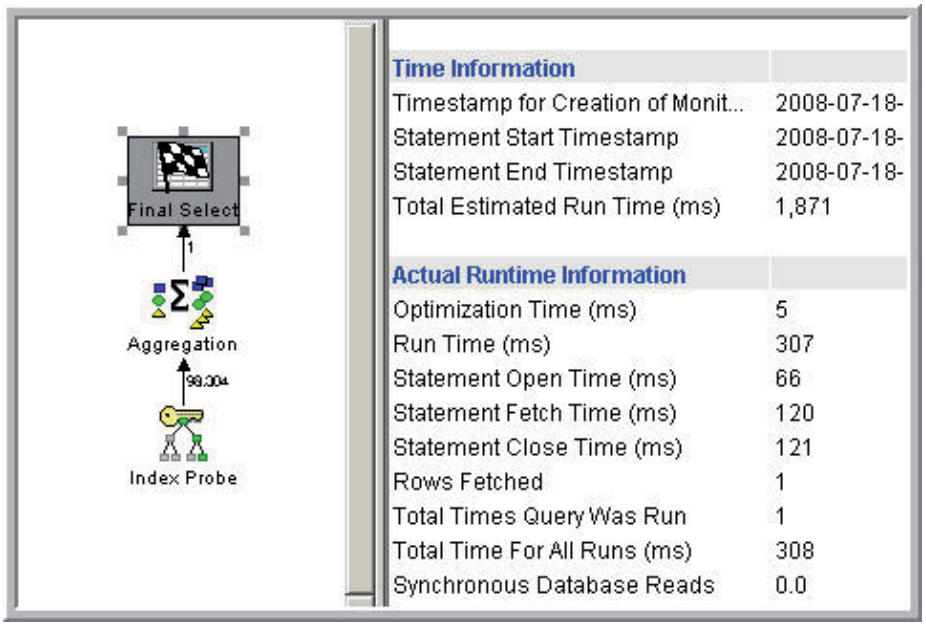
3.4.2.5
Seite 6

Meine Abfrage enthält in diesem Fall in der „WHERE“-Klausel einen mathematischen Ausdruck. Die SQE kann zwar das SQL-Statement ausführen, aber sie muss – wie der folgende Ausführungsplan zeigt - aufgrund der Aggregatfunktion „COUNT“ die Tabelle sequentiell lesen, obwohl zwei eigenständige Indizes für die Abfrage vorhanden sind. Die Auflösung des Ausdruckes erfolgt wieder zur Laufzeit.



Verwendung von Ausdrücken in lokalen Selektionskriterien V5R4

Auch hier ist der zuvor erstellte Index „I_PLUS_BSTQTY_QTY“ nützlich. Der Index enthält die notwendigen Informationen, so dass ein Zugriff auf die Tabelle „PRDKT00“ unnötig wird.



Einsatz abgeleiteter Index mit mathematischer Funktion V6R1

Abgeleitete Indizes können von der SQE auch genutzt werden, wenn Sie die Ausdrücke in „WHERE“-Klauseln ohne Aggregatfunktion verwenden. Hier habe ich aber bisher die Erfahrung gemacht, dass die Indizes für die Spalten „PrdQty“ und „PrdBstQty“ vorher gelöscht werden müssen, damit der neue Index verwendet wird. Grundsätzlich sollten Sie am Anfang sehr genau prüfen, ob die abgeleiteten Indizes den erhofften Nutzen bringen, und zunächst sehr einfache Ausdrücke verwenden. Je komplexer die Ausdrücke aufgebaut sind, desto unwahrscheinlicher wird die Verwendung des Index. Laut IBM spielt in diesen Fällen auch die Spaltenfolge in der „WHERE“-Bedingung eine entscheidende Rolle. Wurde der Index – wie bei mir – mit der Spaltenfolge „PRDQTY“ und „PRDBSTQTY“ erstellt, müssen die Abfragen die Indexspaltenfolge beachten. Diese Erfahrung habe ich bislang nicht machen können. Das kann auf Ihrem Server mit anderer Laufzeitumgebung und anderem Datenbestand anders sein. Probieren Sie es aus! Sie finden sicherlich sehr schnell weitere sinnvolle Einsatzmöglichkeiten für den Einsatz abgeleiteter Indizes und lösen auf diesem Weg das eine oder andere Laufzeitproblem.

IBM hat mit Einführung der abgeleiteten Indizes im Übrigen auch die Möglichkeit geschaffen, abgeleitete logische DDS-Dateien zu ersetzen.

```
***** Datenanfang ***********************************
     R DETAILFMT                 PFILE(ITPSQL/DETAIL)
       DEKURZ
       MAKURZ
       BZKURZ
       BZLANG
     K DEKURZ
```

Logische DDS-Datei mit Satzformatnamen

Dieses DDS-Objekt konnte bislang nicht durch einen SQL-Index ersetzt werden. Hinderlich war der Satzformatname. Bislang war es lediglich möglich den SQL Tabellen eigenständige Satzformatnamen zuzuweisen. Seit Release V6R1 ist die Klausel RCDFMT auch für SQL Indizes vorhanden.

```
CREATE INDEX ITPSQL.I_DEKURZ ON ITPSQL.DETAIL(DEKURZ)
RCDFMT DETAILFMT;
```

3.4.2.5
Seite 8

Des Weiteren besteht mit DDS die Möglichkeit alle Spalten der physischen Datei in das Satzformat der logischen Datei zu übernehmen oder nur bestimmte Spalten im Satzformat zu benennen. Auch diese Möglichkeit kannte der SQL-Index bislang nicht. Mit Release V6R1 kommt daher eine weitere Klausel hinzu:

```
CREATE INDEX ITPSQL.I_DE_C01 ON ITPSQL.DETAIL (DEKURZ  ASC)
RCDFMT DEKURZFMT ADD ALL COLUMNS;
```

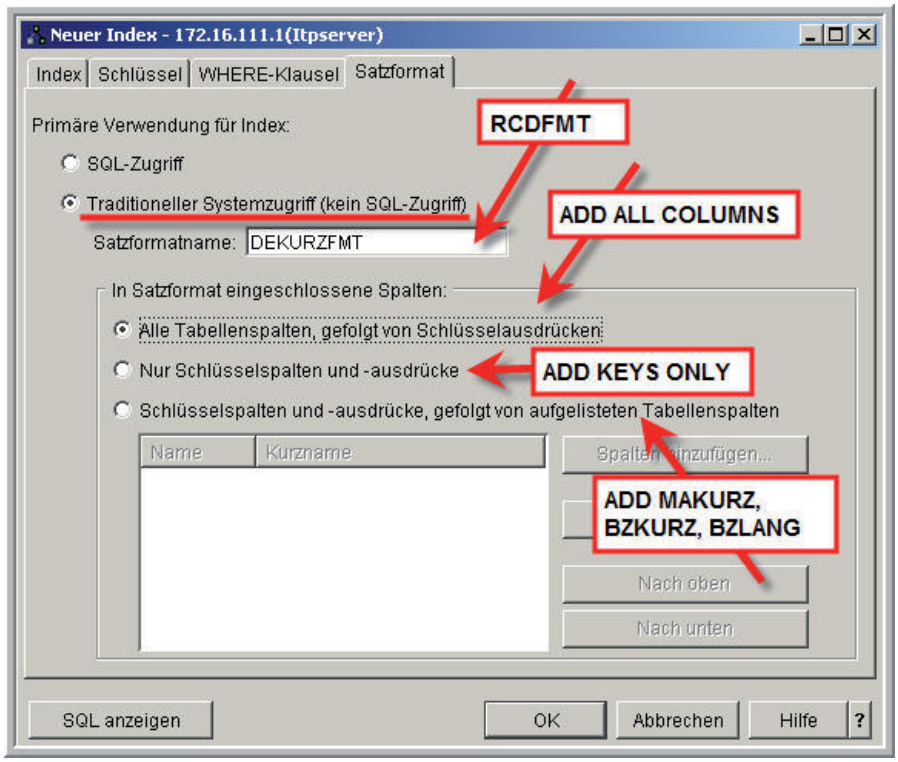
Um alle Spalten ins Satzformat zu übernehmen, müsste die Klausel ADD ALL COLUMS verwendet werden. „ADD ALL COLUMS“ ist im Übrigen der Unterlassungswert. Sollen hingegen lediglich die Schlüsselfelder im Satzformat vorhanden sein, müsste die Klausel ADD KEYS ONLY verwendet werden.

```
CREATE INDEX ITPSQL. I_DE_C02 ON ITPSQL.DETAIL (DEKURZ  ASC)
RCDFMT DEKURZFMT ADD KEYS ONLY;
```

Und letztendlich können Sie die Felder des Satzformates jetzt auch mit SQL individuell bestimmen wie im Ausgangsbeispiel mit DDS gezeigt:

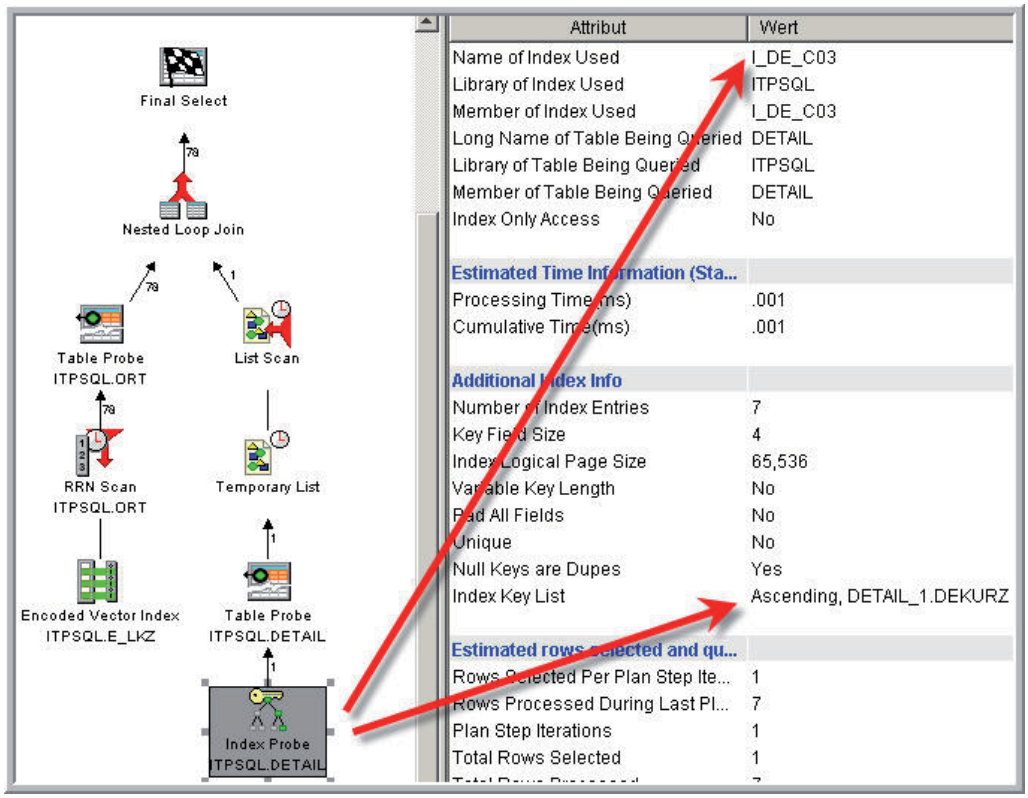
```
CREATE INDEX ITPSQL. I_DE_C03 ON ITPSQL.DETAIL (DEKURZ  ASC)
RCDFMT DEKURZFMT ADD DEKURZ, MAKURZ, BZKURZ, BZLANG;
```

Aufgrund dieser neuen Klauseln wurde im System i Navigator der Dialog „Neuer Index“ um das Register „Satzformat“ erweitert:



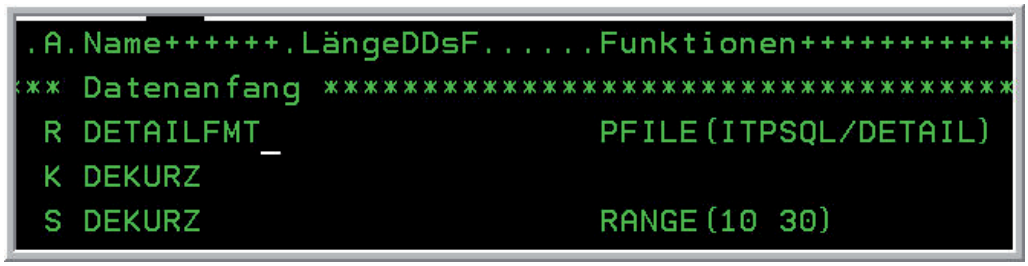
Der Dialog „Neuer Index – Satzformat“ V6R1

Sie können den Satzformatnamen benennen, alle Spalten, nur die Schlüsselspalten oder die Schlüsselspalten und individuell benannte Spalten im Satzformat aufnehmen. Außerdem legen Sie die primäre Verwendung des Index fest. Hier könnte schnell der Eindruck entstehen, dass Indizes, die primär für den traditionellen Systemzugriff erstellt werden, nicht vom SQL Optimizer während des Optimierungsprozesses berücksichtigt werden. Dies ist jedoch nicht der Fall, wie der folgende Ausführungsplan schnell beweist:



SQL Index - Traditioneller Systemzugriff

Im Optimierungsprozess nicht berücksichtigt werden lediglich Indizes, die abgeleitete logische DDS-Dateien ersetzen. Schauen wir uns noch einmal eine DDS-Ausgangssituation an:



Logische DDS-Datei abgeleitet

3.4.2.5
Seite 10

Dieses Mal werden alle Felder der physischen Datei übernommen, aber die logische DDS-Datei enthält Selektionskriterien. Es handelt sich um eine „abgeleitete", logische DDS-Datei. Derartige Selektionen können im Release V6R1 ebenfalls mit SQL abgebildet werden.

```
CREATE INDEX ITPSQL.I_DETAIL02 ON ITPSQL.DETAIL (DEKURZ  ASC)
WHERE DEKURZ  BETWEEN  10 and 30
RCDFMT DETAILFMT ADD ALL COLUMNS;
```

Der Index erhält jetzt eine zusätzliche „WHERE"-Bedingung. Dieser abgeleitete Index wird nicht im Optimierungsprozess berücksichtigt; er soll ausschließlich helfen, DDS-Objekte vollständig abzulösen. Bisher waren abgeleitete logische DDS-Dateien ein Handicap bei der Abschaffung der DDS-Dateien. Dieses Handicap hat IBM im Release V6R1 beseitigt! Es bleibt abzuwarten, ob und wann IBM sich von DDS-Syntaxen im Datenbankumfeld vollständig trennen wird. Sie sollten nicht so lange mit der Umstellung Ihrer Dateibeschreibungen warten, sondern bereits jetzt beginnen sukzessive Ihre DDS-Quellen durch SQL-Syntax ersetzen.

Der Entwurf und die Erstellung relationaler Datenbanken

2. Ergänzung 03/2005

3.5
Seite 2

Allgemeine Gedanken

In den vorhergehenden Kapiteln habe ich Ihnen vermittelt, wie Daten in Datenbanken mit SQL gehandhabt werden können. Nun ist es nicht so, dass diese Informationen jedes einzelne Detail der Thematik abdecken, sondern es findet sich mit Sicherheit noch eine Fülle von Bereichen und Anwendungsmöglichkeiten, die dadurch noch nicht behandelt wurden. Nicht ohne Hintergrund hat die verfügbare Literatur im Zusammenhang mit Datenbanken einen gewaltigen Umfang erreicht. Nun ist es aber auch nicht meine Absicht, Ihnen eine vollständige Referenz zur Verfügung zu stellen. Ich möchte Ihnen eher eine Vorstellung vermitteln, was mit SQL und modernen relationalen Datenbanken machbar ist. Die Frage, die ich in diesem Teil des Buches stellen möchte, lautet: „Wie kann ich mir eine relationale Datenbank aufbauen und welche Betrachtungen sollte ich vor einer Implementierung anstellen?“

3.5.1
Seite 2

Der Entwurf

Der Entwurf einer relationalen Datenbank ist auf den ersten Blick so etwas wie das Problem beim Schreiben eines Briefes. Da sitzt man vor einem leeren Blatt Papier und überlegt, wie fange ich an. Der Grundgedanke beim Entwurf einer Datenbank ist die Abbildung des zu entwerfenden Systems in Form von so genannten Entitäten und Relationen, dem zufolge spricht man auch vom Entity-Relationship- Ansatz. Etwas weniger technisch gesprochen, man versucht ein System durch Dinge, Objekte und logische Einheiten zu beschreiben und untersucht die Verbindungen, die zwischen diesen Einheiten bestehen. Ein geradezu klassisches Beispiel sind die Entitäten <Rechnungskopf> und <Rechnungsposition>, die selbstverständlich auch in einer logischen Verbindung miteinander stehen. In den meisten Fällen wird bereits auf dieser Stufe des Entwurfs überlegt, welche Felder mit welchen Datentypen und dieser oder jener Anzahl von Dezimalstellen in einem Datensatz benötigt werden. Diese Überlegungen gehen aber schon zu weit in die Tiefe des zukünftigen Systems und sehr detaillierte Definitionen auf dieser Ebene erschweren die Arbeit des weiteren Entwurfs. Es besteht die Gefahr, sich früh in Feinheiten zu verlieren und dann den weiteren Ausbau eben diesen bereits vorhandenen Mustern anzupassen. So ist es aus meiner Sicht empfehlenswert, sich erst einmal einen Überblick über das abzubildende System zu verschaffen und zwar auf einer relativ hoch abstrahierenden Ebene. Sprechen Sie noch nicht über Tabellen und Spalten, sondern von Entitäten.

Ist das vielleicht leichter gesagt als getan? Wie können denn Entitäten identifiziert werden?

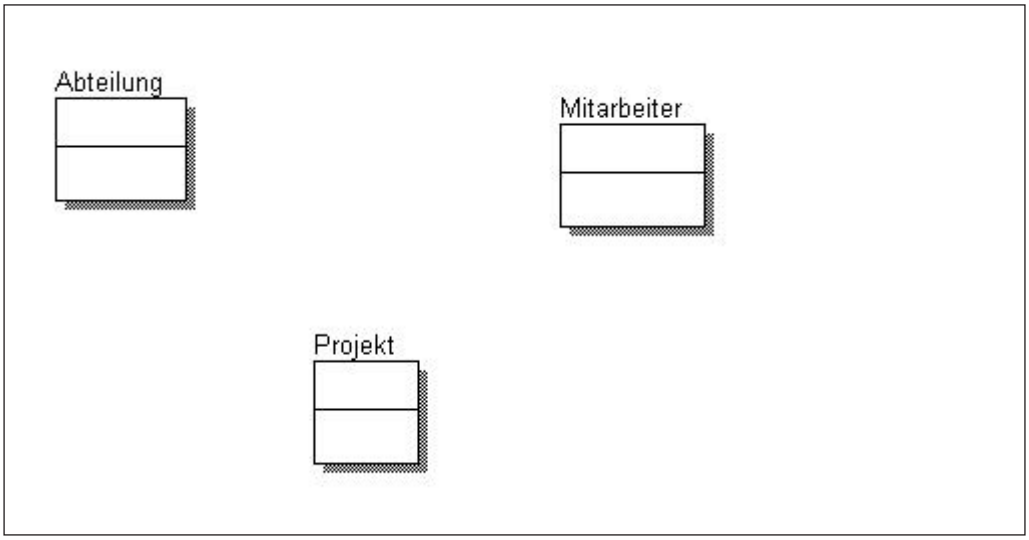
Nun oft liefert eine in Prosa gefasste Beschreibung viele Kandidaten für Entitäten.

3.5.2
Seite 2

Beispiel:

In einer Abteilung sind mehrere Mitarbeiter im Rahmen von Projekten tätig. Einzelne oder mehrere Mitarbeiter kommen im Zuge des Projekts dazu, andere erhalten neue Aufgabengebiete. Eventuell auch in anderen Abteilungen.

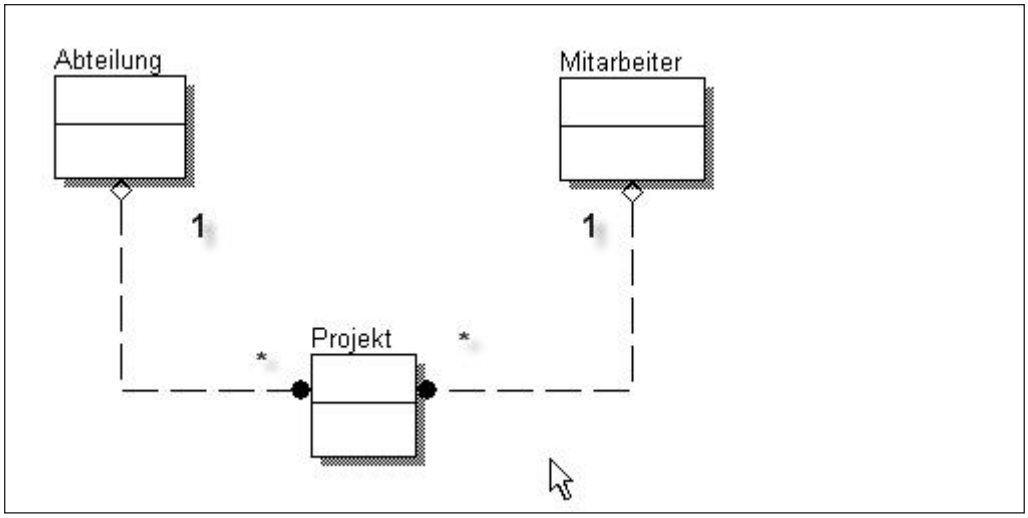
In diesem kurzen Text ist noch nicht die Rede von Feinheiten oder Details der Implementierung. Jedoch lassen sich hier einzelne Entitäten identifizieren. Schauen Sie sich einfach die Hauptworte an. Oft sind das mögliche Entitäten!



Kandidaten für Entitäten

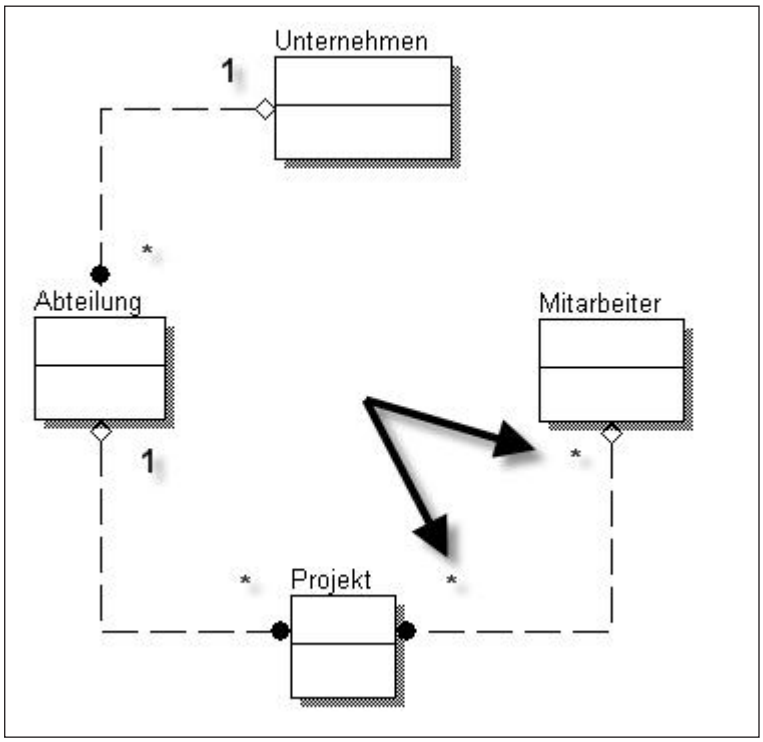
Wir wollen uns an dieser Stelle noch keine Gedanken darüber machen, ob die entdeckten Entitäten das System vollständig beschreiben, zumal es auch aus mathematischer Sicht durchaus zweifelhaft ist, ob Systeme ab einer gewissen Komplexität noch vollständig beschreibbar sind. Setzen wir uns als Ziel eine **möglichst weitgehende** Abdeckung.

Im folgenden Schritt werden Beziehungen und Verbindungen zwischen den Entitäten gesucht. Diese können durch sprachliche Wendungen wie „in einer Abteilung werden mehrere Projekte abgewickelt…“ oder „ein Mitarbeiter kann an mehreren Projekten beteiligt sein...“ ermittelt werden. So kann eine Verbindung identifiziert werden und man erhält auch die Kardinalitäten. Das heißt, welche Anzahl Entitäten stehen an einem Ende der Beziehung (Relation) mit welcher Anzahl Entitäten am anderen Ende in Verbindung.



Kardinalitäten

Denken wir noch ein wenig über die bisher gefundenen Entitäten nach. Eigentlich schwebt unsere Abteilung noch im luftleeren Raum, sie sollte zu einem Unternehmen gehören. Und ein Mitarbeiter kann an einem Projekt mitarbeiten, aber was ist mit einer umgekehrten Verbindung? Auch an einem Projekt wird in der Regel mehr als ein Mitarbeiter beteiligt sein. Die Beziehung einer Abteilung zu einer Firma ist leicht darstellbar. Bei den Projekten und Mitarbeitern sieht es etwas anders aus. Hier befinden sich auf beiden Seiten der Relation unter Umständen mehr als eine zugeordnete Entität. Also eine n : m Beziehung liegt hier vor. So weit so gut, bauen wir also die folgenden Kardinalitäten auf und fügen die Entität Unternehmen noch hinzu.



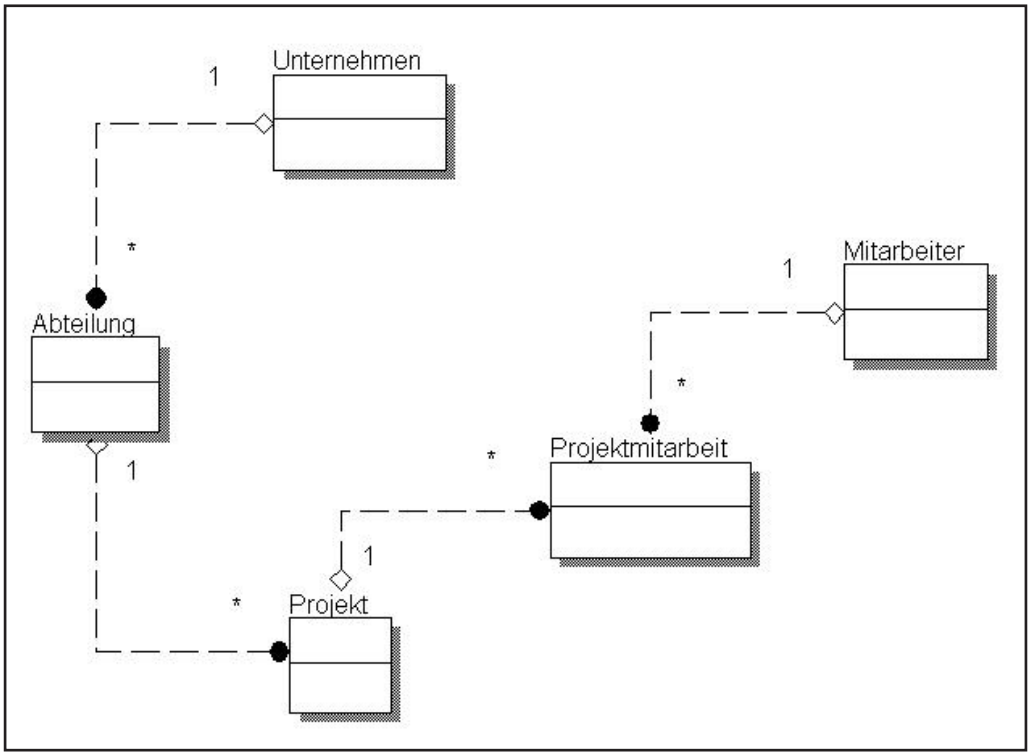
Die Entität Unternehmen kommt hinzu

3.5.2
Seite 4

Diese m : n Kardinalitäten sind im Datenbankentwurf alles andere als selten und sie sollten durch weitere Maßnahmen auch wieder beseitigt werden! Wir können nun bei dem Begriff der Relation noch einen Schritt weiter gehen und feststellen, dass auch Relationen Eigenschaften besitzen können.

Eine Entität macht nur dann einen Sinn, wenn sie Eigenschaften besitzt. Diese Eigenschaften spiegeln sich in letzter Konsequenz durch Spalten in Tabellen wieder. Aber wie kann eine Relation nun Eigenschaften erhalten und warum kann das Sinn machen?

Betrachten wir die mehrfache Verbindung zwischen Mitarbeiter und Projekt. Es wurde schon angemerkt, dass ein Mitarbeiter nach einer gewissen Zeit von einem Projekt abgezogen und einem anderen zugeteilt werden kann. Er ist also nur eine gewisse Zeit an einem Projekt beteiligt. Damit haben wir schon Eigenschaften der Beziehung zwischen den Entitäten, nämlich Beginn und Ende der Mitarbeit am Projekt. Wenn wir nun der Relation noch einen Namen geben, <Projektmitarbeit> ist eine Möglichkeit, so erhalten wir eine weitere Entität und können sie in unser Diagramm eintragen. Als Nebenerscheinung verschwinden auch die Mehrfach-Kardinalitäten an den Beziehungsenden.



Neue Struktur ohne Mehrfach-Kardinalitäten

Ich werde einmal kurz zusammenfassen, was dieses Diagramm aussagt.

- Ein Unternehmen umfasst mehrere Abteilungen.
- Eine Abteilung betreut mehrere Projekte.
- Ein Mitarbeiter kann über die Relation <Projektmitarbeit> in mehreren Projekten arbeiten.
- Ein Projekt erhält über die Relation <Projektmitarbeit> mehrere Mitarbeiter.

Achtung, bisher ist nur die Rede von Entitäten, Kardinalitäten und Relationen, noch haben wir nicht weiter über interne Eigenschaften (Spalten von Tabellen) nachgedacht. Zur Zeit befinden wir uns in einem logischen Modell und nicht in der Implementation. In großen Systemen wird man feststellen können, dass nicht zu jeder logischen Entität auch eine Tabelle gehört. Aber das würde in diesem Zusammenhang den Rahmen sprengen.

Welche Eigenschaften lassen sich nun einem Mitarbeiter zuordnen? Da wären sicher Dinge wie

- Anrede
- Titel
- Vorname
- Name
- Geburtstag
- Geschlecht
- Adresse
- …

Betrachten wir Spalten wie Anrede. Sie wird einmal Inhalte wie „Frau“, „Herr“ und so weiter annehmen. Nur, wenn die Werte sich einmal ändern sollten, muss der gesamte Datenbestand durchsucht werden und die Inhalte der betreffenden Spalte auf den neuesten Stand gebracht werden. So weit sollten die Dinge eigentlich bekannt sein, ebenso die Lösung. In diesen Fällen wird man so genannte Schlüsseltabellen einbauen. Aber welche Schlüsseltabellen und wie viele? Auch hier besteht eine praktikable Möglichkeit der Schlüsselflut Herr zu werden.

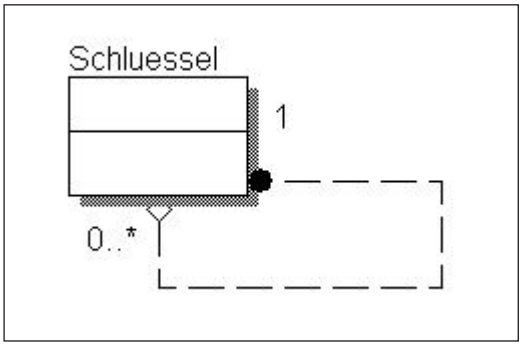
Wie stellt sich ganz allgemein ein Schlüssel dar, wenn man dieses Konzept genauer betrachtet.

Erstens gibt es den Schlüsselbegriff an sich. Also die Anrede, das Länderkennzeichen, Geschlecht und so fort. Zweitens stellt man fest, dass die Schlüssel natürlich bestimmte diskrete Werte annehmen. Zum Schlüsselbegriff <Anrede> existieren die Werte <Frau>, <Herr>, <Firma>…

2. Ergänzung 03/2005

Weiterhin kann es zu einem bestimmten Schlüsselbegriff auch untergeordnete Schlüssel geben, die eine Unterteilung des Oberbegriffs darstellen.

So kann der Schlüsselbegriff <Warengruppe> unterteilt werden in Begriffe für <Handelswaren> und <Dienstleistungen>. Wir halten fest, dass es Relationen innerhalb der Entität <Schlüssel> geben kann (aber nicht zwangsweise geben muss!).



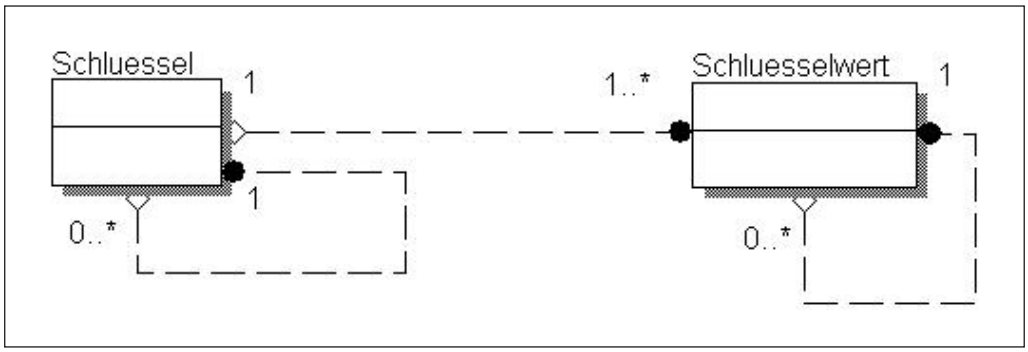
Schlüssel und Unterschlüssel

Durch die Kardinalität 0..* habe ich angedeutet, dass hier ein untergeordneter Schlüssel vorhanden sein kann, aber nicht muss. Eine kleine Anmerkung: Die Notation bietet Ihnen nicht nur eine gute Dokumentation der entstehenden Datenbank, sondern gerade die Aufzeichnung der Kardinalitäten liefert Ihnen Hinweise auf die Merkmale der späteren Spalten. Bei der Kardinalität 0..* wird es darauf hinauslaufen, dass das die logische Verbindung herstellende Feld Null-Werte enthalten kann.

In einer solchen Tabelle lassen sich also unterschiedliche Schlüsselbegriffe aufbewahren. Zu einem einzelnen Schlüsselbegriff wie zum Beispiel Länderkennzeichen gehören nun auch die diversen Schlüsselinhalte. Damit ist bei dem Länderkennzeichen eine Auflistung folgender Art gemeint.

- Argentinien
- Belgien
- China
- Deutschland
- . . .

In diesem Fall wird es außer den vollen Texten auch Kurztexte geben, also Kürzel, die dann beispielsweise in Briefköpfen auftauchen. Diese Inhalte (voller Text und Kürzel) lassen sich in eine weitere Tabelle aufnehmen, deren einzelne Sätze dann natürlich mit den Sätzen in der Tabelle der Schlüsselbegriffe in Beziehung stehen müssen. Hier wird sich zu einem einzelnen Schlüsselbegriff im Allgemeinen mehr als ein Satz in zugeordneten Werten finden, also eine 1 : n Beziehung liegt vor. Nun sollte man zusätzlich berücksichtigen, dass auch zu einem Schlüsselwert Untereinträge vorhanden sein könnten, die Betrachtung erfolgt dann analog zu den Überlegungen bei der eigentlichen Schlüsseltabelle. Aus diesen Überlegungen ergibt sich folgendes Bild.



Schlüsselbegriffe und ihre Inhalte

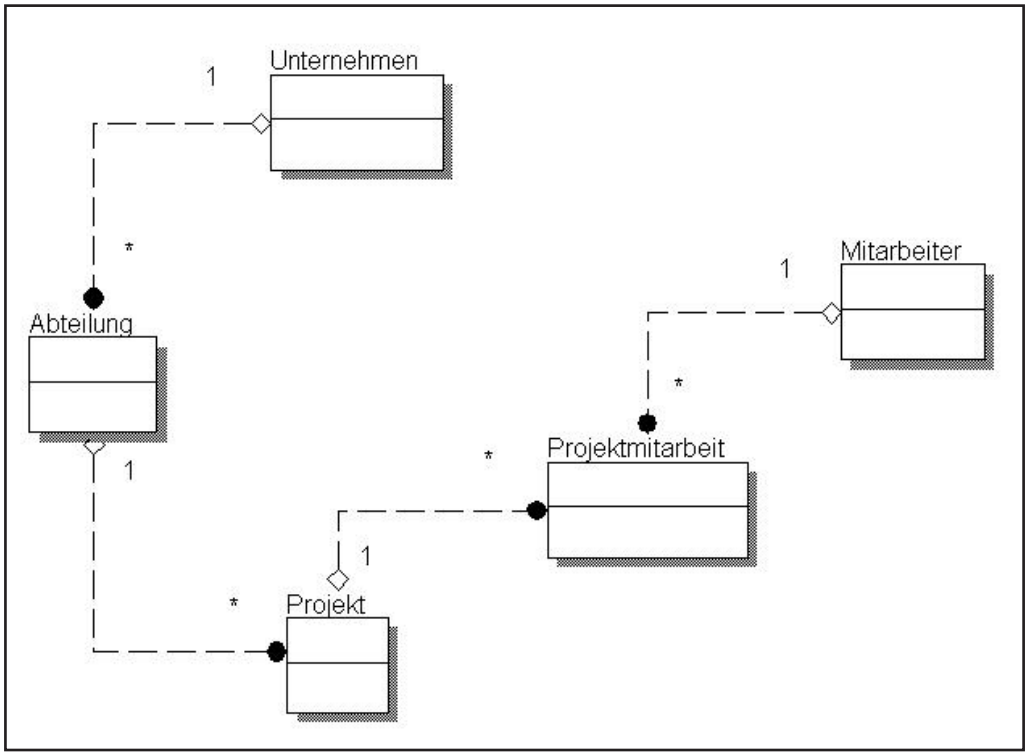
Betrachten wir die bei der neuen Tabelle <Schlüsselwert> angegebenen Kardinalitäten. Der Abbildung entnimmt man, dass zu einem Schlüsselwert kein, einer oder mehrere untergeordnete Schlüsselwerte vorhanden sein können. Aus der Sicht eines untergeordneten Wertes muss allerdings ein entsprechender Oberwert zugeordnet sein.

Wenn auf diese Art und Weise vorgegangen wird, ist es nicht mehr zwingend notwendig, neue Schlüsseltabellen aufzubauen, wenn neue Schlüssel auftauchen. Ich möchte damit sagen, dass es mit dieser oder ähnlichen Vorgehensweisen möglich ist, von der Programmierung neuer Details einer Systemmodellierung wegzukommen und sich mehr in die Richtung der Konfiguration solcher Dinge zu bewegen. Neue abzubildende Konzepte werden also nicht ausschließlich über die Erstellung von Datenbankobjekten gelöst, sondern durch ein Paket von Pflege- beziehungsweise Konfigurationsprogrammen.

2. Ergänzung 03/2005

Ich habe Ihnen bisher das Prinzip gezeigt, solche Schlüssel- oder auch Kategoriebeziehungen aufzubauen. Bei ein wenig weitergehender Betrachtung ist aber festzustellen, dass der innere Aufbau (zumindest der Schlüsselwerttabelle) relativ komplex sein kann. Was ist beispielsweise mit dem Datentyp des Schlüsselwertes, er ist sicherlich nicht durchgehend alphanumerisch wie bei den Länderkennzeichen oder Inhalten für Anreden und Titel. Denken Sie einmal über diese Anforderung nach! Lösungsansätze in diesem Rahmen auszudiskutieren, würde allerdings viel zu weit in die Theorie führen und davon wartet noch genug weiter unten.

Vielleicht warten Sie schon an dieser Stelle auf den Aufbau der Tabellen, also die Definition der einzelnen Spalten. Aber noch sind wir nicht ganz so weit. Betrachten wir noch einmal eine der Abbildungen von oben.



Unternehmensstruktur

Hier empfinde ich eine Entität als störend, und das ist die Entität <Mitarbeiter>. Wir wollen hier zwar nicht eine vollständige Modellierung einer Unternehmensstruktur aufbauen, aber es gibt hier noch eine Möglichkeit, die bisher vorliegende Struktur allgemeiner zu halten. Ein weiteres Prinzip, mit dem Sie Ihre Datenbanken stabiler gegenüber zukünftigen Erweiterungen machen können.

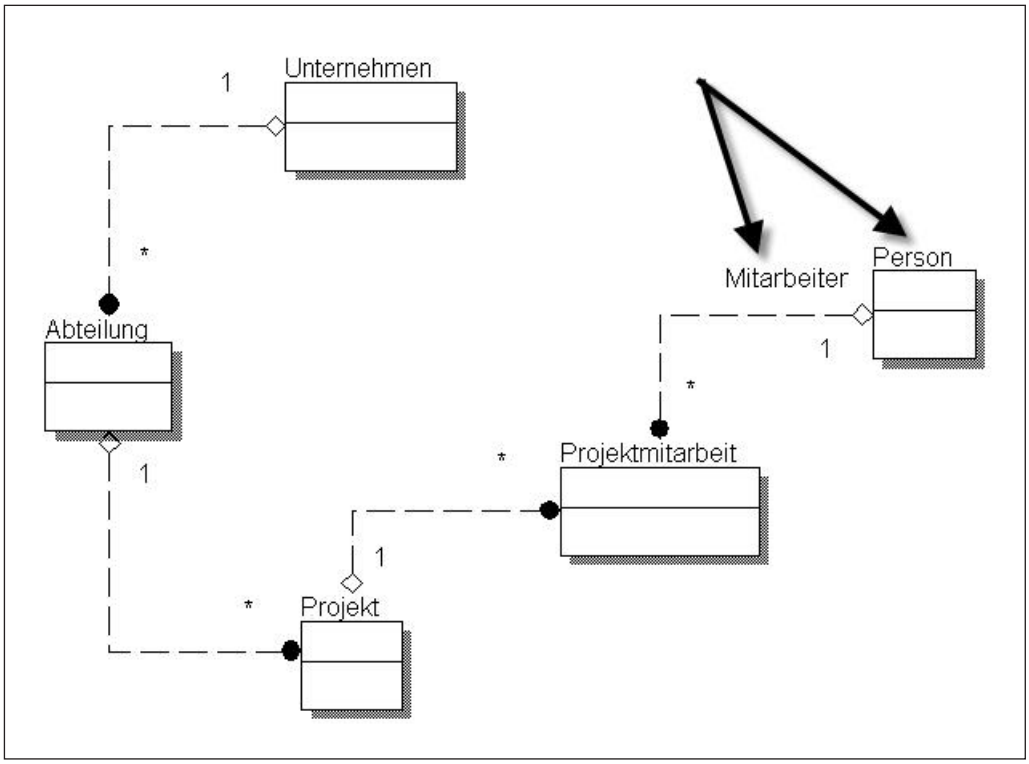
Es wird nicht ausreichen nur von der Entität Mitarbeiter zu reden, früher oder später wird es Entitäten geben, die mit

- Angestellter
- Sachbearbeiter
- Kunde
- Lieferant
- Ansprechpartner
- …

und so weiter und so fort bezeichnet werden. Es ist leicht nachvollziehbar, dass alle diese möglichen Entitäten gewisse Gemeinsamkeiten besitzen. Wenn man im ersten Schritt einmal vernachlässigt, dass für die beiden Entitäten Kunde und Lieferant Firmen stehen können, so sieht man unmittelbar, dass sich hinter diesen Begriffen eigentlich Personen verbergen. Und damit kommen wir zum Begriff der Rolle. Wenn im Sprachgebrauch der Modellbeschreibung Sätze der Form „… ist ein(e)…“ auftauchen, so deutet das auf eine Rolle hin. Dieser Satz ist natürlich nur ein Hinweis und seine Formulierung sollte auch recht flexibel ausgelegt werden. Ich werde es einmal anhand unserer Entität <Mitarbeiter> deutlich machen.

Ich setze als Erstes voraus, das ich keine Entität <Mitarbeiter> erstellen werde, sondern das es eine Entität <Person> geben wird. Aha, aber was ist dann mit dem Mitarbeiter?

Nun, wenn der Mitarbeiter eigentlich eine Person ist, nur eben mit anderen oder zusätzlichen Attributen (Eigenschaften), so sage ich einfach, dass die Person in Bezug auf die Relation <Mitarbeit> die Rolle eines Mitarbeiters spielt. Dieser Sachverhalt stellt sich dann im Diagramm wie folgt dar.



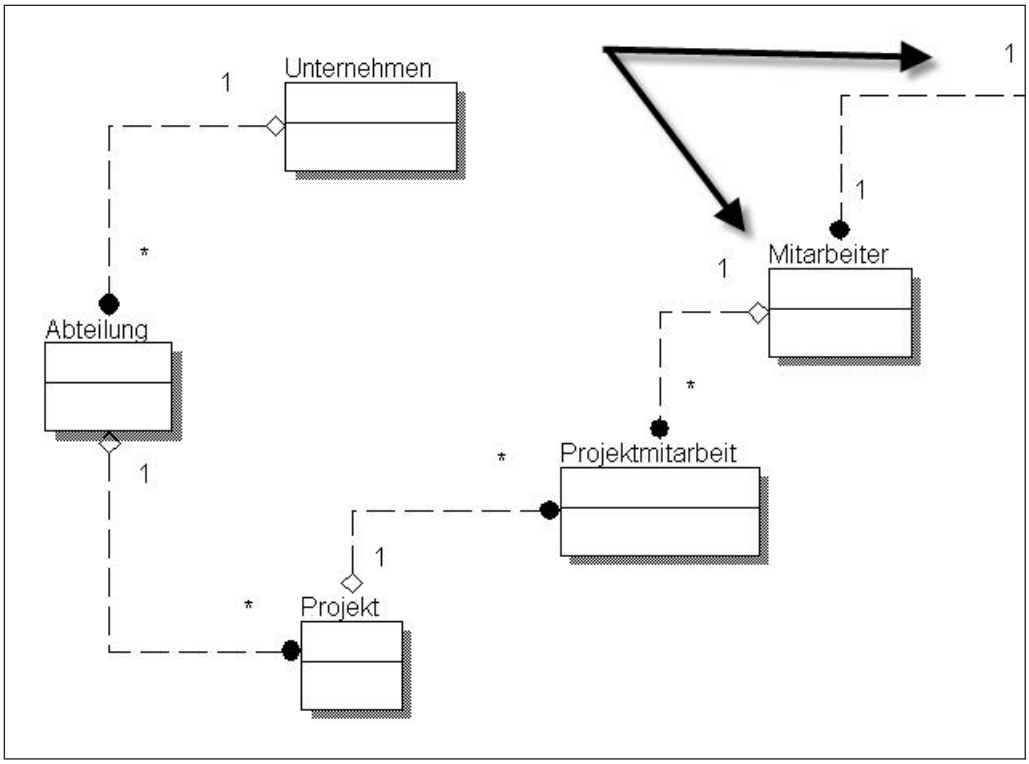
Eine Person und ihre Rolle

Beachten Sie bitte die beiden Pfeile. Ich habe die Entität <Mitarbeiter> durch eine Entität <Person> ersetzt und über die Darstellung der Beziehung „Mitarbeiter“ geschrieben. Diese Lesart bedeutet: Eine Person spielt in Bezug auf <Projektmitarbeit> die Rolle eines Mitarbeiters. Nun gut, bis hier ist das lediglich eine Änderung der Schreibweisen innerhalb des Diagramms. Es ist sozusagen eine andere logische Sicht der Dinge. Das ist nicht nur einfach so dahin gesagt, nein, in der Datenbanktheorie macht man sehr wohl einen Unterschied zwischen einem logischen und einem physikalischen Modell!

Ich möchte aber auch zeigen, wie sich diese Sicht in weiteren Entitäten und/oder Relationen darstellt. Man kann zum Beispiel zwischen <Person> und <Projektmitarbeit> eine weitere Tabelle, ich nenne sie einmal Rollen-Tabelle, schalten.

Es muss nun noch festgestellt werden, welche Kardinalitäten dann vorliegen. Wenn ich von einer Rolle <Mitarbeiter> ausgehe, so sollte diese Rolle nur genau einer Person zugeordnet sein (obwohl, darüber kann man trefflich diskutieren…) und umgekehrt ebenso (auch darüber kann diskutiert werden…).

Dann haben wir folgendes Bild.



Rollenmodellierung (eine Möglichkeit)

Wie man sieht, bleibt die Modellierung der Beziehungen vom Projekt über die Mitarbeit zum Mitarbeiter erhalten, was auch nicht anders erwünscht war. Aber wir haben die Entität <Person> isoliert, wenn man so will.

Hier sollte die Navigation der Beziehungen betrachtet werden oder anders ausgedrückt, welche Tabelle übernimmt die Aufgabe der führenden Tabelle und welche ist die geführte Tabelle.

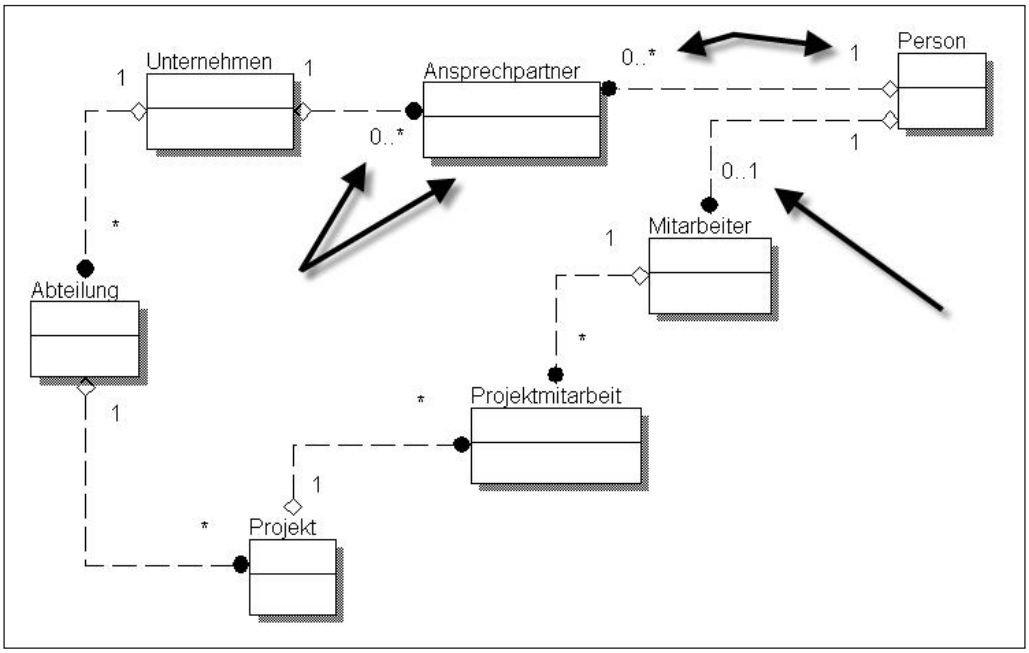
Die Mutter ist hier die Person und nicht der Mitarbeiter. Wenn wir schon einmal von Fremdschlüsseln reden, so wird es in <Mitarbeiter> ein Feld geben, dessen Inhalt auf den Primärschlüssel von <Person> verweist und nicht umgekehrt. Ebenso gibt es in <Projektmitarbeit> einen Verweis auf den Primärschlüssel von <Mitarbeiter>, um die logische Verbindung zwischen Tabellen herzustellen.

Warum ist das so wichtig? Weil jetzt in jeder Entität (auch jeder irgendwann neu hinzugefügten), die Mitarbeiter in der Zuordnung haben soll, einfach ein Verweis auf diesen Mitarbeiter eingebaut werden kann. In einem anderen Fall müsste <Mitarbeiter> um ein Verweisfeld auf die neue Entität erweitert werden, das bedeutet Änderung in bestehenden Tabellen, die höchstwahrscheinlich schon Daten beinhalten.

2. Ergänzung 03/2005

3.5.2
Seite 12

Lassen Sie uns anschließend noch eine kleine Betrachtung zum Vorteil der Rollengestaltung machen. Nehmen wir an, wir möchten jetzt noch im bestehenden Modell die Ansprechpartner des Unternehmens einbauen. Hätten wir nicht die Möglichkeit, hier den Rollengedanken umzusetzen, müssten wir wohl eine weitere Entität <Ansprechpartner> einpflegen. Die Attribute dieser Entität wären aber zum großen Teil mit den Attributen von <Person> übereinstimmend. Also fügen wir eine Rolle <Ansprechpartner> ein, wobei die Rolle von einer <Person> übernommen werden soll.



Die neue Rolle Ansprechpartner

Betrachten wir die Beziehungen. Von der Rolle Ansprechpartner gibt es einen Verweis auf das Unternehmen, in dem die Person diese Rolle spielt. Umgekehrt muss nicht zwingend ein Ansprechpartner dem Unternehmen zugeordnet sein, Kardinalität 0..* beim Ansprechpartner und 1 beim Unternehmen. Ebenso muss eine Zuordnung zu einer Person vorhanden sein, wenn man von einem bestehenden Ansprechpartner ausgeht, Kardinalität 1 bei der Person.

Beachten Sie, das sich die Kardinalität bei Mitarbeiter von 1 zu 0..1 geändert hat. In diesem Kontext würde die Kardinalität 1 ja bedeuten, dass zu jeder Person zwingend eine Rolle als Mitarbeiter existieren muss! Hier liegt aber eine logische ODER-Verwandschaft vor. Entweder die Person besitzt die Rolle Mitarbeiter oder die Rolle Ansprechpartner. Auch die Variante, dass beide Rollen oder keine der beiden in Anspruch genommen werden, ist möglich. Und welche Auswirkungen hätte der Einbau dieser neuen Rolle, wenn die Datenbank in dieser Form bereits existieren würde?

Es wird eine neue Tabelle erstellt, aber die Verweise auf andere Bestandteile der Datenbank werden in diese neue Tabelle eingebaut (Verweisspalten) und nicht in bestehende Tabellen. Die alte Struktur wird nicht berührt!

Auf den ersten Blick scheint diese Art des Modellentwurfs recht aufwendig zu sein. Aber die Erfahrung hat gezeigt, dass sich der Aufwand zu Beginn lohnt, da die Risiken und die Arbeit bei späteren Erweiterungen und Änderungen ganz erheblich reduziert werden kann. Es wird nicht gelingen diesen Spät-Aufwand auf Null zu senken, aber eine Erleichterung in hohem Maß ist allemal realistisch.

Es soll reichen, was die Modellierung eines Systems betrifft. Zumindest bis zu diesem Zeitpunkt. Wir haben auf diesem Weg zum fertigen Entwurf ein wirklich nur kleines Stück des Weges beschritten, der sich noch eine reichliche Strecke weiter durch die hohe Theorie ausweitet. Aber ich denke, dass Sie zumindest einige Ansätze für Ihre Praxis erhalten haben. In den folgenden Absätzen möchte ich nun einige Gedanken zu Schlüsselfeldern und Verweisen mit Ihnen besprechen. Ebenso auch Grundlegendes über die Gestaltung der Struktur von Tabellen.

Was macht eine Entität aus, welchen Hintergrund hat sie? Zusammenfassend kann man sagen, eine Entität stellt ein Konzept aus der realen Welt dar, einen Teil des zu modellierenden Systems. Wie geeignete Kandidaten für Entitäten gefunden werden können habe ich eingangs schon erläutert. So weit wäre eine Entität aber nicht viel mehr als eine leere Hülle. Sie ist erst dann mit Sinn gefüllt, wenn sie Eigenschaften beziehungsweise Attribute erhält. Diese Attribute beschreiben die Entität und sie sollten wirklich spezifisch für sie sein. Nehmen wir als Beispiel die Person. Es ist sicherlich spezifisch für eine Person, dass sie eine Anrede, eventuell einen Titel, Geschlecht und Körpergröße besitzt. Diese Attribute und weitere machen eine Person aus, sind kennzeichnend für sie. Im täglichen Gebrauch einer Datenbank wird im Zusammenhang von Personen auch die Rede von Adressen sein.

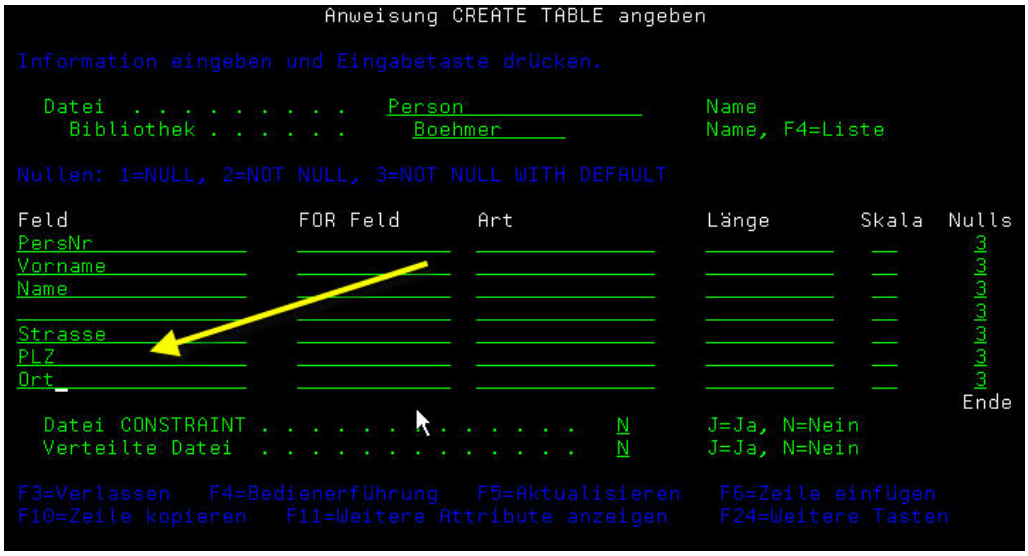
Wenn einer Person nur eine Adresse zugeordnet wird, kann man datenbanktechnisch diese Informationen wohl in einer entsprechend aufgebauten Personentabelle unterbringen. Aber eine Adresse ist kein kennzeichnendes Merkmal wie das Alter oder der Geburtstag. Abgesehen davon wird man einer Person auch mehrere Adressen zuordnen wollen. Vielleicht im Sinne von Wohn-, Rechnungs- und Lieferadresse und so weiter.

Zum einen läuft dieser Weg aber gegen das Prinzip in einer Tabelle, keine sich wiederholenden Daten unterzubringen. Sie kennen das möglicherweise von Spalten wie Name 1, Name 2, Name 3. Irgendwann braucht man Name 4 und schon ist eine bestehende Tabelle zu ändern.

3.5.2
Seite 14

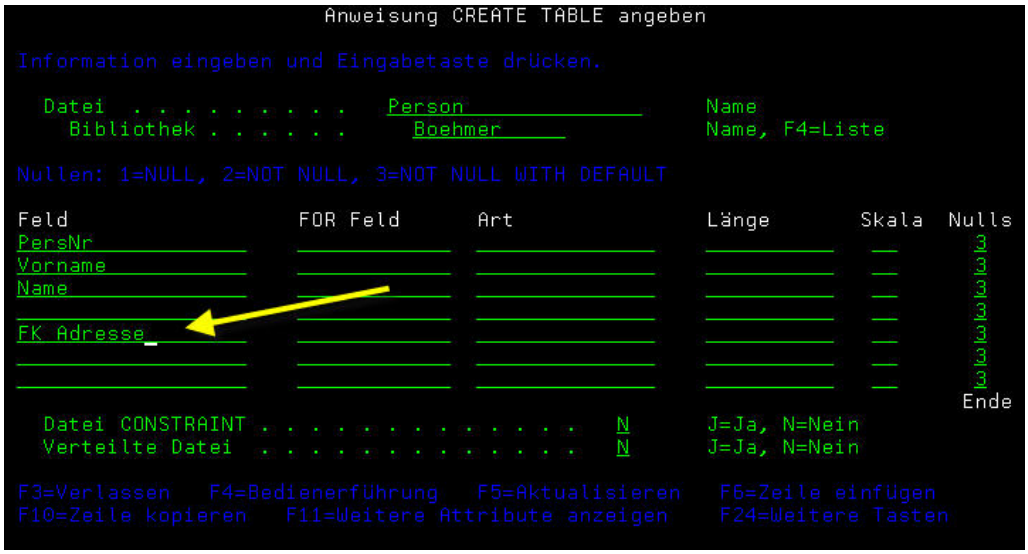
Das ist ein Gesichtspunkt. Der andere ist, dass Adressen nicht nur im Zusammenhang mit Personen sondern auch bei Unternehmen benötigt werden. Auch hier muss dann eine neue Tabelle kreiert werden.

Demzufolge trennen Sie besser die Entitäten deutlich und das gelingt Ihnen besser, wenn Sie darauf achten, wirklich nur die spezifischen Merkmale in eine Entität aufzunehmen.



Person mit eingebundener Adresse

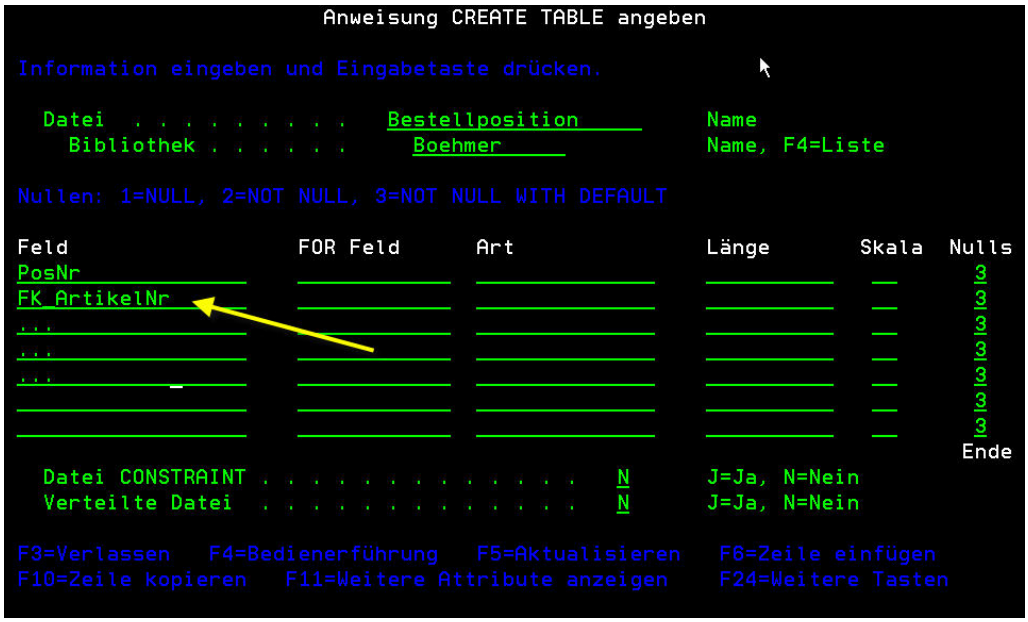
Die Tabelle sollte also nicht wie in der obigen Abbildung aufgebaut werden, sondern eher wie in der folgenden.



Person mit einem Verweis auf Adresse

In der Spalte FK_Adresse ist ein Wert eingetragen, der den Inhalt eines Primärschlüssels in der Adresstabelle darstellt. Damit ist aber noch nicht ein Foreign-Key im Datenbanksinn erstellt, sondern nur eine logische Verbindung!

An dieser Stelle stellt sich auch die Frage, wie kann ein Primärschlüssel gestaltet werden? Bleiben wir bei dem Beispiel unserer Person. Auf den ersten Blick bietet es sich an, als Primärschlüssel eine Personal-Nummer zu verwenden. Nun sei diese Person noch in einer bestimmten Abteilung tätig, also nehmen wir noch die Abteilungs-Kennung mit in den Primärschlüssel. Ähnlich wird recht oft vorgegangen. Denken wir einmal an die Schlüsselung von Artikeln. Da nimmt man teilweise Informationen in die Schlüssel auf, ich denke da an Warengruppen und ähnliches.

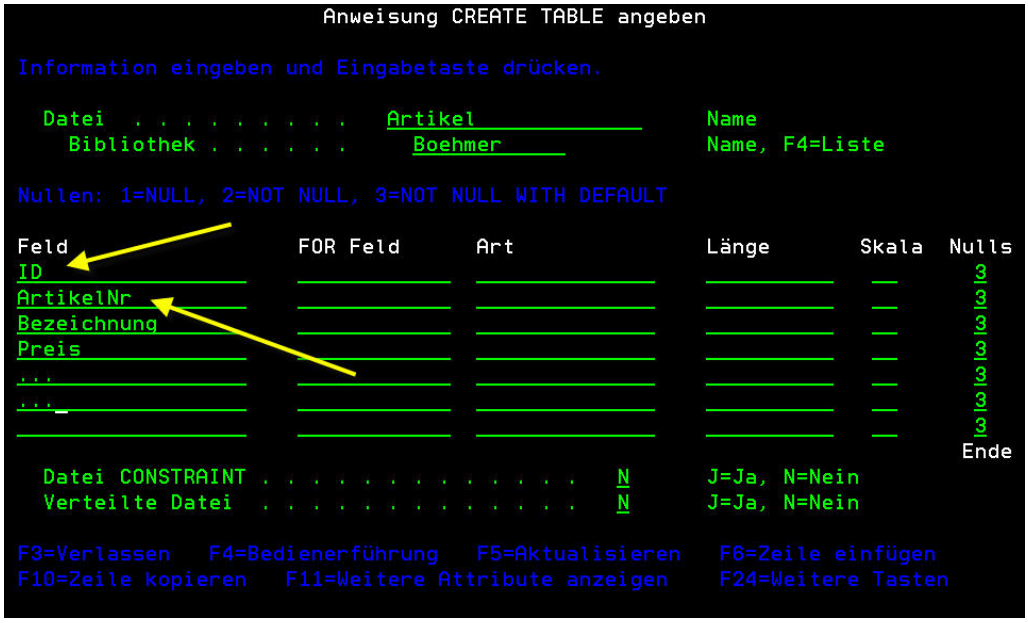


Verweis von einer Bestellung auf Artikel

Hier verweist der Inhalt der Spalte FK_ArtikelNr auf einen Primärschlüssel in der Artikeltabelle, also auf unseren „sprechenden" Schlüssel mit der Warengruppe. Eigentlich nicht gefährlich, sollte man denken.

Nun kommt es aber durchaus vor, dass Schlüsselstrukturen sich ändern. Vielleicht wird intern die Schlüsselung der Warengruppe variiert, der Key wird kürzer oder länger oder noch tragischer, der Datentyp verändert sich. Das hat zur Folge, das sämtliche abhängigen Tabellen untersucht werden müssen und in den dortigen Fremdschlüsselspalten die neuen Primärschlüsselwerte eingetragen werden müssen. Sonst erhalten wir inkonsistente Daten oder sogar Fehler.

3.5.2
Seite 16



Primärschlüssel als laufende Nummer

Vermeidbar ist diese nachteilige Situation dadurch, dass als Primärschlüssel eine Art primitive laufende Nummer eingeführt wird, die für sich allein genommen ohne jede Aussagekraft ist (siehe oberer Pfeil). Die eindeutige Identifizierung des Artikels, die ja nach wie vor nötig ist, erhalten wir durch eine Spalte ArtikelNr (siehe unterer Pfeil), die zugleich als Unique Index angelegt wird.

Verweise von abhängigen Tabellen wie der Bestellposition zielen immer noch auf den Primärschlüssel, das ist nun aber die Spalte ID und nicht die Spalte ArtikelNr!

Wenn sich jetzt etwas in der Struktur der Artikelnummer ändert, ist natürlich immer noch Arbeit da, weil die Inhalte der bestehenden Daten abgeändert werden müssen. Aber die Querverbindungen in der Datenbank bleiben unberührt!

Betrachten wir nun noch einige Möglichkeiten der Implementierung. Für diesen Teil beschränke ich mich auf Mittel der Emulation. Eine Einführung in die Verwendung der graphischen Front_end-Tools folgt noch in einem weiteren Teil.

Die Implementierung

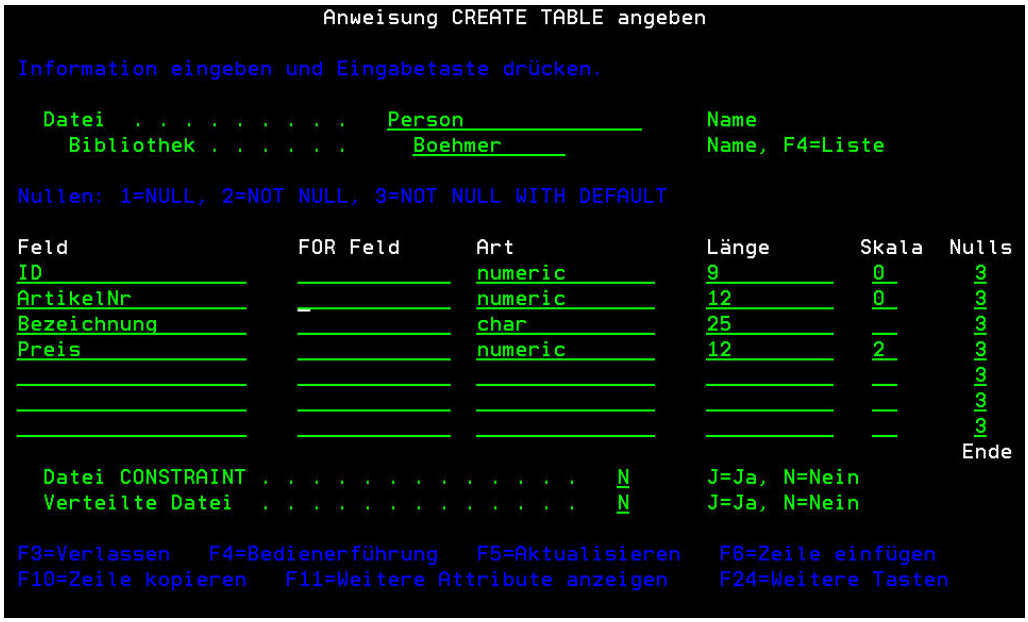
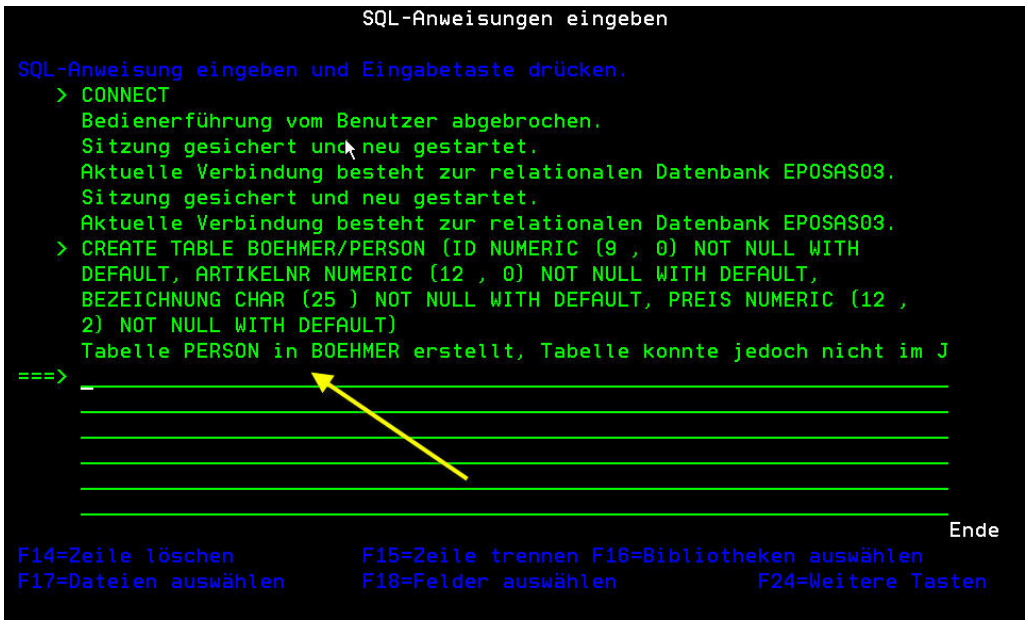


Tabelle anlegen

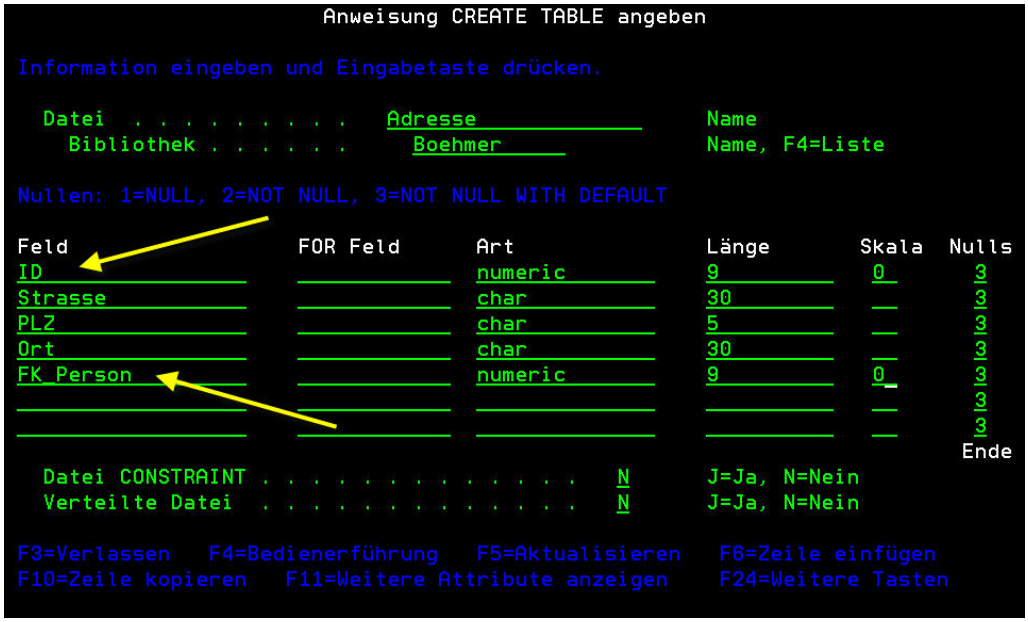
Mit diesem Schritt lege ich eine Tabelle <Person> in meiner Bibliothek an.



Meldung vom System

3.5.3
Seite 2

Einer Person sollen mehrere Adressen zugeordnet werden. Wir benötigen also noch eine Tabelle für Adressen. Die detaillierte Gestaltung der Tabellenspalten will ich hier nicht bis zur letzten Konsequenz ausführen, es geht mir hier mehr um das Werkzeug. Hier also ein möglicher Aufbau der Tabelle, wichtig ist dabei die Spalte FK_Person, die in einem weiteren Schritt als Wurzel eines Foreign-Keys dienen wird.



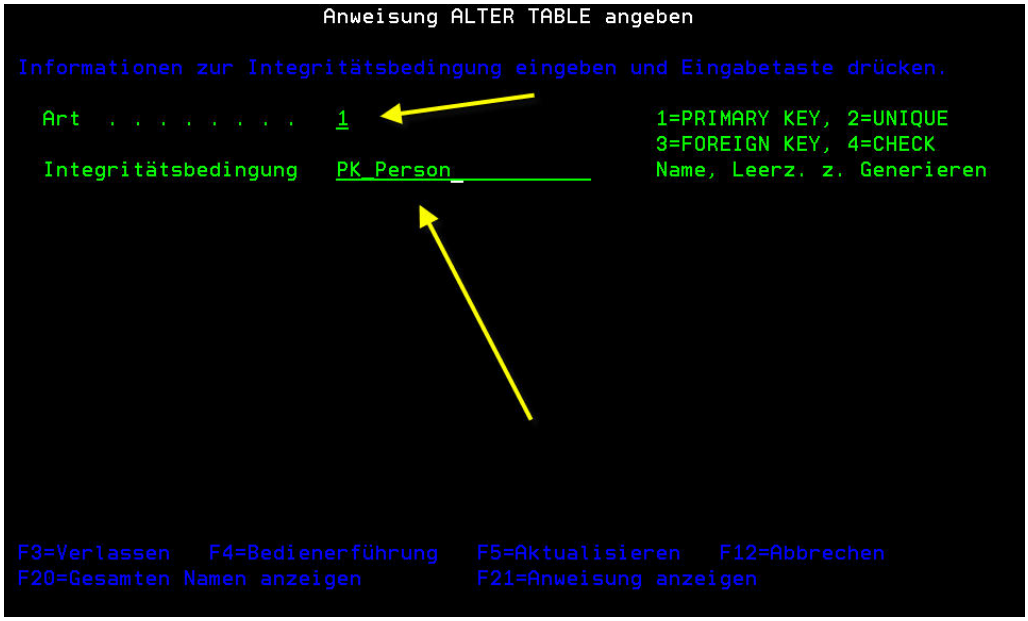
Die Tabelle Adresse

Auch in dieser Tabelle finden wir eine laufende Nummer als Primärschlüssel, der als solcher allerdings erst noch angelegt werden muss (oberer Pfeil).

Legen wir also im folgenden Schritt die Primärschlüssel sowohl für die Tabelle Person als auch für die Tabelle Adresse an. Dafür geben wir in der interaktiven SQL-Umgebung den Befehl „Alter Table“ ein und prompten ihn.

Alter Table

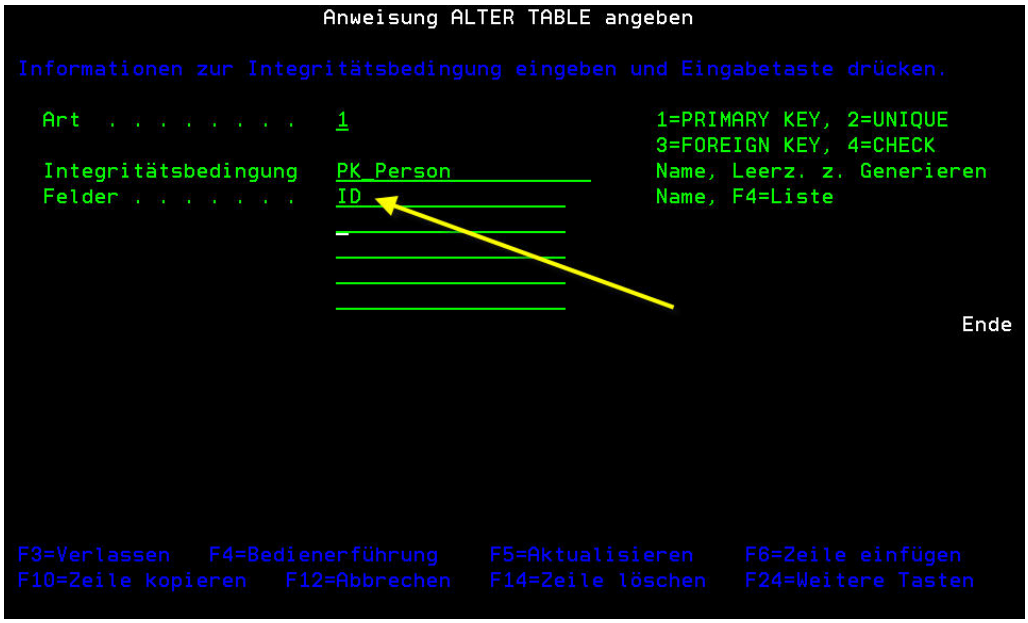
Hier wird die Art der hinzuzufügenden Integritätsbedingung angegeben und dem zukünftigen Primärschlüssel auch ein Name zugewiesen. Sonst legt das System einen generierten Schlüssel an. Wenn jedoch ein aussagefähiger Name vorhanden ist, haben Sie es später leichter, eventuelle Meldungen zu interpretieren.



Prompt des Befehls

3.5.3
Seite 4

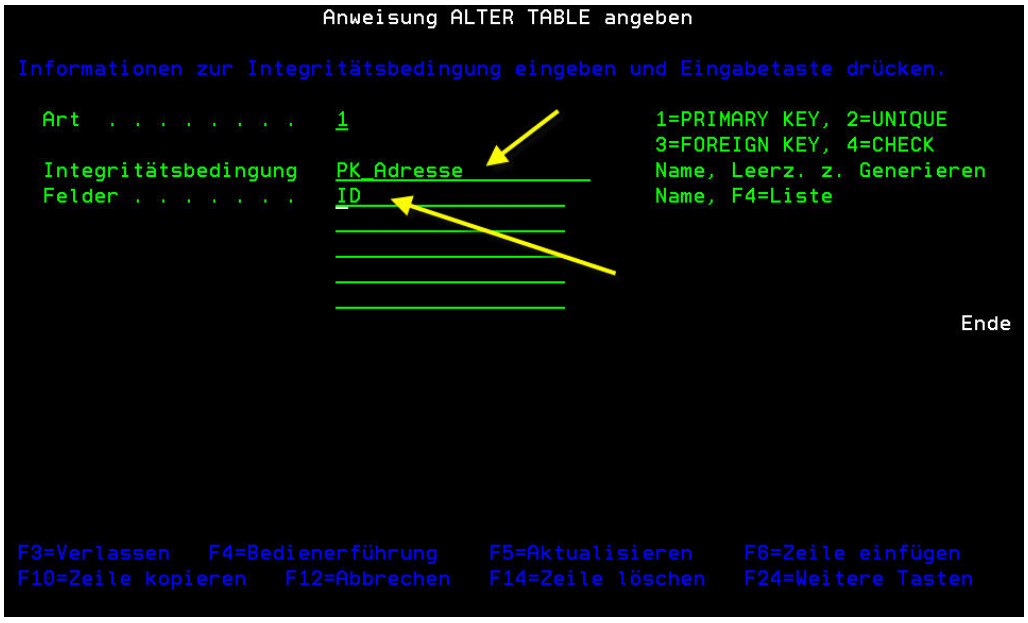
Nach der Datenfreigabe geben Sie noch das Feld an, das zum Primärschlüssel gehören soll.



Feldangabe

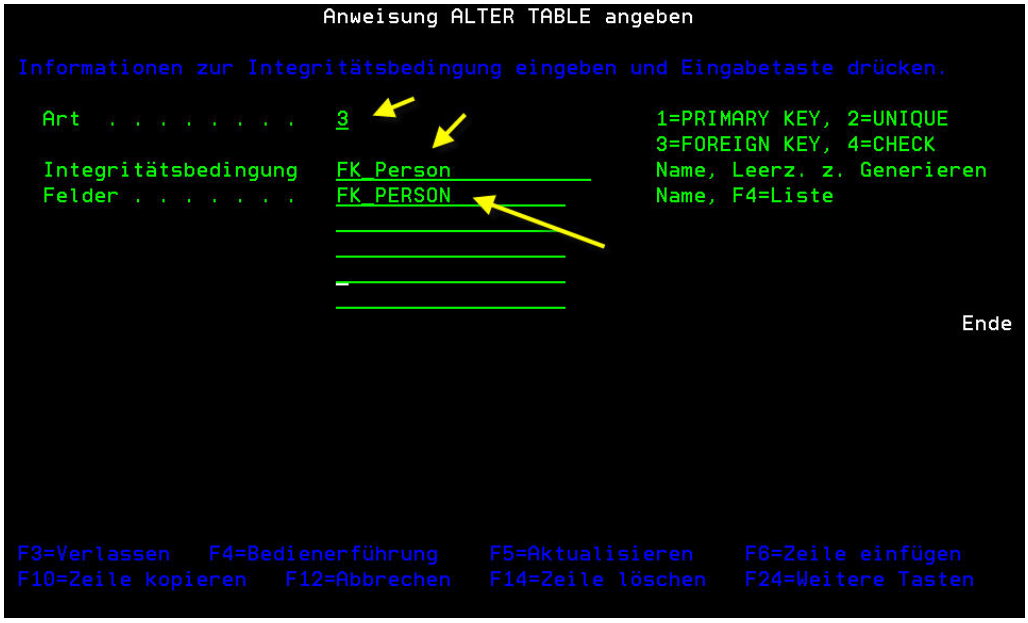
Nach dieser Eingabe drücken Sie Datenfreigabe und der Primärschlüssel wird angelegt. Er ist jetzt ein eigenständiges Objekt in der Datenbank.

Wir legen auch für die Tabelle Adresse einen Primärschlüssel in dieser Art an. Auch in dieser Tabelle beziehen wir uns dabei auf die Spalte ID. Dieser Name für eine Primärschlüsselspalte kann auch zum Standard werden, anschließend muss man nicht mehr überlegen, wie der PK einer Tabelle benannt ist.



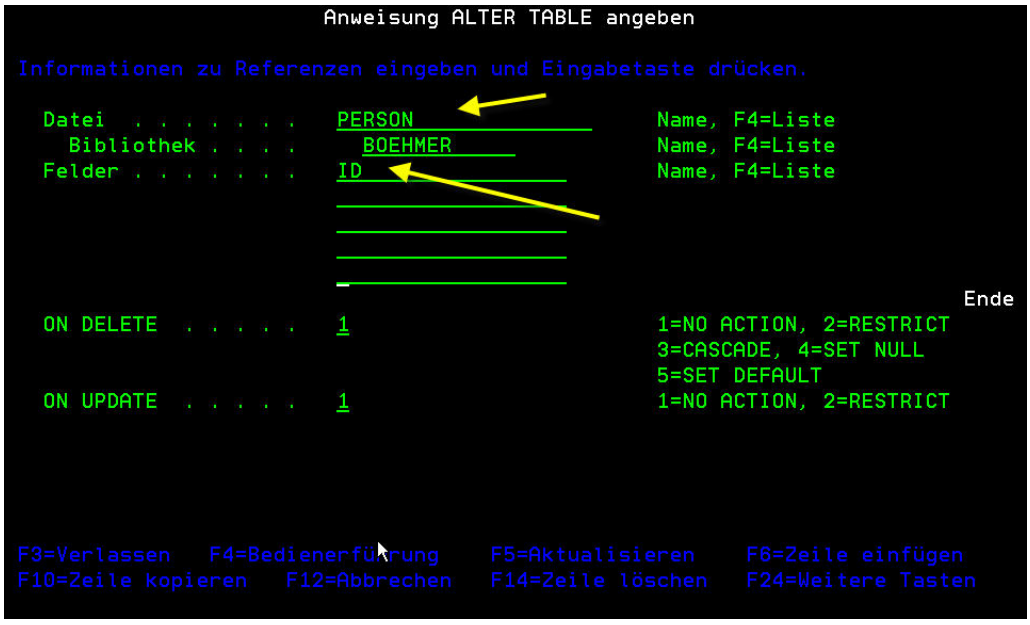
Primärschlüssel für Adresse

Die Primärschlüssel für beide Tabellen sind angelegt. Jetzt wird noch ein Fremdschlüssel ausgehend von Adresse mit Ziel Person benötigt. Dafür wird wieder die Anweisung „Alter Table“ eingegeben. Über die Auswahl 3 wird definiert, dass ein Foreign-Key angelegt werden soll. Der FK bekommt wie ein Primärschlüssel einen Namen und ich definiere, von welcher Spalte der Verweis ausgehen soll.



Anlegen eines Foreign-Keys

Im Anschluss an dieses Bild muss noch angegeben werden, auf welche Tabelle der Fremdschlüssel verweisen soll und welches Feld den Primärschlüssel bildet.



Angabe der Zieltabelle

3.5.3
Seite 6

```
    > ALTER TABLE BOEHMER/ADRESSE ADD CONSTRAINT FK_PERSON FOREIGN KEY
      (FK_PERSON) REFERENCES BOEHMER/PERSON (ID) ON DELETE NO ACTION ON
      UPDATE NO ACTION
      ALTER für Tabelle ADRESSE in BOEHMER beendet.
===>
                                                                        Ende
F14=Zeile löschen          F15=Zeile trennen F16=Bibliotheken auswählen
F17=Dateien auswählen      F18=Felder auswählen          F24=Weitere Tasten
```

Systemmeldung über erfolgte Änderung

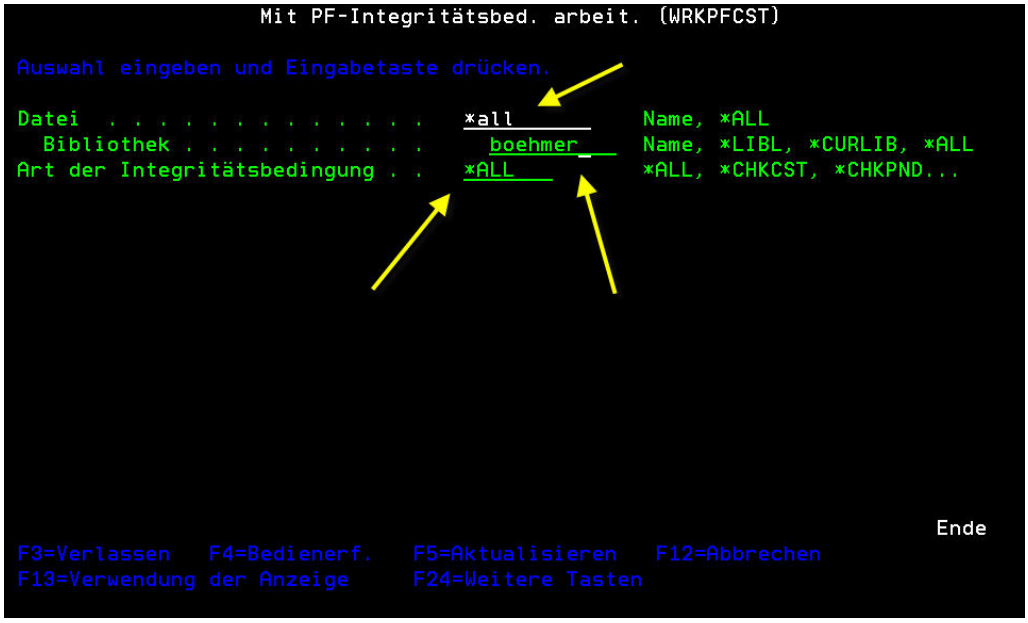
Der Fremdschlüssel wurde erfolgreich angelegt. Abgesehen von der Eingabe über die interaktive SQL-Umgebung besteht auch die Möglichkeit, mit einigen Befehlen aus der Kommandozeile heraus zu arbeiten.

Geben Sie einmal den Befehl WrkPFCst ein und prompten Sie ihn.

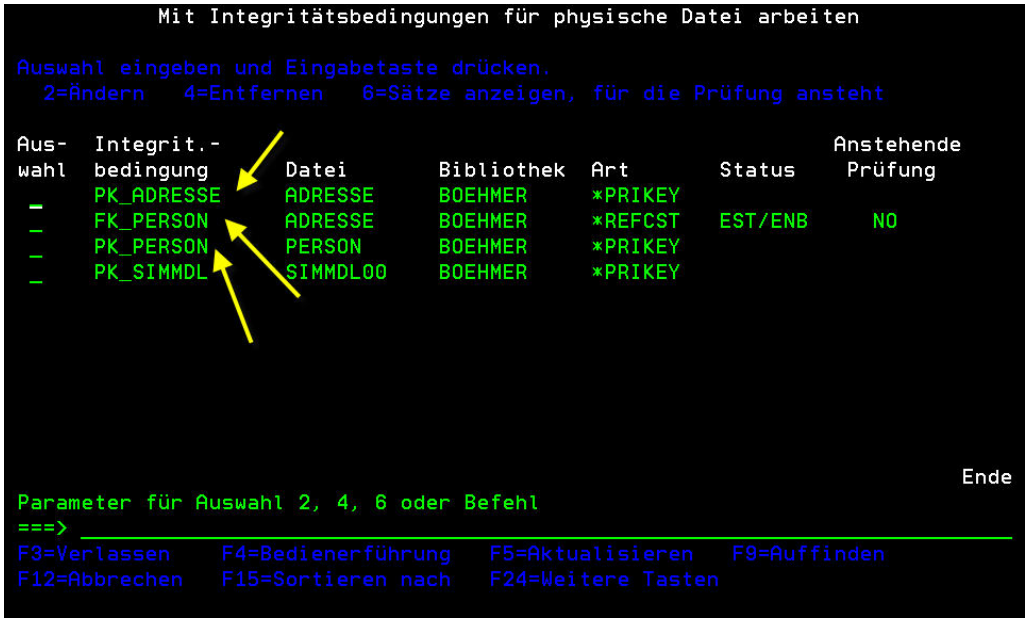
```
MAIN                          OS/400-Hauptmenü
                                                    System:    EPOSAS03
Auswahlmöglichkeiten:

     1. Benutzeraufgaben
     2. Büroaufgaben
     3. Allgemeine Systemaufgaben
     4. Dateien, Bibliotheken und Ordner
     5. Programmierung
     6. Datenfernverarbeitung
     7. System definieren oder ändern
     8. Problembehandlung
     9. Menü anzeigen
    10. Unterstützende Informationen - Auswahlmöglichkeiten
    11. Client Access/400-Aufgaben

    90. Abmelden

Auswahl oder Befehl
===> wrkpfcst
F3=Verlassen   F4=Bedienerführung   F9=Auffinden   F12=Abbrechen
F13=Unterstützende Informationen    F23=Anfangsmenü festlegen
```

WrkPFCst in der Befehlszeile

Geben Sie an, dass Sie mit allen Integritätsbedingungen aller Dateien in der Bibliothek arbeiten wollen.



Nach dem Prompt mit F4



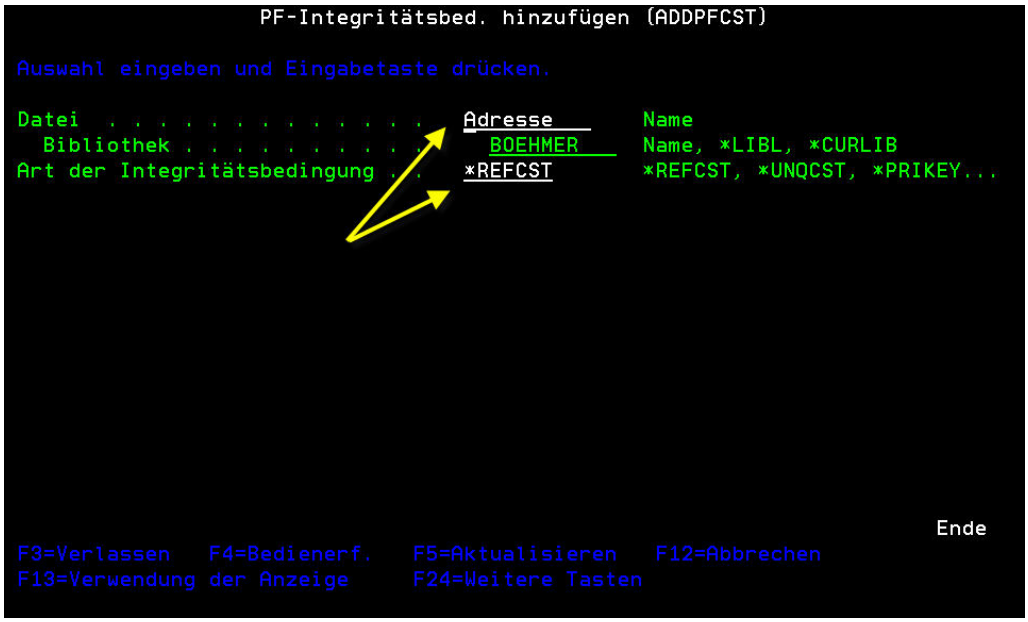
Vorhandene Integritätsbedingungen

3.5.3
Seite 8

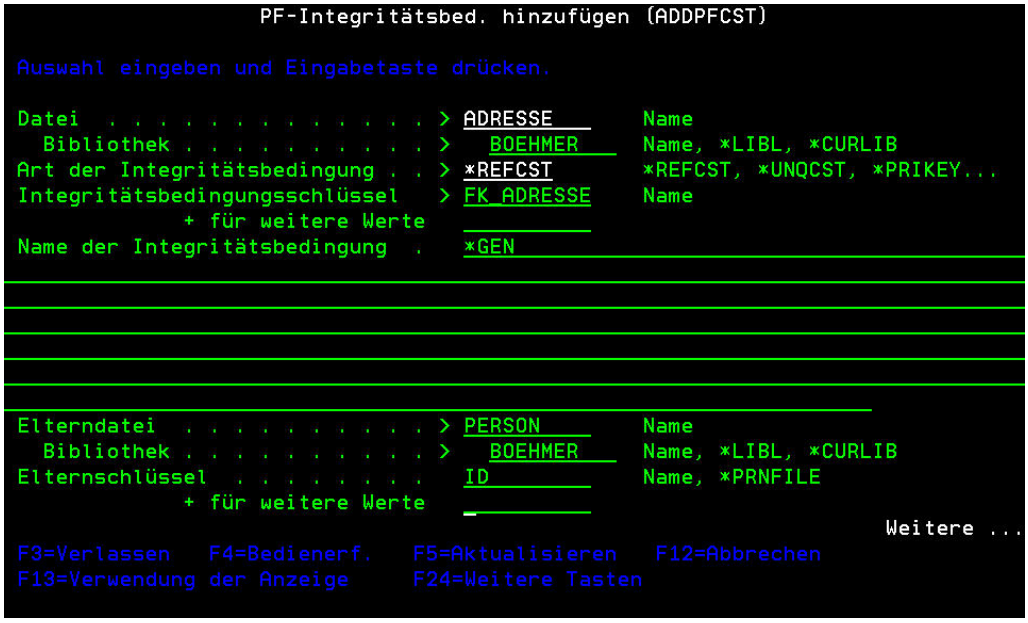
Die mit Pfeilen markierten Integritätsbedingungen habe ich in den vorhergehenden Schritten angelegt. Die Art *PRIKEY kennzeichnet einen Primärschlüssel, die Art *REFCST einen Fremdschlüssel. Bei dem Foreign-Key FK_Person sehen wir noch den Status EST/ENB. Dies bedeutet, dass der Verweis eingerichtet (EST) ist und auch aktiviert wurde (ENB). Ab sofort wird die Datenbank prüfen, ob die Zuordnung von Adress-Sätzen zu Personen korrekt ist. Es ist nicht mehr möglich, mit irgendwelchen Tools oder anderen Mitteln Sätze in der Tabelle Adresse anzulegen, wenn kein passender Satz in Person vorhanden ist. Die Validierungsaufgabe muss nun nicht mehr von einer Software übernommen werden, sondern obliegt der Datenbank!

Das Anlegen und Entfernen von Integritätsbedingungen können Sie von der Befehlszeile aus auch mit den Befehlen AddPFCst (Hinzufügen) und RmvPFCst (Entfernen) erledigen. Die Prompts zu diesen Befehlen sind selbsterklärend.

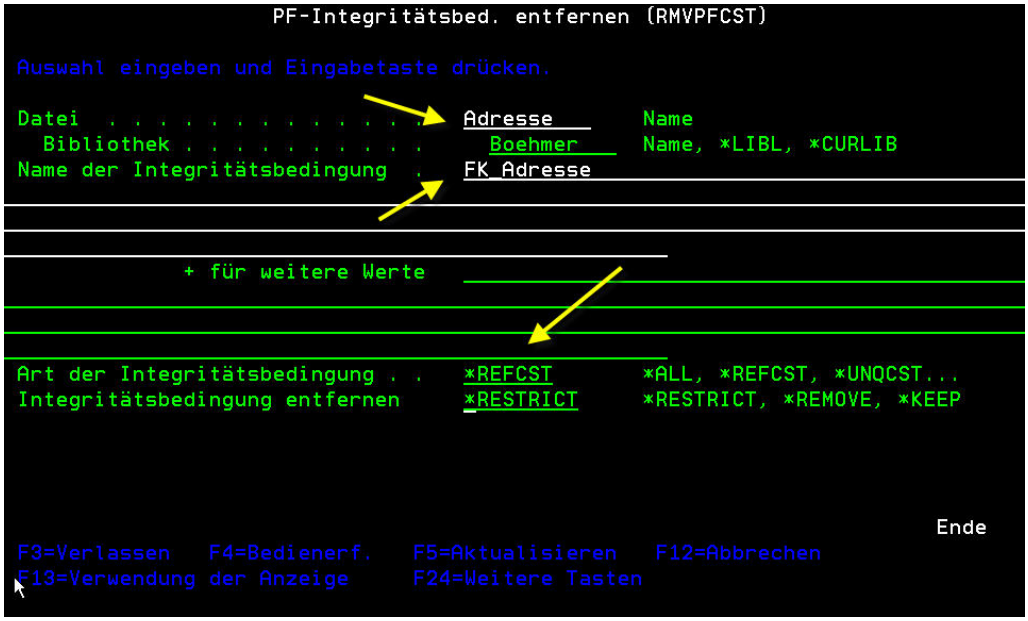
Nun mag man auf die Idee verfallen, bei bestehenden Dateisystemen (ich sage hier bewusst Dateisystem…) nachträglich solche Integritätsbedingungen nachzupflegen. Ist das denn möglich? Ja, diese Aktionen sind möglich. Aber was passiert, wenn bereits inkonsistente Datenbestände vorliegen? In diesen Fällen wird Arbeit auf Sie warten, denn wie gesagt, die Datenbank lässt keine inkonsistenten Daten in dieser Form zu.



Der Befehl ADDPFCST



Die ausgefüllten Parameter



Der Befehl RMVCST

3.5.3
Seite 10

Datenbankimplementation mit modernen Front-End-Tools

Mit der Umgebung für die interaktive Eingabe von SQL-Statements wird dem Anwender schon ein recht praktisches Werkzeug in die Hand gegeben. Die Umsetzung des Datenbankentwurfs in eine richtige Datenbank kann damit vorgenommen werden. Jedoch ist man durch den Einstieg über eine 5250-Emulation natürlich an die Einschränkungen dieser Oberfläche gebunden.

Wesentlich mehr Komfort und damit letztlich auch mehr Effizienz bei der Umsetzung bieten jedoch Funktionalitäten, die Bestandteil des WDSC sind. Auch der iSeries-Navigator bietet recht gute Möglichkeiten, mit Datenbanken zu arbeiten.

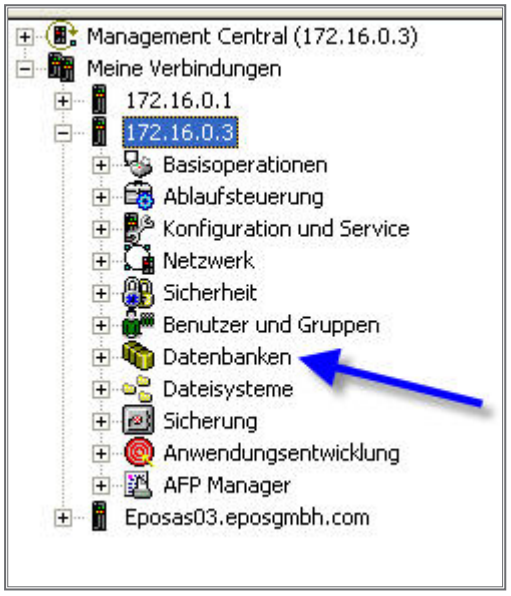
Auf den folgenden Seiten werde ich Ihnen Wege zeigen, wie sowohl mit dem Navigator als auch mit den Daten-Projekten des WDSC mit den einzelnen Objekten einer Datenbank gearbeitet werden kann. Diese Werkzeuge liefern aus meiner Sicht mehr als ausreichende Funktionen im Zusammenhang mit Datenbanken.

Ich möchte mit dem iSeries-Navigator beginnen. Da es mir in diesem Kapitel darauf ankommt, Ihnen verschiedene Werkzeuge vorzustellen und auf die einzelnen Schritte auf dem Weg zur implementierten Datenbank aufzuzeigen, werde ich mich bei der Erläuterung des Entwurfs zurückhalten. Hier verweise ich Sie auf die ersten Kapitel im Abschnitt Datenbanken. Also machen wir uns auf den Weg!

Starten Sie den iSeries-Navigator

3.5.4
Seite 2

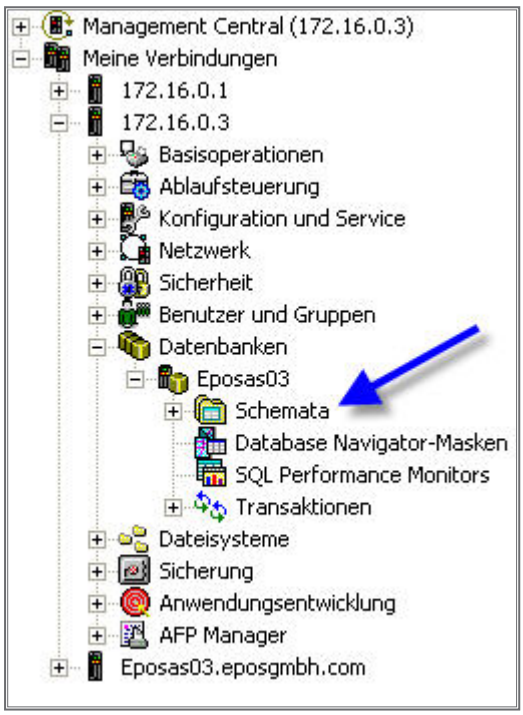
Erweitern Sie eine Ihrer Verbindungen oder erstellen Sie eine, wenn Sie im Navigator noch kein Endpunktsystem angelegt haben.



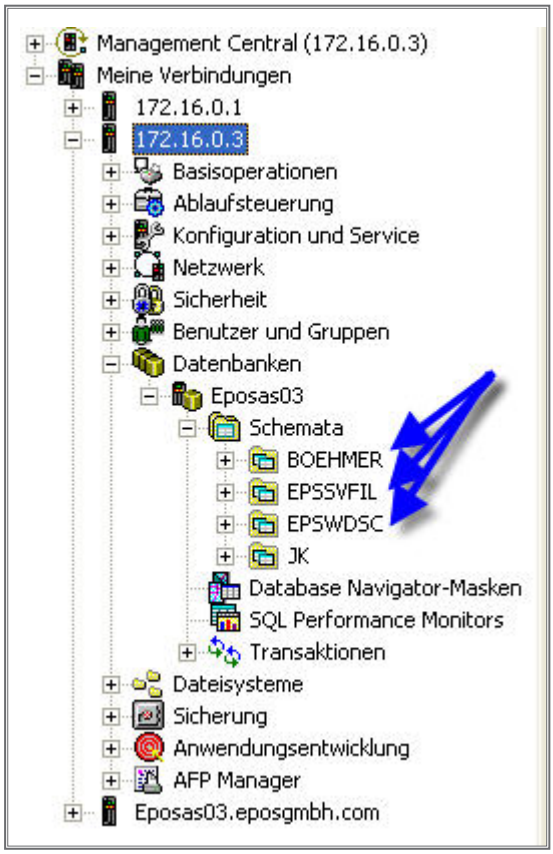
Eine expandierte Verbindung

Die Ansichten, die hier wiedergegeben sind, erhalten Sie, wenn Sie den iSeries-Navigator in der abgebildeten Version verwenden. Bei anderen Versionen kann das Bild leicht variieren.

Im Anschluss öffnen Sie den Unterpunkt <Datenbanken> und erhalten die anschließende Darstellung.



Öffnen Sie nun bitte <Schemata>. Die hier erscheinenden Unterpunkte werden Sie eventuell als eine Art Bibliotheksliste interpretieren. Ganz falsch liegen Sie damit nicht.

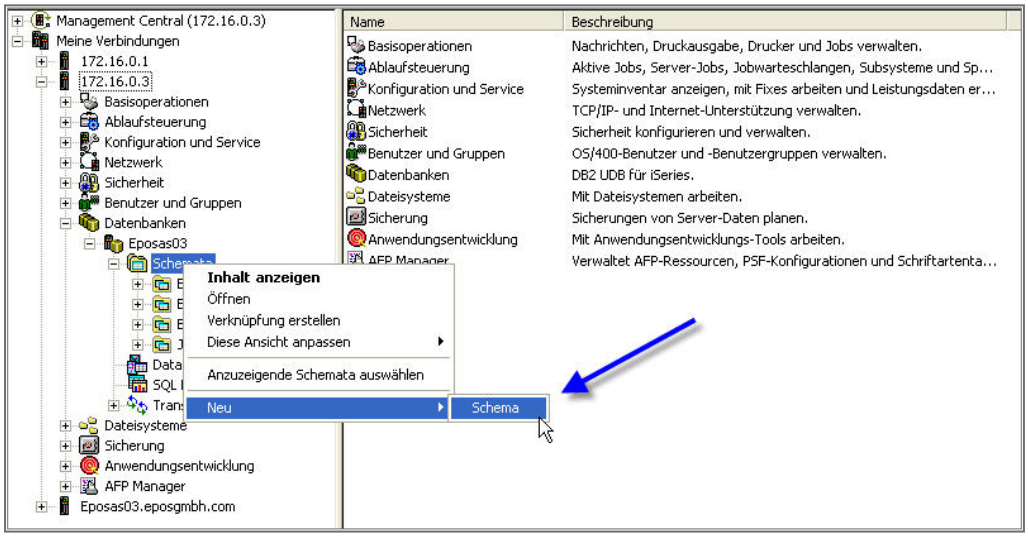


Schemata, die zur Zeit gelistet werden.

Unter den Namen, die in dieser Teilauflistung erscheinen, finde ich auf der entsprechenden Maschine natürlich Bibliotheken gleicher Bezeichnung. Die einzelnen Benennungen in diesem Bereich des iSeries-Navigators sind dem allgemein üblichen Sprachgebrauch im Zusammenhang mit Datenbanken angepasst worden. Lassen Sie sich also bitte nicht verwirren!

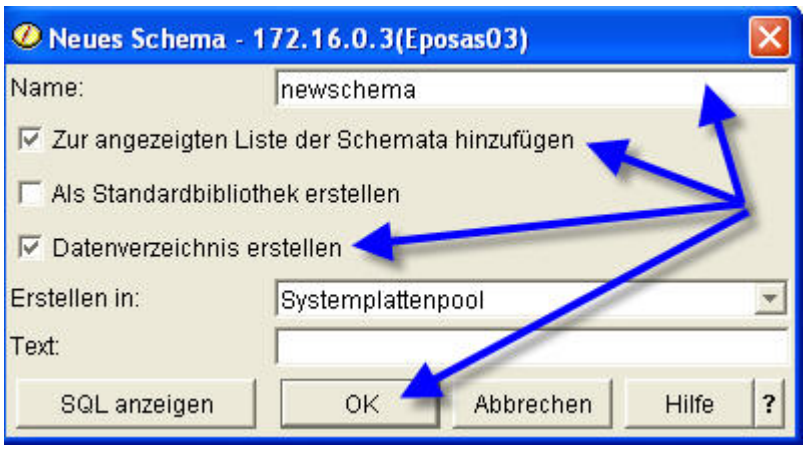
Verfolgen Sie nun bitte die einzelnen Schritte, die zur Erstellung eines neuen Schemas notwendig sind. Vielleicht denken Sie an dieser Stelle auch, dass der Autor doch eigentlich eine Datenbank erstellen wollte… Bitte bedenken Sie noch einmal: die eigentliche Datenbank ist das System und unsere neu zu erstellenden Objekte werden in einem neuen Schema untergebracht. „Grün“ gesprochen, würden wir eine neue Bibliothek erstellen und anschließend die physischen und logischen Dateien. Hier sollten wir aber besser die Datenbank-Terminologie verwenden.

Nach einem Klick mit der rechten Maustaste auf <Schemata> öffnet sich ein Kontextmenu. Selektieren Sie nun die Option <Neu>/<Schema>.



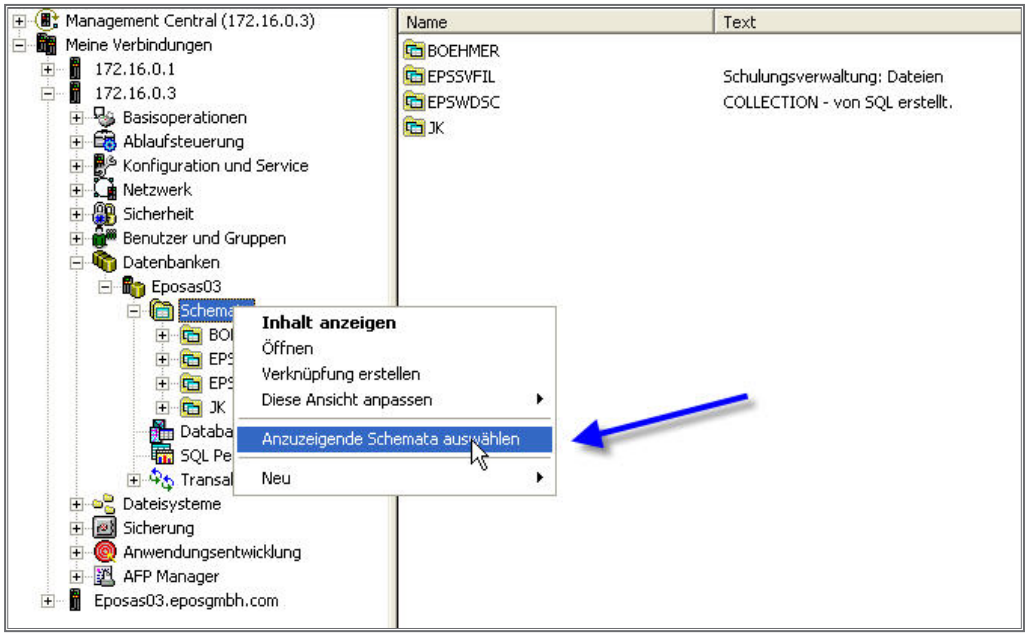
Im folgenden Dialog definieren Sie das neue Schema. Analog zur Erstellung einer neuen Bibliothek muss der gewünschte Name des Schemas auf dem System eindeutig sein. Aktivieren Sie die Check-Box, die das neue Schema in die angezeigte Liste übernimmt. Ein Datenverzeichnis sollte ebenfalls erstellt werden, auch hier aktivieren Sie bitte die entsprechende Check-Box.

Bestätigen Sie die Angaben bitte mit dem OK-Button.



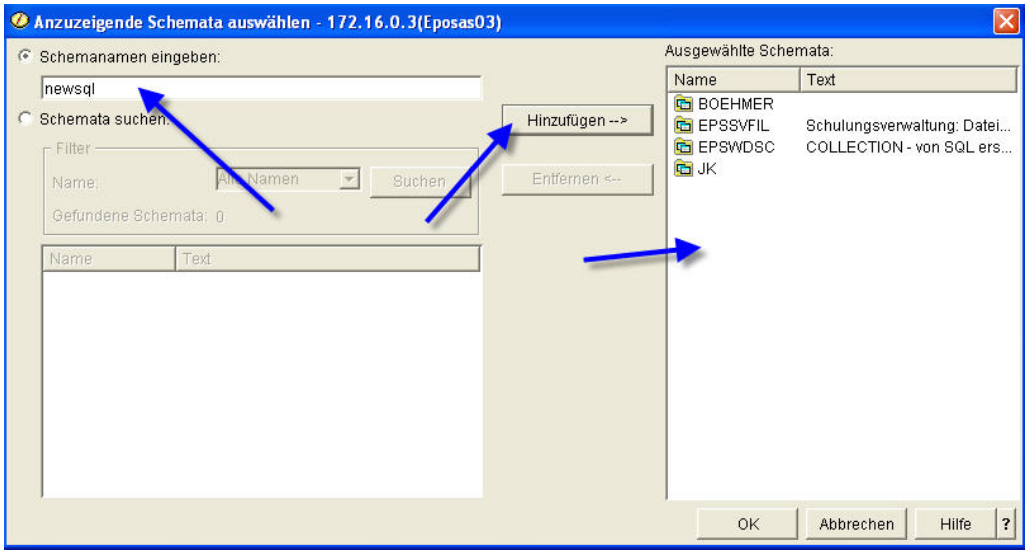
Dialog zur Definition eines neuen Schemas.

Sollte Ihr Schema danach nicht in der Schemaliste erscheinen, klicken Sie bitte noch einmal mit der rechten Maustaste auf <Schemata>. Im Kontextmenu selektieren Sie <Anzuzeigende Schemata auswählen>.

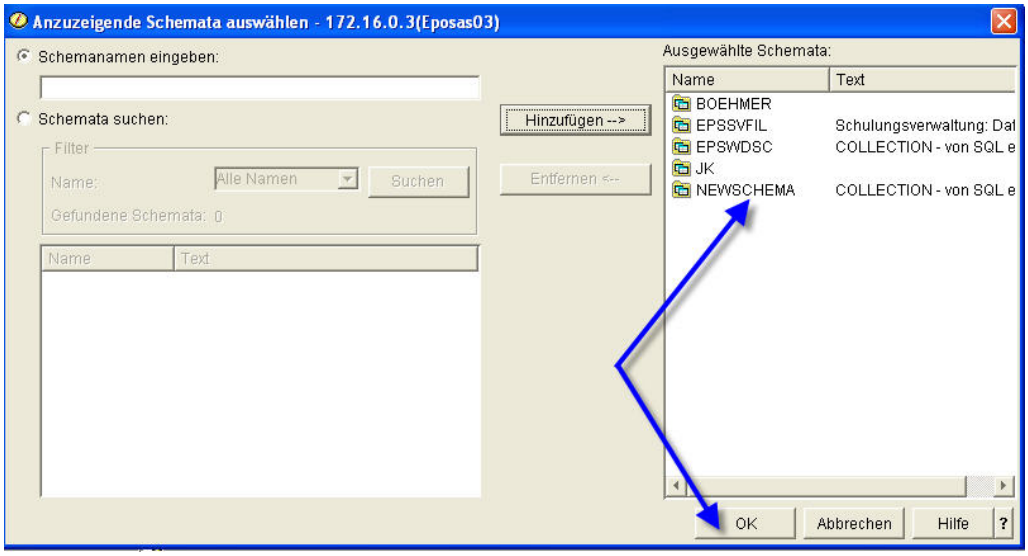


Liste der Schemata bearbeiten

Unter <Schemaname> tragen Sie die Bezeichnung Ihres Schemas ein oder suchen Sie optional, indem Sie den unteren Radiobutton aktivieren. Mit einem Klick auf <Hinzufügen> wird das Schema in den Bereich der ausgewählten Schemata übernommen. Anschließend klicken Sie wieder auf <OK>, damit die Änderungen übernommen werden.

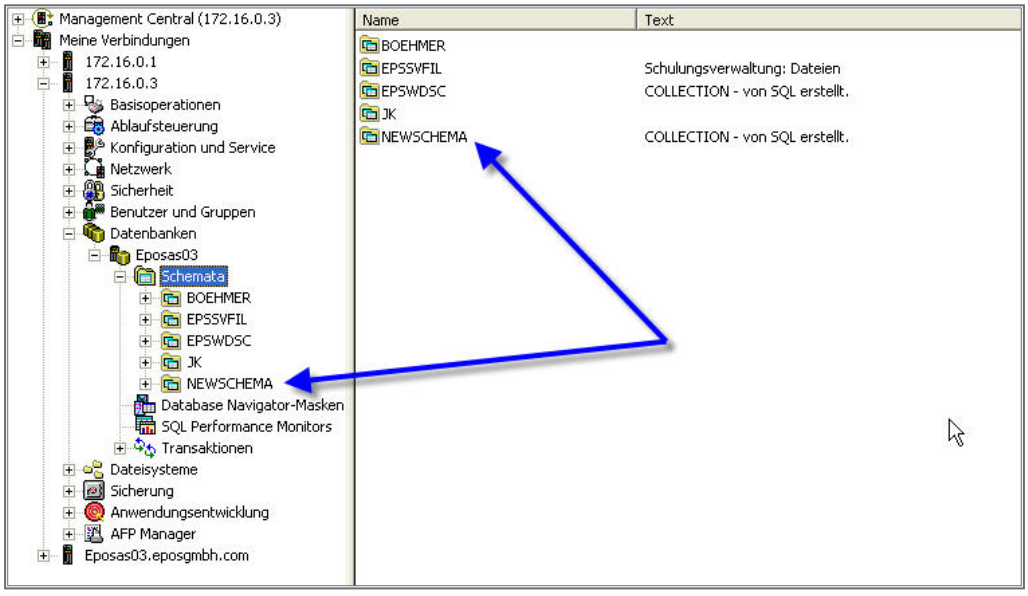


Neues Schema übernehmen



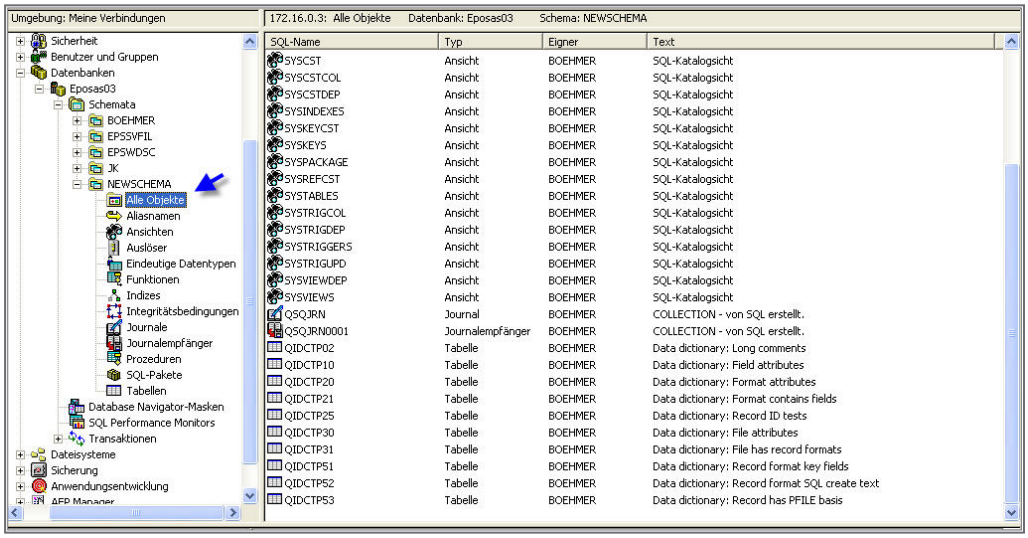
Nach der Übernahme durch <Hinzufügen>

Nach diesen Aktionen sollte Ihr Schema in der Liste mit aufgeführt sein.



Geänderte Schemaliste

Bitte expandieren Sie das von Ihnen erstellte Schema und markieren dann den Punkt <alle Objekte>. Wenn Sie an die Erstellung einer Collection weiter oben im Datenbankteil dieses Buches denken, werden Sie sicherlich nicht sehr überrascht sein, in Ihrem Schema bereits allerlei Objekte zu finden…



Ein neues Schema…

Damit sind wir nun in der Lage, unser Schema mit Leben zu füllen, sprich Tabellen, Sichten, Schlüssel und so weiter können angelegt werden. Wie versprochen werde ich Sie nicht mit dem Entwurf einer neuen Datenbank aufhalten, sondern ich beschränke mich auf ein Minimalsystem. Folgender Gedanke liegt dahinter: Wir legen Rechnungen in der Form ab, dass eine der berüchtigten (Rechnungs)-Kopfdateien erstellt wird. Die einzelnen Positionen einer Rechnung legen wir in der zugehörigen Positionstabelle ab. Im Anschluss daran legen wir noch für jede der beiden Tabellen einen Primärschlüssel an und die Positionstabelle wird über einen Fremdschlüssel mit der Rechnungskopfdatei verbunden.

Legen wir fest, welche Informationen in der Rechnungskopfdatei und in der Positionsdatei enthalten sein sollen.

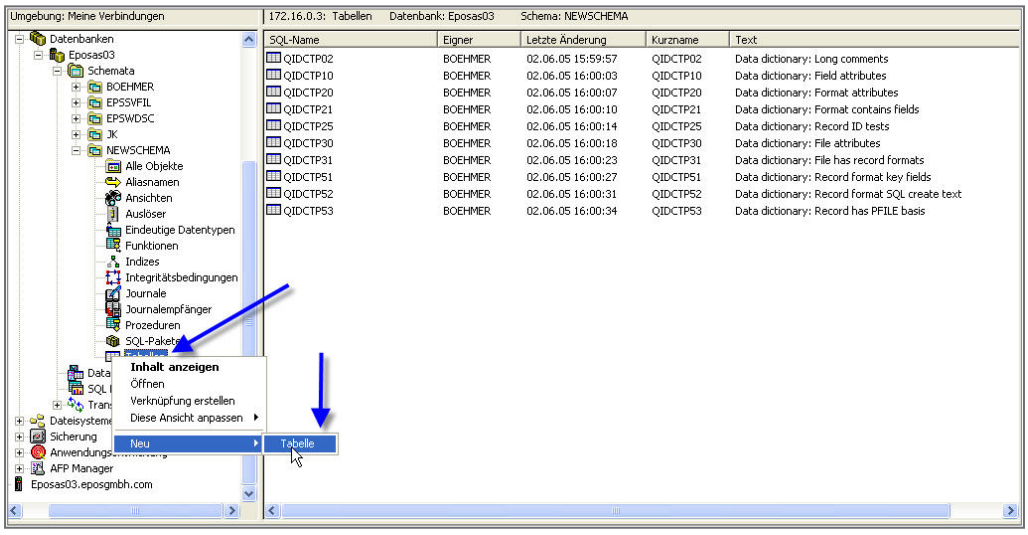
Aufbau der Kopfdatei…:

- ID, 9-Stellen, numerisch und Primärschlüssel
- RECHNR, Rechnungsnummer, 9-Stellen, numerisch und eindeutig
- RECHDAT, Rechnungsdatum, Date

Aufbau der Positionsdatei…:

- ID, 9-Stellen, numerisch und Primärschlüssel
- RECHNR, 9-Stellen numerisch, dient als Zuordnungsfeld zum Rechnungskopf
- SUMME, 12-Stellen, numerisch mit zwei Nachkommastellen

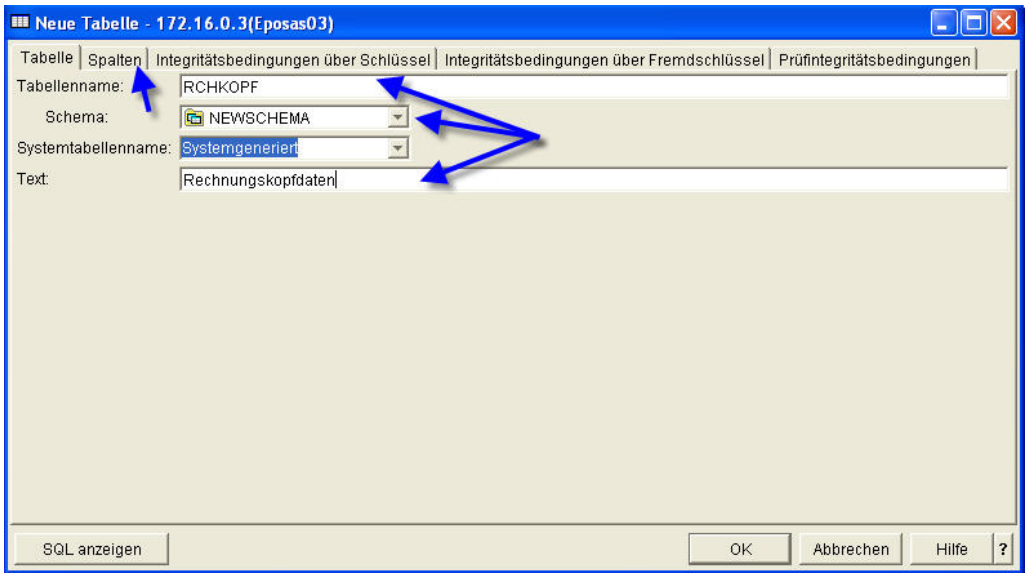
Im ersten Schritt soll die Kopfdatei erstellt werden. Dazu klicken Sie einfach den Unterpunkt <Tabellen> mit der rechten Maustaste an. Im folgenden Kontextmenu selektieren Sie die Option <neu> / <Tabelle>.



Anlegen einer neuen Tabelle

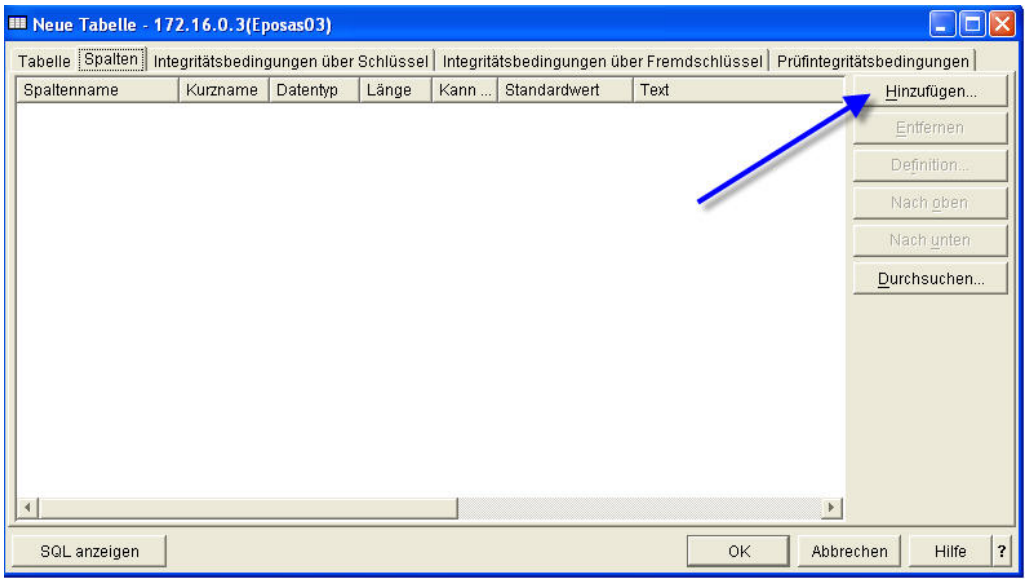
3.5.4
Seite 10

Der folgende Dialog gibt uns die Möglichkeit, die zu erstellende Tabelle zu definieren. Geben Sie hier den gewünschten Tabellennamen und das Schema an, in dem die Tabelle erstellt werden soll. Optional können Sie auch einen die Tabelle beschreibenden Text angeben. Wenn die Eintragungen vorgenommen worden sind, klicken Sie den Reiter <Spalten> an.



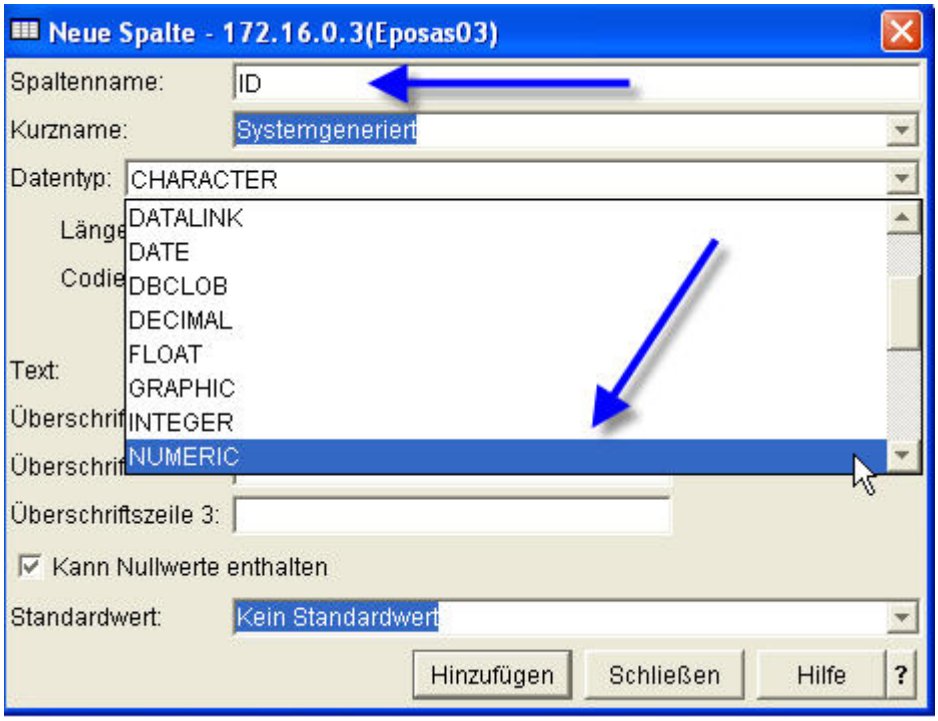
Tabellenname und Ablageschema angeben

In diesem Bereich können Sie über den Button <Hinzufügen> die einzelnen Spalten definieren.



Reiter für die Definition der Spalten

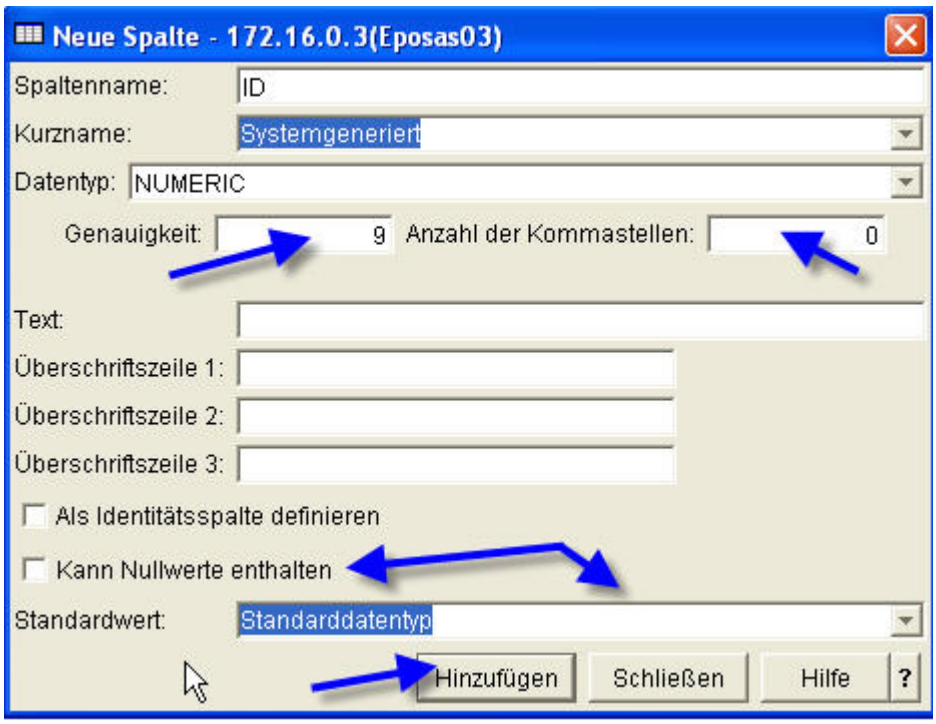
Die erste anzulegende Spalte erhält den Namen ID. Tragen Sie den Namen im Feld Spaltenname ein. Die Combo-Box für Datentypen wird aufgeklappt und in der Typliste wählen Sie <numeric> aus. Achtung, die Spaltendefinition ist damit noch nicht erledigt!



Spaltendefinition

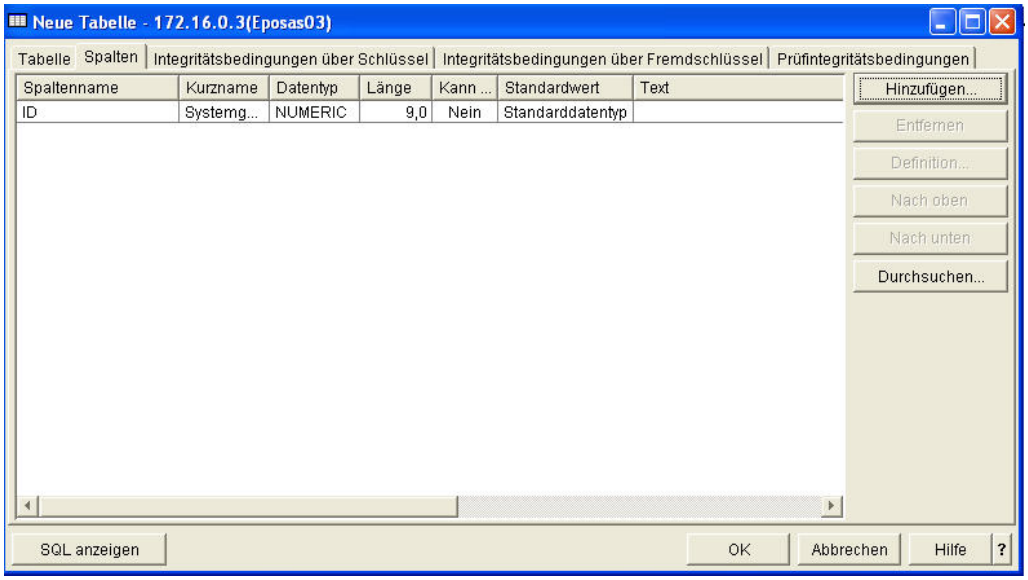
3.5.4
Seite 12

Die Genauigkeit wird vom Tool auf fünf Stellen gesetzt. Wir ändern diesen Wert auf neun Stellen ab. Die Anzahl der Nachkommastellen belassen wir auf der vorgegebenen Einstellung. Die folgenden Eingabefelder für beschreibenden Text und Überschriften sind optional und brauchen nur bei Bedarf ausgefüllt zu werden. Entfernen Sie die Häkchen aus den Check-Boxen für Identität und Null-Werte. Den Standardwert sollten Sie auf <Standarddatentyp> setzen.



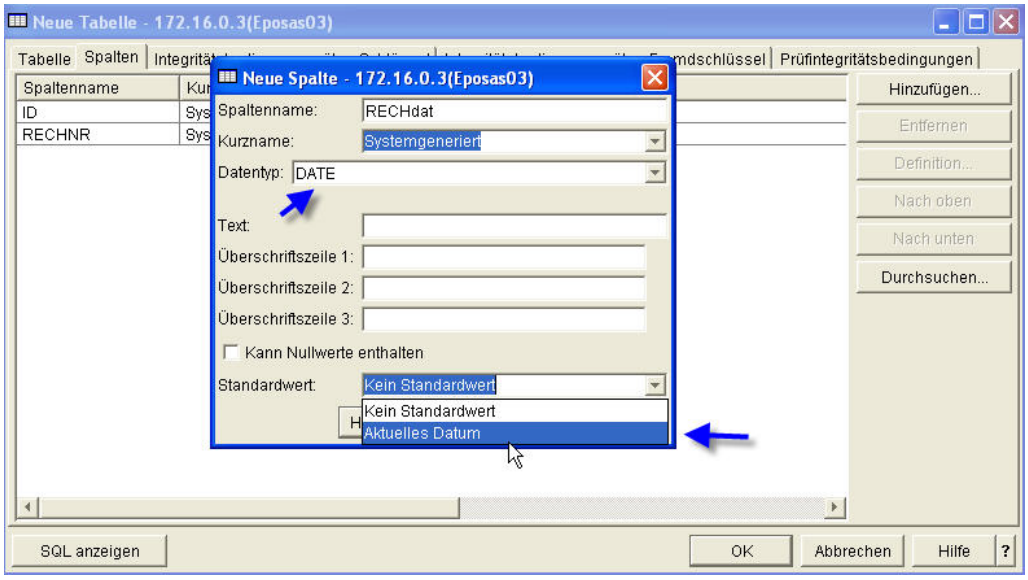
weitere Angaben zur Spalte

Durch Anklicken des Buttons <Hinzufügen> wird die eben definierte Spalte zur Spaltenliste der Tabelle hinzugefügt. Der Dialog selbst wird dadurch nicht geschlossen.



registrierte Spalte

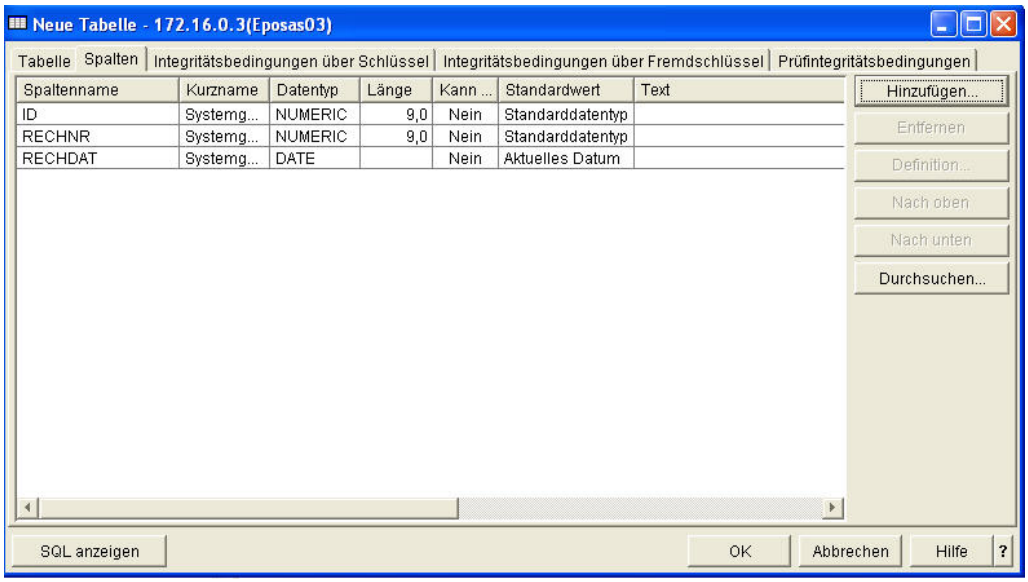
Legen Sie die die weiteren Spalten in dieser Weise an, aber achten Sie auf die Spalte für das Rechnungsdatum. Hier sollten Sie als Standardwert das aktuelle Tagesdatum festlegen.



Spalte Rechnungsdatum

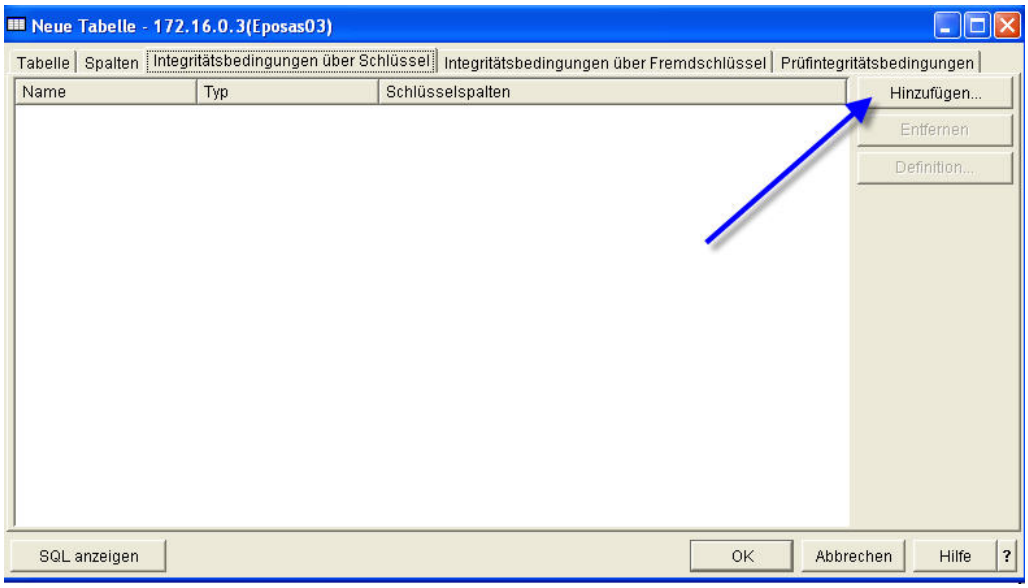
3.5.4
Seite 14

Nach der Definition der Spalten für diese Tabelle sollte die Spaltenliste etwa folgendes Bild bieten. Übrigens, noch ist diese Tabelle nicht wirklich erstellt worden, wie am Text im Titel des Dialogs zu erkennen ist.



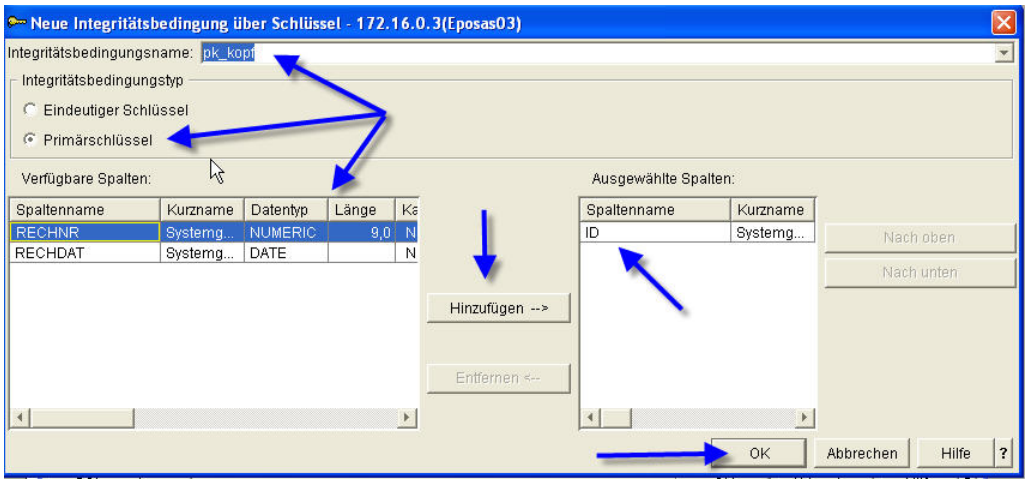
Letzte Fassung der Spaltenliste

In der Definition fehlt nun noch der Primärschlüssel für die Rechnungskopftabelle. Klicken Sie dafür den Reiter <Integritätsbedingungen über Schlüssel> an. Im dann erscheinenden Bereich klicken Sie den <Button <Hinzufügen> an.



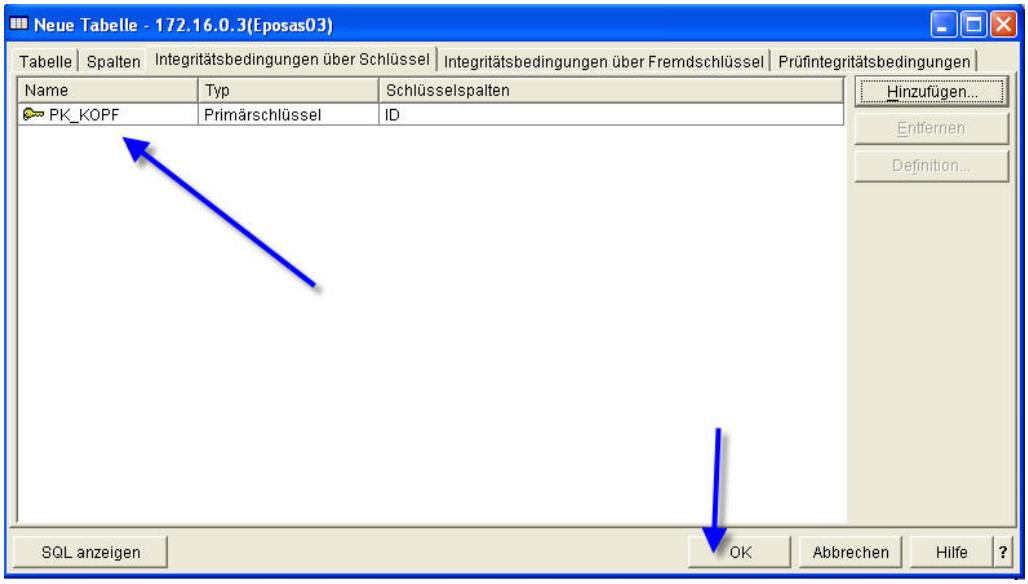
Reiter zur Definition von Schlüsseln

Für die Bezeichnung eines Schlüssels empfehle ich einen möglichst aussagefähigen Namen. In diesem Fall trage ich unter Integritätsbedingungsname „PK_KOPF“ ein. Da hier ein Primärschlüssel angelegt werden soll, muss der Radiobutton <Primärschlüssel> selektiert werden. Klicken Sie nun das Feld <ID> in der Liste der verfügbaren Spalten an, da <ID> für den Schlüssel vorgesehen ist. Durch den Button <Hinzufügen> wird <ID> in die Liste der ausgewählten Spalten übernommen. Anschließend verlassen Sie diesen Dialog mit einem Klick auf <OK>.



Primärschlüssel definieren

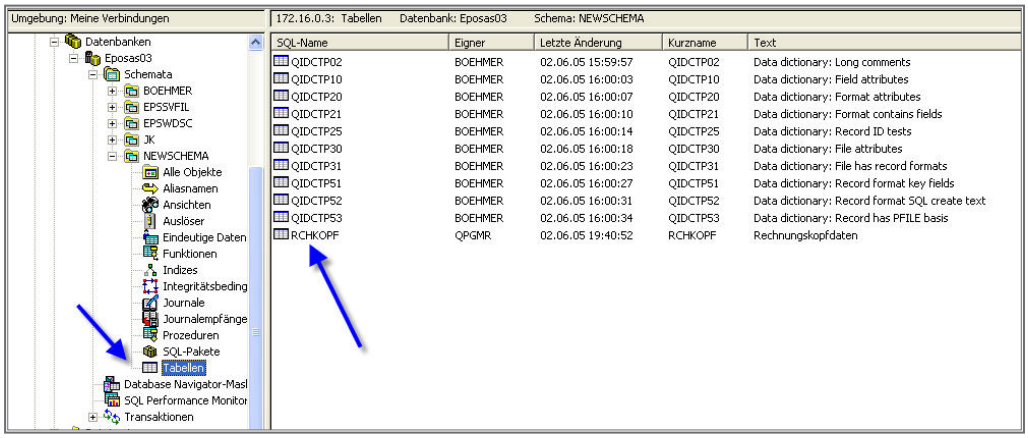
Der Schlüssel wurde nun übernommen. Der Definitions-Dialog für die Tabelle kann jetzt verlassen werden. Ein Fremdschlüssel ist nicht angelegt werden, da noch keine weiteren Tabellen von uns in diesem Schema definiert wurden.



Primärschlüssel

3.5.4
Seite 16

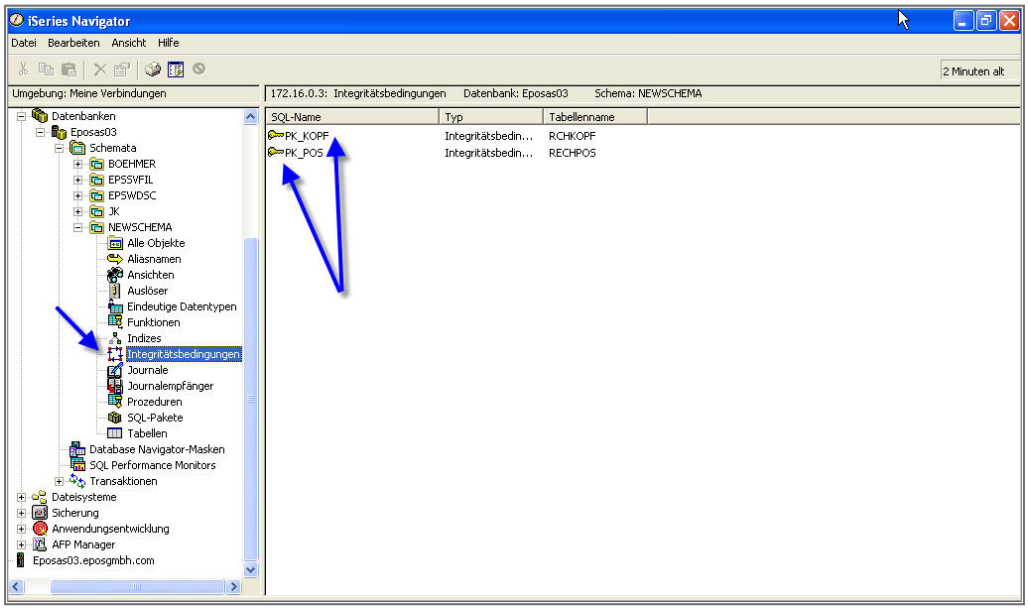
Das System wird nun die Tabelle mit der gewünschten Definition im ausgewählten Schema erstellen.



Angelegte Tabelle

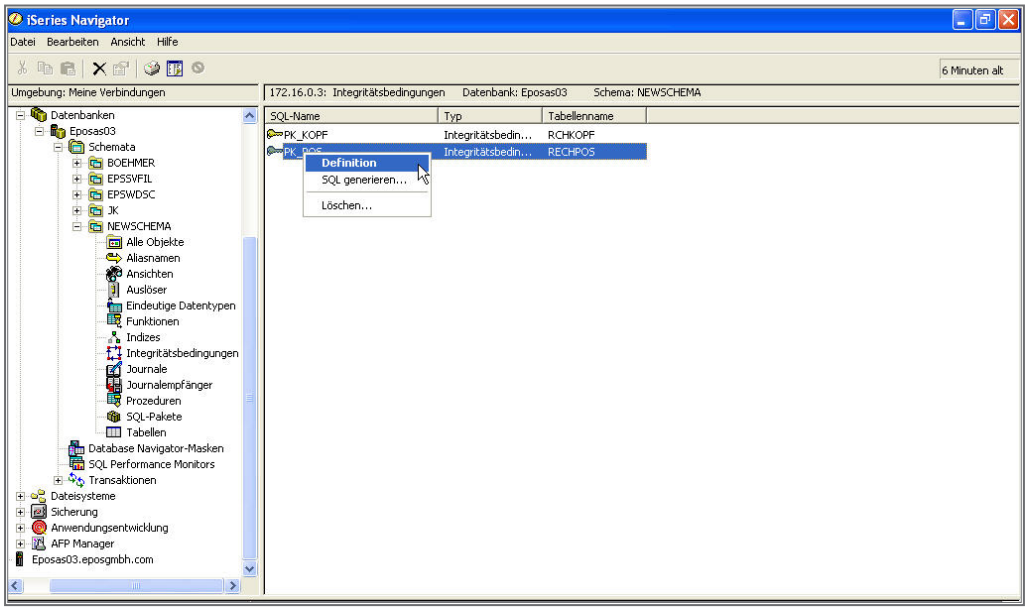
Wir müssen nun noch die Positionstabelle erstellen. Dazu wiederholen wir die vorhergehenden Schritte noch einmal.

Ich markiere nun einmal mit der linken Maustaste den Punkt <Integritätsbedingungen> und folgendes Bild wird sich ergeben.



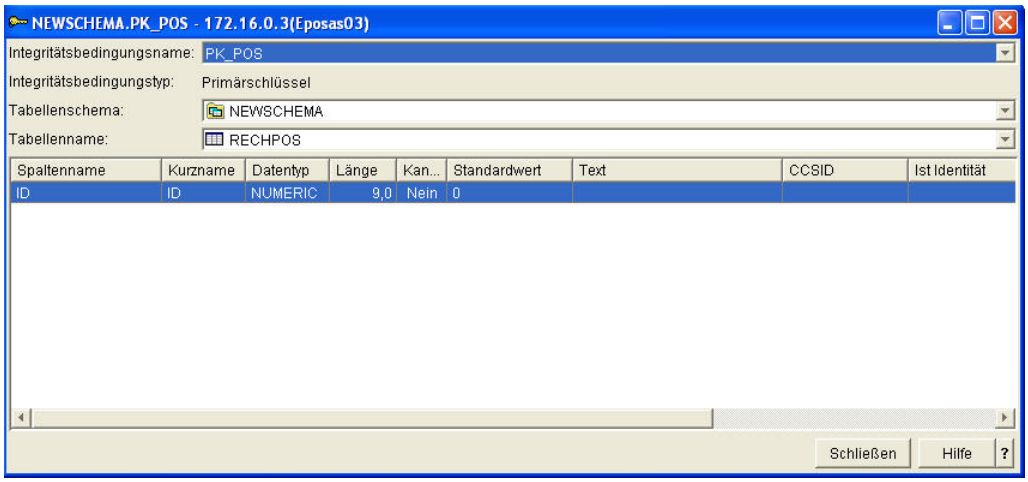
Integritätsbedingungen

Bei der Definition der Positionstabelle habe ich natürlich auch einen Primärschlüssel definiert, den ich PK_POS genannt habe. Da ich ihn angelegt habe, wissen Sie nicht, welche Spalte ich für ihn gewählt habe. Aber da lässt sich ein Ausweg finden. Markieren Sie den Schlüssel PK_POS mit der rechten Maustaste und schauen sich die Definition an.



Schlüsseleigenschaften abrufen

Der Informationsdialog mit der Definition der Integritätsbedingung.

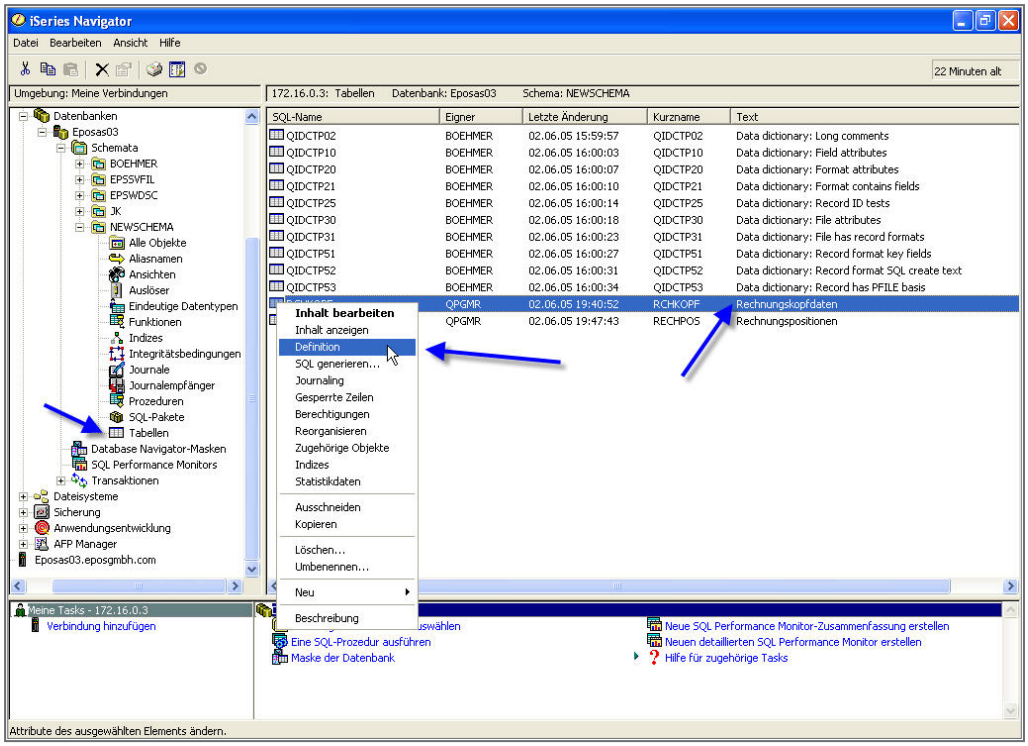


RI-Eigenschaften

3.5.4
Seite 18

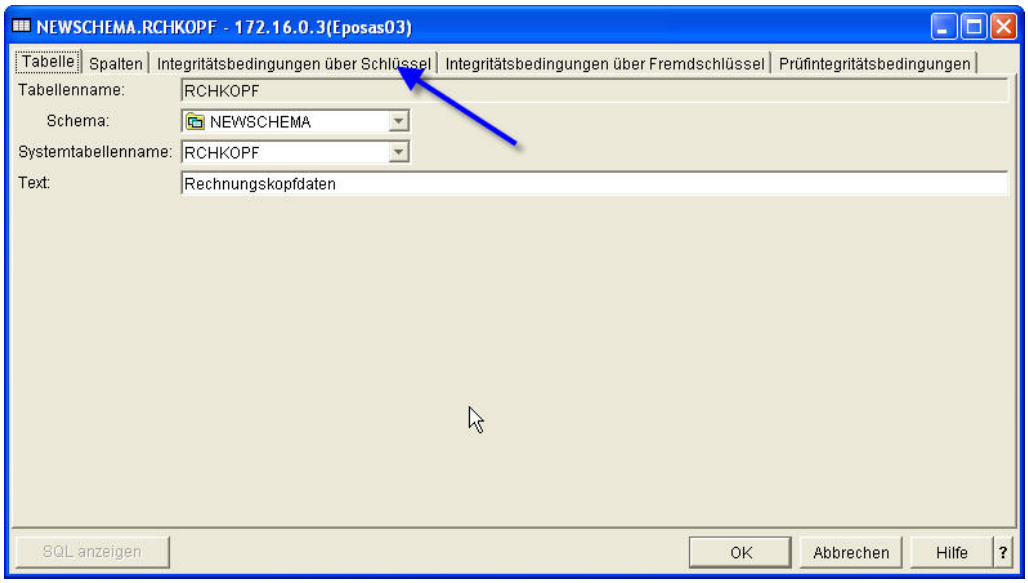
Es fehlt noch der Fremdschlüssel, mit dem von der Positionstabelle auf die Kopftabelle verwiesen wird und die Angabe der Eindeutigkeit für die Rechnungsnummer in der Rechnungstabelle. Diese Definitionen können natürlich im Nachhinein angelegt werden.

Dazu selektieren wir wieder den Unterpunkt <Tabellen>, klicken anschließend mit der rechten Maustaste auf den Tabelleneintrag der Positionstabelle und wählen im Kontextmenu die Option <Definition> aus.

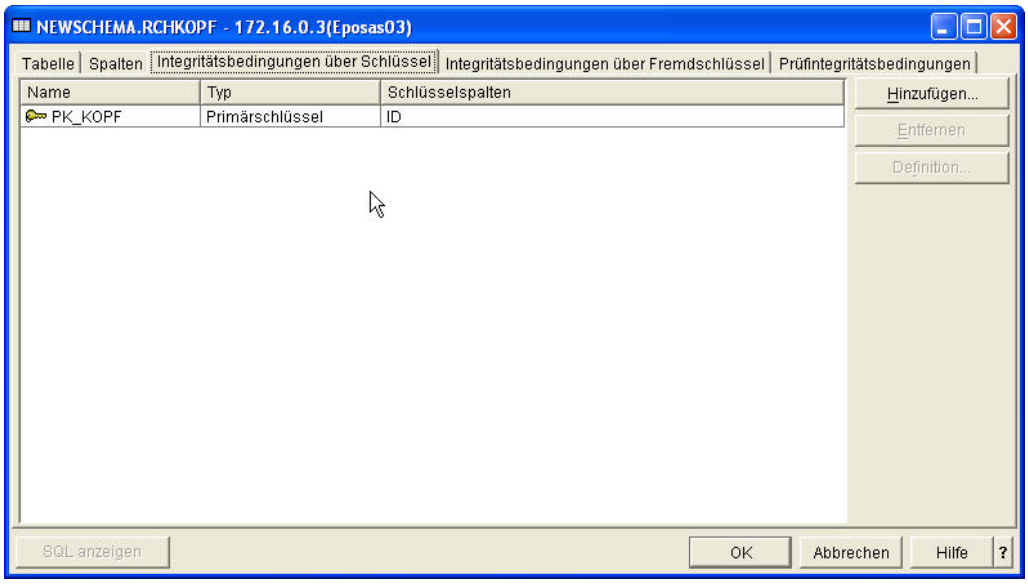


Eindeutigkeit definieren

In diesem Dialog (kein anderer als der bereits bekannte…) klicken wir den Reiter für die Integritätsbedingungen über Schlüssel an.



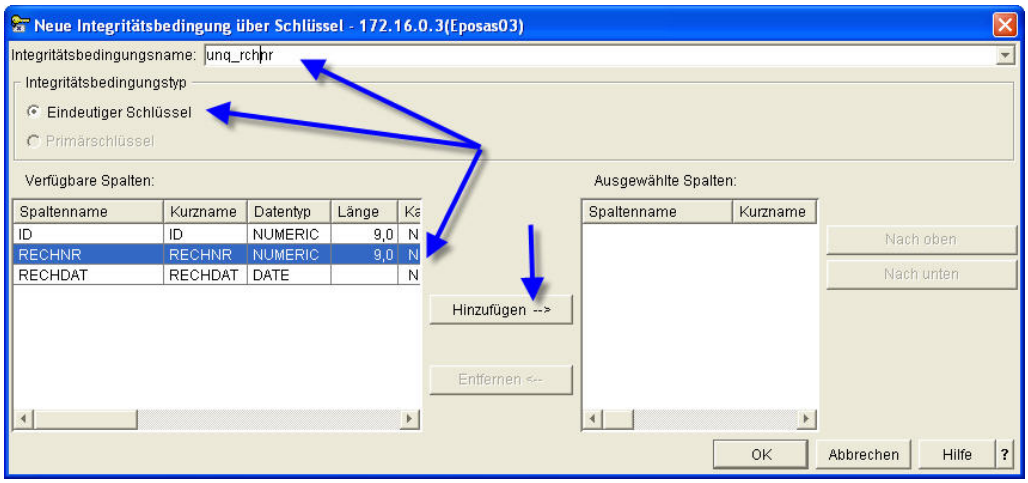
Tabellendefinition



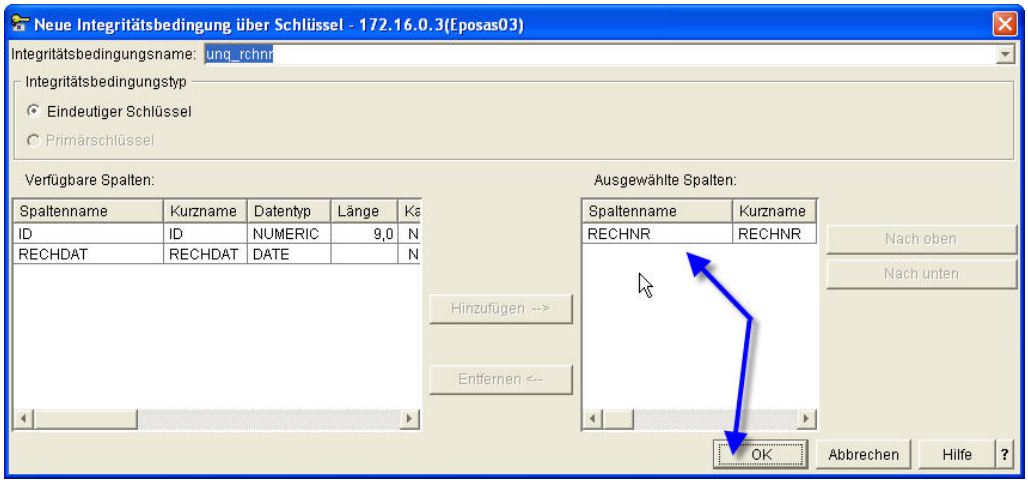
Schlüsseldefinition

3.5.4
Seite 20

Wir fügen eine neue Bedingung hinzu, indem wir auf <Hinzufügen> klicken. Dann sollte wieder ein sprechender Name vergeben werden. Es wurde festgelegt, dass der Inhalt der Spalte RECHNR eindeutig sein soll. Also wird diese Spalte ausgewählt und der Liste der ausgewählten Spalten angegliedert.

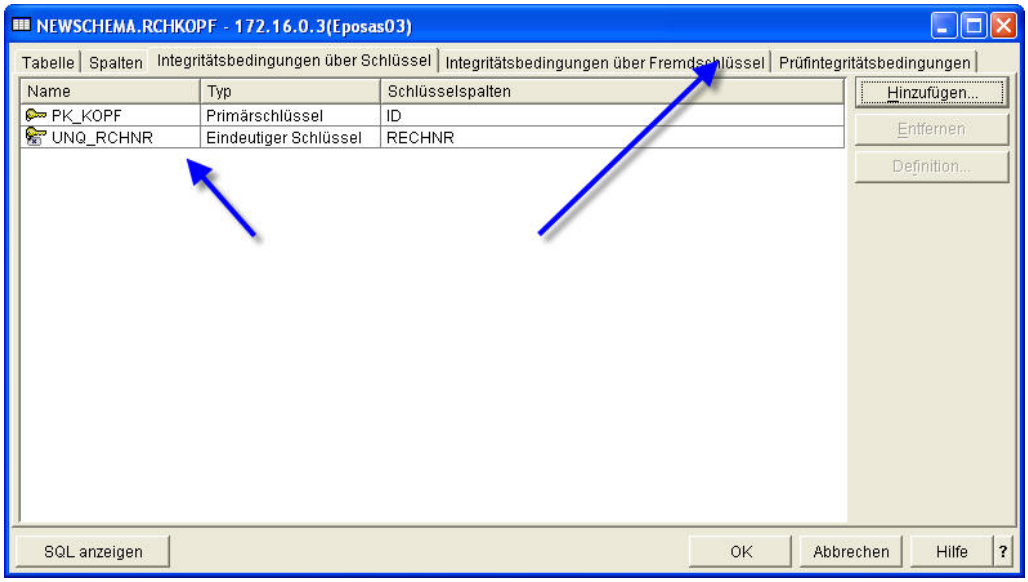


Bedingung definieren



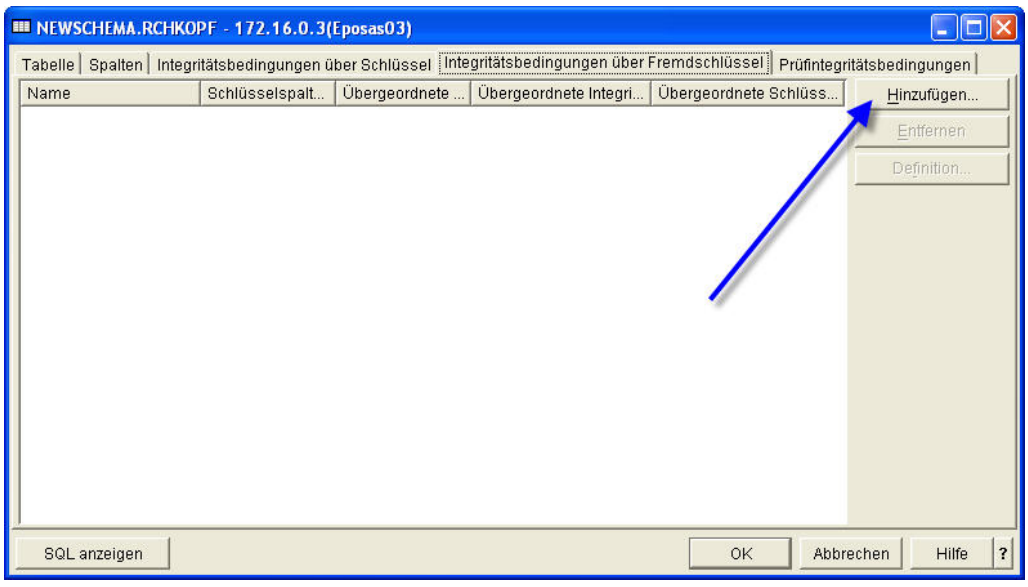
und übernehmen

Anschließend stellen sich die Integritätsbedingungen wie folgt dar:



Liste der Bedingungen

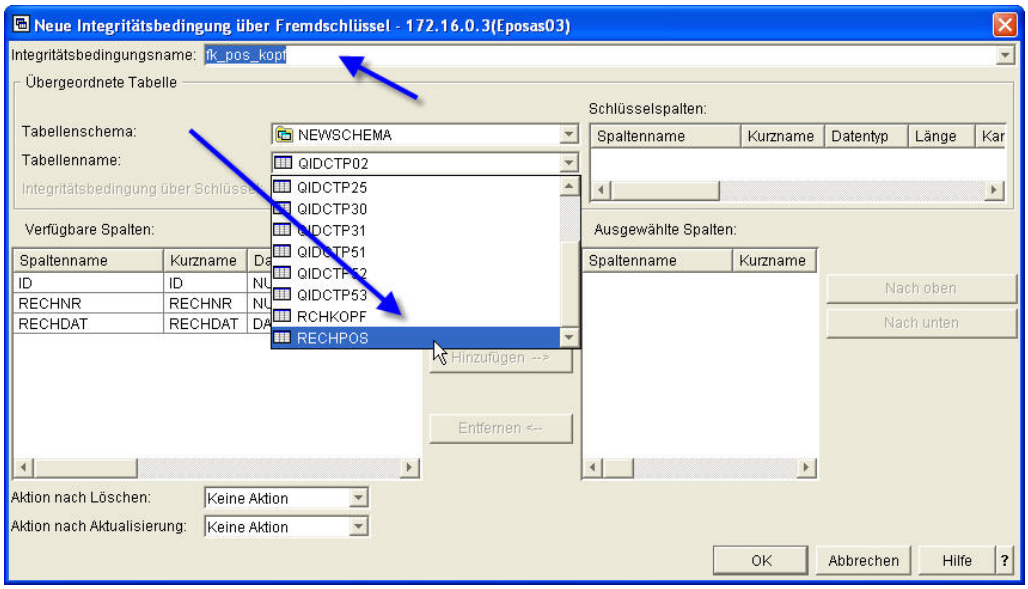
Nun soll noch der Fremdschlüssel angelegt werden. Dazu klicken Sie auf den Reiter <Integritätsbedingungen über Fremdschlüssel> und anschließend auf <Hinzufügen>.



Fremdschlüssel erstellen

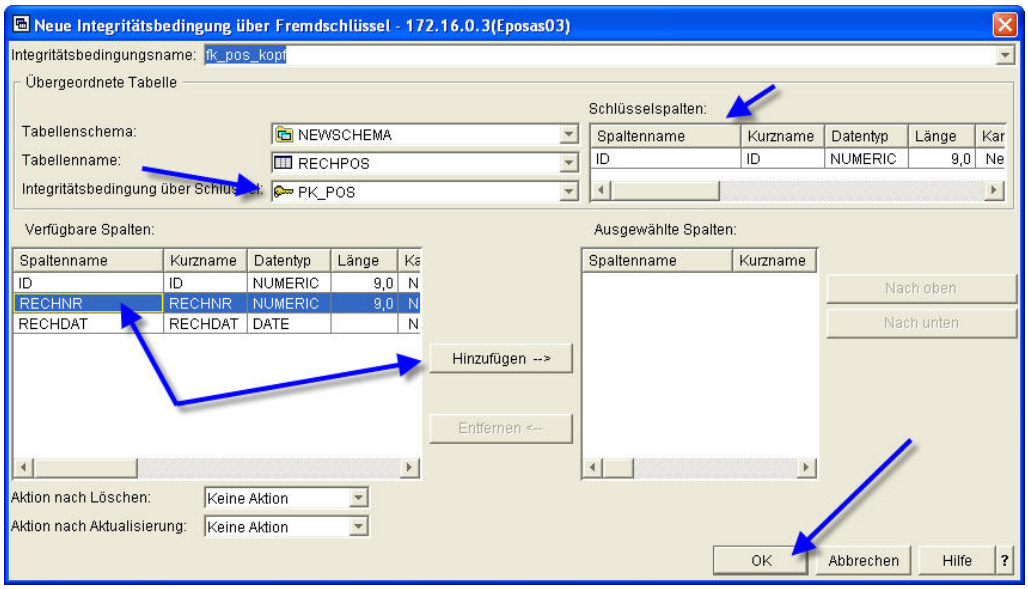
3.5.4
Seite 22

Auch hier wird wieder ein aussagefähiger Name für den Fremdschlüssel vergeben. Wählen Sie die übergeordnete Zieltabelle aus dem Listenanteil der Combo-Box aus, in diesem Fall die Kopftabelle.



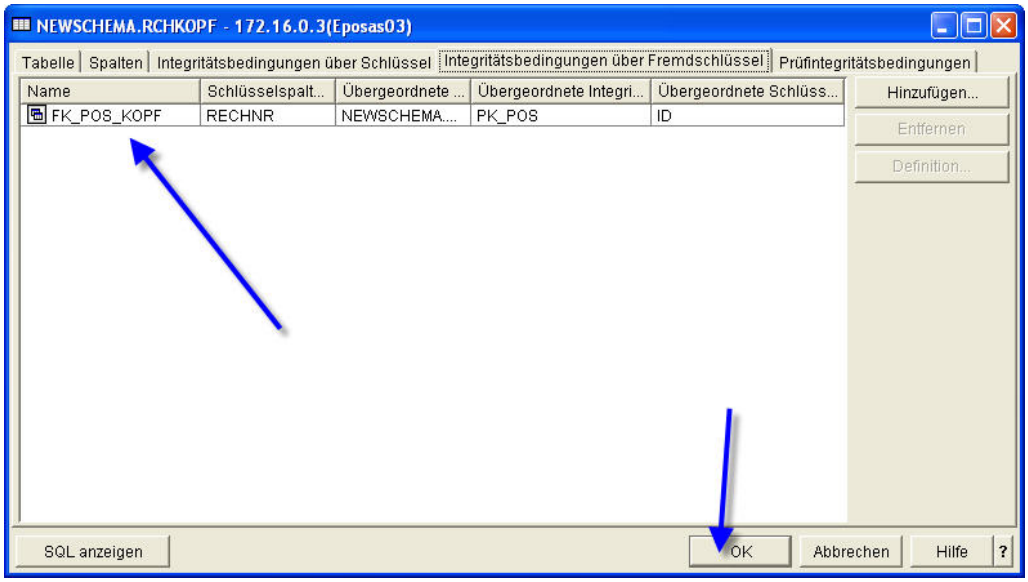
Auswahl der Zieltabelle

Nach der Auswahl der übergeordneten Tabelle wird automatisch die Integritätsbedingung über Schlüssel (Primärschlüssel) dieser Tabelle vervollständigt, wie an den Eintragungen PK_POS und ID im oberen Teil des Dialogs zu erkennen ist. Selektieren Sie nun die Spalte RECHNR und klicken Sie <Hinzufügen> an. Durch <OK> bestätigen Sie noch Ihre Wahl.

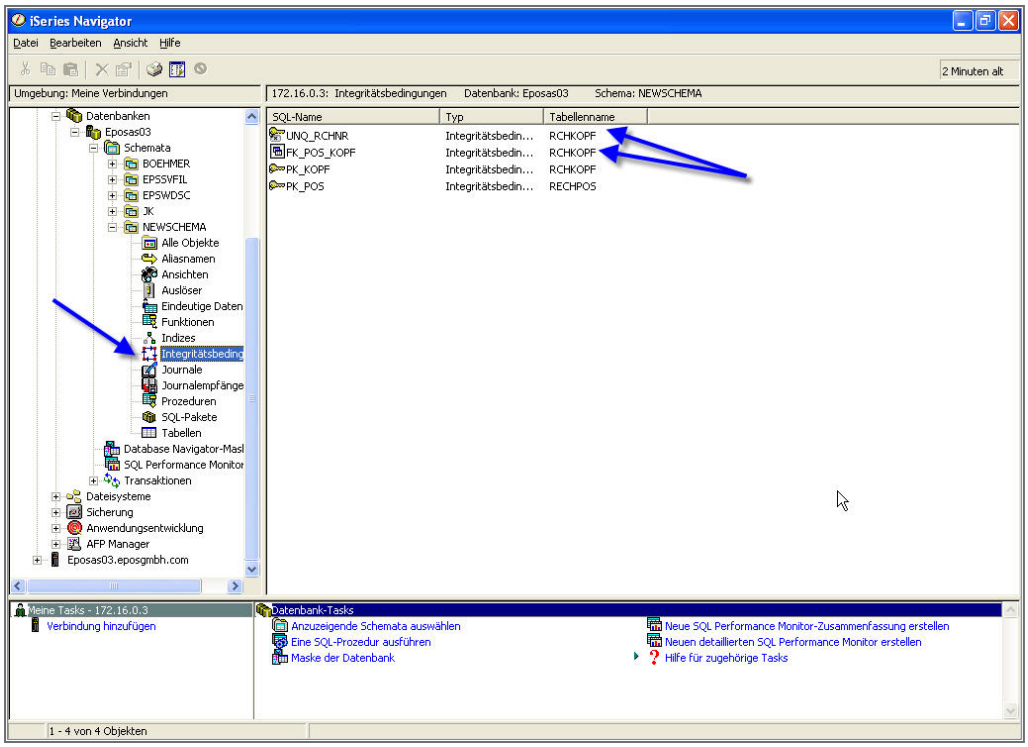


Feldwahl

Das System kehrt zum Definitionsdialog zurück und zeigt den eben definierten Fremdschlüssel an. Um den Schlüssel endgültig in der Datenbank anzulegen, klicken Sie bitte noch einmal auf den Button <OK>.



Der definierte Fremdschlüssel



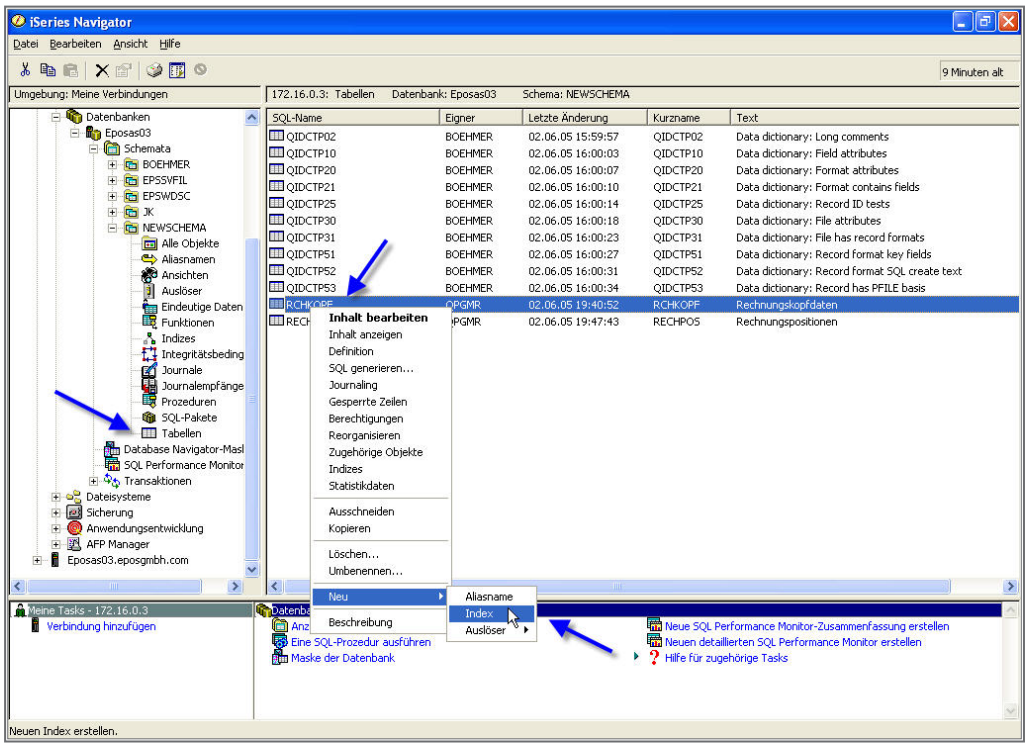
Neue Integritätsbedingungen

Anschließend kontrollieren Sie Ihre Aktion, indem Sie die Integritätsbedingungen selektieren. Im rechten Bereich erscheinen dann die soeben angelegten referenziellen Bedingungen.

3.5.4
Seite 24

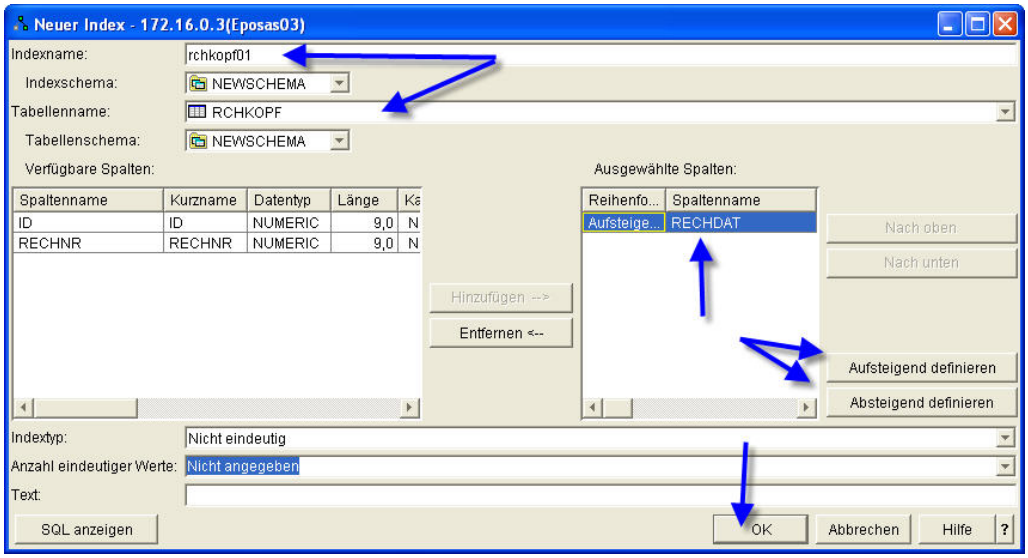
Zum Abschluss überlegen wir uns noch, dass sicherlich oft über die Spalte Rechnungsdatum gesucht wird. Vielleicht soll auch aus einem Programm heraus über das Datum auf Sätze der Rechnungskopftabelle zugegriffen werden. Die Frage ist also, wie lege ich einen Index an?

Selektieren Sie die Tabellen und klicken die Rechnungstabelle mit rechts an. Im Kontextmenu wählen Sie die Option zur Neuanlage eines Index.



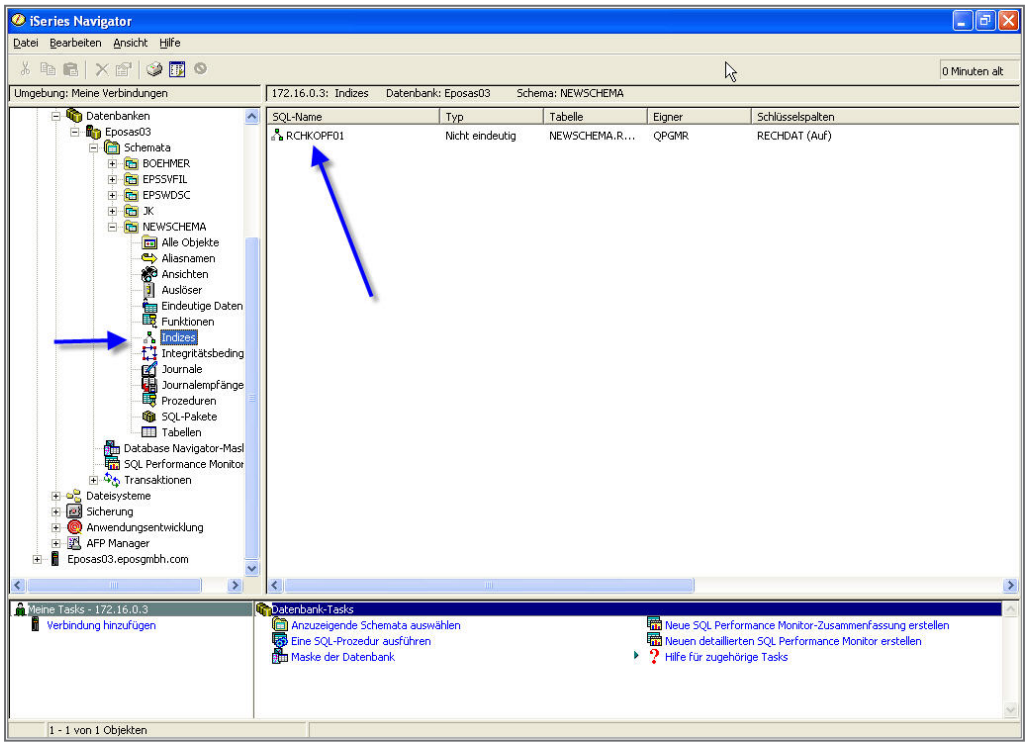
Zur Anlage eines neuen Index

Beginnen Sie mit der Vergabe des Namens des Index. Danach wählen Sie die gewünschte Spalte und fügen Sie diese zu den ausgewählten Spalten hinzu. Beachten Sie, dass Sie hier auch die Folge definieren können. Mit <OK> werden Ihre Angaben übernommen.



Definition

Auch hier können Sie noch kontrollieren, ob Ihre Aktionen erfolgreich waren.



Der angelegte Index

3.5.4
Seite 26

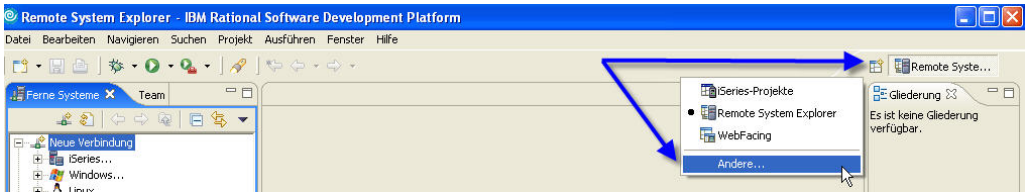
So weit soll es reichen, was die Datenbankfunktionen des iSeries-Navigators angeht. Das bedeutet nicht, dass sich damit auch sein Funktionsspektrum in diesem Umfeld erschöpft; ganz im Gegenteil…

Dazu habe ich einen kleinen Vorschlag.

Begeben Sie sich mit der rechten Maustaste auf eine Reise durch die Datenbankwelt des Navigators und staunen Sie!

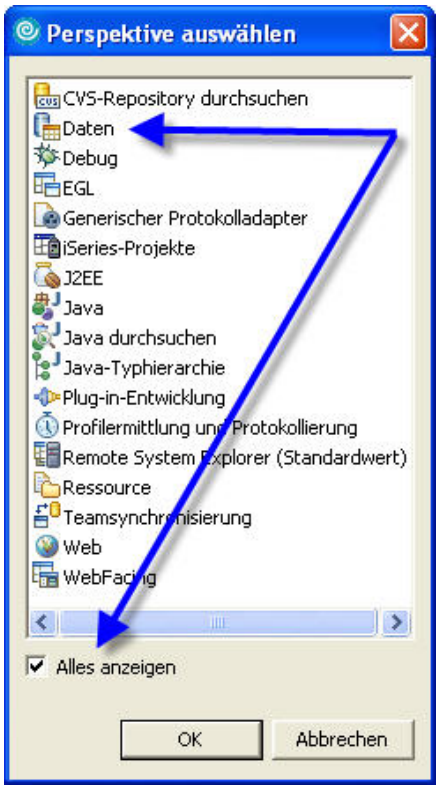
Datenbankimplementierung mit dem Websphere Developmentstudio Client

Neben dem iSeries-Navigator haben wir mit dem WDSC ein weiteres Tool zur Verfügung, mit dem ebenfalls Datenbanken erstellt und bearbeitet werden können. Auch unterschiedliche SQL-Statements können verwendet werden und zu ihrer Erstellung stehen auch mehrere Assistenten zur Verfügung. Tasten wir uns hinein in das Tool, indem zu Beginn die Arbeitsumgebung eingerichtet wird. In diesem Kapitel wird die Version 6 des WDSC verwendet; wundern Sie sich bitte nicht über die Gestaltung der Oberfläche, sie weist erhebliche Unterschiede im Vergleich zu ihren Vorgängerversionen auf. Ausgangspunkt des Rundgangs ist die Perspektive des Remote System Explorers. Von dort aus wird erst einmal die Datenperspektive geöffnet.



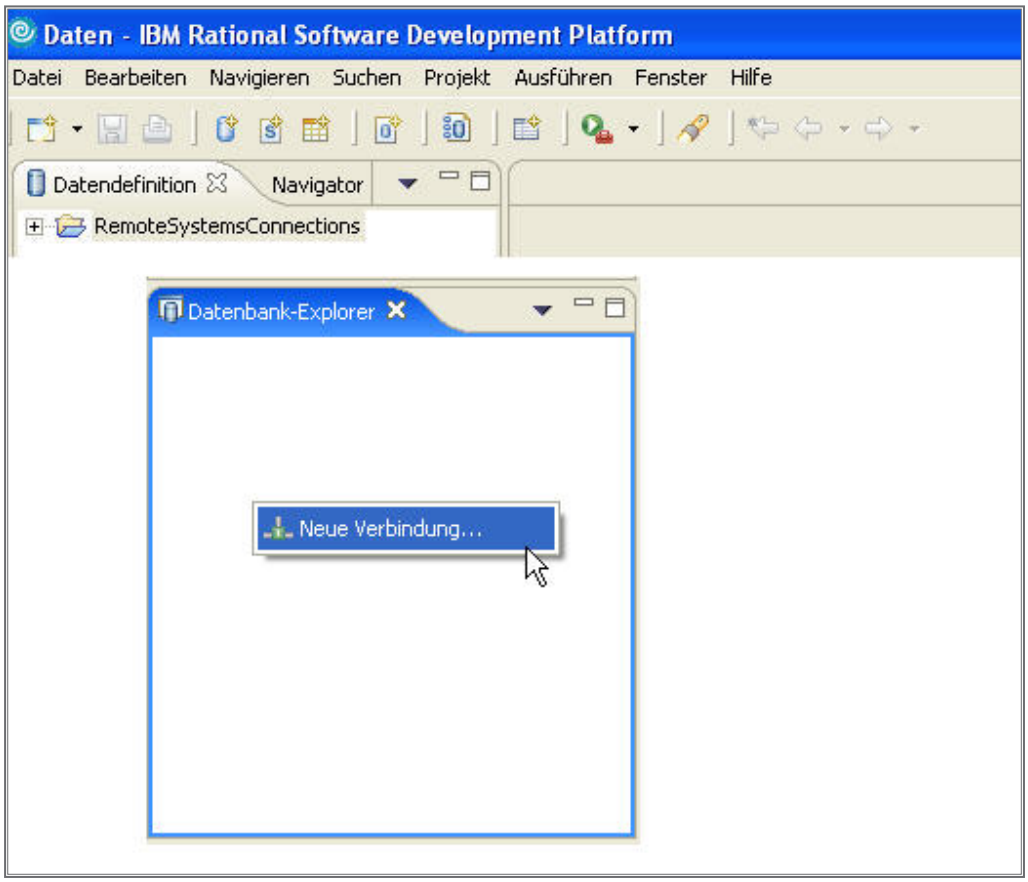
Öffnen einer zusätzlichen Perspektive

Im folgenden Dialog sollten Sie die CheckBox unten links mit einem Haken versehen, da sonst die Perspektive „Daten“ nicht in der Liste auswählbarer Perspektiven erscheinen wird.



Auswahl der Datenperspektive

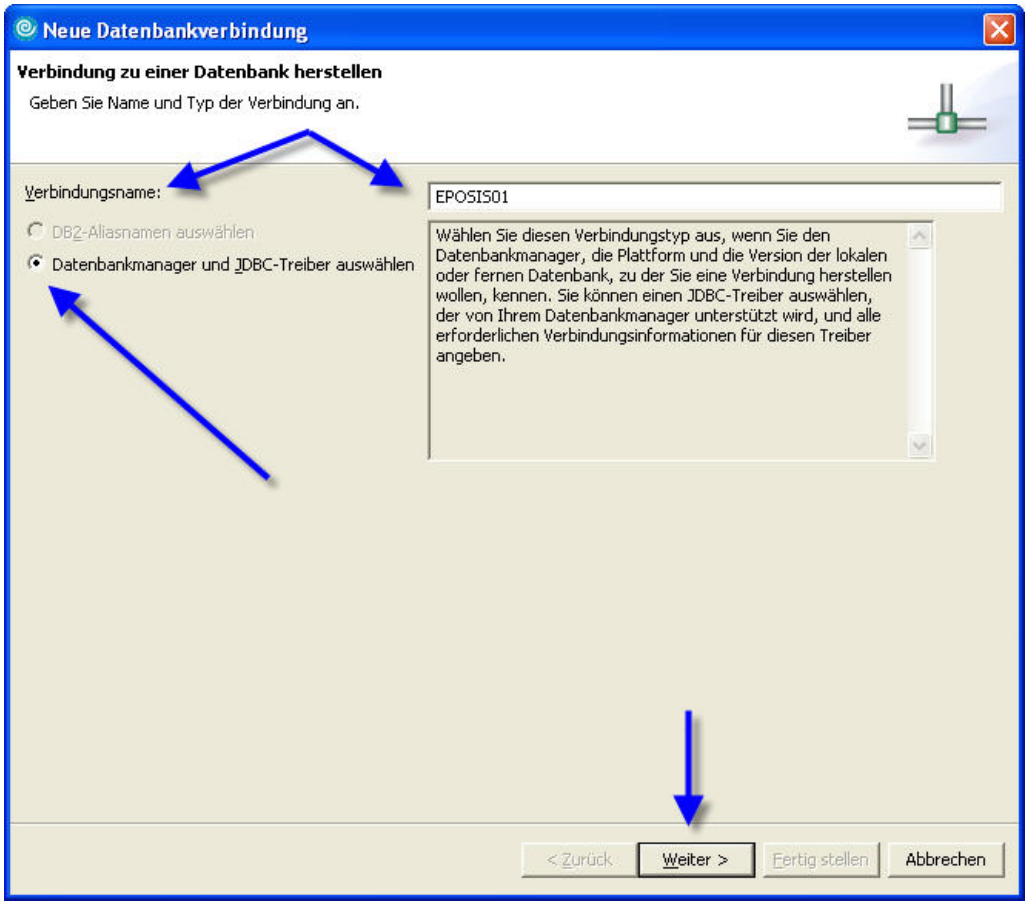
Im nächsten Schritt werde ich eine Verbindung zu einer Datenbank herstellen; in diesem Fall zu einem iSeries-System. Sie können aber Verbindungen zu den meisten der namhaften Datenbanken herstellen. Nachdem sich die Datenperspektive geöffnet hat (das kann je nach Rechnerausstattung eine gewisse Zeit in Anspruch nehme), öffne ich mit der rechten Maustaste ein Kontextmenu in der Sicht „Datenbank-Explorer". Diese Sicht befindet sich in der unteren linken Ecke der Datenperspektive.



DB_Verbindung herstellen, erster Schritt

Im nächsten Dialog vergeben Sie einen Namen für Ihre neue Datenbankverbindung. Welche Bezeichnung hier gewählt wird, ist nicht ganz so wichtig, aber dem Typ sollten Sie Beachtung schenken. Sollten Sie eine DB2 auf Ihrem Rechner installiert haben oder auch eine ferne DB2 (auf iSeries oder aber PC-basiert) und ist diese Datenbank katalogisiert, so können Sie den ersten Radio-Button aktivieren. Die Verbindung kann dann über den so genannten Alias-Namen hergestellt werden.

Über den zweiten Radio-Button haben Sie die Möglichkeit, das Ziel-Datenbanksystem und den passenden JDBC-Treiber selbst zu wählen. Ich werde hier die zweite Alternative wählen.

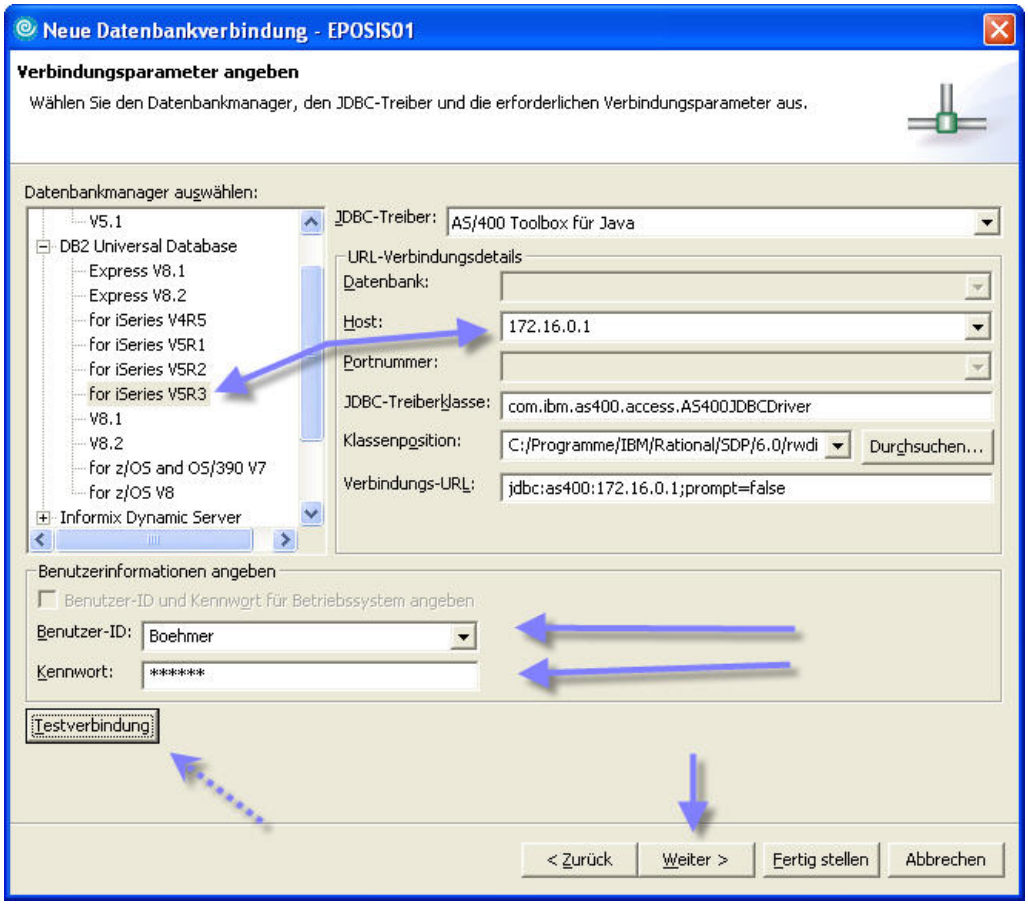


DB_Verbindung herstellen, zweiter Schritt

Nachdem Sie die Schaltfläche „Weiter“ angeklickt haben, werden Sie anschließend Ihre Datenbankverbindung weiter parametrisieren. So können Sie im linken Bereich des neuen Dialogs das Datenbankmanagementsystem Ihrer Ziel-Datenbank angeben. Sie werden bemerken, dass viele der namhaften Datenbanken hier schon zu Ihrer Verfügung stehen.

Im rechten Bereich wird das System einen passenden JDBC-Treiber vorschlagen und den entsprechenden Aufbau der Verbindungszeichenfolge vorschlagen. Im Großen und Ganzen müssen Sie nur noch den Ziel-Host definieren, in dem Sie zum Beispiel die IP-Adresse oder den DNS-Namen angeben. Die Verbindungszeichenfolge wird automatisch durch den Assistenten angepasst.

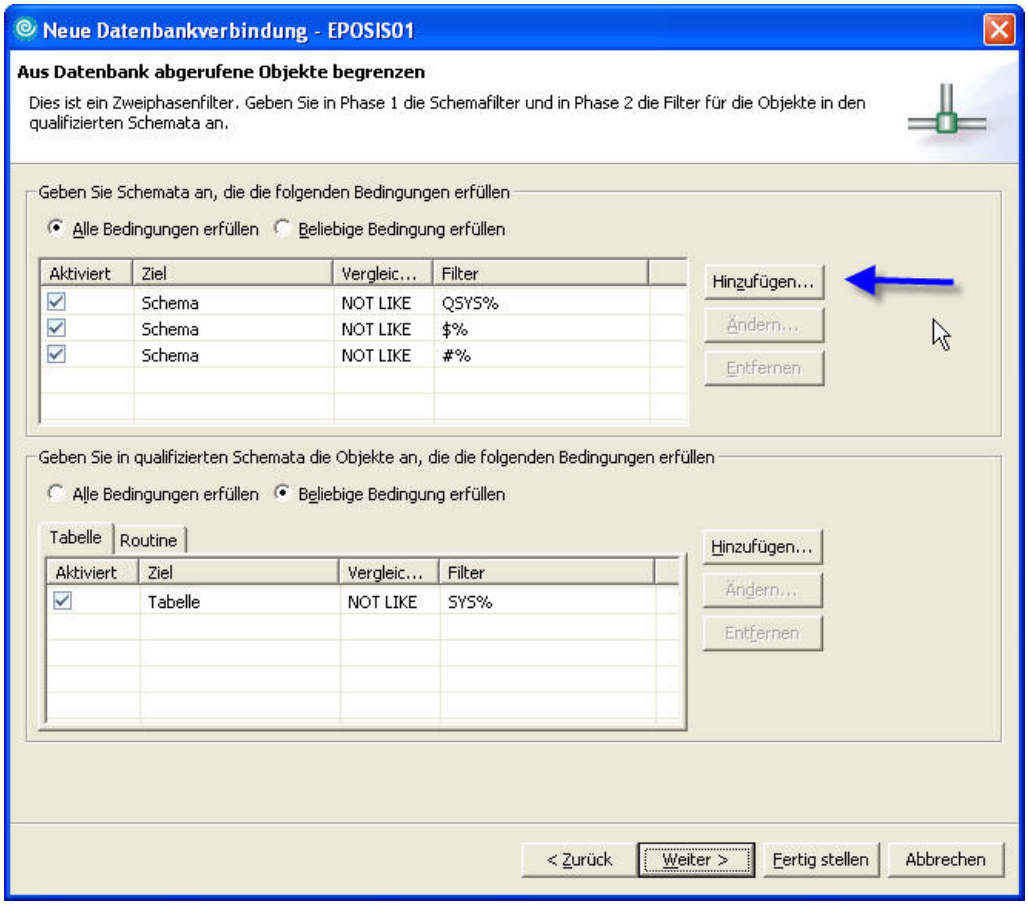
Im unteren linken Bereich geben Sie bitte ein gültiges Profil der Zieldatenbank mit Kennwort an. Anschließend sollten Sie die jetzt bestehende Konfiguration über die Schaltfläche „Testverbindung“ überprüfen; unbedingt notwendig ist das allerdings nicht. Anschließend klicken Sie auf „Weiter“.



Angeben der Verbindungsparameter

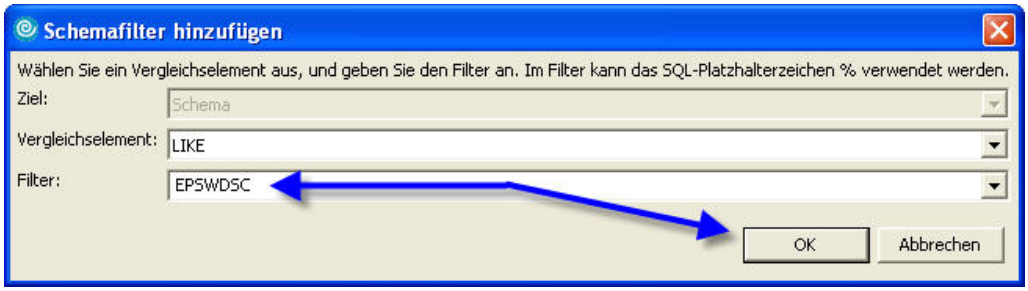
Das System wird diverse Informationen von der Zieldatenbank abrufen. Der Umfang dieser Information kann unter Umständen ganz erhebliche Ausmaße annehmen. Wenn, wie in unserem Beispiel, die Zieldatenbank eine iSeries-DB2 sein soll, denken Sie einmal daran, dass das gesamte System in diesem Fall die Datenbank ist. Ohne eine Filterung wird also die gesamte Maschine auf Datenbankinformationen untersucht.

Ich werde also dafür Sorge tragen, dass erst einmal nur ein einzelnes Schema abgerufen wird.



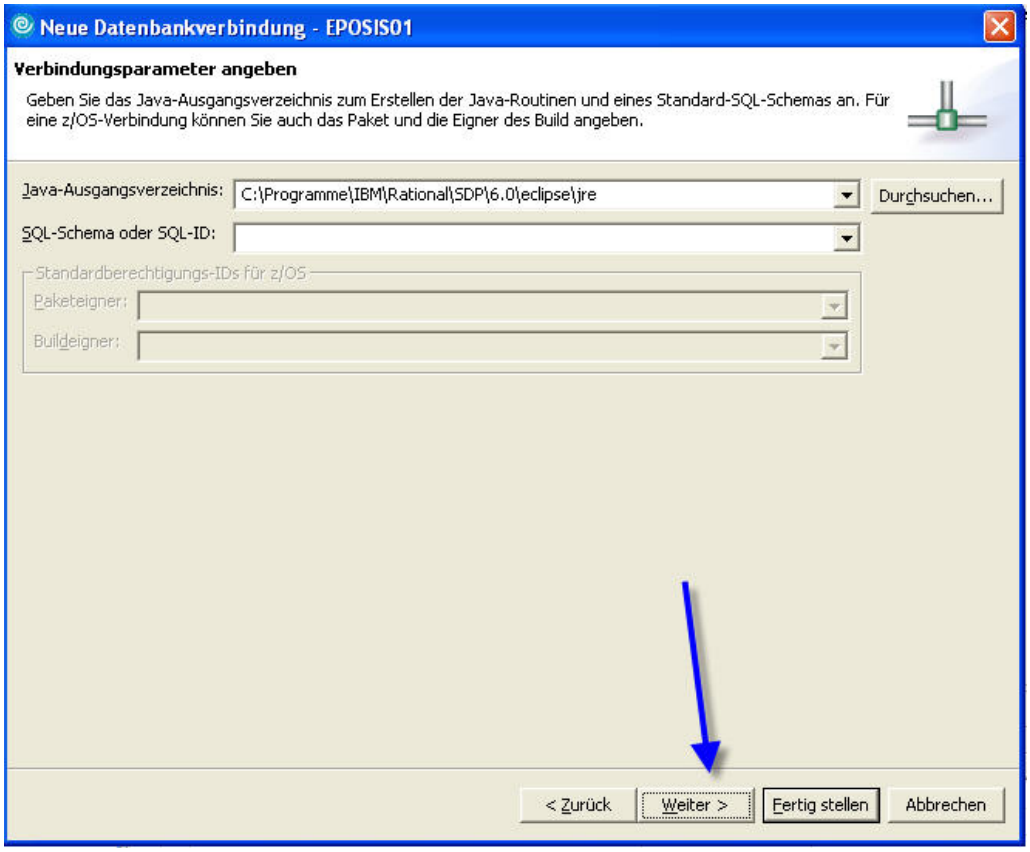
Abzurufende Informationen filtern

Durch die Filtervorgaben werden schon die Systemschemata ausgeblendet. Jedoch bleiben danach mit Sicherheit noch genügend andere übrig, deren Metadaten abgerufen werden würden. Klicken wir auf „Hinzufügen“. In der dann erscheinenden Abfrage definiere ich, dass nur Informationen zum Schema EPSWDSC abgerufen werden sollen.



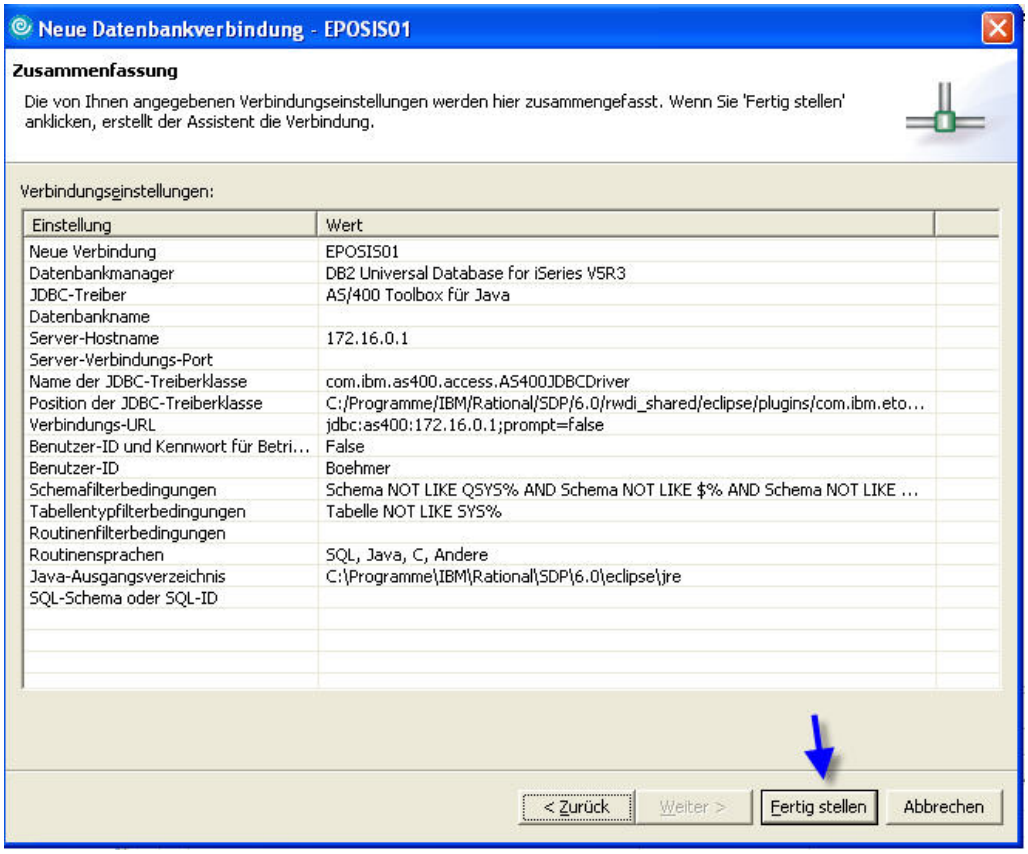
Abzufragendes Schema angeben

Nachdem ich auf „OK“ geklickt habe, bin ich wieder im Ausgangsdialog (siehe Abbildung "*Abzurufende Informationen filtern*"), in dem jetzt zusätzlich das eben angegebene Filterkriterium eingetragen wurde. Nach einem Klick auf „Weiter“ erscheint dieser Dialog, den ich in diesem Fall einfach mit „Weiter“ verlasse.

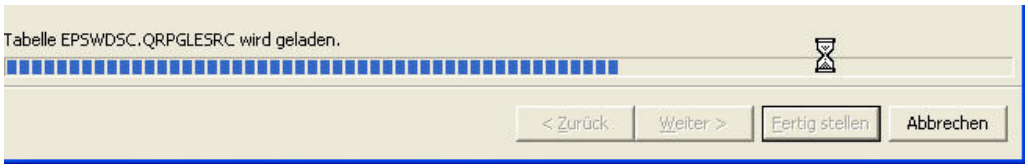


Weitere Parameter

Die nun folgende Zusammenfassung der einzelnen Konfigurationsschritte sollte noch einmal kontrolliert werden. Wenn man mit dem Ergebnis einverstanden ist, kann die Erstellung der Datenbankverbindung fertig gestellt werden. Anschließend werden Sie folgendes Bild erhalten:



Zusammenfassung der bisherigen Einstellungen

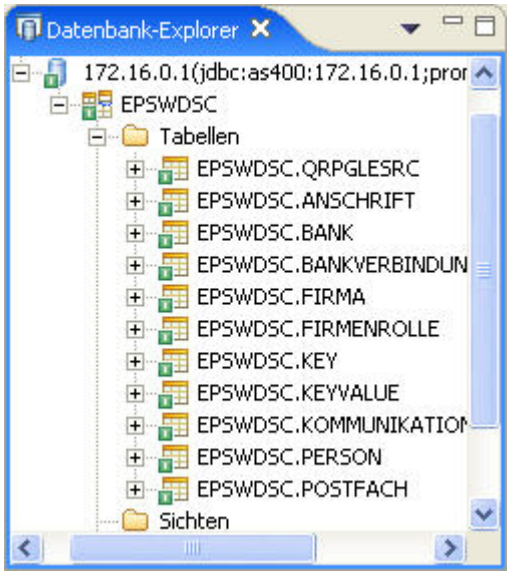


Während der Verbindungsherstellung

Übernahme von Metadaten

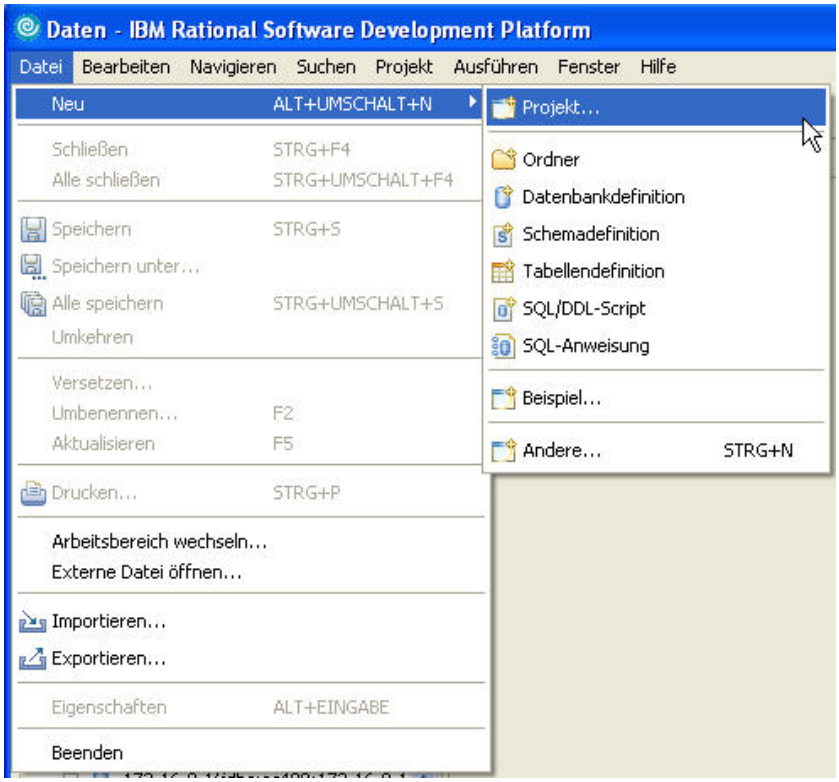
In dem letzten Dialog möchte ich nicht, dass die ermittelten Metadaten direkt in einen Projektordner kopiert werden, weil ich die zu importierenden Strukturen selbst definieren möchte. Die CheckBox werde ich also nicht aktivieren.

Im Anschluss daran befinde ich mich wieder in der bekannten Workbench-Umgebung. Ein Blick auf die Sicht „Datenbank-Explorer“ zeigt dann folgendes Bild:



Zur Verfügung stehende Daten

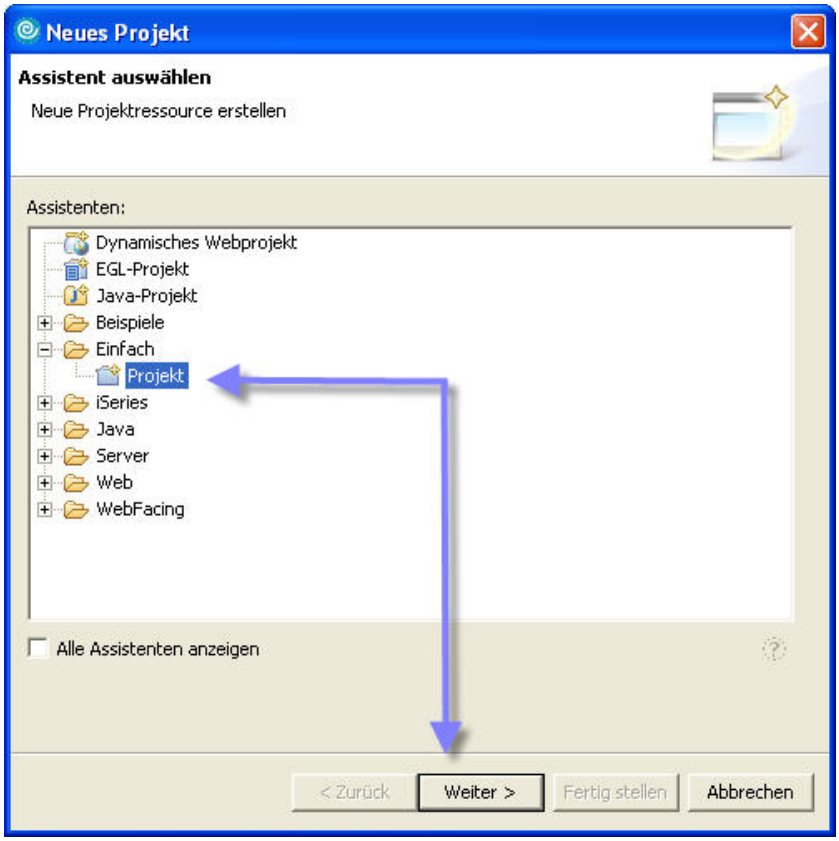
Unter diesen Tabellen, die im Schema EPSWDSC bereits existieren, können jetzt die ausgewählt werden, mit denen im Rahmen eines Datenbankprojektes gearbeitet werden soll. Wenn jedoch nur neue Tabellen erstellt werden sollen, so ist das natürlich auch möglich. Die Datenbankverbindung dient nämlich dazu, auch neu erstellte Datenbankstrukturen auf das Zielsystem zu übertragen. Beide Möglichkeiten möchte ich Ihnen vermitteln. Gehen wir zuerst den Weg, Tabellen in ein Datenbankprojekt zu importieren. Es kann nicht schaden, sich eine saubere Ordnerstruktur aufzubauen; die Dinge sind dann erheblich einfacher aufzufinden. Ich werde zu Beginn ein neues Projekt unter dem Namen „DB_Import“ anlegen.



Anlegen eines neuen Projekts

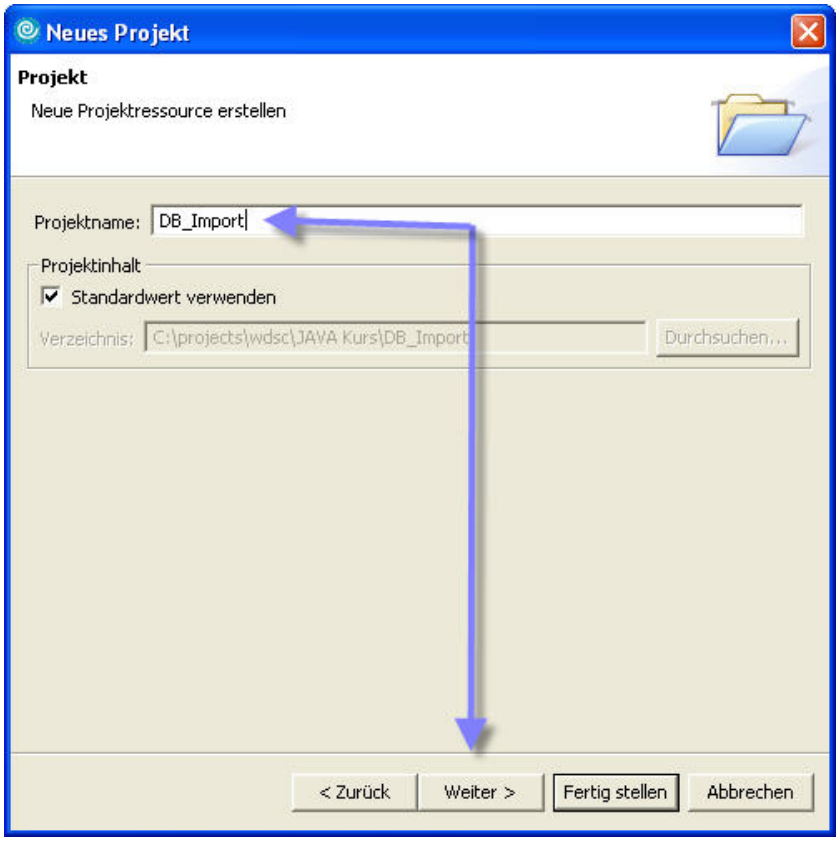
Die Grundlage für die folgende Arbeit soll ein einfaches Projekt sein. Dieser Projekttyp wird im nächsten Dialog ausgewählt. Es wäre auch möglich die gewünschten Tabellen in den bestehenden Ordner „RemoteSystemsConnections“ zu übernehmen. Aber wie gesagt, die Organisation wäre mir zu undurchsichtig.

3.5.5
Seite 10



Auswählen der Projektart

Der Assistent wird mit „Weiter“ fortgesetzt und im nächsten Schritt muss die Bezeichnung des neuen Projekts angegeben werden.



Das neue Projekt benennen

Eventuelle Projektverweise

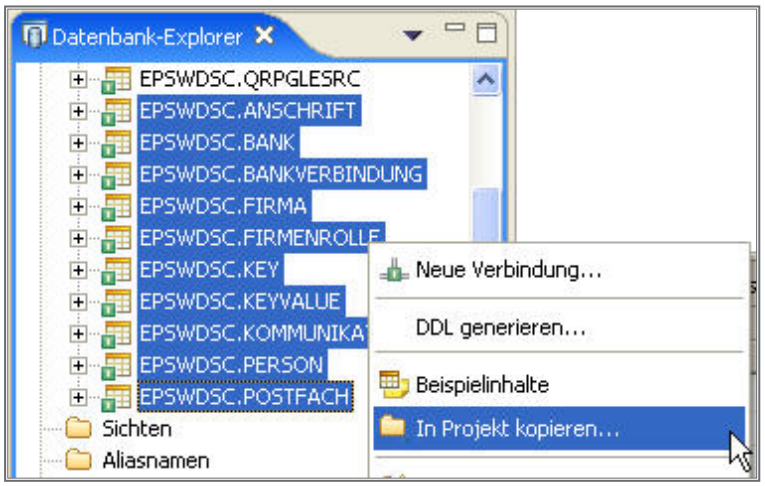
Hier kann noch auf abhängige Projekte verwiesen werden. Ich werde diesen Schritt hier jedoch ignorieren und auf „ Fertigstellen“ klicken.

Die Workbench zeigt dann in der Datendefinition folgendes Bild.

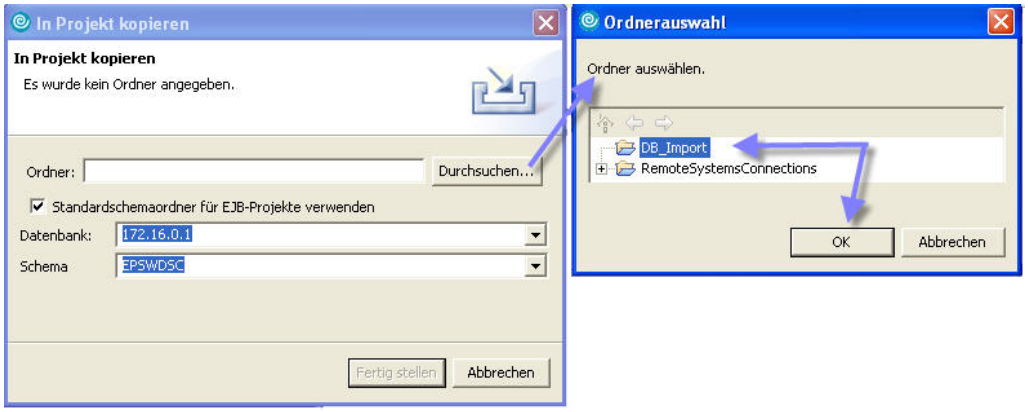
3.5.5
Seite 12

Die Sicht Datendefinition

Um alle im Datenbank-Explorer angezeigten Tabellen in dieses Projekt zu importieren, werden sie markiert und mit der rechten Maustaste wird ein Kontextmenu aufgerufen. Dort wird die Menuoption „In Projekt kopieren“ angewählt.



Tabellen importieren

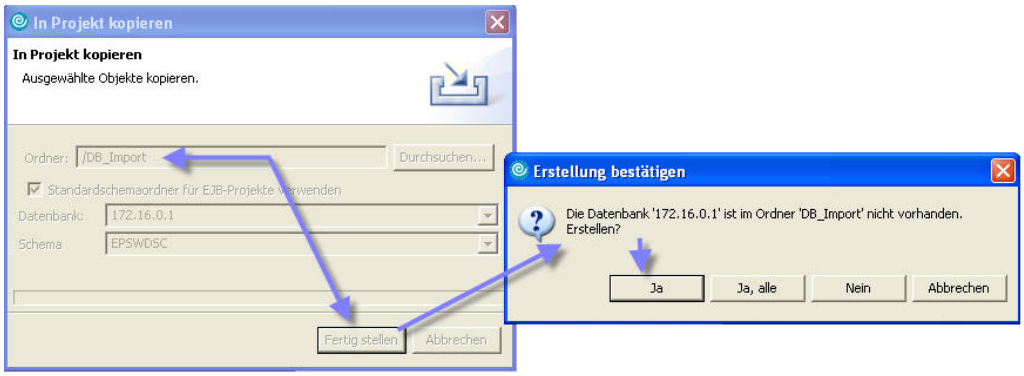


Zielprojekt bestimmen

Sie können das gewünschte Zielprojekt angeben, indem die vorhandenen Projektordner durchsucht werden oder der Name eines bestehenden Projekts wird direkt in das Feld „Ordner“ eingetragen. In diesem Beispiel /DB_Import.

Jetzt kann die Aktion durchgeführt werden. Im Zielprojekt ist noch keine Datenbank definiert worden. Eine entsprechende Information erfolgt und ich werde den Assistenten die benötigte Datenbank erstellen lassen. Sollte das Schema für die zu kopierenden Tabellen nicht vorhanden sein, wird der Assistent auch hier nachfragen.

Lassen Sie sich bitte nicht verwirren von diesen verschiedenen Kopier- und Erstellungsvorgängen. Hier werden keine neuen Objekte auf der iSeries erstellt, sondern lediglich Informationen und Konfigurationen in die Arbeitsumgebung übernommen.

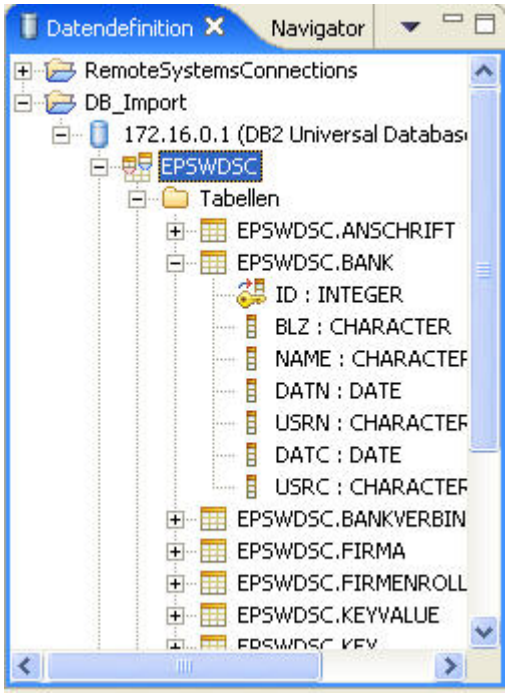


Eine Datenbankdefinition erstellen lassen

Erstellung des Zielschemas

Die Sicht „Datendefinition“ zeigt nun eine Kopie der Ansicht des Datenbank-Explorers. Die folgende Abbildung zeigt den Aufbau einer Tabelle, nachdem einzelne Knoten der Baumansicht in der Datendefinition expandiert wurden. Die Datenbank kann nun auf vielfältige Weise bearbeitet werden. Wenn die endgültige Struktur vorliegt, kann das Ergebnis natürlich auch wieder übertragen werden. Noch einmal: bis zur Rückübertragung der geänderten Informationen auf das Ursprungssystem bleibt die dort vorliegende Datenbank unverändert im alten Zustand.

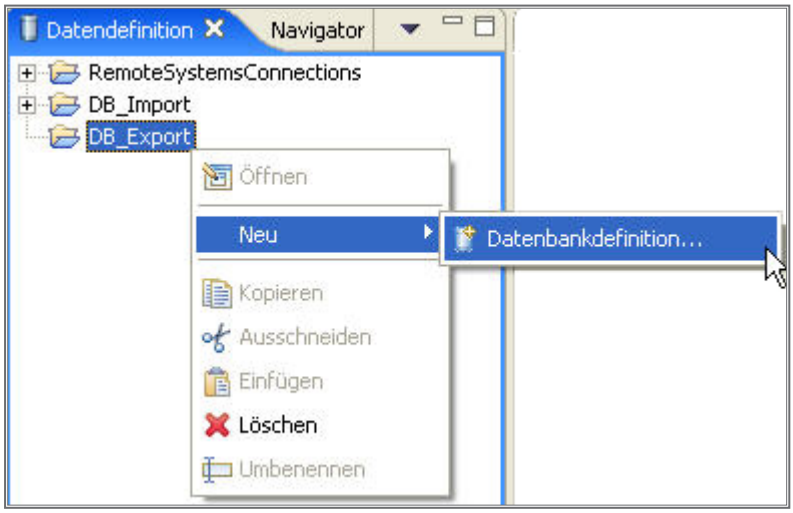
Ich werde die bestehenden Tabellen allerdings unberührt lassen und in den nächsten Schritten ein weiteres Datenbankprojekt aufbauen. Dies wird sich zwar auch auf das Schema EPSWDSC beziehen, aber ich möchte dort neue Tabellen erstellen. So kann ich Ihnen verdeutlichen, dass die Arbeit innerhalb eines Schemas durchaus weiter strukturiert werden kann, ohne bestehende Ordnungen im Umfeld berücksichtigen zu müssen.



Die importierten Tabellen

Die Frage ist, wie kann vorgegangen werden, wenn eine neue Datenbank erstellt werden soll. Nehmen wir an, ich möchte eine Reihe von Schulungen in der neuen Datenbank verwalten. Dort sollen Informationen über Bezeichnung, Thema, Dauer und Preis einer Schulung festgehalten werden, so wie auch Daten über Veranstaltungstermine und Orte. Das neue Projekt soll den Namen DB_Export erhalten, und Sie werden schon ahnen, welchen Hintergrund dieser Name haben wird.

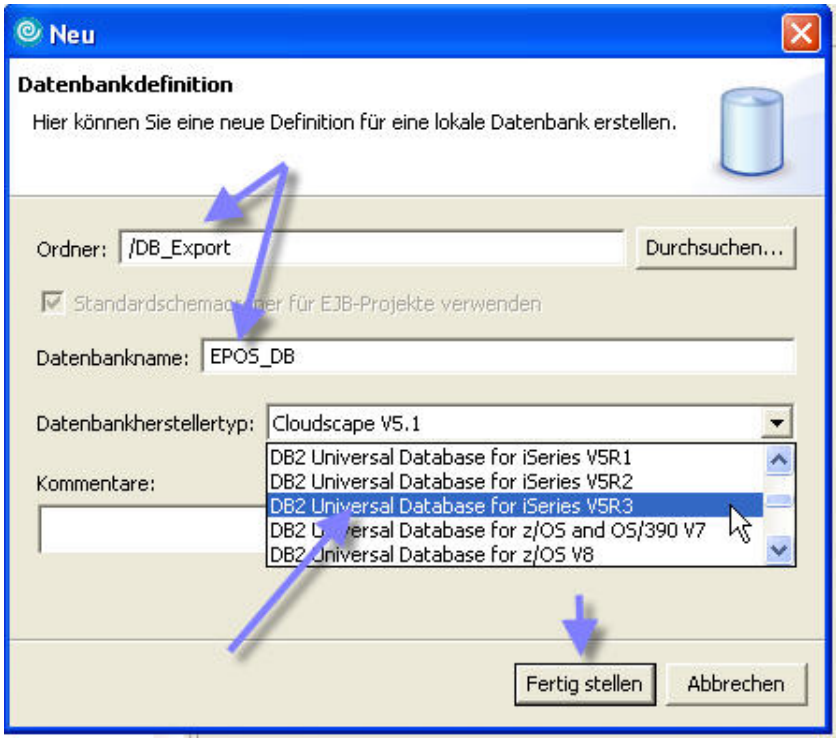
Ähnlich wie in der oben beschriebenen Vorgehensweise werde ich ein neues Projekt erstellen. Allerdings werde ich keine bestehenden Tabellendefinitionen einbinden, sondern eine neue Datenbankdefinition generieren. Diese Aufgabe wird in Angriff genommen, indem wieder über die rechte Maustaste ein Kontextmenu geöffnet wird und die Definition über die Option „Neu... Datenbankdefinition…“ gegangen wird.



Erstellen einer neuen Datenbankdefinition

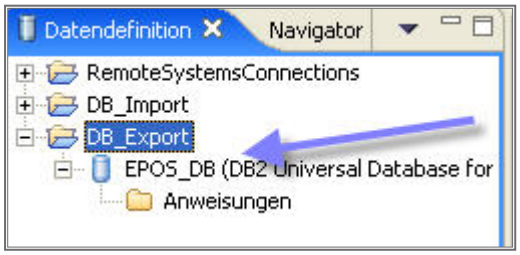
3.5.5
Seite 16

Folgender Dialog erscheint:



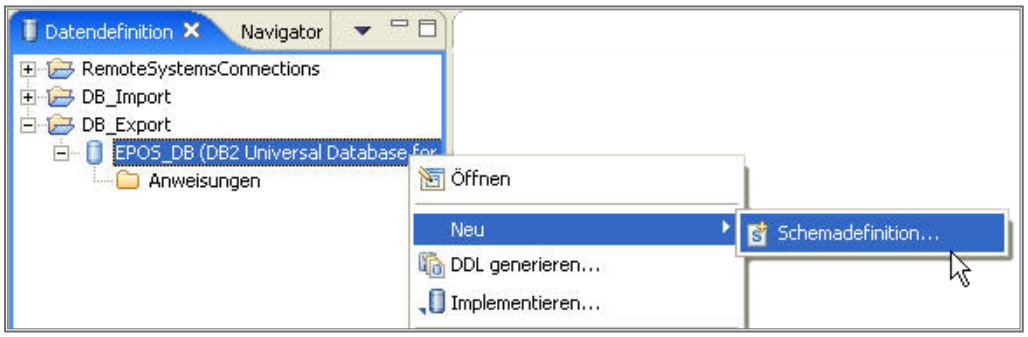
Festlegen der Datenbankdefinition

Hier muss darauf geachtet werden, dass das gewünschte Projekt im Feld „Ordner“ erscheint und dann vergebe ich einen Namen für die Datenbank. Dieser Name ist nur für die Arbeit in der Workbench von Bedeutung. Aus der Liste der Datenbankhersteller wird nun noch der passende Zieldatenbanktyp herausgesucht und wenn gewünscht, kann auch noch ein beschreibender Kommentar zu der Datenbank eingesetzt werden. Wenn das soweit geschehen ist, kann diese Aktion abgeschlossen werden. Anschließend bietet sich in der Sicht „Datendefinition“ folgendes Bild:



Sicht Datendefinitionen

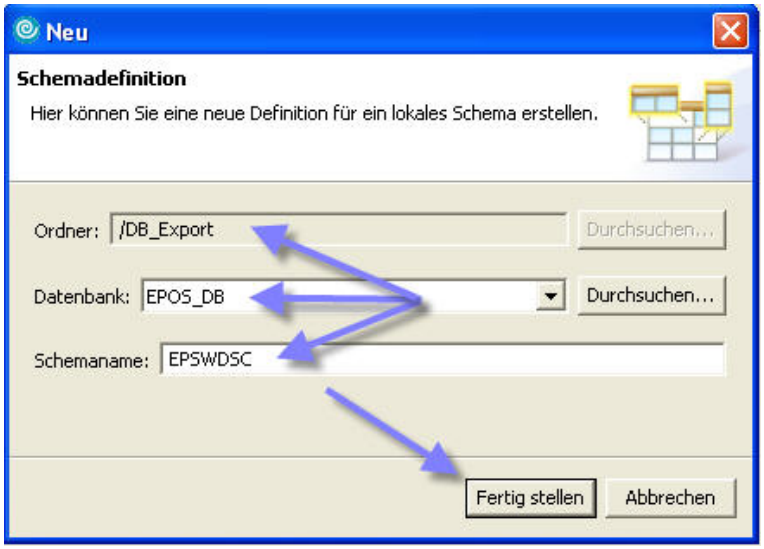
Die Datenbankdefinition liegt vor, aber es kann noch keine Tabelle definiert werden, da ein Schema als Container vorhanden sein muss. Die Datenbankdefinition wird mit der rechten Maustaste angeklickt und über den abgebildeten Weg erstelle ich ein Schema.



Erstellung eines Schemas

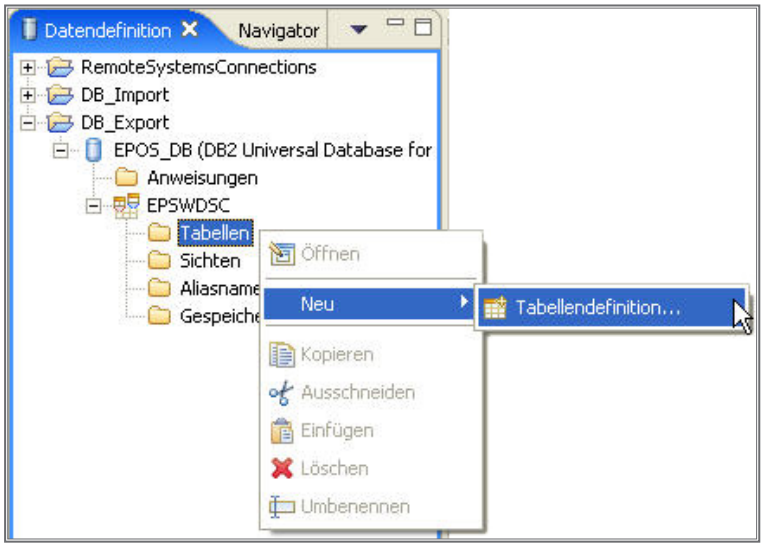
Das neue Schema wird auch den Namen EPSWDSC erhalten. Seltsam möglicherweise? Ein Schema mit diesem Namen existiert doch schon? Richtig! Der Gedanke dabei ist, dass ich beim abschließenden Übertragen der neuen Strukturen auf meine iSeries eben KEIN neues Schema erstellen möchte, sondern die Tabellen in das bestehende Schema EPSWDSC schreiben will.

3.5.5
Seite 18

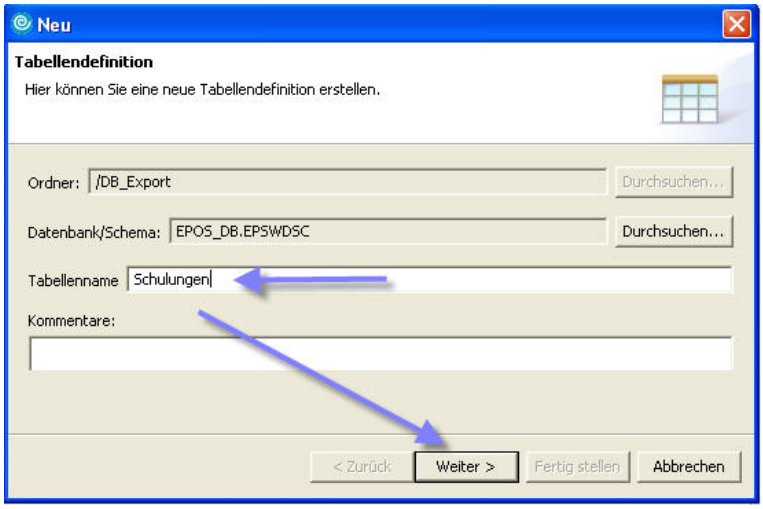


Benennung des neuen Schemas

Auch hier bitte wieder darauf achten, dass Projekt und Datenbank richtig angewählt worden sind. Wieder werfen wir einen Blick auf den aktuellen Aufbau der Datendefinition und klicken den Unterordner „Tabellen“ mit links an.

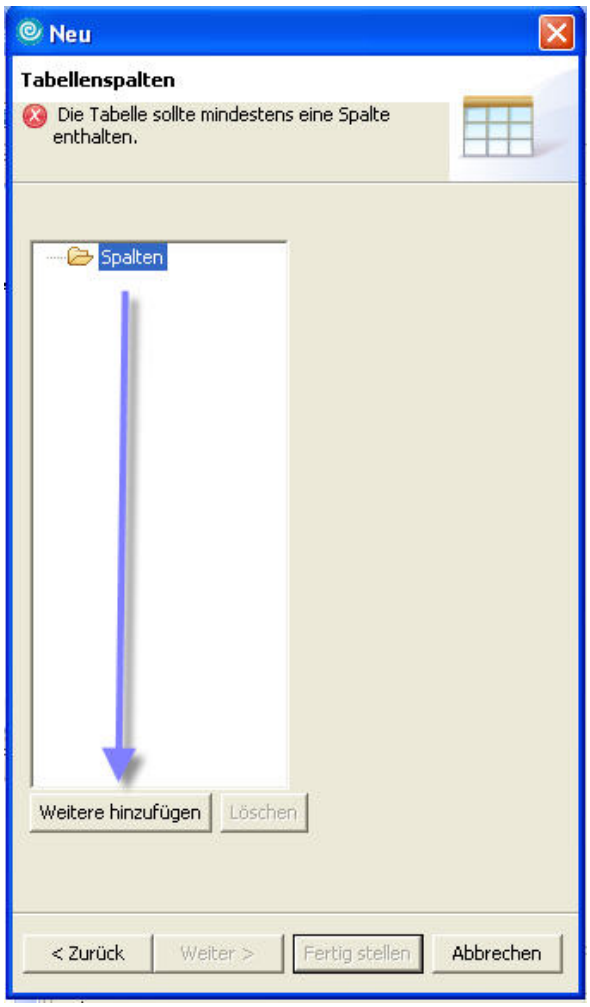


Nach Anlage eines neuen (lokalen) Schemas



Neue Tabellendefinition angeben

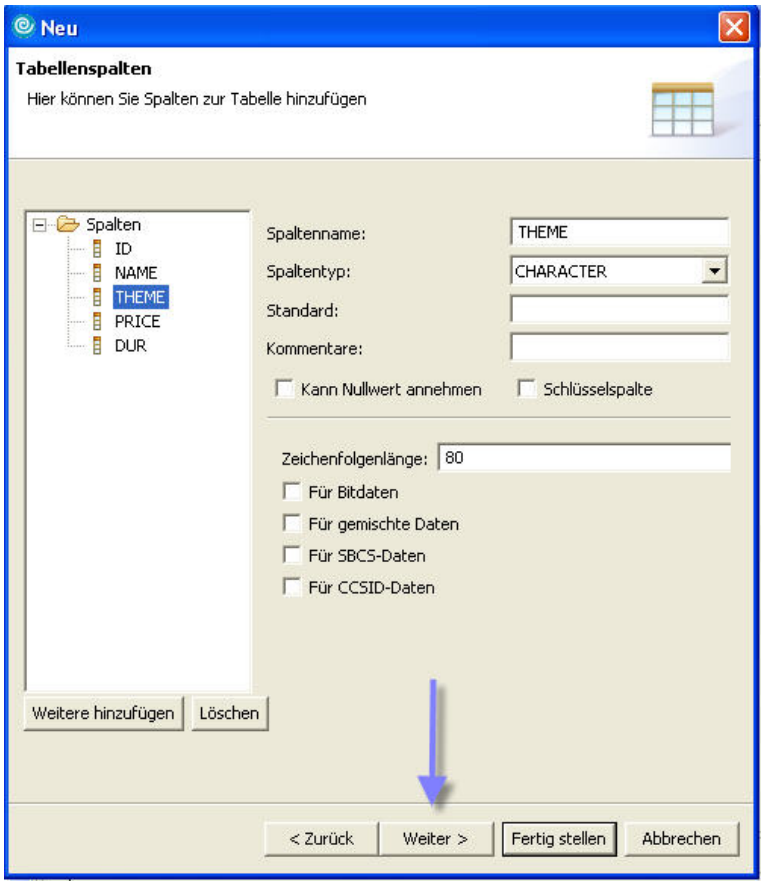
Hier trage ich nur den Namen einer neuen Tabelle, in diesem Fall Schulungen, ein und klicke „Weiter“ an. Die nächsten Dialoge befassen sich mit dem Aufbau der Tabelle.



Spalten erstellen

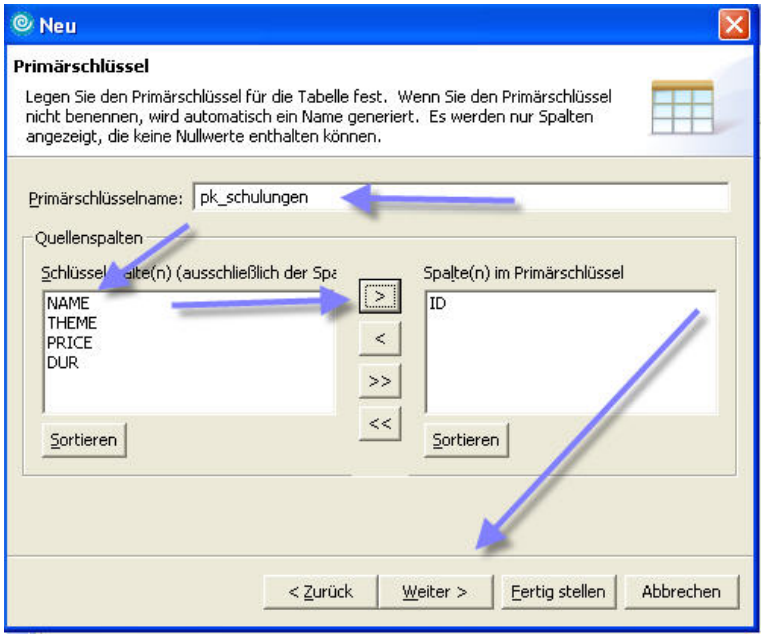
3.5.5
Seite 20

Mit Hilfe der Schaltfläche „Weitere hinzufügen“ können nun die einzelnen Spalten definiert werden. Änderungen sind allerdings auch im Nachhinein noch möglich, allerdings mit gewissen Einschränkungen.



Definition der Spalten

Der folgende Dialog übernimmt die Vergabe eines Primärschlüssels. Wenn im Feld „Primärschlüsselname“ kein Eintrag vorgenommen wird, generiert das System einen Namen. Ich plädiere hier wie immer für die Vergabe eines so genannten sprechenden Namens. Aus der angezeigten Liste der vorher erstellten Felddefinitionen wird das Feld markiert, das den Primärschlüssel repräsentieren soll und dann mit der durch einen Pfeil bezeichneten Schaltfläche übernommen. Auf diese Weise können auch zusammengesetzte Schlüssel erstellt werden. Danach geht es weiter zum nächsten Dialog des Assistenten.

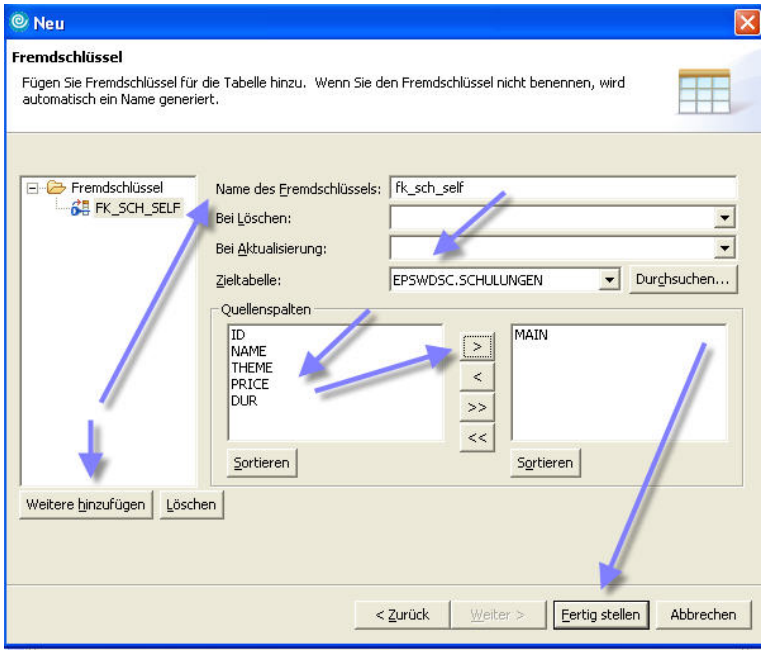


Primärschlüssel angeben

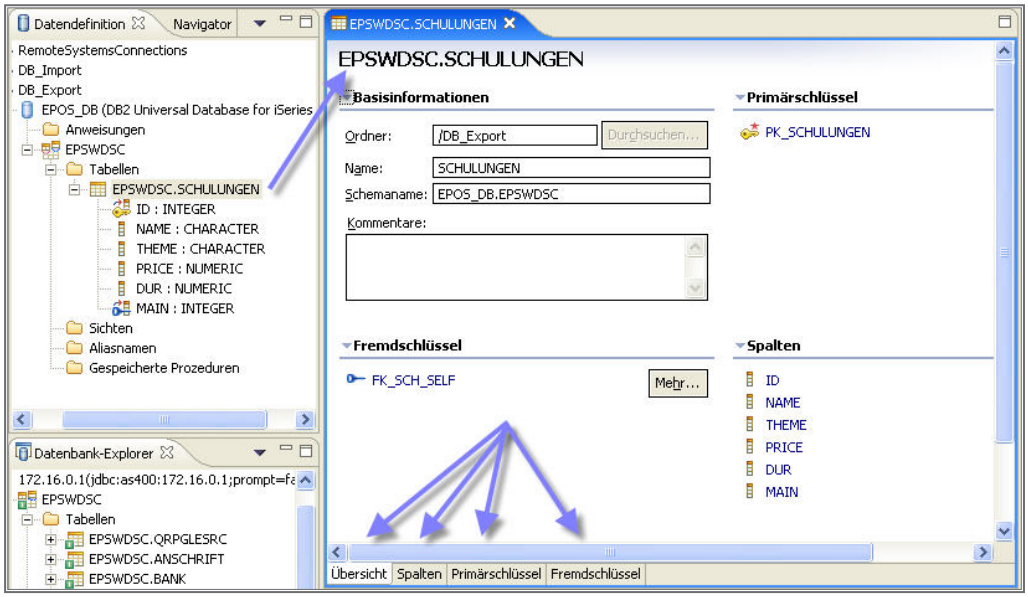
Jetzt können Fremdschlüssel erstellt werden. Sind mehrere Fremdschlüssel erforderlich, so wird die Schaltfläche „Weitere hinzufügen“ verwendet mit dem Funktionsprinzip, wie es oben bei der Definition der einzelnen Spalten beschrieben wurde. Der Name des Schlüssels sollte angegeben werden (sonst wird auch er generiert), natürlich die Zieltabelle, die hier über eine Combo-Box ausgesucht werden kann. Über die Schaltfläche „Durchsuchen“ kann auch ein Verweis auf eine Tabelle in einem anderen als dem aktuellen Datenbankprojekt hergestellt werden. Versuchen Sie sich einmal daran; dargestellt habe ich diesen Weg hier nicht.

Wenn nun alle Fremdschlüssel, die benötigt werden definiert sind, kann die Tabellendefinition in das Projekt übernommen werden (Fertigstellen). Die Sicht „Datendefinitionen“ zeigt nun auch die eben erstellte Tabelle und ein Doppelklick auf den Tabellennamen öffnet eine weitere Sicht, in der die Tabelle weiter bearbeitet werden kann. Ich zeige nur die Hauptsicht, denn eine Darstellung aller Varianten würde den Umfang dieses Buches bei Weitem sprengen.

3.5.5
Seite 22



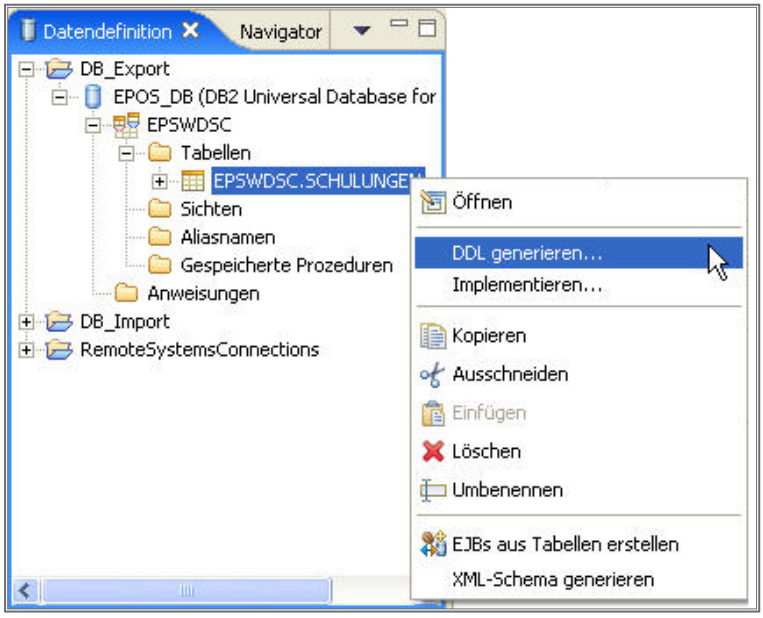
Fremdschlüssel angeben



Nach dem Doppelklick

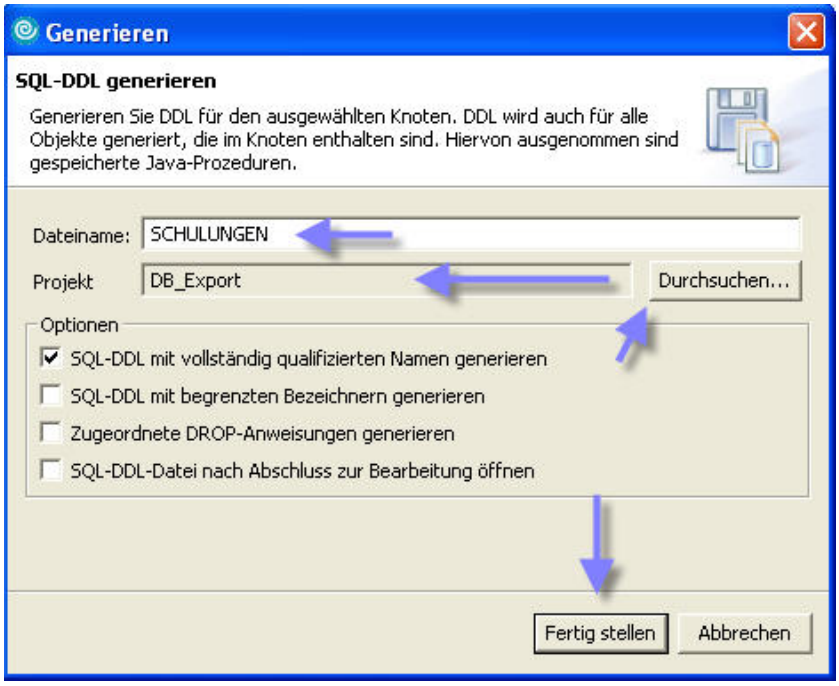
Wir haben hier zwar (noch?) nicht die Möglichkeit, direkt über die Gestaltung von Datenbankdiagrammen die Struktur einer Datenbank vollständig visuell zu entwerfen und zu bearbeiten, aber die Möglichkeiten des Tools sind schon sehr hilfreich bei der Arbeit in diesem Zusammenhang, denke ich. Wenn man sich für einen Überblick über die Strukturen der einzelnen Tabellen und anderen Objekte in seiner Datenbank interessiert und nicht relativ unkomfortabel in jede einzelne Definition hineinschauen möchte, so besteht die Möglichkeit, sich ein Script generieren zu lassen (DDL). Dies kann natürlich auch dazu verwendet werden, die Datenbank auf anderen Systemen zu implementieren. Wie aber geht das?

Bitte markieren Sie die Objekte Ihrer Datenbank, für die ein Script erzeugt werden soll und klicken Sie anschließend mit der rechten Maustaste darauf. Im folgenden Kontextmenu wählen Sie die Option „DDL generieren…“ aus.



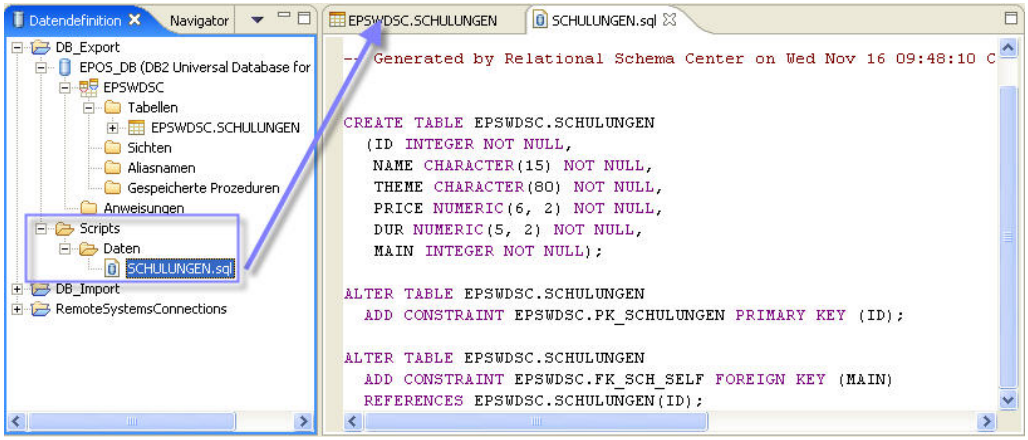
DDL generieren

Danach geben Sie noch an, welche Bezeichnung die Datei erhalten soll und in welchem Projekt sie abgelegt werden soll. Projekte können auch hier wieder mit der Schaltfläche „Durchsuchen…“ markiert werden.



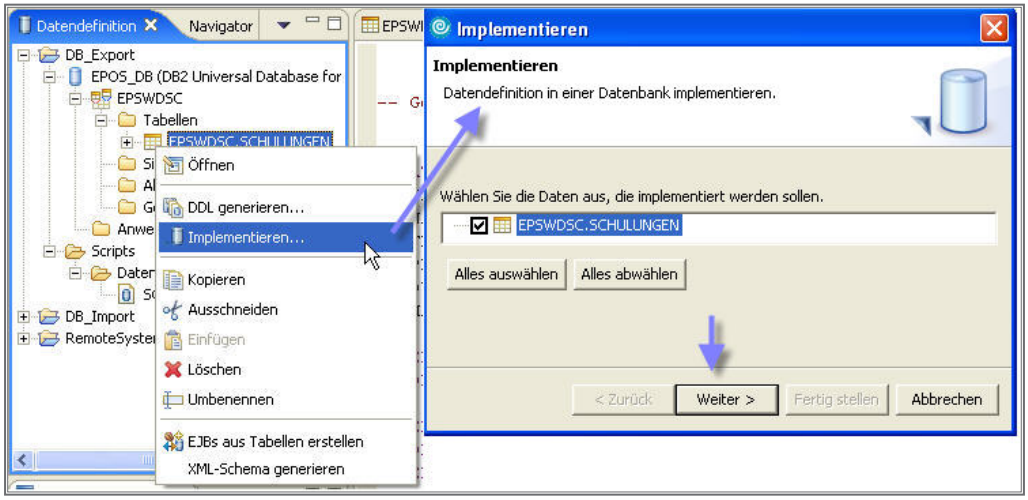
Datei benennen

In der Sicht Datendefinition werden Sie innerhalb Ihres Datenprojekts nun einen neuen Ordner „Scripts“ finden, der die generierte DDL-Datei beinhaltet. Nach einem Doppelklick öffnet sich eine Editorsicht, in der die Quelle auch bearbeitet werden kann. Bei der Erstellung von DDL müssen Sie sich nicht auf einzelne Tabellen beschränken. Auch für das gesamte Schema können Sie die eben beschriebenen Schritte anwenden.



Geöffnete DDL-Datei

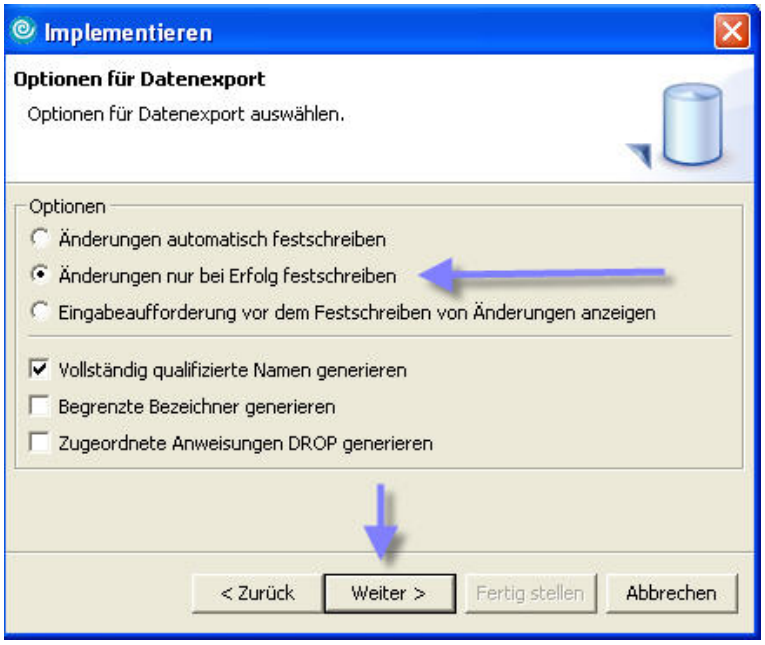
Als nächstes möchte ich die Tabelle auf meiner iSeries erstellen. Die erste Variante läuft über die Definition selbst. Die zu übertragende Definition wird markiert, ein Kontextmenu geöffnet und die Option „Implementieren…“ ausgewählt. Dabei ergibt sich folgendes Bild: Nachdem die Liste überprüft wurde, klicke ich auf „Weiter…“.



Tabellendefinition implementieren

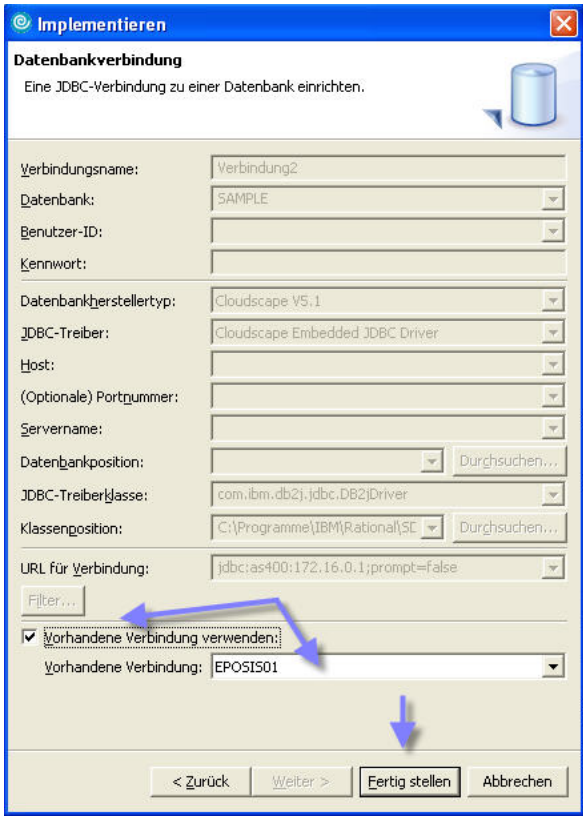
Im nächsten Dialog sollten Sie die Änderungen nur bei Erfolg endgültig festschreiben lassen.

Der letzte Radio-Button in der Liste sollte aktiviert werden, wenn die zu implementierende Tabelle (oder das Objekt) bereits in der Zieldatenbank besteht. Danach kann dieser Dialog mit „Weiter...“ verlassen werden.



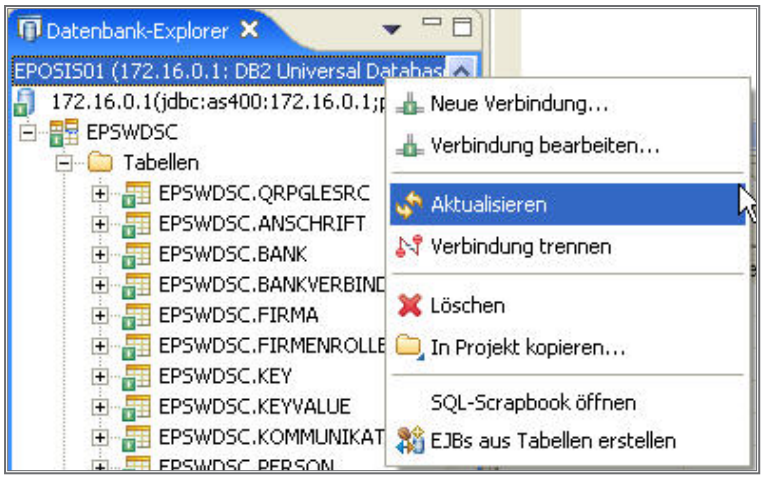
Weitere Optionen angeben

Nun muss noch die Datenbankverbindung angegeben werden, über die die Implementierung erfolgen soll. In diesem Fall besteht schon eine Verbindung, die ich beim Import weiter oben verwendet habe. Über die CheckBox „Vorhandene Verbindung verwenden…“ kann ich mir die Arbeit erleichtern.

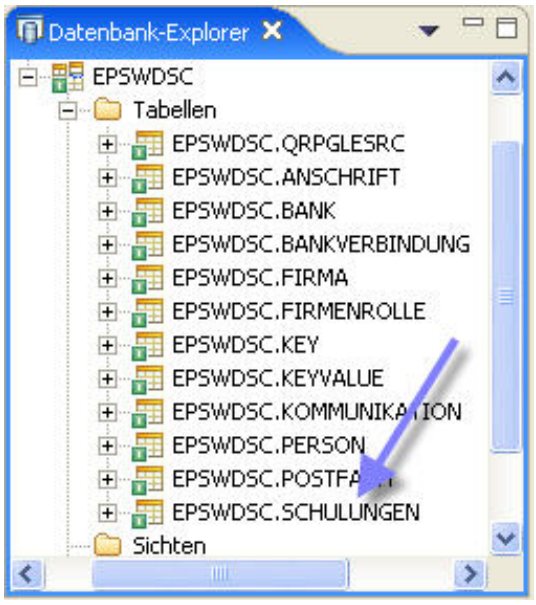


Verbindung angeben

Wenn nach dem Klick auf „Fertig stellen“ keine Fehlermeldungen angezeigt wurden, ist die Tabelle auf dem Zielsystem, das durch die Verbindung definiert wurde, erstellt worden. Herzlichen Glückwunsch! Wie kann das nun kontrolliert werden? Man braucht nur die Datenbankverbindung zu aktualisieren. Die Tabelle muss dann in der Objektliste in der Sicht Datenbank-Explorer erscheinen. Markieren Sie die Verbindung und aktualisieren Sie über ein Kontextmenu.



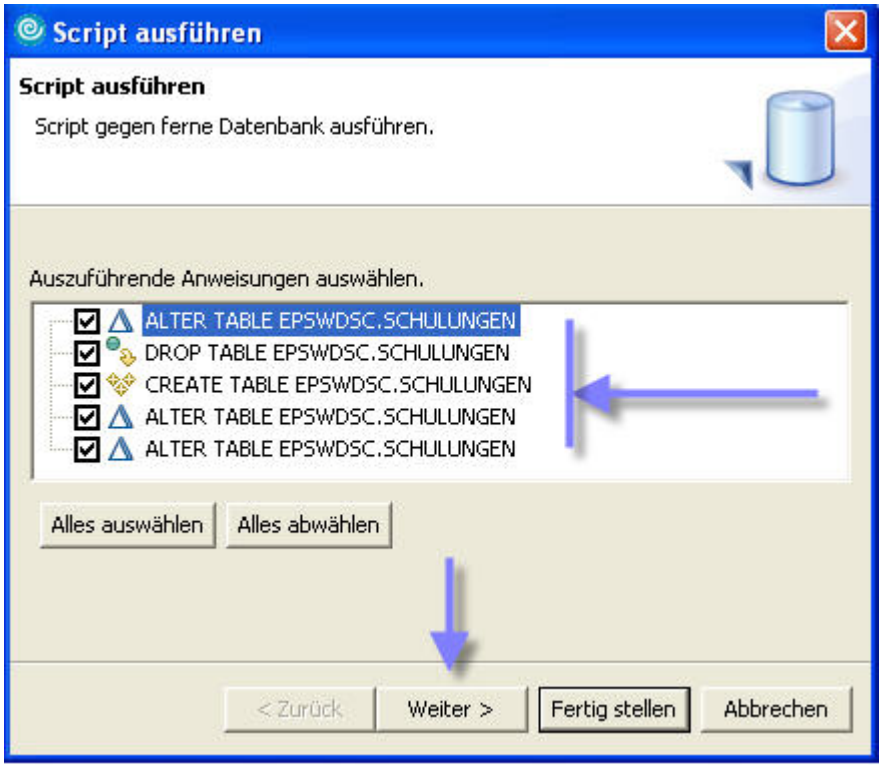
Verbindung aktualisieren



Die neue Tabellenliste nach der Aktualisierung

3.5.5
Seite 28

Ich möchte nun die Implementierung über das Script vornehmen. Dazu erstelle ich es noch einmal, allerdings nehme ich diesmal die Drop-Anweisungen in die Datei mit auf, da die Tabelle ja schon existiert. Das Script wird nun ebenfalls über das Kontextmenü implementiert. Der erste Dialog unterscheidet sich von der Abfolge, die eben beschrieben wurde. Hier können noch verschiedene Aktionen des Scripts deaktiviert werden, ganz nach Wunsch.



Aktionen wählen

Der weitere Ablauf ist dann mit der Implementierung über die Definition der Tabelle selbst zu vergleichen. Auch kann zum Abschluss eine Aktualisierung der Datenbankverbindung vorgenommen werden.

Eclipse Datenbank-Tools

Die Tatsache, dass System-i-Anwendungsentwickler mit den Nachfolgeprodukten von WDSC – etwa mit RDi (Rational Developer for IBM i) – keine Möglichkeit mehr haben, Datenbanken bzw. deren Inhalte direkt zu bearbeiten, darf als unglücklich bezeichnet werden. Wunderlich mutet diese Tatsache auch deshalb an, weil IBM nicht ohne Grund die Verwendung SQL-basierter Datenbanken der Nutzung klassischer, auf physischen Dateien basierender Datenbanken vorzieht.

Während in WDSC die Datenperspektive zum Einsatz von SQL genutzt werden konnte, fehlt diese in RDi-Paketen.

Nun hat IBM die Entwicklerwerkzeuge auf der Basis der Eclipse-Funktionen und Plug-in-Technologien erstellt. Das bietet uns die Möglichkeit, Erweiterungen von Basisprodukten – wie RDi – durch die Installation zusätzlicher Plug-ins zu nutzen.

Ein solches Plug-in wird direkt von Eclipse geliefert und bietet unterschiedliche Funktionen zum Arbeiten mit Datenbanken und SQL: Dabei ist der Einsatz des Eclipse Data Tools nicht allein auf die Verwendung mit der DB2/UDB des System i beschränkt, sondern auch für die Nutzung mit anderen Datenbanken geeignet.

Die Eclipse Data Tools Platform (kurz DTP) kann kostenlos von den Eclipse-Webseiten heruntergeladen werden. Der folgende Link verweist auf die entsprechenden Eclipse-Webseiten:

http://www.eclipse.org/datatools/downloads.php

Achten Sie UNBEDINGT darauf, dass Sie die korrekte Version von DTP herunterladen und installieren. Insbesondere die Kompatibilität mit der installierten Eclipse-Version ist wichtig. Die aktuell genutzte Version können Sie in RDi wie folgt ermitteln:

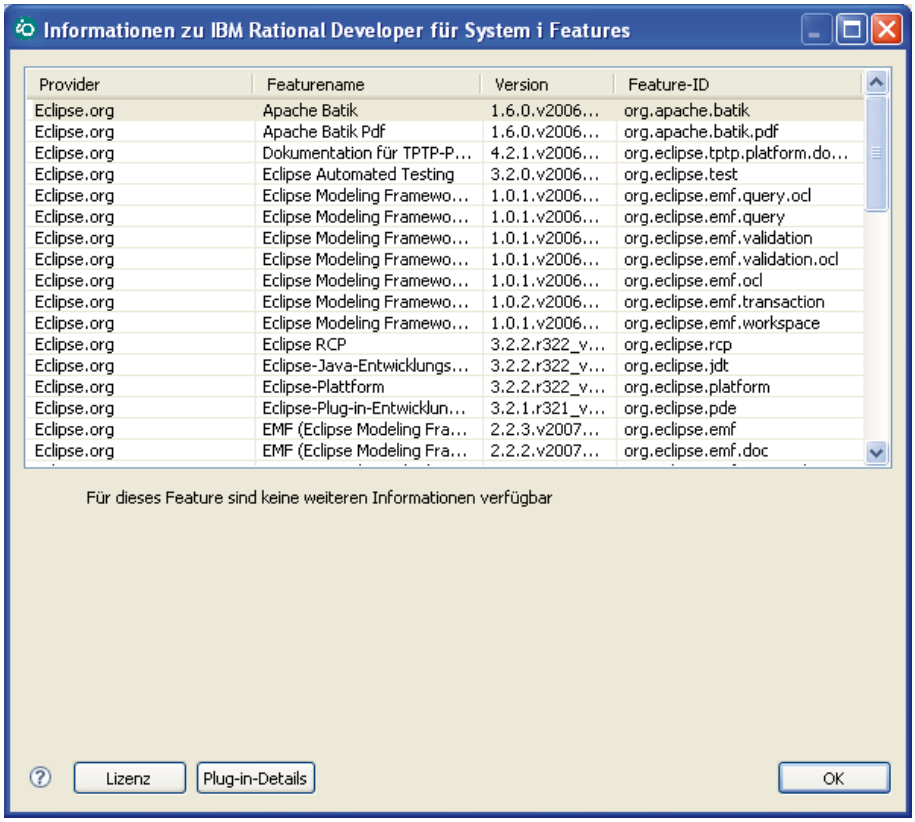
- Selektieren Sie die Menüoption „Hilfe / Info über IBM Rational Developer for System i“.

3.5.5.1
Seite 2

Daraufhin erscheint eine Anzeige wie sie in der folgenden Abbildung zu sehen ist:

Installierte Version

- Wählen Sie die Option „Featuredetails“.
  Damit erhalten Sie eine Übersicht der installierten Features.
  Unter anderem finden Sie dort auch die installierte Eclipse-Version.
  In unserem Fall ist dies die Version 3.2.2.



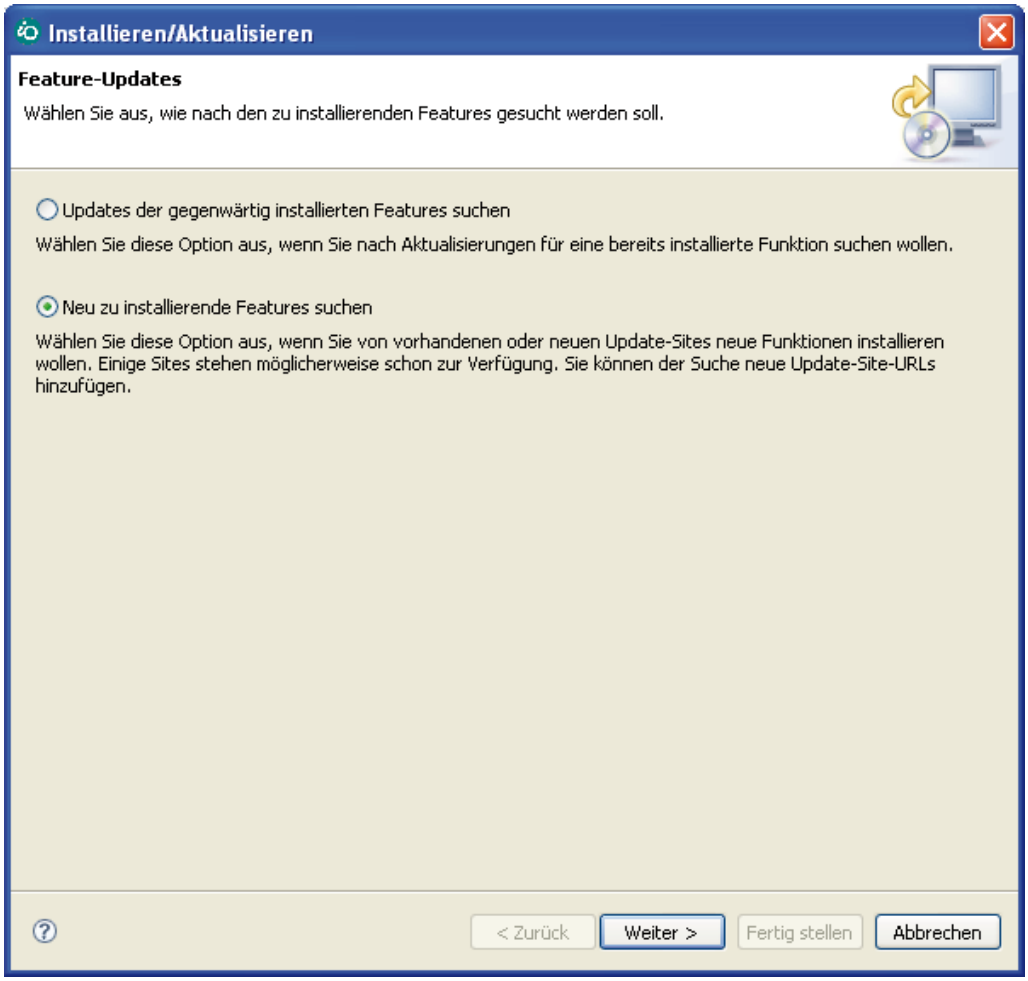
Eclipse-Version

Das bedeutet, dass wir bei dieser RDi-Installation eine DTP-Version nutzen können, die mit der Eclipse-Plattform 3.2.2 kompatibel ist.

Nach dem Herunterladen des DTP muss es entpackt werden. Das Ziel für das Entpacken kann beliebig bestimmt werden.

Nun bereiten wir das Installieren vor. Dazu wählen wir zunächst innerhalb von RDi die Menüoption „Hilfe / Software Updates / Suchen und installieren".

Daraufhin wird ein neues Fenster geöffnet, in dem wir die Option „Neu zu installierende Features suchen" anklicken.



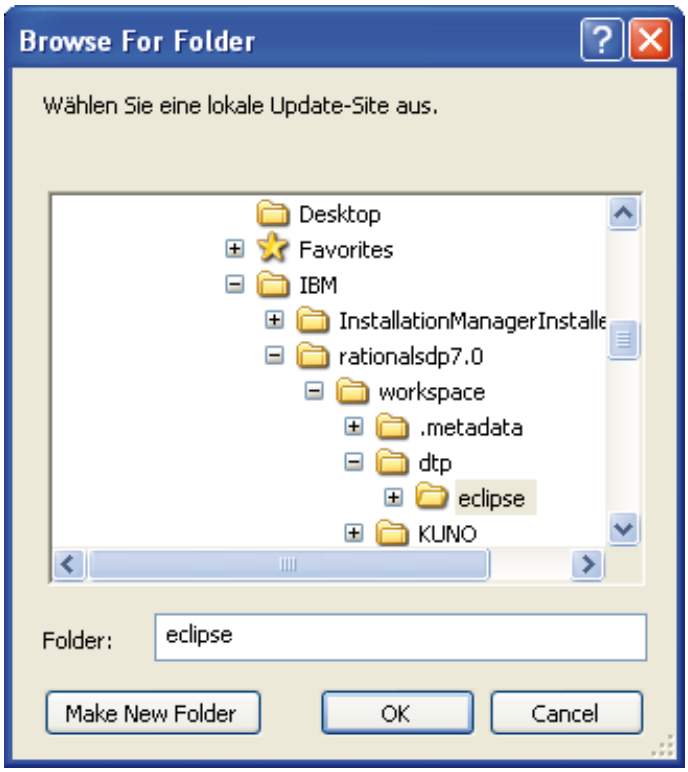
Neu zu installierende Features suchen

Bestätigen Sie die Auswahl mit einem Klick auf die Schaltfläche „Weiter".

Nun werden die verfügbaren Update-Bereiche angezeigt. Dort fehlt der DTP-Bereich noch. Durch einen Klick auf die Schaltfläche „Neue lokale Site" kann dieser für die Installation hinzugefügt werden.

3.5.5.1
Seite 4

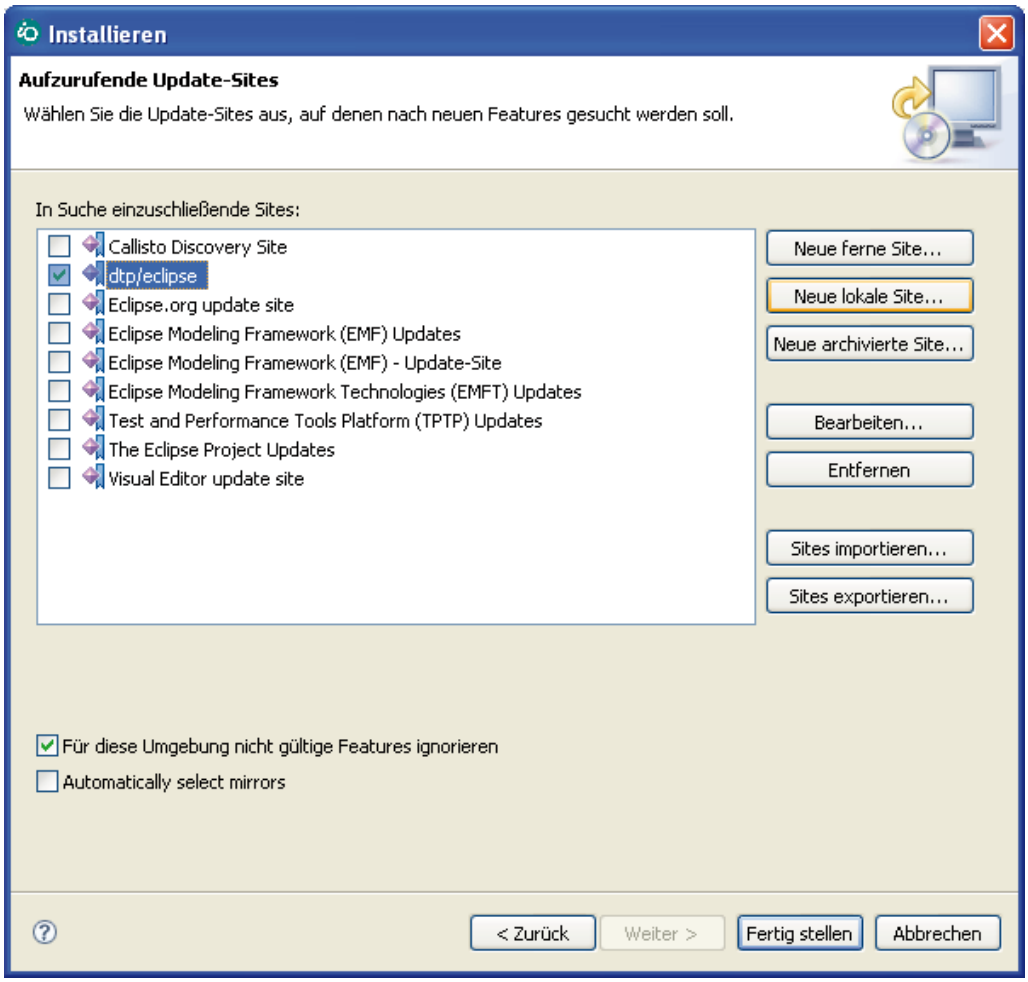
- Klicken Sie auf die Schaltfläche „Neue lokale Site“.
- Wählen Sie den Speicherort der entpackten Installationsdaten für DTP aus. Wählen Sie als Verzeichnis den Folder „Eclipse“.



Auswahl Eclipse-Installation

- Bestätigen Sie die Auswahl mit einem Klick auf „OK“.

Damit erscheint jetzt der DTP-Bereich auch in der Auflistung der zu installierenden Sites:

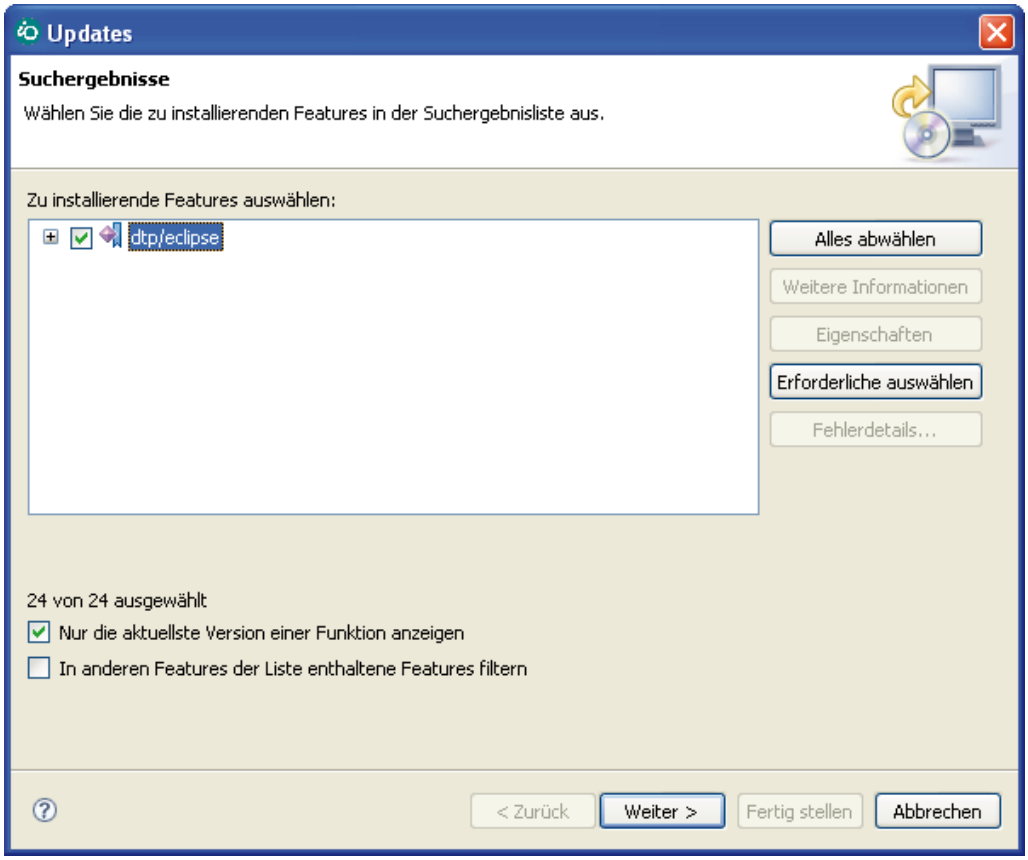


Neue Update Site

Der Installationsprozess wird jetzt mit einem Klick auf die Schaltfläche „Fertig stellen“ gestartet.

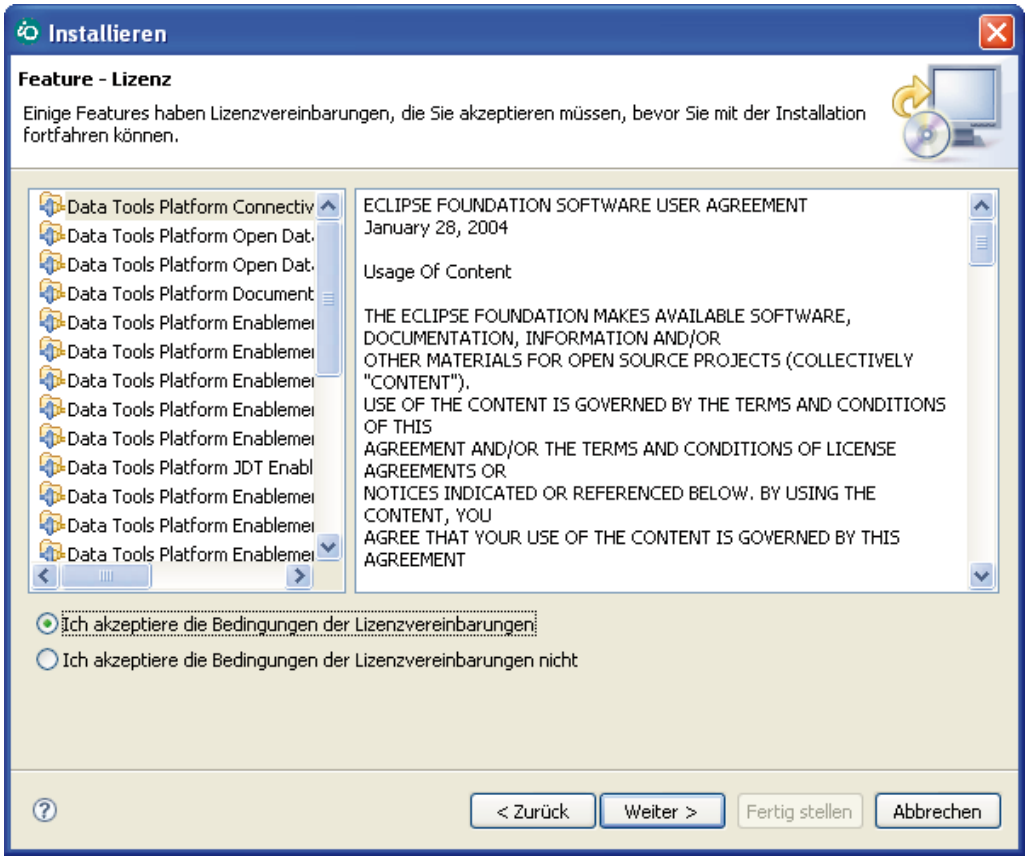
3.5.5.1
Seite 6

Nach einem kurzen Vorbereitungslauf wird das neu zu installierende Feature aufgelistet. Zur Installation muss es noch markiert werden. Dazu genügt ein Klick in das Kästchen links neben der Bezeichnung des Features.



Zu installierende Features

Mit einem Klick auf die Schaltfläche „Weiter“ werden zunächst die Lizenzbestimmungen für die Eclipse Data Tools angezeigt. Diese müssen für die Fortsetzung der Installation akzeptiert werden.

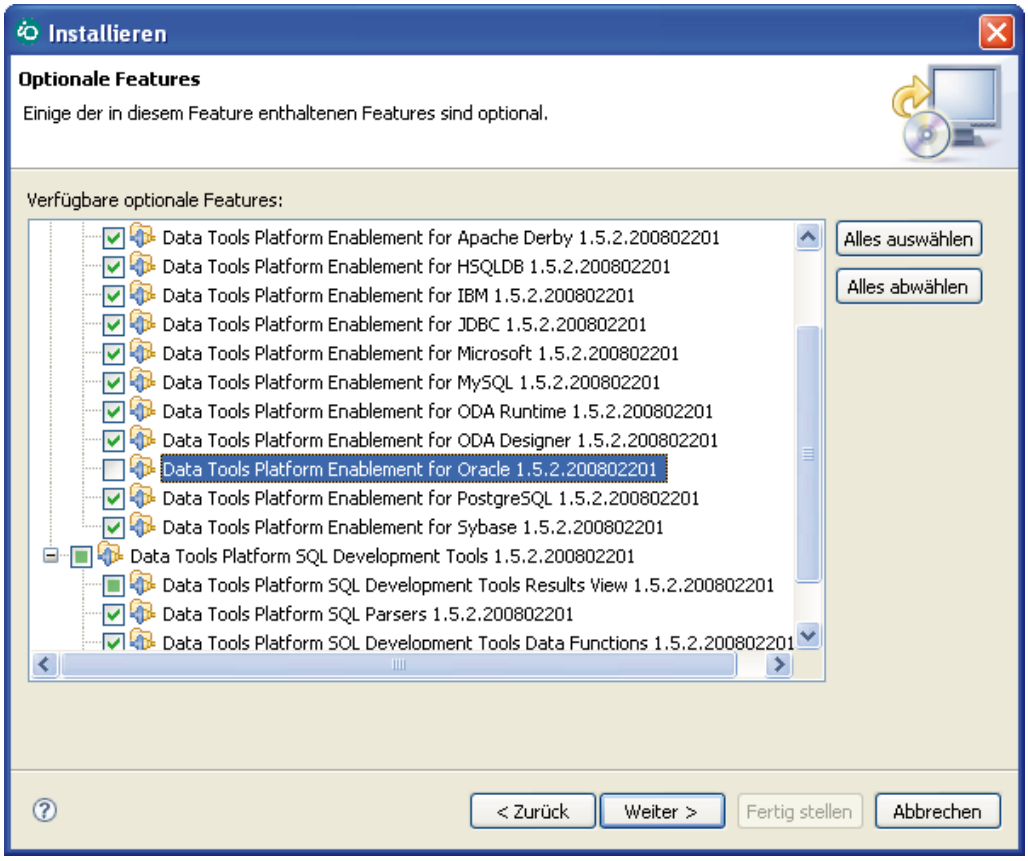


Lizenz

- Akzeptieren Sie die Lizenzbestimmungen
- Bestätigen Sie mit einem Klick auf „Weiter“.

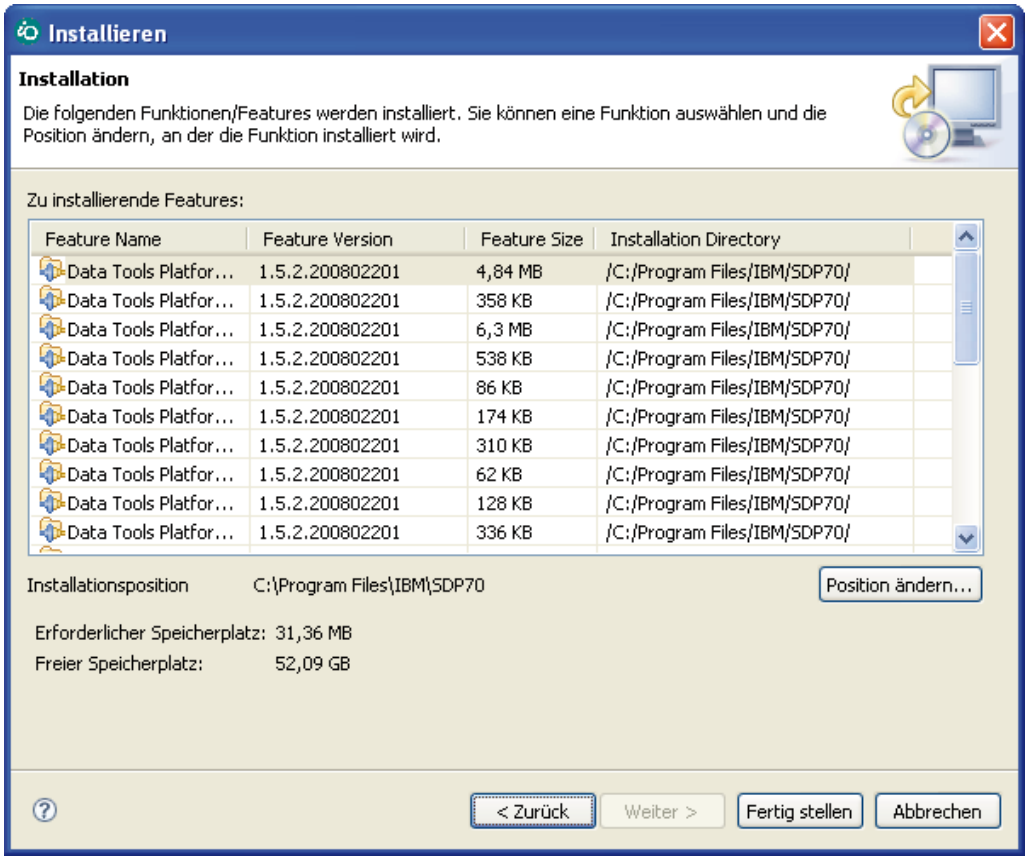
Die optional zu installierenden Komponenten werden aufgelistet. Unter anderem finden wir hier verschiedene Datenbankoptionen. Sollten Sie beispielsweise keine Oracle-Datenbanken mit RDi nutzen wollen, dann kann diese Option deaktiviert werden. So habe ich es exemplarisch auch in der folgenden Abbildung gemacht.

3.5.5.1
Seite 8



Data-Tools-Auswahl

Mit einem Klick auf „Weiter“ werden nun die zu installierenden Details angezeigt. Der eigentliche Installationsprozess wird mit einem abschließenden Klick auf die Schaltfläche „Fertig stellen“ gestartet.



Installationsposition

Nach der Installation muss RDi erneut gestartet werden. Dies kann wahlweise sofort oder auch später erfolgen. Um die neuen Funktionen nutzen zu können, muss der Neustart aber erfolgt sein.

Damit ist die Grundinstallation von DTP abgeschlossen. Jetzt müssen wir die Treiber für die zu verwendenden Datenbanken definieren. Diese werden über die Benutzervorgaben gesteuert. Die Verwaltung der Benutzervorgaben erfolgt über den Menüeintrag „Fenster / Benutzervorgaben“.

- Wählen Sie die Menüoption „Fenster / Benutzervorgaben“.

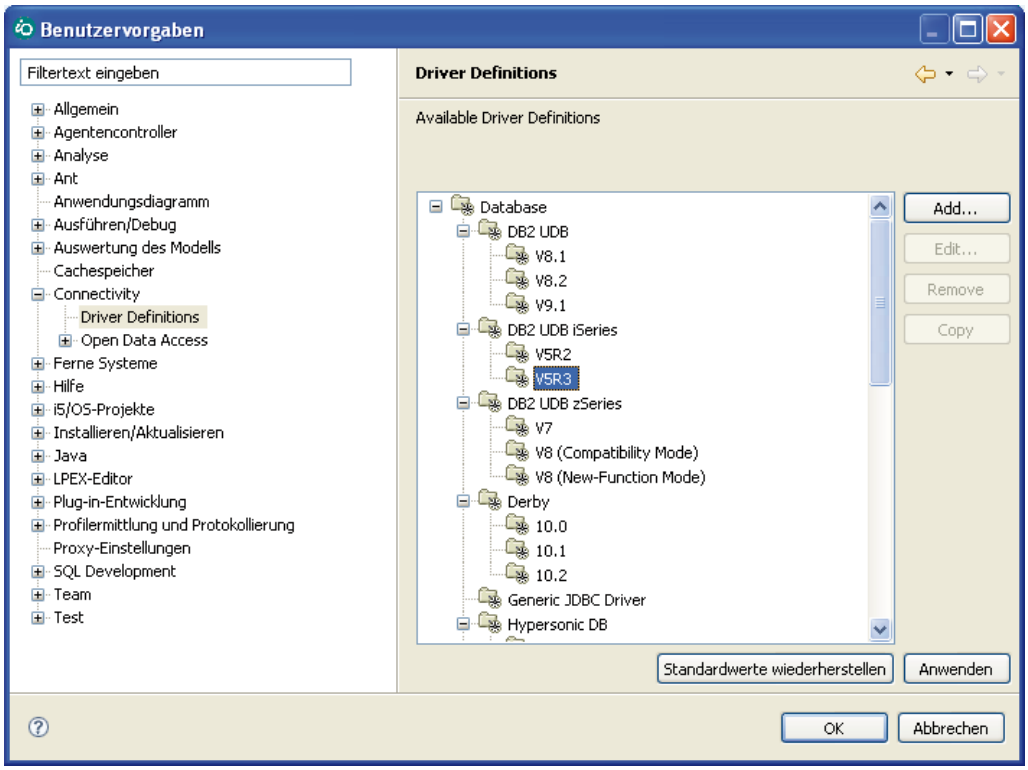
In den Benutzervorgaben sind verschiedene Einstellungen möglich. Die Treiberangaben für DTP finden wir unter dem Abschnitt „Connectivity / Driver Definitions“.

- Erweitern Sie den Eintrag „Connectivity“.
- Wählen Sie die Option „Driver Definitions“.
- Selektieren Sie die für Ihr System zutreffende Version der DB2/UDB für System i bzw. iSeries.

3.5.5.1
Seite 10

Anmerkung

Je nach verwendeter Version von DTP kann es sein, dass nicht die aktuelle Version der DB2/UDB angezeigt wird. Nutzen Sie in einem solchen Fall den aktuellsten der aufgelisteten Datenbanktreiber!

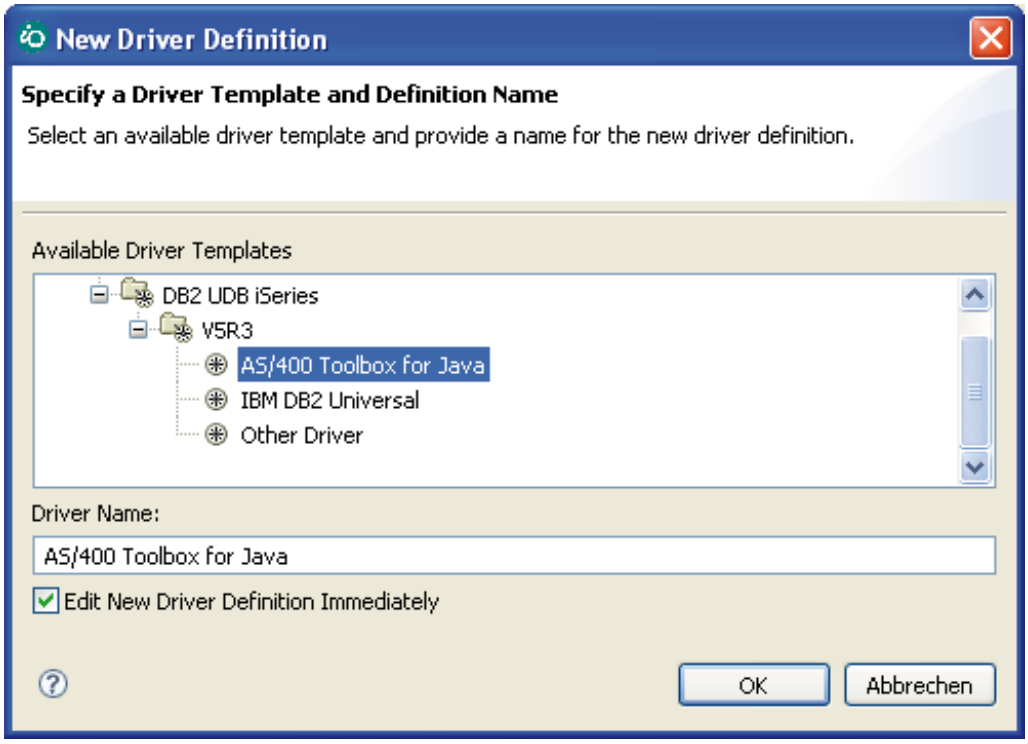


DB2 UDB iSeries

- Klicken Sie nach der Auswahl des Treibers auf die Schaltfläche „Add“.

Nun werden die verfügbaren Treiber aufgelistet. In unserem Beispiel stehen uns verschiedene Treiber zur Verfügung. Wir wählen den Treiber „AS/400 Toolbox for Java“ aus.

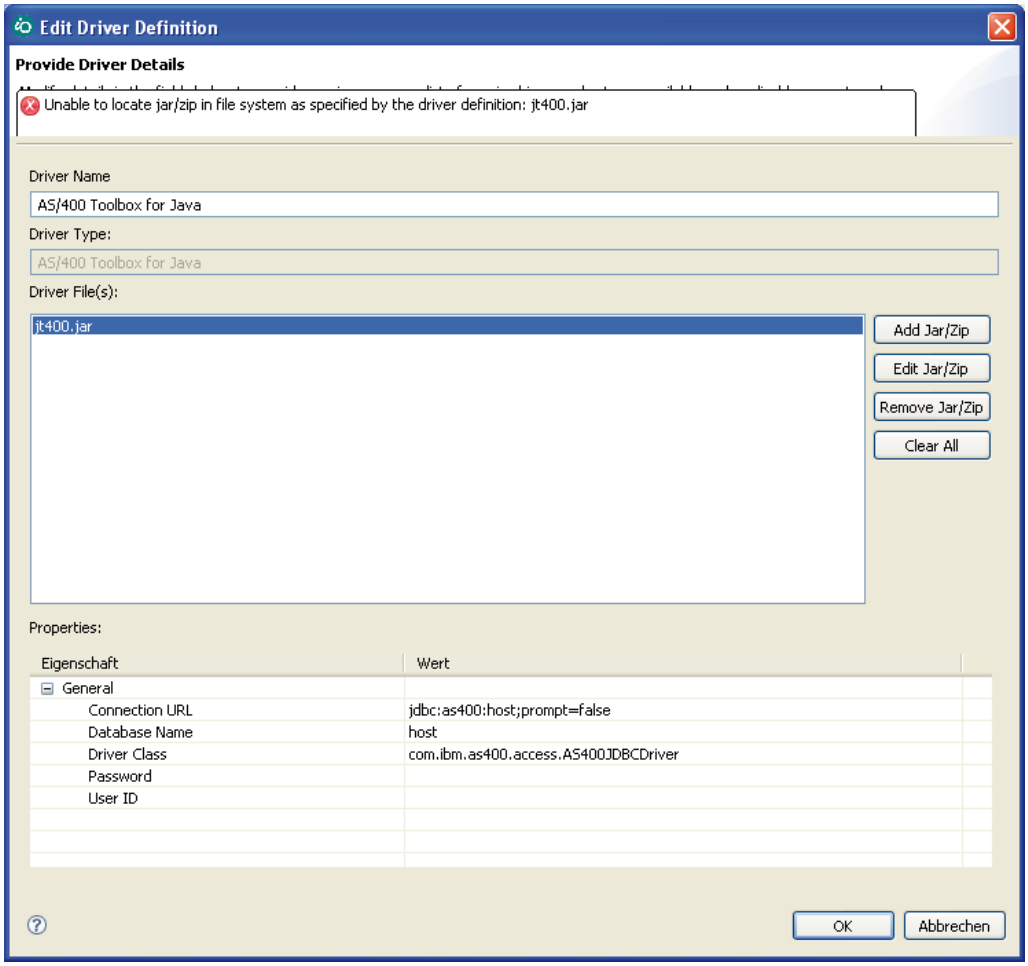
- Erweitern Sie den Eintrag der Version der DB2/UDB (in unserem Beispiel V5R3) durch einen Klick auf das „+“-Zeichen.
- Selektieren Sie in der Auflistung der verfügbaren Treiber den Treiber „AS/400 Toolbox for Java“. Damit wird dieser in das Feld „Driver Name“ übernommen.
- Aktivieren Sie die Option „Edit New Driver Definition Immediately“.
- Bestätigen Sie die Auswahl mit einem Klick auf die Schaltfläche „OK“.



Toolbox für Java

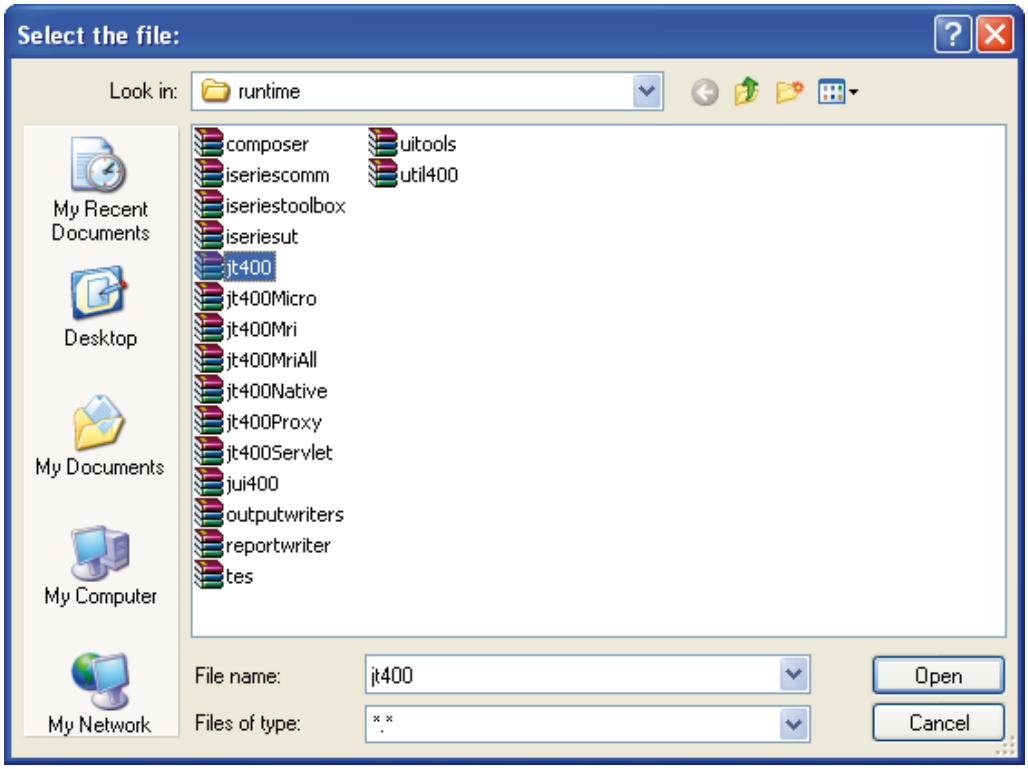
Leider ist die Location dieses Treibers variabel. Das erklärt auch die Fehlermeldung, die wir in der nächsten Anzeige sehen. „Unable to locate jar/zip in file system“ sagt hier aus, dass die Location nicht angegeben wurde. Das ändern wir jetzt. Dazu geben wir den Speicherort der jt400.jar an.

3.5.5.1
Seite 12



Fehlerhinweis

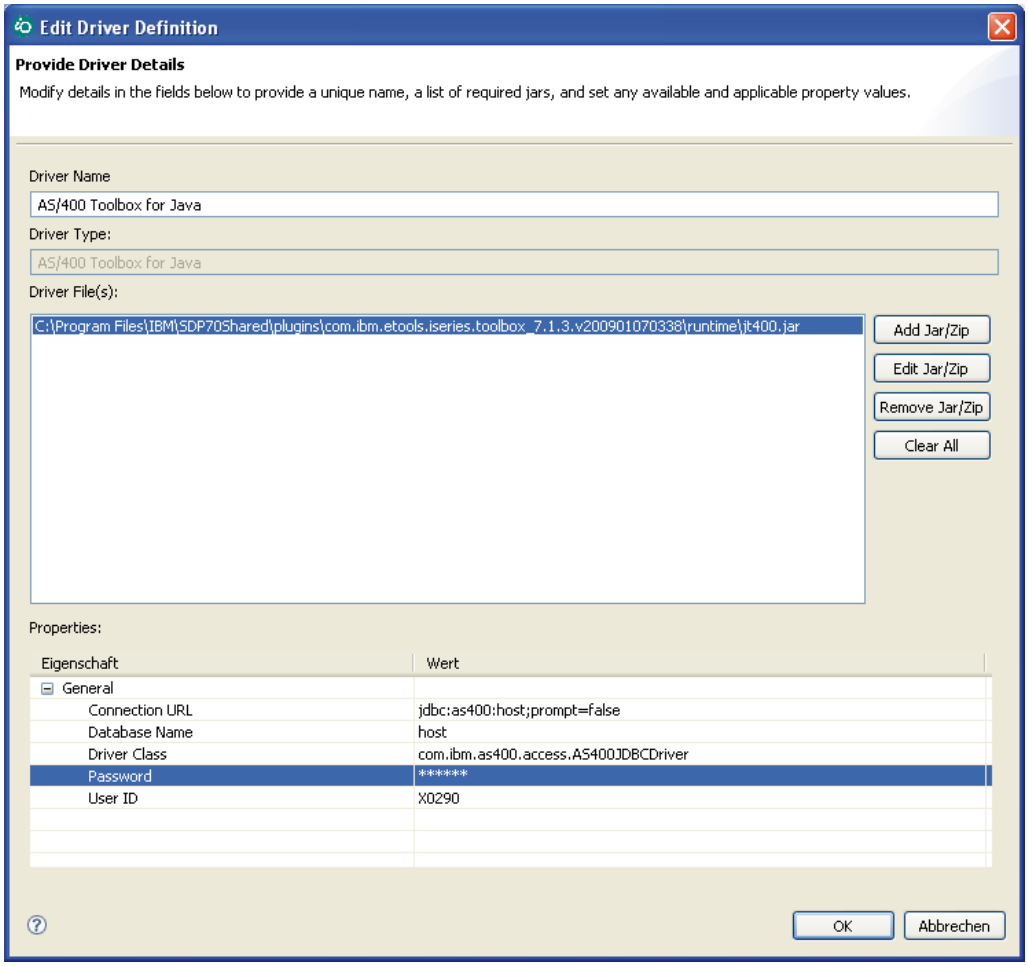
- Markieren Sie den Eintrag „jt400.jar“ in der Auflistung der Driver Files.
- Klicken Sie auf die Schaltfläche „Edit Jar/Zip“.
- Suchen Sie eine möglichst aktuelle Version der jt400.jar und übernehmen Sie diese.



jt400-Auswahl

Jetzt muss noch der Zugriff auf die DB2/UDB definiert werden. Dazu geben wir den Datenbanknamen, den Benutzer und das zugehörige Kennwort im Bereich „Properties“ ein. Ein Beispiel dazu finden Sie in der folgenden Abbildung.

3.5.5.1
Seite 14



Treiberdetails

- Bestätigen Sie die Eingaben mit einem Klick auf „OK“.

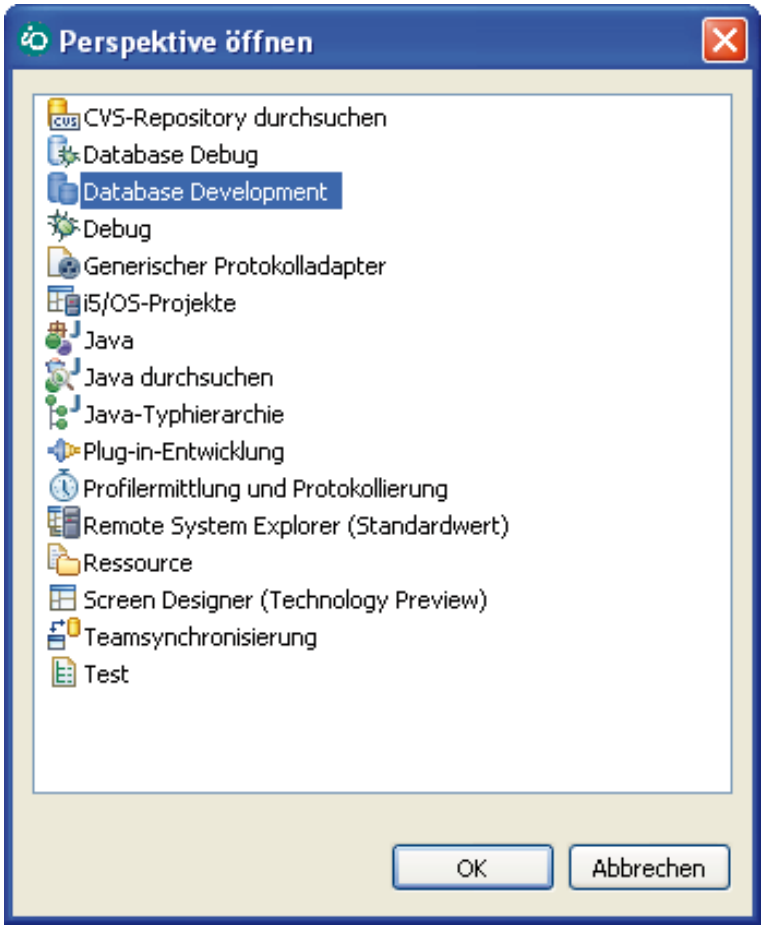
In der Anzeige der Benutzervorgaben („Driver Definition“) klicken wir abschließend auf die Schaltfläche „Anwenden“ und dann auf „OK“.

Damit haben wir die Voraussetzungen für den Zugriff auf die DB2/UDB des System i abgeschlossen.

Unter anderem finden wir jetzt in der Übersicht der Perspektiven die Perspektive „Database Development“, die wir für die weitere Arbeit nutzen werden.

Wählen Sie die neue Perspektive „Database Development“ aus.

Das können Sie z. B. über die Menüoption „Fenster / Perspektive öffnen / andere“ durchführen.



Perspektive auswählen

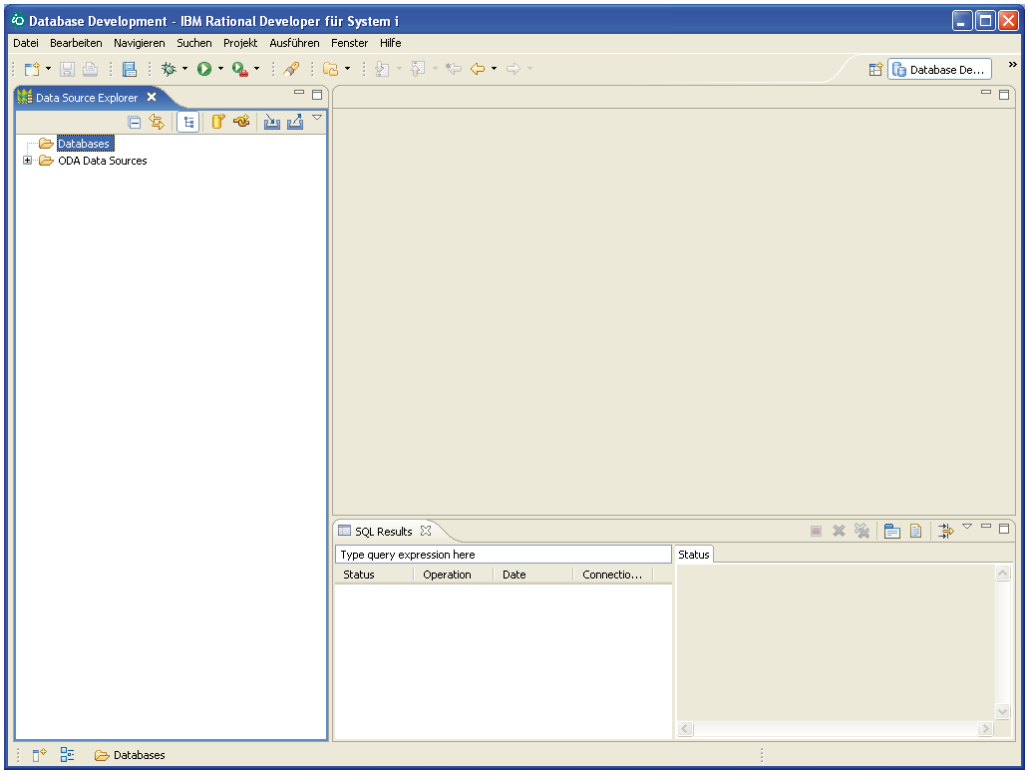
Die neue Datenperspektive besteht aus verschiedenen Anzeigenbereichen. Links befindet sich der „Data Source Explorer“ – die Grundlage für das Arbeiten mit den Inhalten einer Datenbank.

SQL-Result-Anzeigen liefern Ergebnisse zu SQL-Anweisungen. Und im Editierbereich können wir SQL-Eingaben vornehmen und SQL-Quellcodes bearbeiten.

3.5.5.1
Seite 16

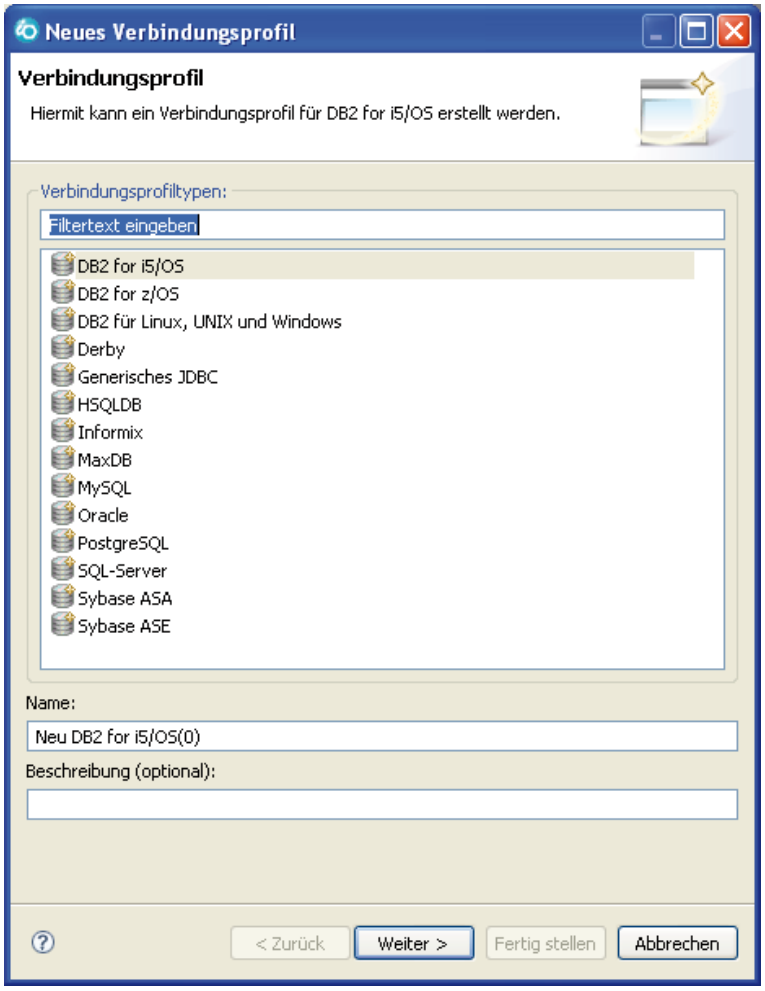
Erstellen eines Verbindungsprofils

Als Erstes müssen wir in dieser Perspektive ein Verbindungsprofil „Connection Profile“ erstellen.



Data Source Explorer

- Wählen Sie in der Sicht „Data Source Explorer“ den Eintrag „Databases“ mit der rechten Maustaste aus.
- Selektieren Sie die Option „Neu“.
  Mit dieser Auswahl gelangen wir in eine Übersicht der verfügbaren oder besser gesagt der in RDI installierten Datenbanktreiber für die unterschiedlichen Datenbanken. Unter anderem finden wir dort auch die notwendigen Treiber für die DB2/UDB für i5/OS.
- Selektieren Sie den Eintrag „DB2 for i5/OS“.

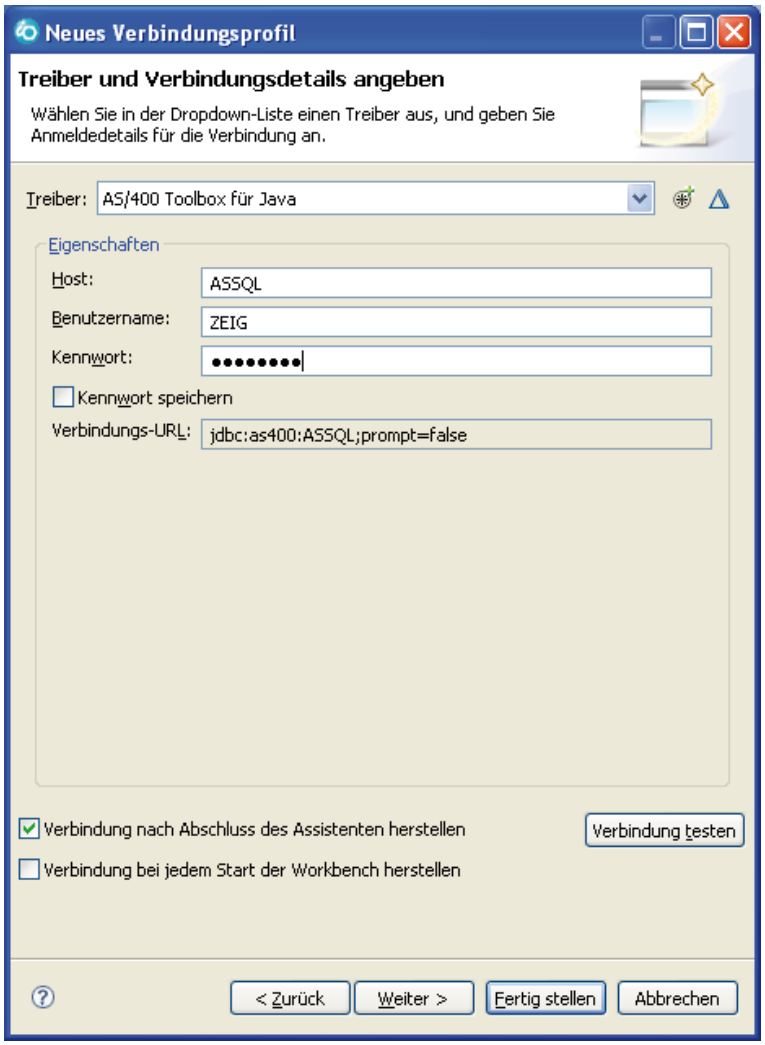


Verbindungsprofil wählen

- Bestätigen Sie die Auswahl mit einem Klick auf „Weiter“.

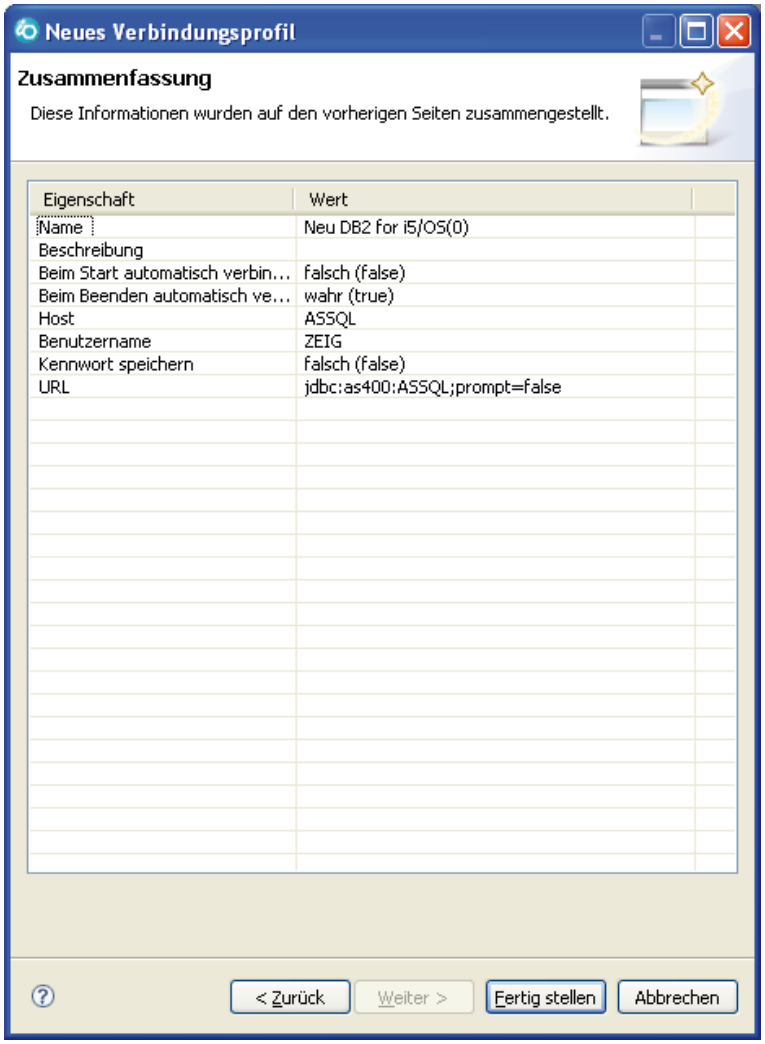
3.5.5.1
Seite 18

In der nächsten Anzeige werden wir zur Eingabe der Verbindungsmerkmale in Form des Hostnamens und eines Benutzers samt Kennwort aufgefordert. Geben Sie hier den Hostnamen für das System i und ein IBM-i-Benutzerprofil an.



Verbindungsdetails

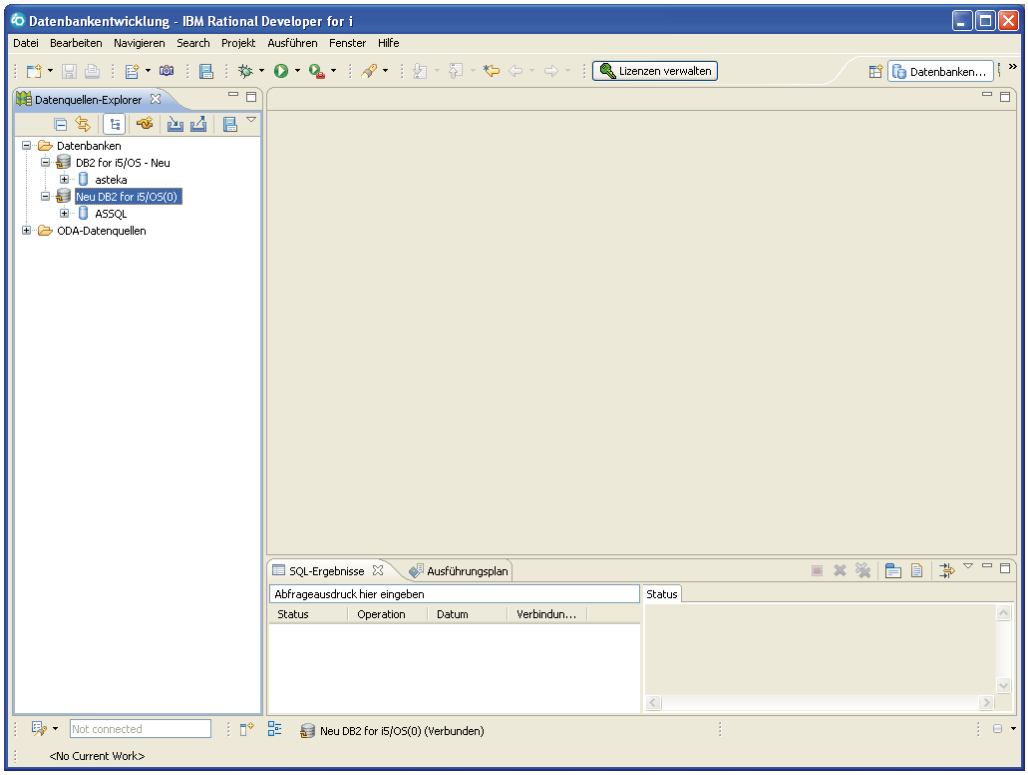
- Mit einem Klick auf „Weiter“ können wir die Angaben nochmals einsehen. Anschließend erscheint eine Anzeige, wie sie in der nachfolgenden Abbildung zu sehen ist.



Zusammenfassung

- Mit einem Klick auf „Fertigstellen“ wird die neue Datenbankverbindung generiert. Das Ergebnis sehen wir in der Navigationsübersicht. Dort sollte sich nun die neue Datenbankverbindung befinden.

3.5.5.1
Seite 20

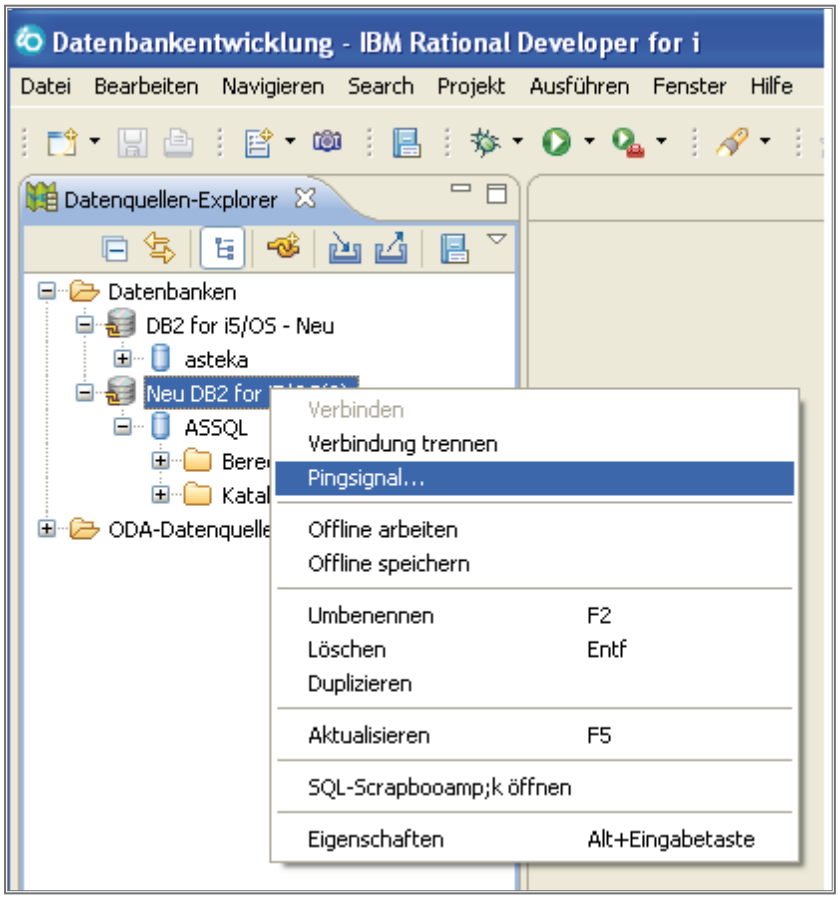


Neue Verbindung wurde definiert

Es bietet sich an, die Verbindung zu testen. Dazu können wir uns einer Pingfunktion bedienen, die in RDI integriert ist.

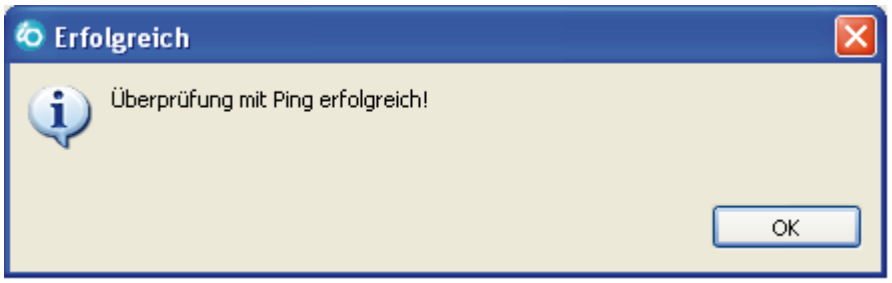
Die Verbindung testen wir, indem wir im linken Navigationsbereich mit der rechten Maustaste auf den neuen Eintrag klicken. Daraufhin erscheint eine Auswahl.

Selektieren Sie den Eintrag „Pingsignal ...“.



Pingsignal

Wenn die Konfiguration IP-seitig in Ordnung ist, sollte eine Nachricht erscheinen, wie sie in der folgenden Abbildung zu sehen ist.



Erfolgreiche Verbindung

Mit der Erstellung des Verbindungsprofils sind die ersten Konfigurationsschritte für das Arbeiten mit dem Datentool von Eclipse abgeschlossen. Es bildet die Grundlage für das Arbeiten mit der jeweiligen Datenbank.

Auf Basis der „jeweiligen Datenbank“ lassen sich die eigentlichen Arbeiten ausführen, die den Anwendungsentwickler in die Lage versetzen, die Funktionen und Werkzeuge in diesem Tool für die Anwendungsentwicklung und die Arbeit mit der DB2/UDB des System i zu verwenden.

3.5.5.1
Seite 22

Arbeiten mit SQL

Die Voraussetzungen für das Arbeiten mit der Datenbank des System i haben wir mit den vorhergehenden Schritten geschaffen. Jetzt gilt es, die SQL-Tätigkeiten mit den RDi-Funktionen nutzen zu können.

Achtung!

Grundsätzlich an dieser Stelle der Hinweis, dass RDi bzw. die Datenperspektive mit allen Inhalten der Datenbank arbeiten. Damit geht einher, dass bei der Verbindungserstellung und beim Arbeiten mit den Datenbankinhalten unter Umständen eine größere Anzahl von Informationen zwischen dem Hostsystem, auf dem die Datenbank läuft, und dem RDi-Client übertragen werden muss.

Eclipse bietet für diesen Fall die Möglichkeit, auf ein Subset der Datenbankinformationen zuzugreifen. Dieses Subset kann man individuell mit Hilfe von Filtern definieren. Der Vorteil: Schnellere Zugriffszeiten bei der Beschaffung der Grundinformationen zur Datenbank bzw. zu deren Elementen.

Mit Hilfe von Filtern lassen sich über die Eigenschaften der Datenbankverbindung auch gezielt Einschränkungen für die benötigten Inhalte der Datenbank definieren.

Unter anderem stehen in der Datenperspektive folgende Filter für unterschiedliche Datenbankbereiche zur Verfügung:

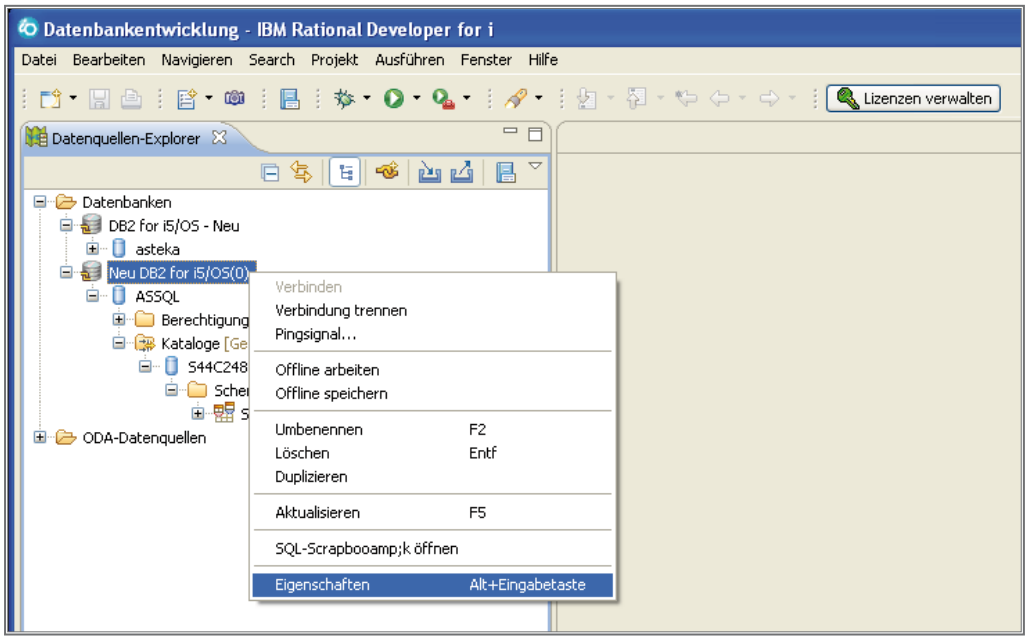
- Schemafilter
- Tabellenfilter

Gehen wir einmal davon aus, dass wir zunächst mit dem Inhalt einer Bibliothek bzw. eines Schemas arbeiten wollen. Dazu können wir einen Schemafilter verwenden, der als Einschluss- oder Ausschlussfilter definiert werden kann.

Den Filter erstellen wir auf Basis des Verbindungsprofils. In dessen Eigenschaften finden wir u. a. auch die verschiedenen Filterdefinitionen.

Erstellen eines Filters

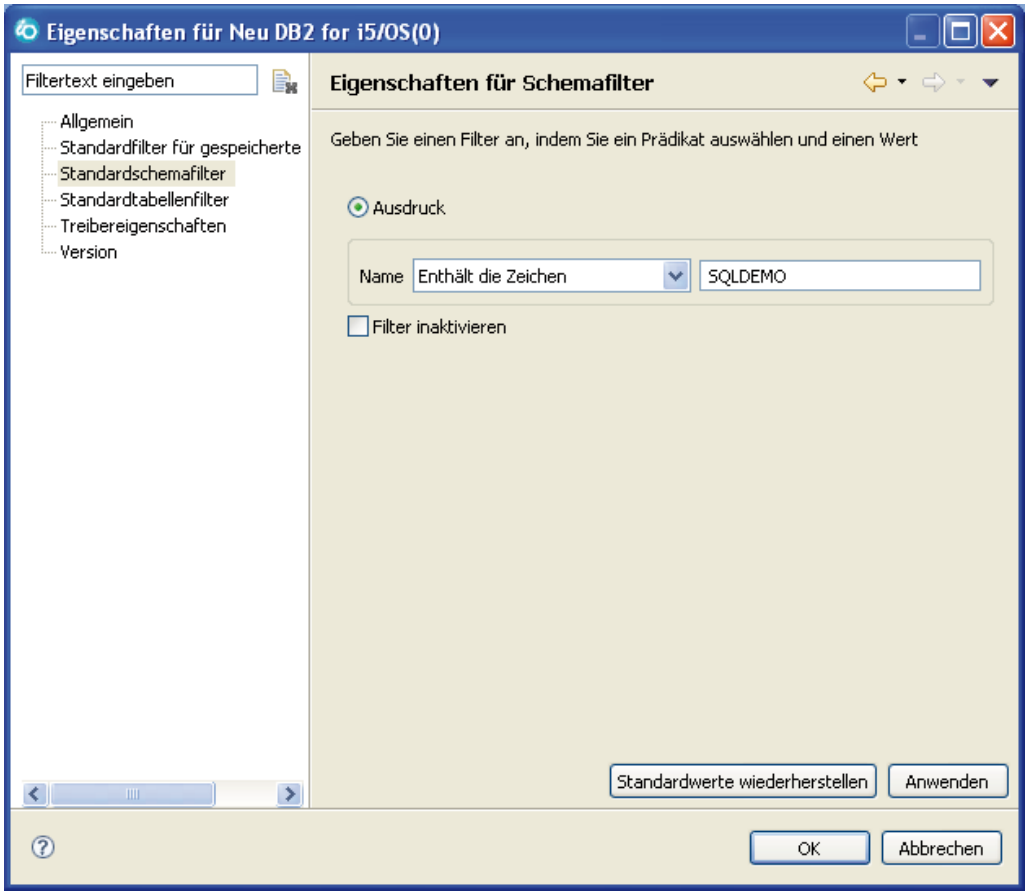
Um einen Schemafilter zu erstellen, klicken wir mit der rechten Maustaste im Navigationsbereich auf das Verbindungsprofil. Es öffnet sich ein Fensterbereich, in dem wir u. a. auch die Eigenschaften des Profils verwalten können.



Eigenschaften des Verbindungsprofils

- Klicken Sie in der Navigationssicht mit der rechten Maustaste auf das Verbindungsprofil.
- Wählen Sie im neuen Anzeigenbereich die Option „Eigenschaften“.
- Selektieren Sie im nächsten Fenster den Bereich „Standardschemafilter“.
- Deaktivieren Sie die Option „Filter inaktiviert“.
- Wählen Sie die Option Name mit „Enthält die Zeichen“.
- Geben Sie im Filterdefinitionsbereich den Namen eines Schemas an. In unserem Beispiel habe ich das Schema „SQLDEMO“ verwendet, das die Beispieltabellen und SQL-Definitionen beinhaltet.

3.5.5.1
Seite 24



Schemafilter

Anmerkung

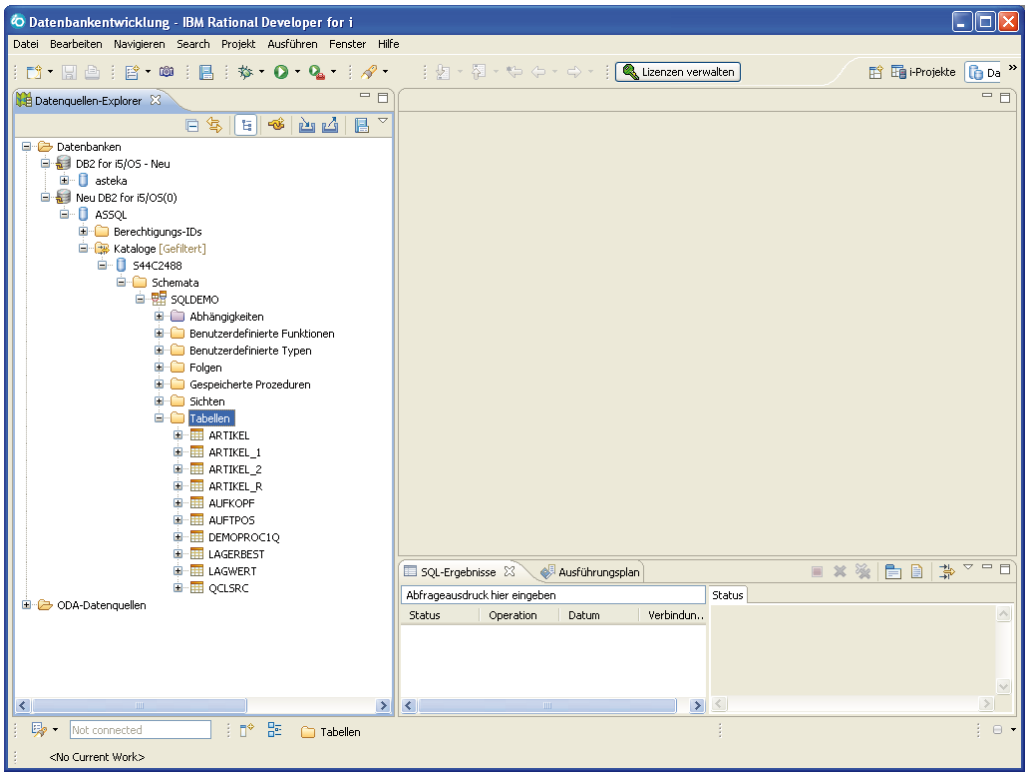
RDI bietet neben der Option „Enthält die Zeichen“ auch noch andere Selektionsoptionen, so z. B.:

- Beginnt mit den Zeichen
- Endet mit den Zeichen
- Beginnt nicht mit den Zeichen
- Endet nicht mit den Zeichen
- Enthält nicht die Zeichen

Bestätigen Sie die Auswahl mit einem Klick auf die Schaltfläche „Anwenden“ und anschließend auf „OK“.

Mit diesen Teilschritten verändert sich der Eintrag in der Navigationsleiste leicht. Wenn Sie den entsprechenden Bereich aktualisieren, finden Sie im Unterordner „Kataloge“ den Hinweis „gefiltert“. Daran erkennt man sofort, dass in diesem Bereich nur ein Subset der Datenbankinhalte dargestellt wird. Unterhalb des Bereichs „Schema“ finden wir nun die Inhalte der Schemata, die den Selektionskriterien entsprechen. In unserem Beispiel ist dies der Inhalt des Schemas „SQLDEMO“. Hier finden wir z. B. unterhalb des Abschnitts „Tabellen“ die in dem Schema SQLDEMO enthaltenen Tabellen.

- Erweitern Sie den Eintrag „Kataloge“.
- Erweitern Sie den Eintrag für das Schema (in unserem Beispiel SQLDEMO).
- Erweitern Sie den Eintrag „Tabellen“.

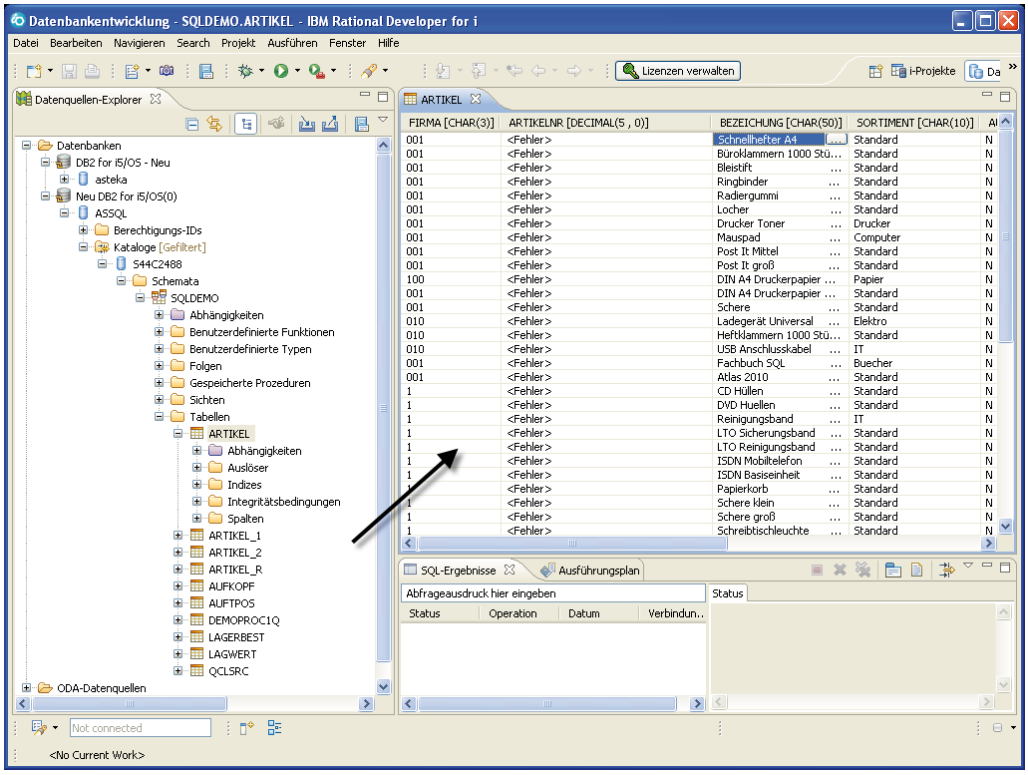


Tabellen in einem Schema

Anmerkung

Durch einen definierten Tabellenfilter kann der hier angezeigte Bestand an Tabellen auch eingeschränkt sein. Die Tabellenfilter werden ebenfalls im Bereich „Eigenschaften“ des Verbindungsprofils definiert. Die Definition verläuft dabei vergleichbar mit der Definition der Schemafilter.

Als erstes Beispiel für die Arbeit mit der Datenbank beschäftigen wir uns mit dem Inhalt einer Tabelle. Diese wird mit der rechten Maustaste markiert, anschließend lässt sich festlegen, wie mit dem Inhalt der Tabelle verfahren werden soll. RDI bietet u. a. die Möglichkeit, mit dem Inhalt der Tabelle direkt zu arbeiten. Die folgende Abbildung zeigt den zum Bearbeiten ausgewählten Inhalt der Tabelle „Artikel“.

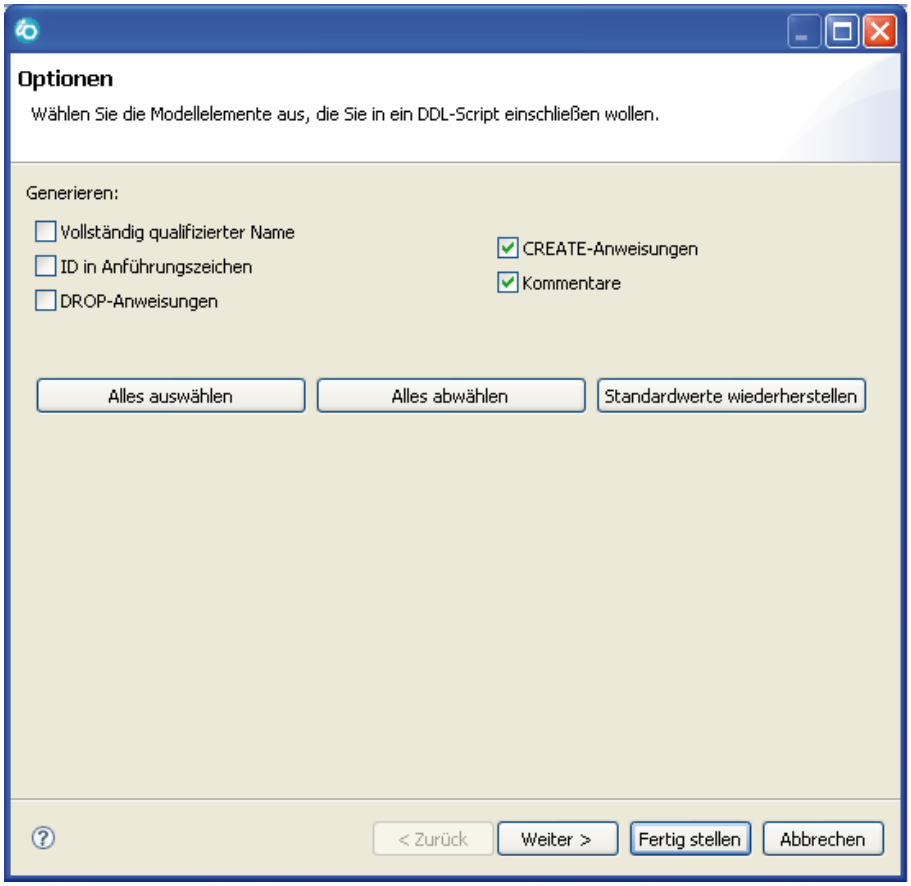


Bearbeiten des Inhalts einer Tabelle

Die Datenbank-Tools von Eclipse bieten verschiedene Werkzeuge für das Arbeiten mit einer Datenbank. System-i-Anwender werden den SQL-Editor mit den nativen Werkzeugen der 5250-Ebene vergleichen und sich gewiss mit dessen Verwendung anfreunden können. Alternativ dazu steht ein weiteres Tool in Form des SQL-Scrapbook zur Verfügung.

Schauen wir uns den SQL-Editor bzw. den Assistenten für SQL einmal genauer an. Dazu erstellen wir zunächst eine einfache Tabelle.

- Klicken Sie mit der rechten Maustaste auf den Eintrag „Tabellen“ in der Navigationssicht.
- Wählen Sie die Option „Neue Tabelle“.
- Im Definitionsbereich für die Optionen können wir unterschiedliche Vorgaben machen, wie die nachfolgende Abbildung zeigt. Passen Sie die Vorgaben an Ihre Wünsche bzw. Anforderungen an.

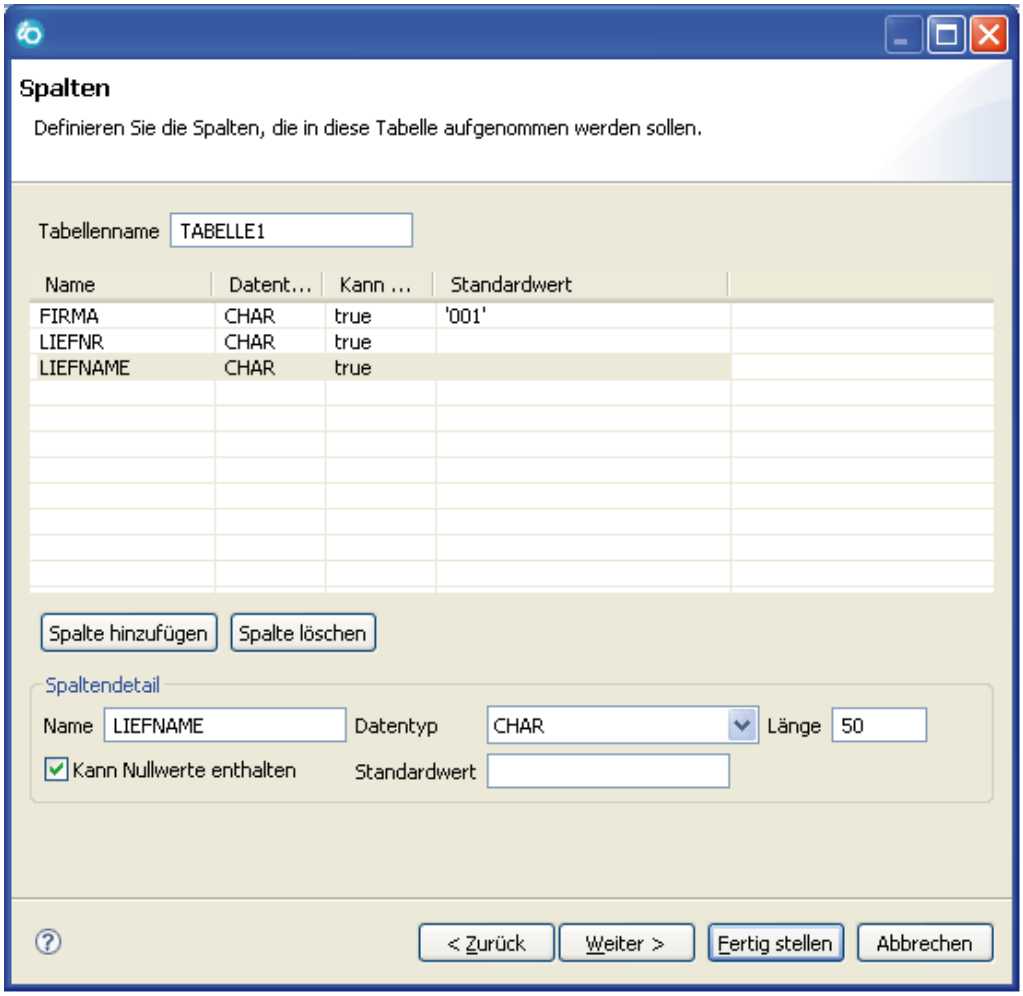


SQL-Optionen

Bestätigen Sie die Angaben mit einem Klick auf die Schaltfläche „Weiter“.

Damit gelangen wir in den Anzeigenbereich des Assistenten für die Erstellung einer neuen Tabelle. Hier müssen wir zunächst den Namen der Tabelle angeben. In unserem Beispiel verwende ich „Tabelle1“. Über die Schaltfläche „Spalte hinzufügen“ können wir die gewünschten Spalten angeben. Diese verfügen natürlich über die gewohnten Merkmale wie Spaltenname, Art und Länge.

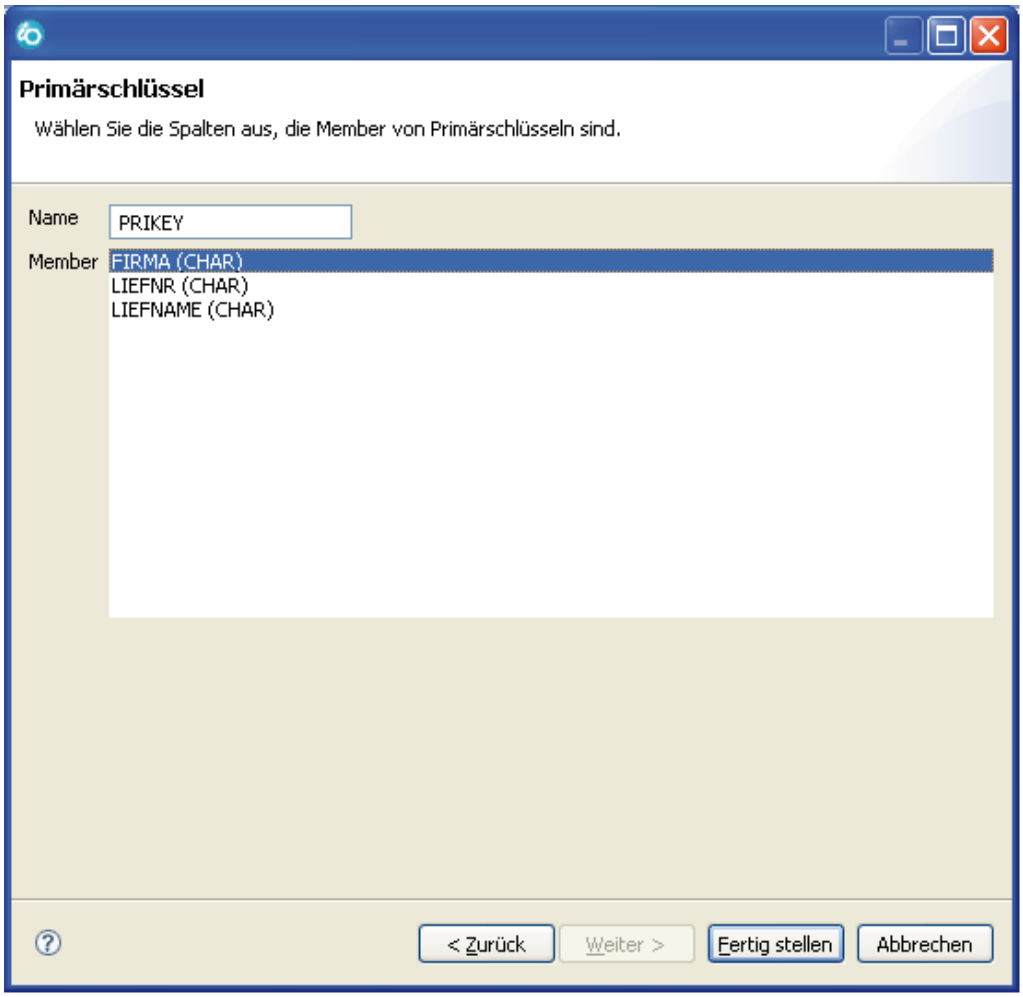
3.5.5.1
Seite 28



Erstellen einer neuen Tabelle

- Geben Sie als Tabellennamen „Tabelle1“ an.
- Klicken Sie auf die Schaltfläche „Spalte hinzufügen“.
- Überschreiben Sie im Feld „Name“ den vorgeschlagenen Spaltennamen mit dem gewünschten Namen für die neue Spalte.
- Geben Sie einen Datentyp und die zugehörige Länge ein.
- Wenn alle Spalten angegeben wurden, klicken Sie auf „Weiter“.

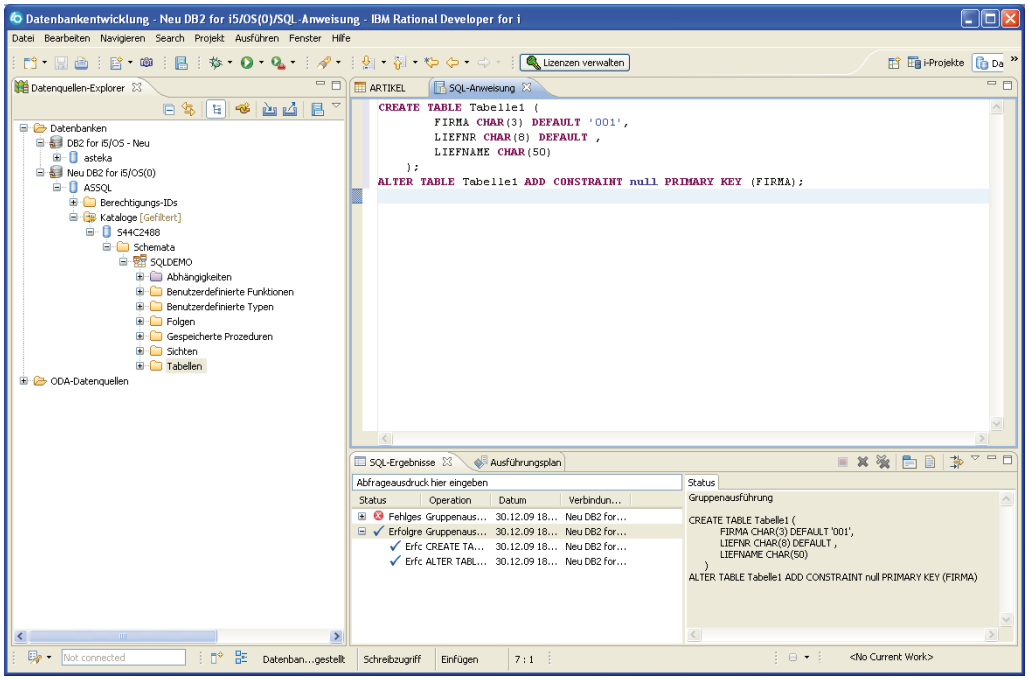
In der nächsten Anzeige bietet uns der Erstellungsassistent die Option, für die neue Tabelle einen Primärschlüssel zu definieren.



Primärschlüssel

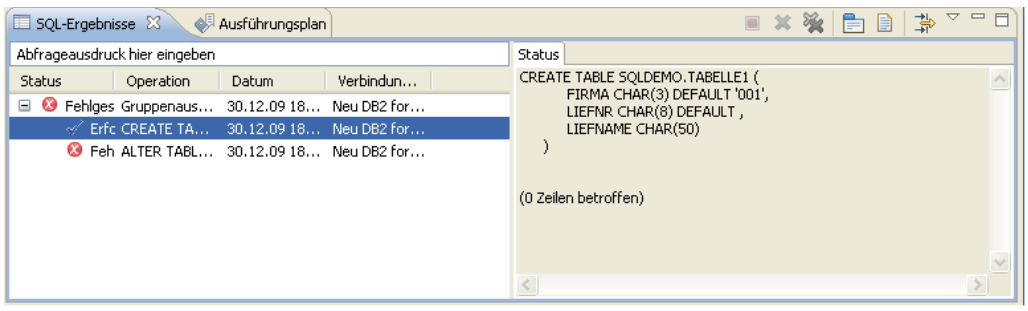
- Wählen Sie die Firmennummer als Primärschlüssel.
- Bestätigen Sie die Auswahl mit einem Klick auf „Fertig stellen“.

Damit verlassen wir den SQL-Assistenten und gelangen in den SQL-Editorbereich. Dieser beinhaltet unsere vorherigen Definitionen, die bei Bedarf hier noch angepasst werden können.



SQL-Syntax im Editor

Mit den Vorgaben lässt sich jetzt die neue Tabelle erstellen. Dazu kann man die Option „Alle Ausführen“ wählen. Damit wird die SQL-Anweisung auf dem Hostsystem (dem System i) abgesetzt und ausgeführt. Ein Statusbereich im unteren Abschnitt der Anzeige liefert das Ergebnis der SQL-Ausführung. Fehler werden mittels eines roten Symbols dargestellt. Mit einem Klick auf den jeweiligen Eintrag erhalten wir rechts im Statusbereich die Details zu dem selektierten Eintrag.

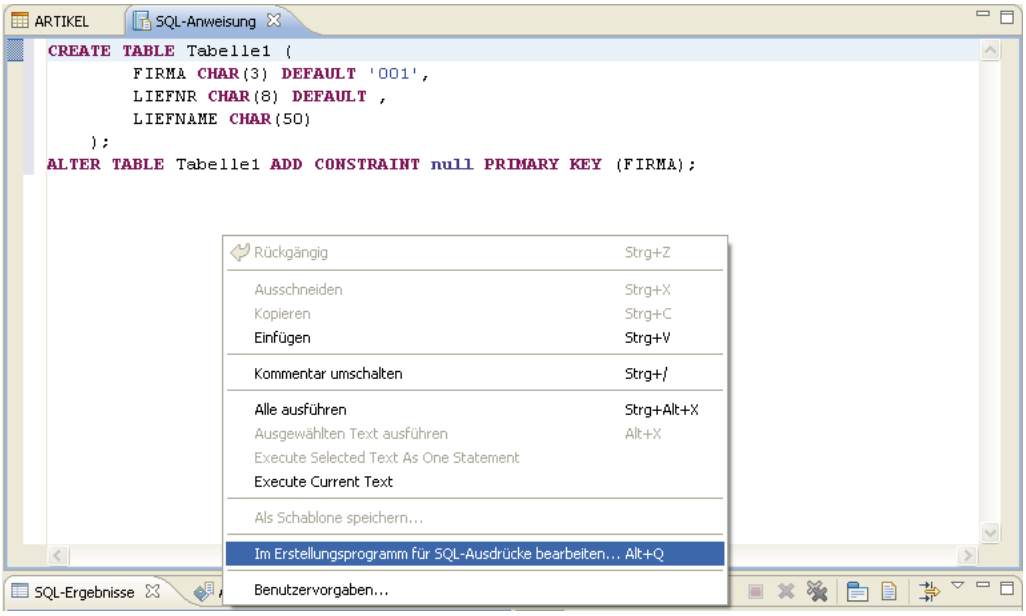


Ergebnis der SQL-Ausführung

SQL-Erstellungsprogramm

Alternativ zum SQL-Editor bietet die Datenbankperspektive auch ein SQL-Erstellungsprogramm, das hier und da auch als „SQL Query Builder" bezeichnet wird. Dieses Erstellungsprogramm (eine andere Form des Erstellungsassistenten) bietet verschiedene Zusatzfunktionen, die insbesondere SQL-Neulingen die Arbeit erleichtern dürften.

Das SQL-Erstellungsprogramm kann aus dem SQL-Editorbereich heraus gestartet werden. Dazu genügt ein Klick mit der rechten Maustaste und die Auswahl der Option „Im Erstellungsprogramm für SQL-Ausdrücke bearbeiten".



Erstellungsprogramm für SQL-Ausdrücke starten

Damit gelangen wir in die Startseite des SQL-Erstellungsprogramms. Hier ist per Standard der Assistent für die Verwendung einer SELECT-Anweisung gestartet. Natürlich kann auch jede andere Art einer SQL-Anweisung mit dem Erstellungsprogramm verwendet werden.

3.5.5.1
Seite 32

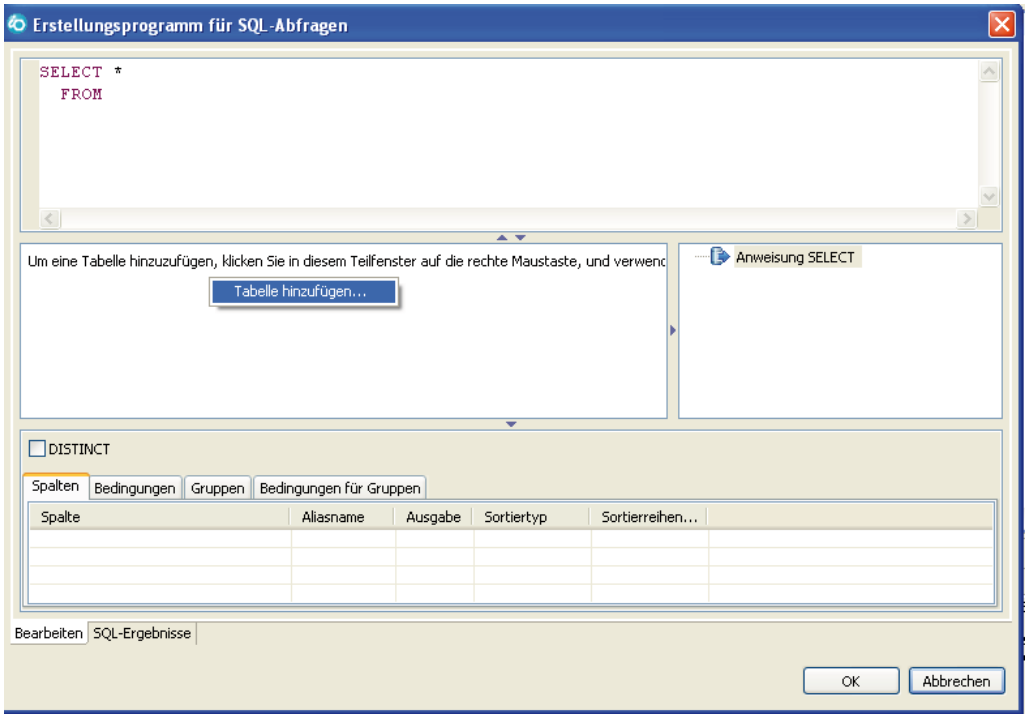


Tabelle auswählen

Schauen wir uns nun den Assistenten am Beispiel einer einfachen SELECT-Anweisung an, die den Inhalt der Tabelle „Artikel“ betrifft.

- Positionieren Sie den Cursor in den Abschnitt „Um eine Tabelle hinzuzufügen...“ und klicken Sie dort mit der rechten Maustaste.
- Erweitern Sie in der nächsten Anzeige das Schema.

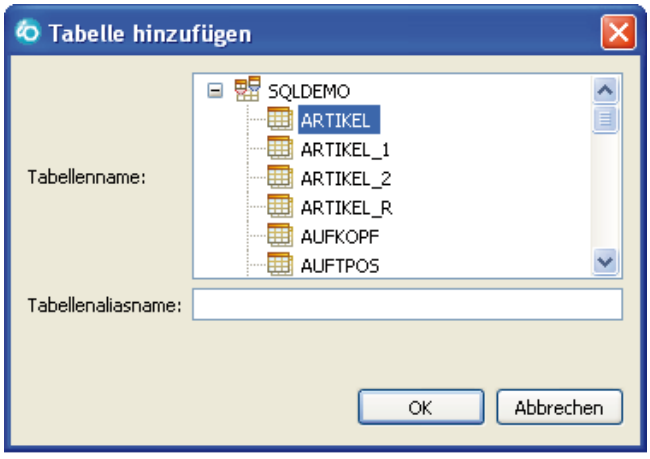
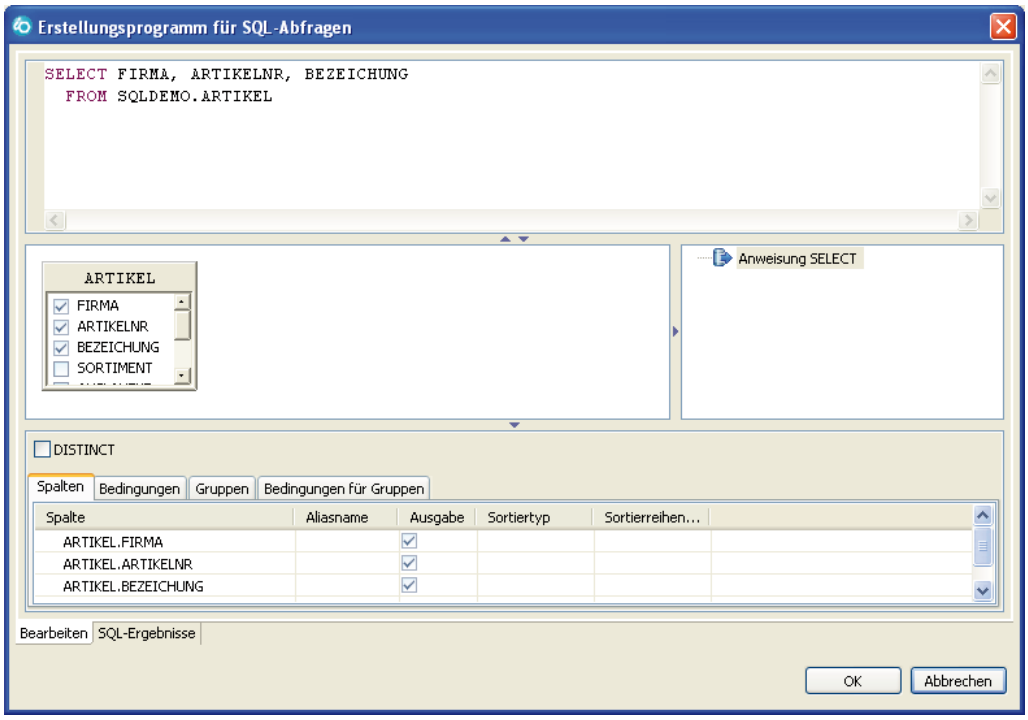


Tabelle auswählen

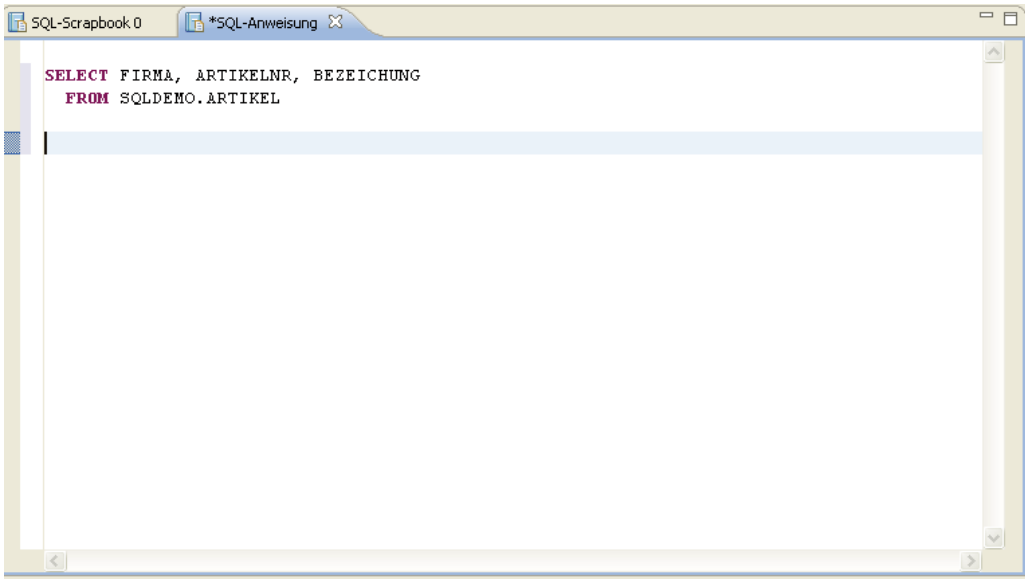
- Markieren Sie die Tabelle „Artikel“.
- Bestätigen Sie die Auswahl mit einem Klick auf „OK“.
  Damit werden die Spalten der Tabelle in diesem Abschnitt angezeigt.
  Diese lassen sich hier zur weiteren Verarbeitung selektieren.
- Wählen Sie einige Spalten mit einem Mausklick aus. Beachten Sie dabei die Veränderung in den Bereichen mit der eigentlichen SELECT-Anweisung und im Abschnitt „Spalten“ im unteren Bereich der Anzeige.



Markierte Spalten

- Klicken Sie auf „OK“.
  Damit wechseln wir wieder in den SQL-Editorbereich, in dem jetzt die SELECT-Anweisung mit den ausgewählten Bereichen zu sehen ist.

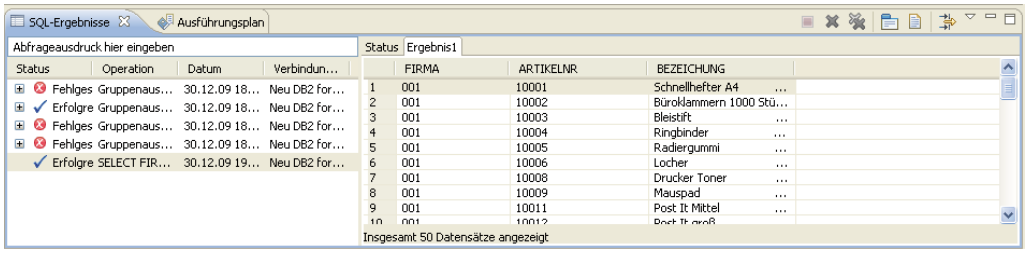
3.5.5.1
Seite 34



Select-Anweisung

Mit der Tastenkombination „Strg + Alt + x“ kann der SQL-Code ausgeführt werden. (Alternativ dazu steht auch hier die Option „Alle ausführen“ nach einem Klick mit der rechten Maustaste zur Verfügung.)

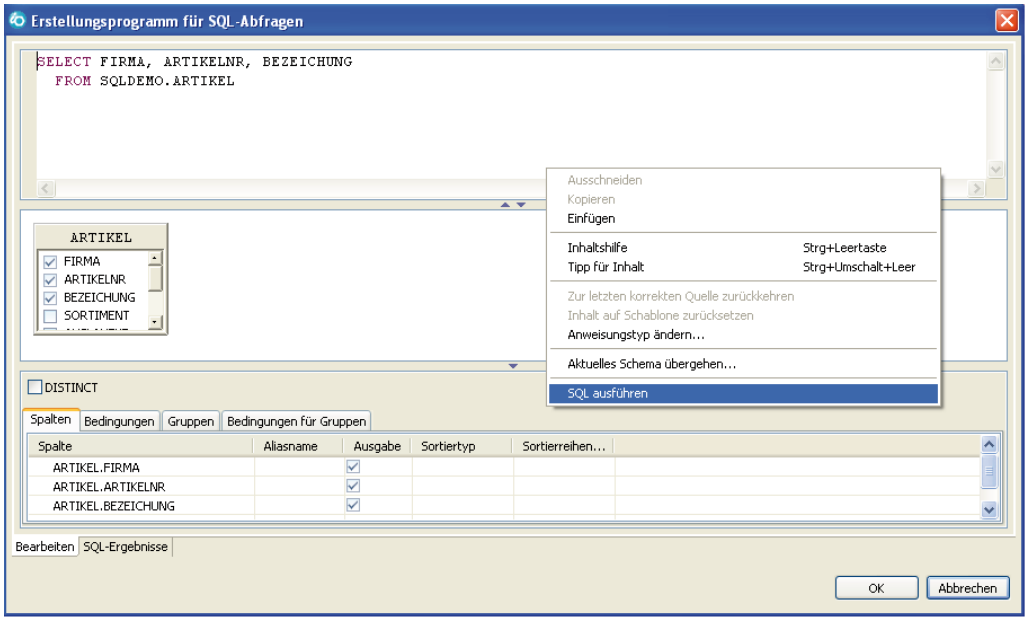
Wählen Sie die Tastenkombination „Strg + Alt + x“. Das Ergebnis der Ausführung sehen wir unten rechts in der Ergebnissicht:



Ergebnis der SELECT-Anweisung

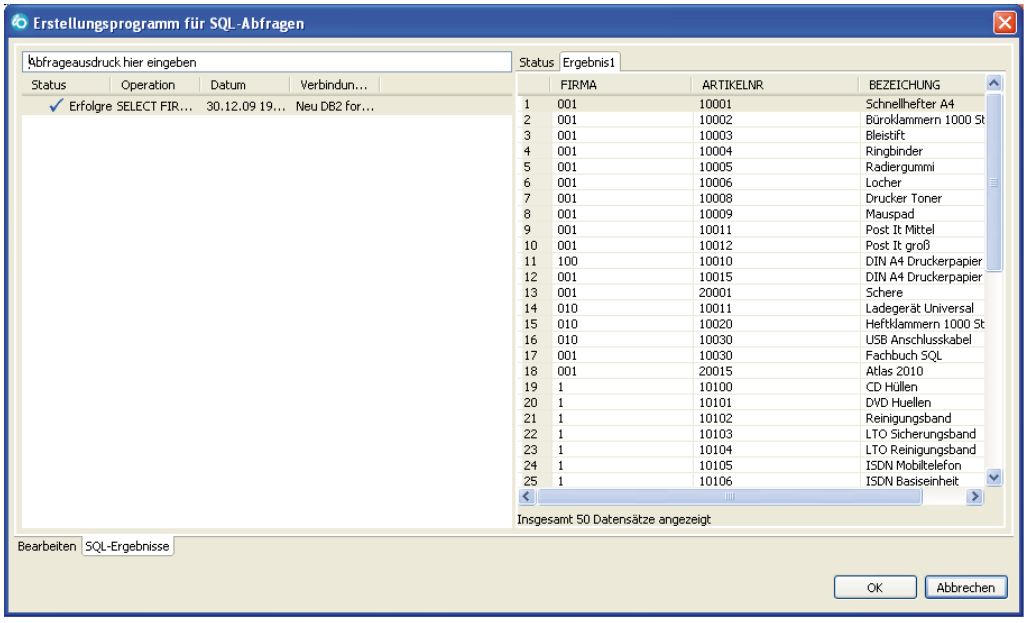
Anmerkung

Alternativ zu dieser Form der SQL-Ausführung kann im Erstellungsprogramm der SQL-Code nach einem Klick mit der rechten Maustaste in einem beliebigen Bereich innerhalb des Fensters für die Erstellung der SQL-Abfragen und der Auswahl der Option „SQL ausführen“ ausgeführt werden.



SQL ausführen

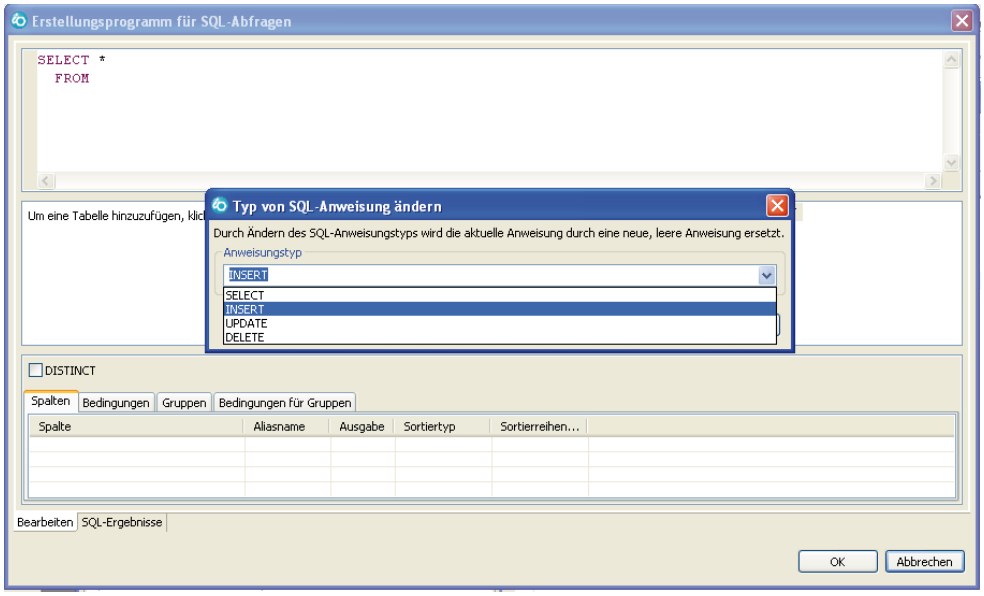
Das Ergebnis wird dann wie folgt ausgegeben:



Ergebnis der SQL-Ausführung

Um im Erstellungsprogramm eine andere Anweisung anzugeben, klickt man einfach mit der rechten Maustaste in den mit SELECT vorbelegten Bereich und wählt die Option „Anweisungstyp ändern“. Daraufhin erscheint ein Auswahlfenster mit den möglichen SQL-Anweisungen.

3.5.5.1
Seite 36



Anweisungstyp ändern

Anmerkung

Natürlich kann das Erstellungsprogramm auch zum Bearbeiten von SQL-Code genutzt werden. Dazu öffnet man den SQL-Code im Standard-SQL-Editor, markiert den SQL-Code und wählt nach einem Klick mit der rechten Maustaste innerhalb des Fensters der SQL-Abfragedefinition die Option „Im Erstellungsprogramm für SQL Ausdrücke bearbeiten“. Alternativ dazu kann man auch die Tastenkombination „Alt + Q“ nutzen.

Datenbankjournalisierung, ein Sicherheitsgurt Ihrer Datenbank

In diesem Kapitel werde ich mich mit dem Thema Journalisierung von Datenbanktabellen befassen. Wir werden die Frage klären, was Journalisierung überhaupt ist: Wie kann ich sie einrichten und wie kann ich sie verwenden, um Daten nach einer unter Umständen missglückten Transaktion wiederherzustellen. Die Möglichkeiten, die sich in diesem Umfeld finden, sind (wen wird es bei diesem System auch wundern?) sehr reichhaltig. Man möge es mir nachsehen, wenn ich auf den einen oder anderen Punkt nicht oder nicht sehr detailliert eingehe. Beispielsweise würde die Beschreibung von Programmen, die Sie selbst erstellt haben und die Journaleinträge selbst senden oder empfangen können, über den verfügbaren Platz in diesem Abschnitt hinausgehen. Ich werde eher den grundsätzlichen Umgang mit Journalen, Empfängern und anderen beteiligten Objekten besprechen.

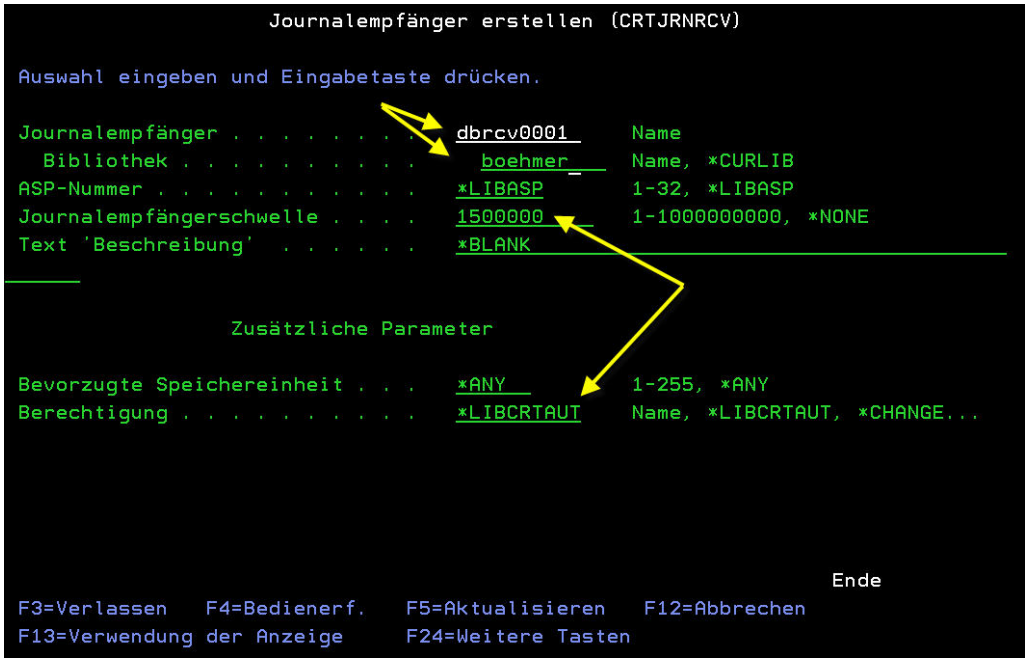
Wozu dient nun die Aufzeichnung in einem Journal. Vorweg sei gesagt, dass nicht nur Dateien, besser Tabellen, aufgezeichnet werden können, sondern auch eine ganze Reihe anderer Objekte. Aufzeichnen bedeutet, dass Änderungen an einem journalisierten Objekt protokolliert werden und im Bedarfsfall wiederhergestellt werden können. Diese Möglichkeit liefert natürlich eine gehörige Portion Sicherheit für den Fall, dass das System einmal ausfällt, aus welchen Gründen auch immer. Da wir hier den Fokus stark auf die Haltung der Daten an sich und die Konsistenz der gespeicherten Informationen legen, interessiert an dieser Stelle die Absicherung gegenüber abgebrochenen oder fehlgeschlagenen Transaktionen. Eine Transaktion fassen Sie bitte als eine Folge von Änderungen im Datenbestand auf. Aufgezeichnet werden nun nicht nur Änderungen von Informationen, sondern auch das Öffnen und Schließen von Dateien, das Starten und Beenden der Aufzeichnung an sich usw.

Man kann sich sicherlich leicht vorstellen, dass mit dieser Aufgabe einige Arbeit auf das System zukommt und dass demzufolge durchaus die Performance beeinflusst werden kann. Natürlich ganz nach Maschine und Auslastung. Auch die Zeit, die für ein IPL benötigt wird, kann sich erhöhen, speziell wenn eine nicht reguläre Systembeendigung vorliegt.

Gehen wir nun davon aus, dass eine Planung vorliegt, welche Dateien journalisiert werden sollen. Dann machen wir uns auf den Weg, die benötigten Dinge zu erstellen.

Ich verwende den Befehl CRTJRNRCV zur Erstellung eines Journalempfängers. Mit den Parametern dieses Befehls gebe ich an, wo der Empfänger selbst gespeichert werden soll, welcher (Speicher-) Schwellenwert verwendet werden soll und wer auf den Empfänger zugreifen darf.

3.5.6
Seite 2

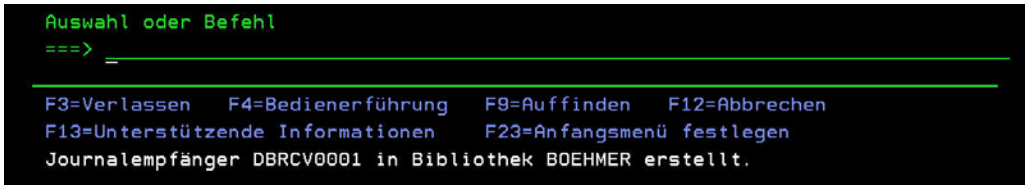


Befehl CRTJRNRCV

Die vier Ziffern am Schluss der Empfängerbezeichnung kommen nicht von ungefähr. Man kann das System dazu veranlassen, einen neuen Empfänger zu erstellen, wenn der aktuelle seine maximale Größe erreicht hat. Liegt eine Bezeichnung in dieser Form vor, kann man Empfänger mit einer Art fortlaufender Nummerierung erzeugen lassen.

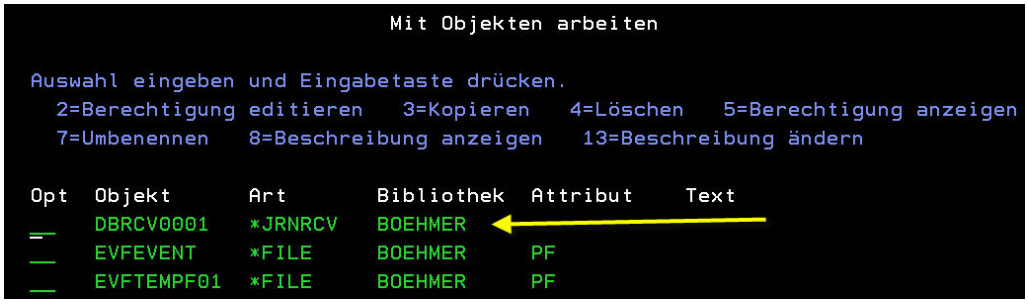
Über den Parameter „Journalempfängerschwelle“ wird die Aufnahmekapazität definiert. Wenn dieser Schwellenwert erreicht wird, führt das System bestimmte Aktionen durch, die unter anderem davon abhängen, wie das steuernde Journal definiert ist.

Drücken Sie „Datenfreigabe“ und der Empfänger wird erstellt.



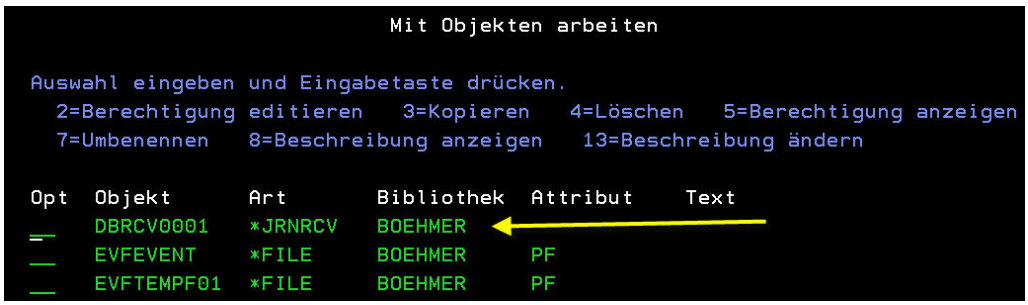
Nach der Datenfreigabe

Werfen wir einen Blick in die Bibliothek. Der Empfänger taucht dort als Objekt der *JRNRCV auf.



Das Empfängerobjekt

Der Empfänger ist für sich genommen noch ohne Funktion. Mit Leben erfüllt wird er erst, wenn er eine feste Beziehung mit einem Journal eingegangen ist. Ein solches fehlt allerdings noch. Bevor ich jedoch ein Journal erstelle, noch ein Wort zu der Zugriffsberechtigung des Empfängers. Schauen wir uns die Objektberechtigung über die Auswahl 2 einmal an.



Objektberechtigung

Schön, mögen Sie sich denken, das erscheint auf den ersten Blick nicht weiter dramatisch. Wo liegt der Pferdefuß?

Empfänger zeichnen die Satzabbilder der aufgezeichneten Dateien auf. Wenn ein Anwender nicht zur Gruppe der Administratoren gerechnet wird, so bleiben ihm nach der Abbildung aber immer noch die *CHANGE-Rechte, das bedeutet, er kann sich durchaus die Satzabbilder ansehen. Und zwar auch dann, wenn er in die eigentliche Tabelle NICHT hineinschauen darf. Unter Umständen FATAL!

Also bitte auch hier nicht den Sicherheitsgedanken vernachlässigen.

Nun gut, erstellen wir das Journal. Zuvor machen wir uns noch ein paar Gedanken zur Vergabe von Namen für Journale. Stellen Sie sich vor, es existieren zwei Journale mit gleichem Namen in verschiedenen Bibliotheken und diese Journale werden beschädigt. Beim Wiederherstellen des Speichers werden die Journale umbenannt, wenn sie in die Bibliothek QRCV (Bibliothek zur Wiederherstellung des Speichers) gestellt werden.

3.5.6
Seite 4

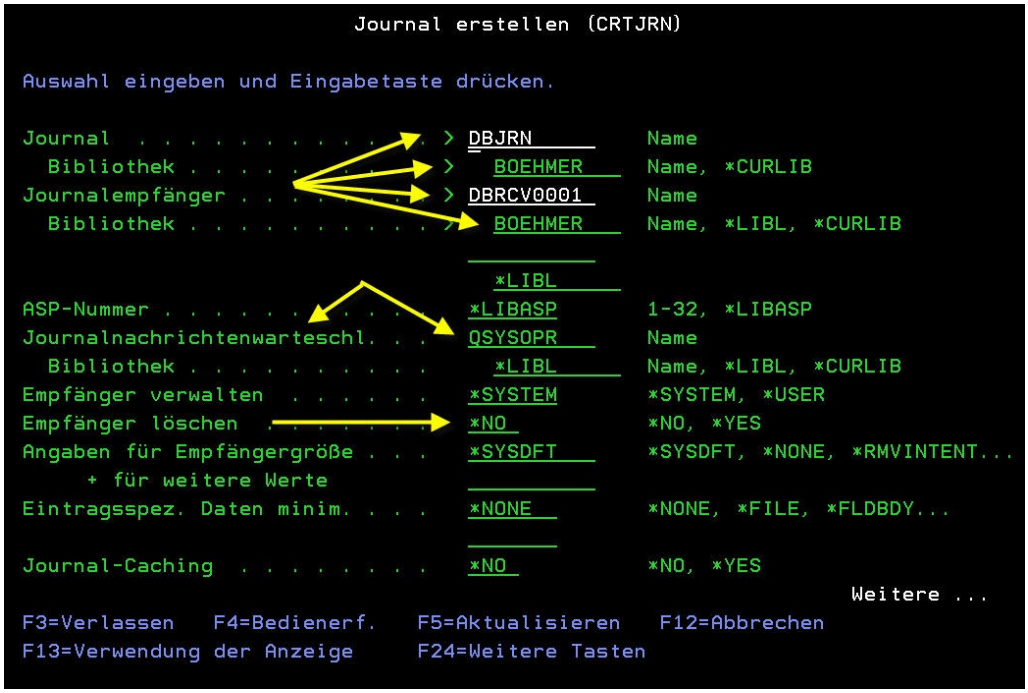
QRCV

Die Wiederherstellung unter altem Namen kann aus QRCV heraus dann unter Umständen nicht vorgenommen werden, da eine Änderung des Objektnamens nicht möglich ist. Ein Grund dafür Sorge zu tragen, dass der Journalname systemweit eindeutig ist. Ähnliche Argumente gelten auch für die Namensvergabe für Empfänger.

Auch die Reihenfolge beim Zurücksichern darf nicht außer Acht gelassen werden. Befinden sich das Journal und die aufgezeichneten Objekte in ein und derselben Bibliothek, so speichert das System automatisch in der korrekten Sequenz zurück.

Findet man die Journale und ihre zugeordneten Objekte in unterschiedlichen Bibliotheken, so kann man durch Beachtung der alphanumerischen Sortierfolge bei Vergabe der Namen die korrekte Folge erzwingen.

Unser Journal wird nun mit dem Befehl CRTJRN erstellt, der in der folgenden Abbildung im Prompt gezeigt wird.



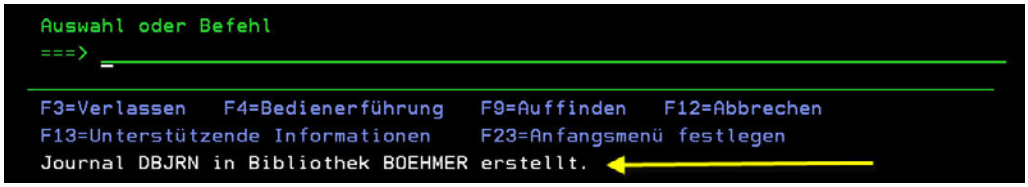
CRTJRN

Name des Journals mit Bibliothek angeben und im Parameter für den Empfänger gebe ich den im vorhergehenden Schritt erstellten Receiver an.

Auch die Nachrichtenwarteschlange an die das System Nachrichten senden soll kann hier angegeben werden. Über den Parameter „Empfänger verwalten“ wird bestimmt, wie das System reagiert, wenn bei dem Empfänger der Schwellenwert erreicht wird. Wenn hier *USER angegeben ist, wird das System eine Nachricht in die Nachrichtenwarteschlange senden. Dort kann sie eventuell von einem speziellen Programm abgearbeitet werden. In unserem Beispiel setze ich den Wert auf *SYSTEM. Das hat zur Folge, das bei Erreichen des Schwellenwertes der Empfänger abgehängt und ein neuer mit identischen Attributen erstellt und angehängt wird.

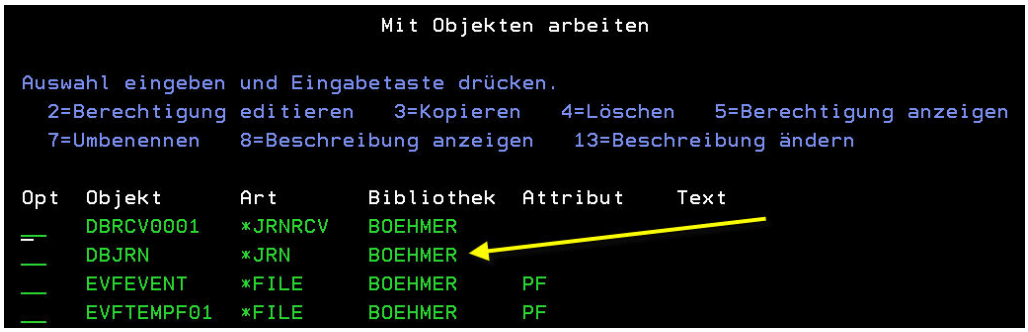
Der folgende Parameter „Empfänger löschen“ kann dafür eingesetzt werden, die abgehängten (und nicht mehr benötigten) Empfänger zu löschen => *YES. In diesem Fall habe ich den Wert auf *NO belassen.

Nach Datenfreigabe wird das Journal-Objekt erstellt.



Nach der Datenfreigabe

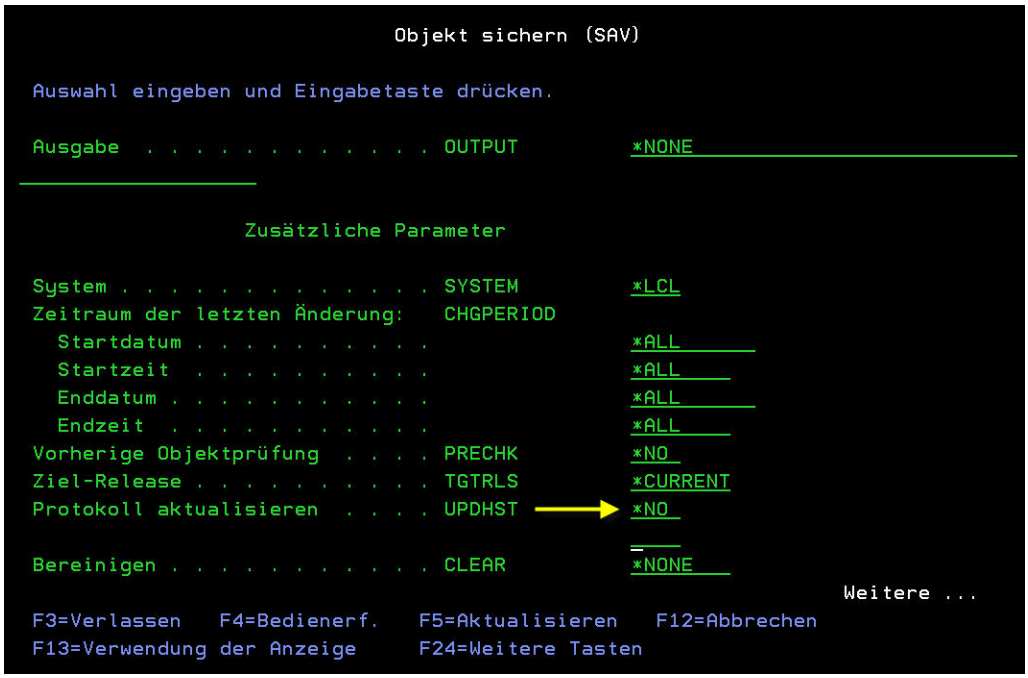
Auch hier werfen wir einen erneuten Blick in die Bibliothek.



Das erstellte Journal

Die Objektart für Journale lautet *JRN.

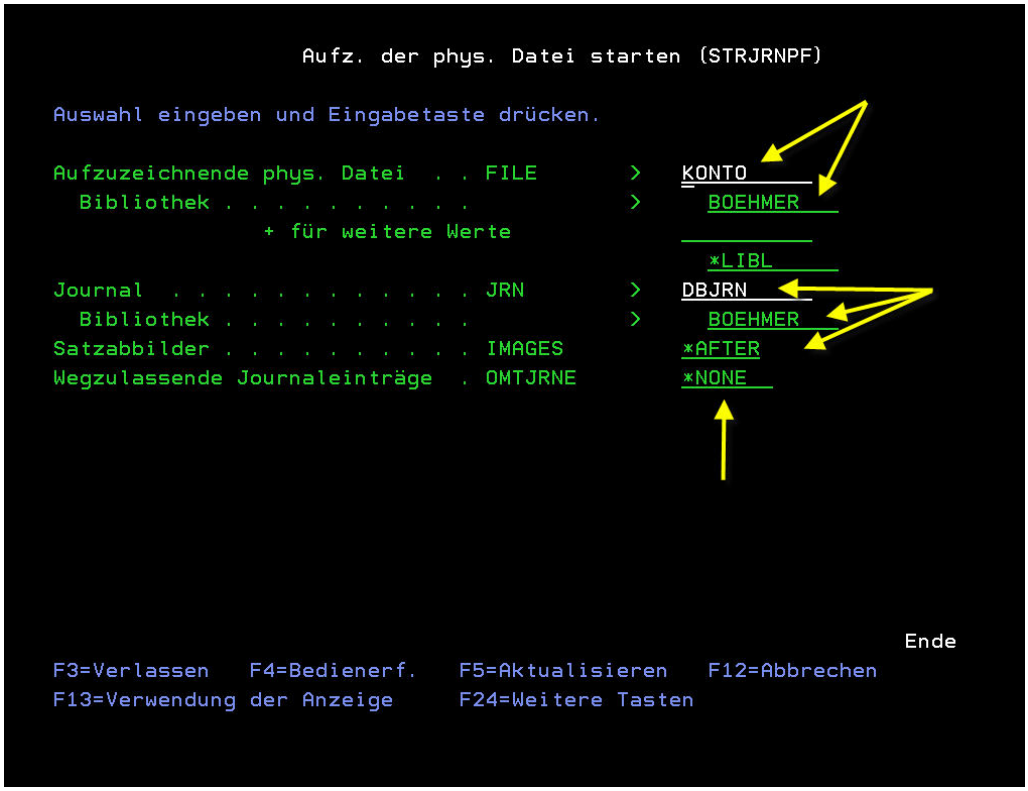
In den nächsten Schritten kann nun die Aufzeichnung der gewünschten Objekte gestartet werden. Dabei sollten Sie jedoch eine bestimmte Reihenfolge von Schritten einhalten. Zuerst starten Sie die Aufzeichnung und im Anschluss daran sollten sie die aufgezeichneten Objekte sichern, und zwar mit einer Aktualisierung der Objekthistorie. Diese Protokollaktualisierung wird von den üblichen Sicherungsbefehlen automatisch mit vorgenommen, nicht jedoch beim Befehl „SAV“. Die dortige Standardeinstellung für den Parameter UPDHST ist *NO, wie in der Abbildung zu sehen ist.



Ausschnitt aus dem Befehl SAV

Den Wert sollten Sie auf *YES ändern.

Wir können nun die Aufzeichnung der gewünschten Datenbankdateien starten, indem wir den Befehl STRJRNPF benutzen. In diesem Befehl kann angegeben werden, ob nur die Daten nach einem Zugriff gesichert werden sollen oder zusätzlich auch die Informationen, wie sie vor der Änderung vorlagen. Man spricht dabei von Nachabbildern beziehungsweise Vorabbildern. Normalerweise werden Dateioperationen wie Schließen und Öffnen mit den Datenoperationen aufgezeichnet. Diese Protokollierung kann aber abgeschaltet werden, wenn die Anzahl der Einträge im Journal verringert werden soll. Bitte berücksichtigen Sie aber, dass bei einer Abschaltung nicht mehr aus dem Journal ermittelt werden kann, wer auf die Datei zugegriffen hat.



Der Befehl STRJRNPF

Sollen sowohl Vorabbilder als auch Nachabbilder aufgezeichnet werden, kann der Parameter „IMAGES“ auf *BOTH gesetzt werden. Über die Angabe von *OPNCLO im Parameter „OMTJRNE“ wird veranlasst, dass Dateioperationen nicht mitregistriert werden.

3.5.6
Seite 8

Die Protokollierung wurde nach der Datenfreigabe gestartet.

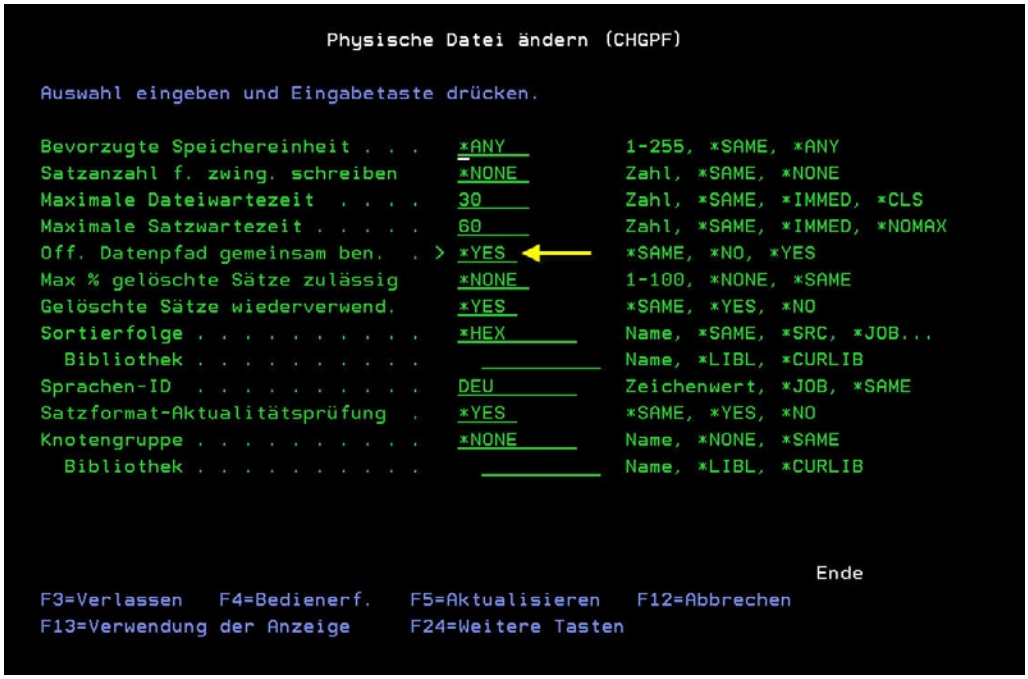
Eine weitere Möglichkeit, die Informationsflut einzugrenzen besteht darin, den Parameter „SHARE“ bei Dateien auf *YES zu setzen. In diesem Fall erstellt das System nur einen Eintrag pro Öffnen-, Schließen-Operation.

Zur kurzen Veranschaulichung sehen Sie die Befehls-Prompts für CRTPF und CHGPF.

```
                    Physische Datei erstellen (CRTPF)

Auswahl eingeben und Eingabetaste drücken.

Speicher zuordnen . . . . . . . ALLOCATE       *NO
Zusammenhängender Speicher . . . CONTIG         *NO
Bevorzugte Speichereinheit . . . UNIT           *ANY
Satzanzahl f. zwing. schreiben   FRCRATIO       *NONE
Maximale Dateiwartezeit . . . . WAITFILE       *IMMED
Maximale Satzwartezeit . . . . . WAITRCD        60
Off. Datenpfad gemeinsam ben. . . SHARE         *NO
Max % gelöschte Sätze zulässig   DLTPCT         *NONE
Gelöschte Sätze wiederverwend.   REUSEDLT       *NO
Sortierfolge . . . . . . . . . . SRTSEQ        *SRC
  Bibliothek . . . . . . . . . .
Sprachen-ID . . . . . . . . . . LANGID         *JOB
Zeichensatz-ID . . . . . . . . . CCSID         *JOB
Aktualisieren zulässig . . . . . ALWUPD        *YES
Löschen zulässig . . . . . . . . ALWDLT        *YES
Satzformat-Aktualitätsprüfung . LVLCHK         *YES
                                                                  Weitere ...
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

CRTPF

```
                    Physische Datei ändern (CHGPF)

Auswahl eingeben und Eingabetaste drücken.

Physische Datei . . . . . . . . > KONTO          Name
  Bibliothek . . . . . . . . . . >   BOEHMER      Name, *LIBL, *CURLIB
System . . . . . . . . . . . . . .   *LCL        *LCL, *RMT, *FILETYPE
```

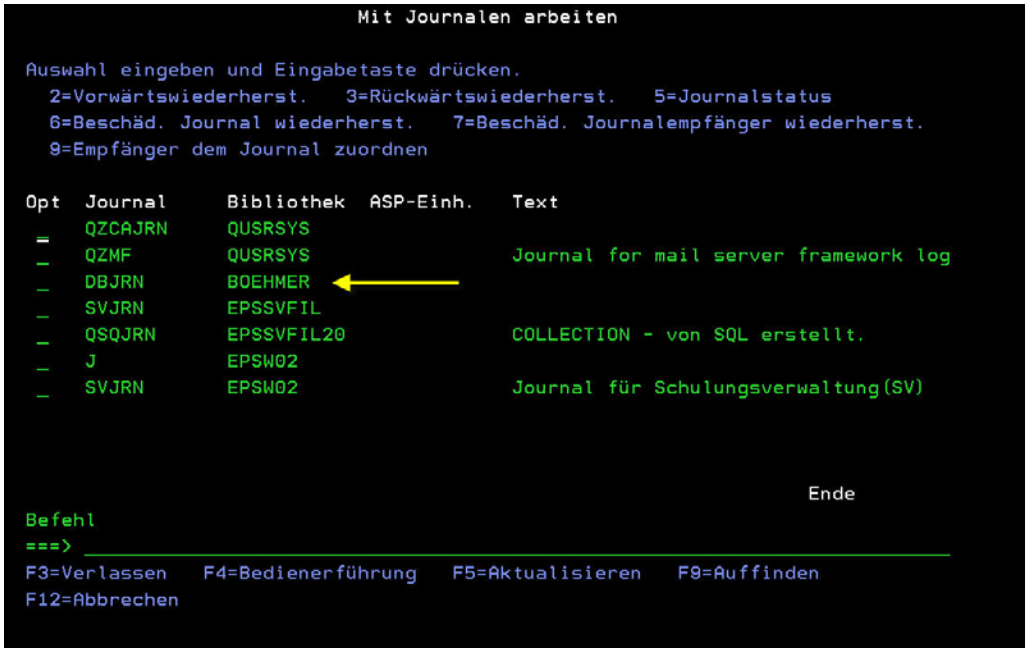


Mit der Einrichtung eines Umfelds zur Aufzeichnung haben Sie schon einen großen Schritt in Richtung Sicherheit und Datenkonsistenz getan. Aber damit allein ist es nicht getan. Denn im Laufe der Zeit wird sich die Zahl der aufgezeichneten Dateien sicherlich ändern und Sie sind gefordert, aktuelle Unterlagen zu erstellen. Denn sonst würde unter Umständen ein hoher Zeitaufwand entstehen, um die Umgebung wieder zu restaurieren. Zum Beispiel, wenn Journale aus irgendwelchen Gründen gelöscht oder vielleicht auch beschädigt wurden. Natürlich können Sie die Dinge durch ein Zurücksichern der betreffenden Objekte wieder ins Reine bringen, aber möglicherweise ist das erneute Starten der Aufzeichnung schneller erledigt als eine Wiederherstellung.

Die folgenden Abbildungen zeigen, wie man die aktuelle Umgebung relativ einfach dokumentieren kann.

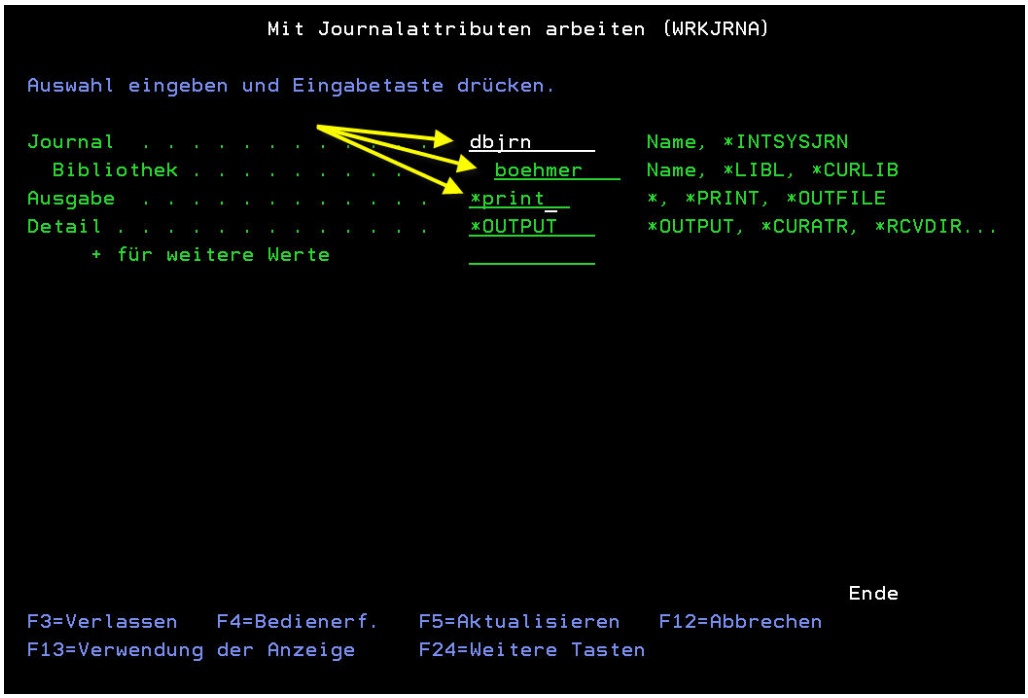
Zuerst stellen Sie mit dem Befehl WRKJRN fest, mit welchen Journalen Sie es zu tun haben.

3.5.6
Seite 10



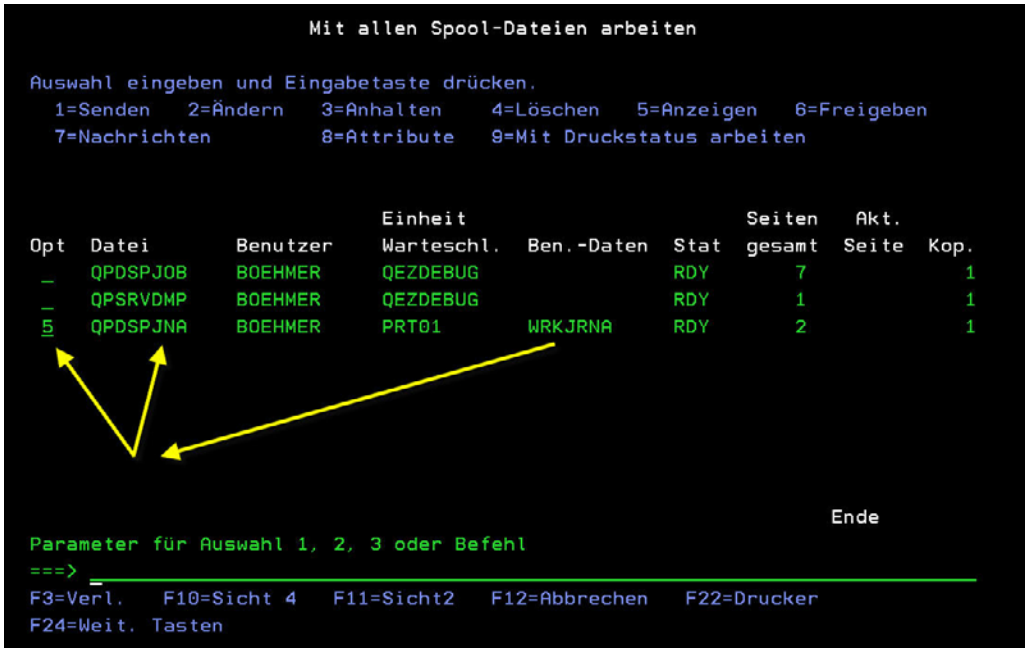
WRKJRN

Nachdem hier eine Übersicht gewonnen wurde, welche Journale vorhanden sind, kann mit dem Befehl WRKJRNA fortgefahren werden.

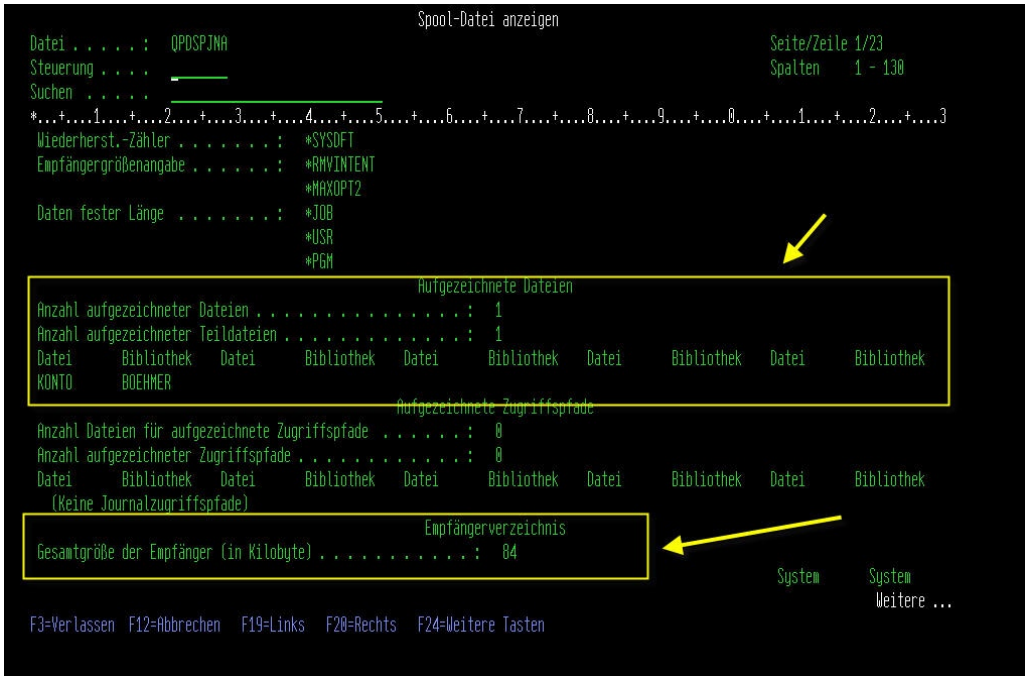


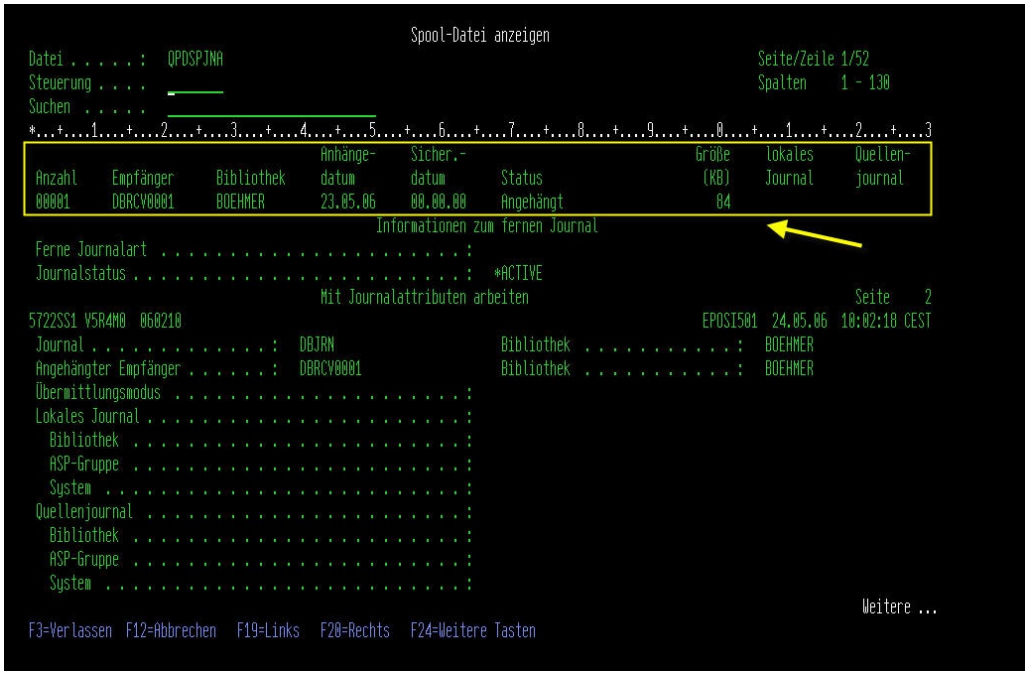
WRKJRNA

Durch die Angabe von *PRINT im Parameter „Ausgabe“ können die Informationen auf einen Drucker gelegt werden, um Listen für die Dokumentation zu erhalten. Hier einige Ausschnitte aus der erzeugten Druckausgabe.



*Ausgabe mit *PRINT*





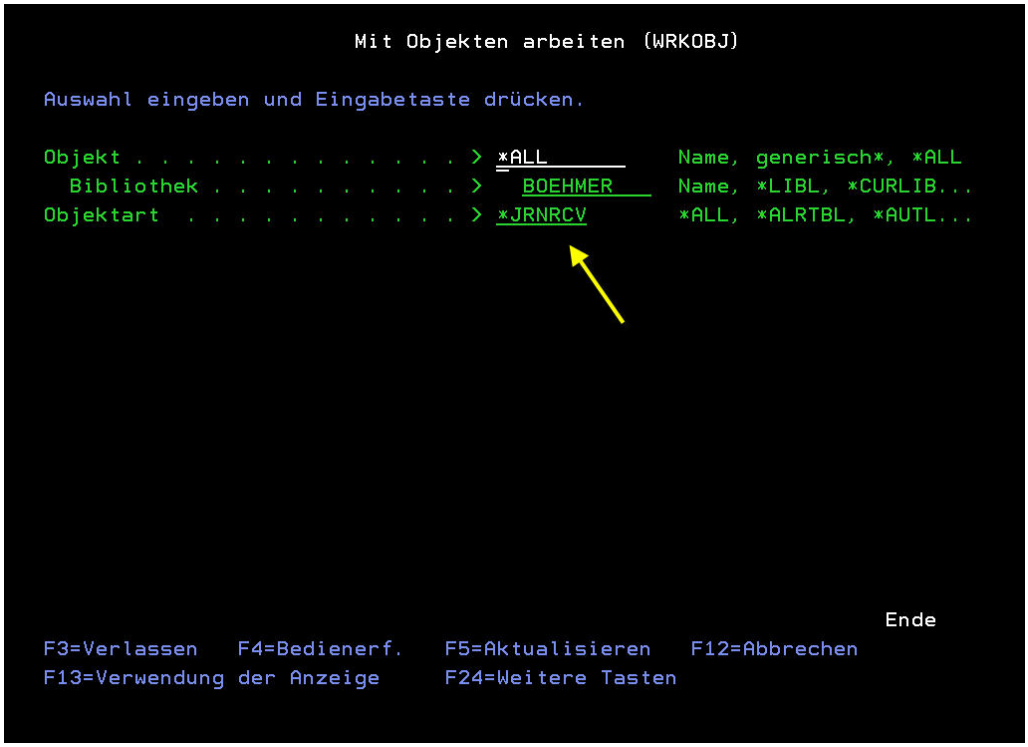
Ausschnitte aus dem Ergebnis

Die bestehenden Journale und Empfänger können, nachdem sie erstellt und in den Einsatz geschickt wurden, natürlich auch mit mehreren Befehlen geändert werden. Mit dem Befehl CHGJRN können Sie die Eigenschaften eines Journals beeinflussen oder auch Receiver an- und abhängen.

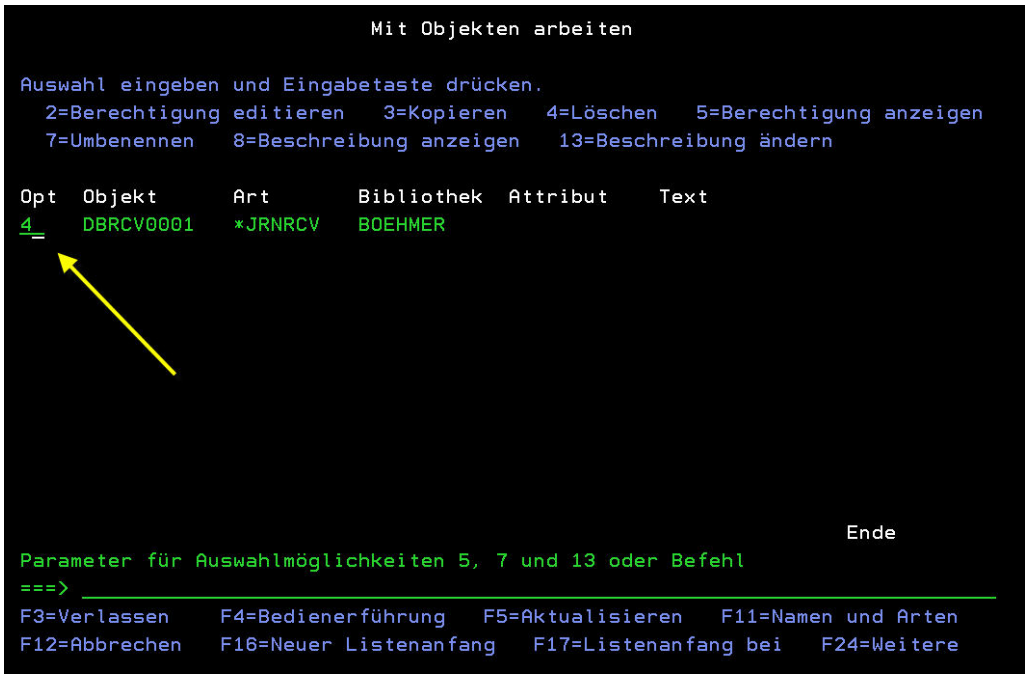
CHGJRN

Empfänger, die ihren Schwellwert erreicht haben und vom System abgehängt wurden (wenn das Journal entsprechend eingerichtet wurde), bleiben auf der Platte und nehmen auf Dauer erheblichen Speicherplatz in Anspruch. Wenn Sie der Meinung sind, dass die betreffenden Objekte nicht mehr zur Wiederherstellung benötigt werden, können die Empfänger auch gelöscht werden. Sie sollten sich trotzdem überlegen, ob die zu löschenden Objekte nicht noch gesichert werden sollten. Sie können natürlich über WRKOBJ gehen, um sich die Objekte herauszusuchen.

3.5.6
Seite 14



WRKOBJ



Die gefundenen Receiver

Oder Sie verwenden WRKJRNA, um dann über das Empfängerverzeichnis (Funktionstaste 15) mit einer Auswahloption die betreffenden Receiver zu löschen.

```
                  Mit Journalattributen arbeiten (WRKJRNA)

Auswahl eingeben und Eingabetaste drücken.

Journal  . . . . . .  dbjrn_____   Name, *INTSYSJRN
  Bibliothek . . . . .  boehmer___  Name, *LIBL, *CURLIB
Ausgabe  . . . . . .  *_______     *, *PRINT, *OUTFILE
Detail . . . . . .    *OUTPUT___   *OUTPUT, *CURATR, *RCVDIR...
     + für weitere Werte           __________
```

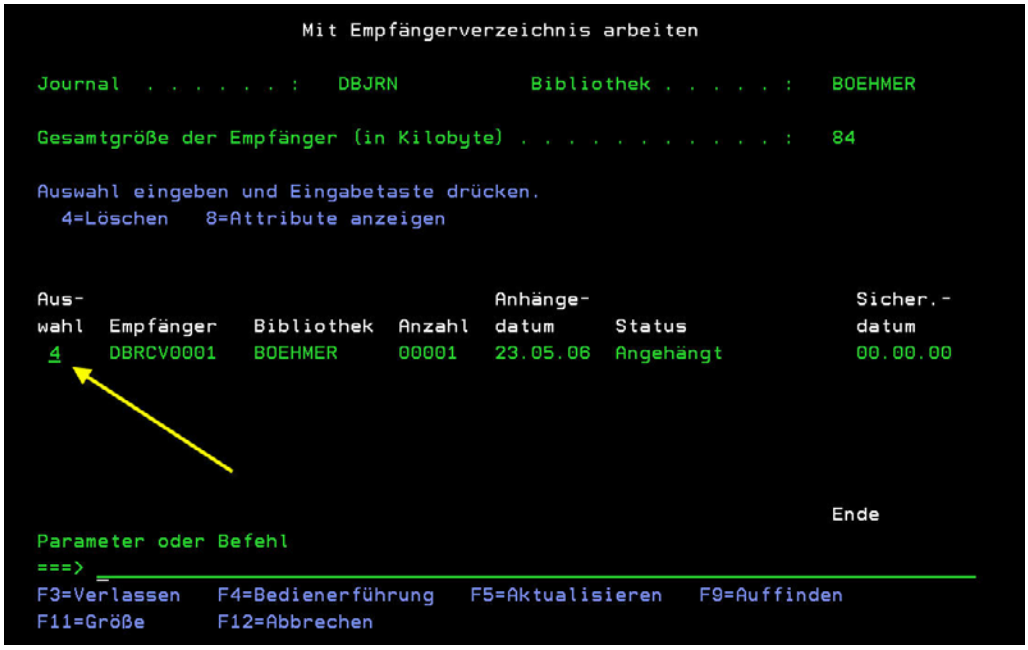
WRKJRNA

Nach der Bestätigung durch Datenfreigabe erhält man folgendes Bild.

```
_                      Mit Journalattributen arbeiten

 Journal  . . . . . :   DBJRN          Bibliothek . . . . . :   BOEHMER

 Angehängter Empfänger:  DBRCV0001     Bibliothek . . . . . :   BOEHMER

 Text . . . . . :   *BLANK

 ASP  . . . . . :   1              Aufgezeichnete Objekte:
 Nachr.warteschlange  :  QSYSOPR     Aktuell  . . . . . :          3
   Bibliothek . . . . :    *LIBL     Maximum  . . . . . :     250000
 Empfänger verwalten  :  *SYSTEM   Wiederherst.-Zähler  :  *SYSDFT
 Empfänger löschen  . :  *NO       Empfängergrößenangabe:  *RMVINTENT
 Journal-Cache  . . . :  *NO                               *MAXOPT2
 Verwaltungsverzög. . :  10        Daten fester Länge . :  *JOB
 Löschverzögerung . . :  10                                *USR
 Journalart . . . . . :  *LOCAL                            *PGM
 Journalstatus  . . . :  *ACTIVE
 Eintragsdaten minim. :  *NONE

                                                           Ende
 F13=Aufgez. Dateien   F15=Mit Empfängerverzeichnis arbeiten
 F19=Aufgez. Objekte anzeigen   F24=Weitere Tasten
```

Von hier aus erhält man über die markierte Funktionstaste das Empfängerverzeichnis und kann die gewünschten Receiver entfernen.

3.5.6
Seite 16

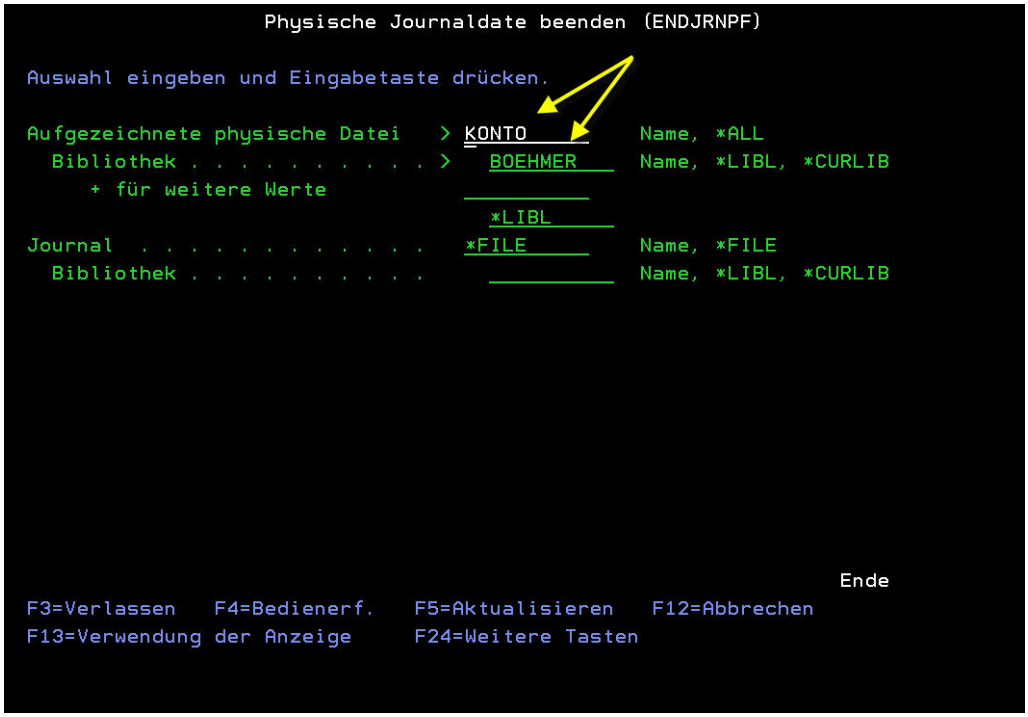


Bevor ein Empfänger gelöscht werden kann, muss er von seinem Journal abgehängt werden. Das wird mit CHGJRN erledigt.

Wenn es auf dem System Journale gibt, die nicht mehr verwendet werden, so sollten sie ebenfalls entfernt werden. Einerseits verbrauchen sie natürlich Plattenplatz, das ist das Eine.

Die andere Seite ist aber auch ein erhöhter Zeit- und Ressourcenbedarf bei IPLs, speziell bei nicht normalen Systembeendigungen.

Sollte die Commit-Steuerung aktiv sein und ein Journal ist der Steuerung zugeordnet, dann beendet man die Commit-Steuerung. Anschließend wird die Aufzeichnung aller Dateien beendet (die von diesem Journal protokolliert werden).



Aufzeichnung beenden

Nun können die gewünschten Receiver gelöscht werden. Wie das mit dem Empfängerverzeichnis gemacht werden kann, ist oben beschrieben. Ich rufe hier einfach die Bedienerführung des Befehls DLTJRNRCV auf.

Mit diesem Parameterinhalt werden Empfänger generisch entfernt.

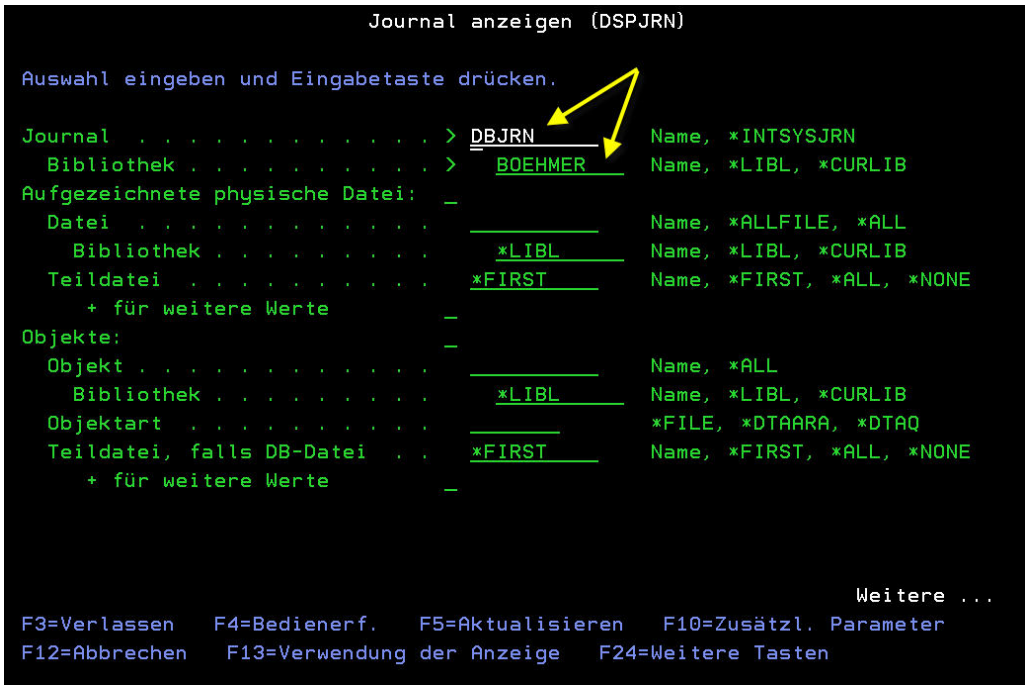
DLTJRNRCV

Um Journale und Receiver zu sichern, können auch verschiedene Befehle verwendet werden. So zum Beispiel geänderte Objekte sichern, Objekte sichern, Bibliothek sichern (SAVCHGOBJ, SAVOBJ, SAVLIB). Beim Zurücksichern von Objekten sollte eine Reihenfolge eingehalten werden:

- Journale
- Physische Dateien
- Logische Dateien
- Und die Receiver

Die Empfänger können im Grunde schon restauriert werden, sobald die Journale wiederhergestellt wurden. Wenn die einzelnen Objekte sich jedoch in verschiedenen Bibliotheken befinden, beachten Sie bitte die Information zu diesem Fall weiter oben.

Wir sollten uns einmal ansehen, welche Informationen im Journal auftauchen, wenn Änderungen an aufgezeichneten Dateien vorgenommen werden. Dazu schreibe ich mit SQL einige Sätze in eine Datei und lasse mir anschließend mit dem Befehl DSPJRN den Inhalt des Journals anzeigen.



DSPJRN

```
                    Journaleinträge anzeigen

Journal . . . . . :   DBJRN          Bibliothek . . . . . :   BOEHMER
Höchste Folgenummer in dieser Anzeige . . . . . : 00000000000000000009
Auswahl eingeben und Eingabetaste drücken.
  5=Gesamten Eintrag anzeigen

Ausw.    Folge   Code  Art   Objekt      Bibliothek  Job         Uhrzeit
 _           1    J     PR                           QZDASOINIT  12:13:25
 _           2    D     JF    KONTO       BOEHMER     QZDASOINIT  13:05:58
 _           3    F     JM    KONTO       BOEHMER     QZDASOINIT  13:05:58
 _           4    F     OP    KONTO       BOEHMER     QPADEV0001  13:06:51
 5           5    R     PT    KONTO       BOEHMER     QPADEV0001  13:06:52
 _           6    F     CL    KONTO       BOEHMER     QPADEV0001  13:06:52
 _           7    F     OP    KONTO       BOEHMER     QPADEV0001  13:07:11
 5           8    R     PT    KONTO       BOEHMER     QPADEV0001  13:07:11
 _           9    F     CL    KONTO       BOEHMER     QPADEV0001  13:07:14

                                                                    Ende
F3=Verlassen    F12=Abbrechen
```

Anzeige nach Datenfreigabe

Ich habe die aufgezeichneten Satzoperationen (PT) in der Abbildung mit einem kleinen Rahmen markiert und möchte mir die beiden Einträge über die Auswahl 5 genauer anschauen.

```
_                       Journaleintrag anzeigen

Objekt . . . . . :   KONTO          Bibliothek . . . . . :   BOEHMER
Teildatei . . . . :   KONTO
Unvollständige Daten :   Nein       Minim. Eintragsdaten :   *NONE
Folge . . . . . :   5
Code . . . . . :   R  - Operation mit bestimmtem Satz
Art . . . . . :   PT - Satz hinzugefügt

            Eintragsspez. Daten
Spalte      *...+....1....+....2....+....3....+....4....+....5
00001       '    Privatkonto                                  '
00051       '    '
```

Inhalt des ersten Eintrags

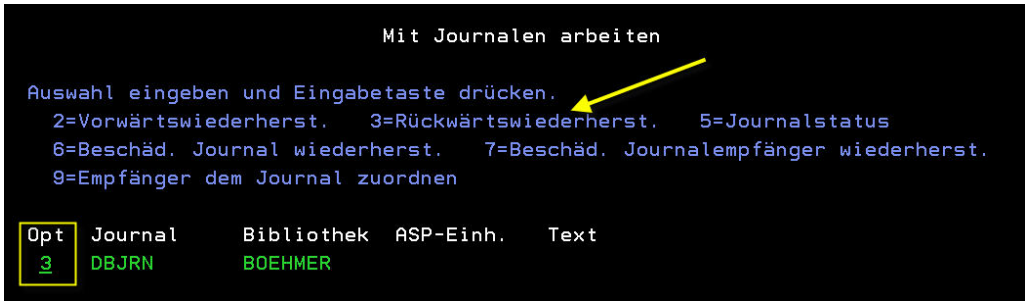
```
            Eintragsspez. Daten
Spalte      *...+....1....+....2....+....3....+....4....+....5
00001       '    Baukonto                                     '
00051       '    '
```

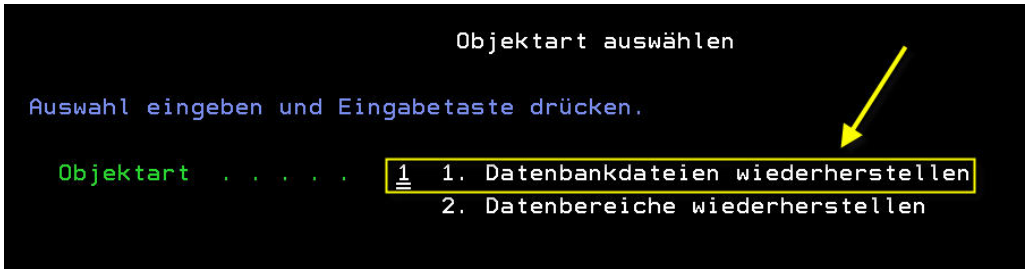
Und des zweiten Eintrags

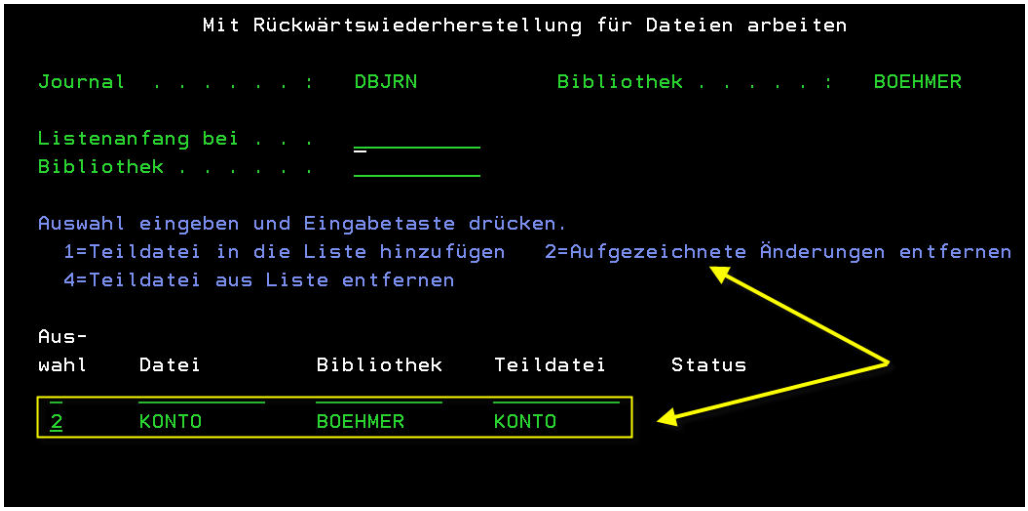
So stellt sich der aktuelle Dateiinhalt dar.

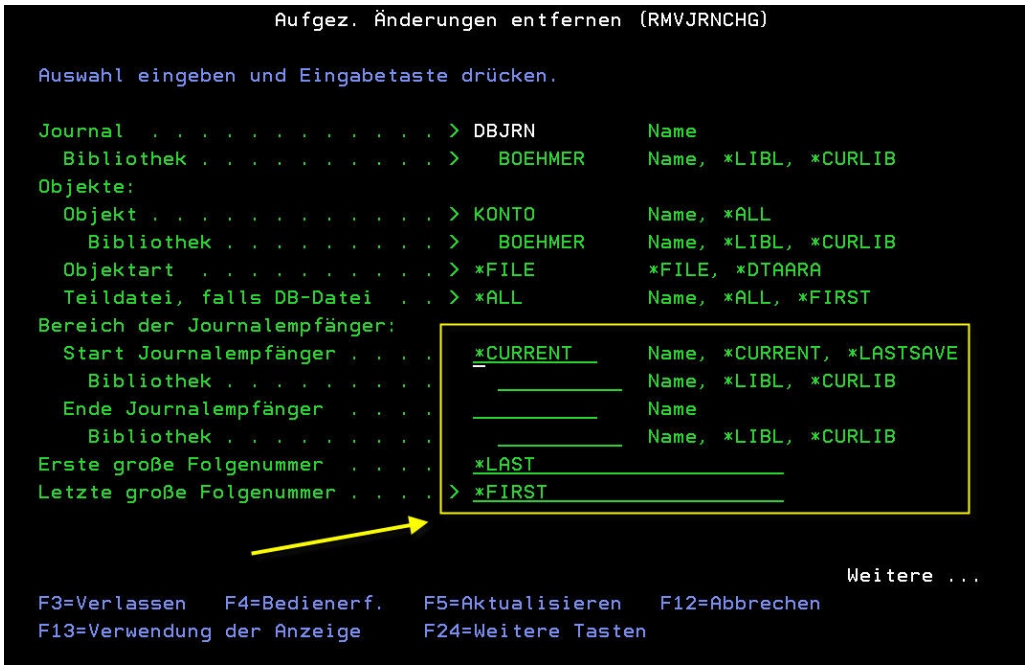
Ich möchte nun die Änderungen wieder rückgängig machen. Dazu werde ich das Journal veranlassen, vom aktuellen Zustand aus rückwärts gehend Eintrag für Eintrag die aufgezeichneten Änderungen zu entfernen. Das wähle ich über den Befehl „Journal“ aus. Die Abbildungsfolge erhält man nach der Parameter- und Auswahlangabe, die ich markiert habe.

WRKJRN







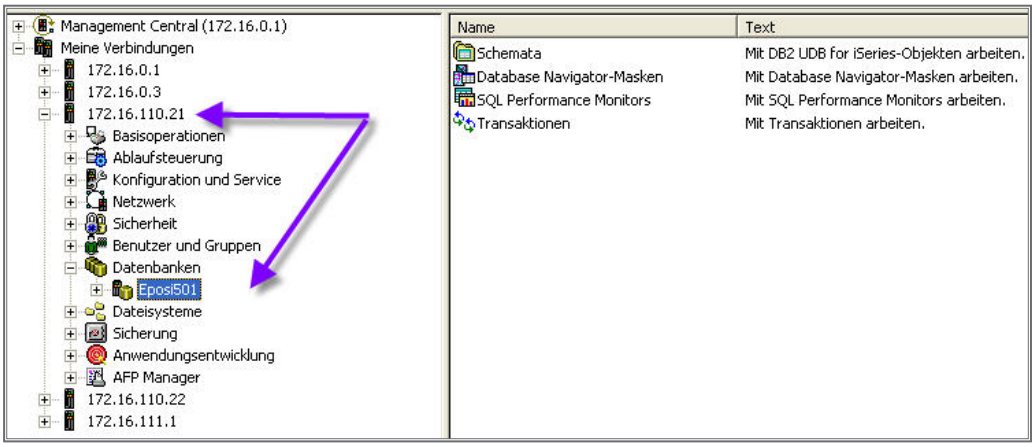


In diesem Bildschirm kann nun noch angegeben werden, auf welche Empfänger zurückgegriffen werden soll und durch welche Folgenummern der Einträge die zurückzusetzenden Änderungen eingegrenzt werden. Ich belasse es bei dem aktuellen Receiver und nach der Datenfreigabe werden die Änderungen rückgängig gemacht.

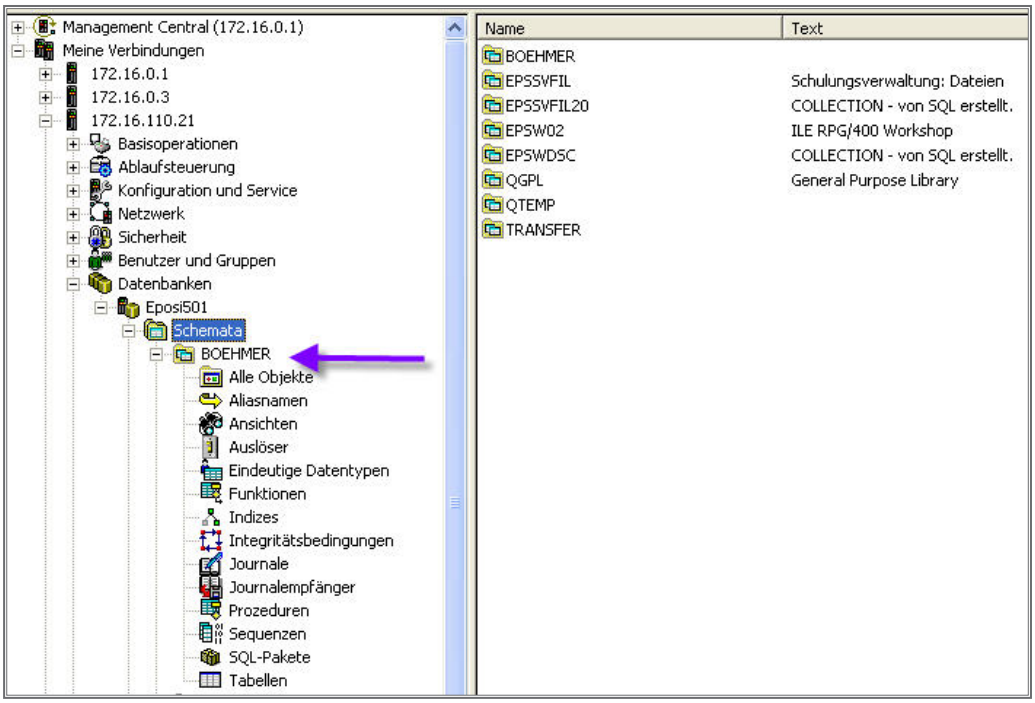
Beachten Sie bitte, dass eine Rückwärts-Wiederherstellung voraussetzt, dass die betreffenden Dateien mit Vor- und Nachabbildern aufgezeichnet werden.

Zum Abschluss dieses Kapitels möchte ich Ihnen noch einige Einstiegsmöglichkeiten über iSeries Navigator zeigen. Öffnen Sie einfach über „meine Verbindungen“ den Punkt „Datenbanken“.

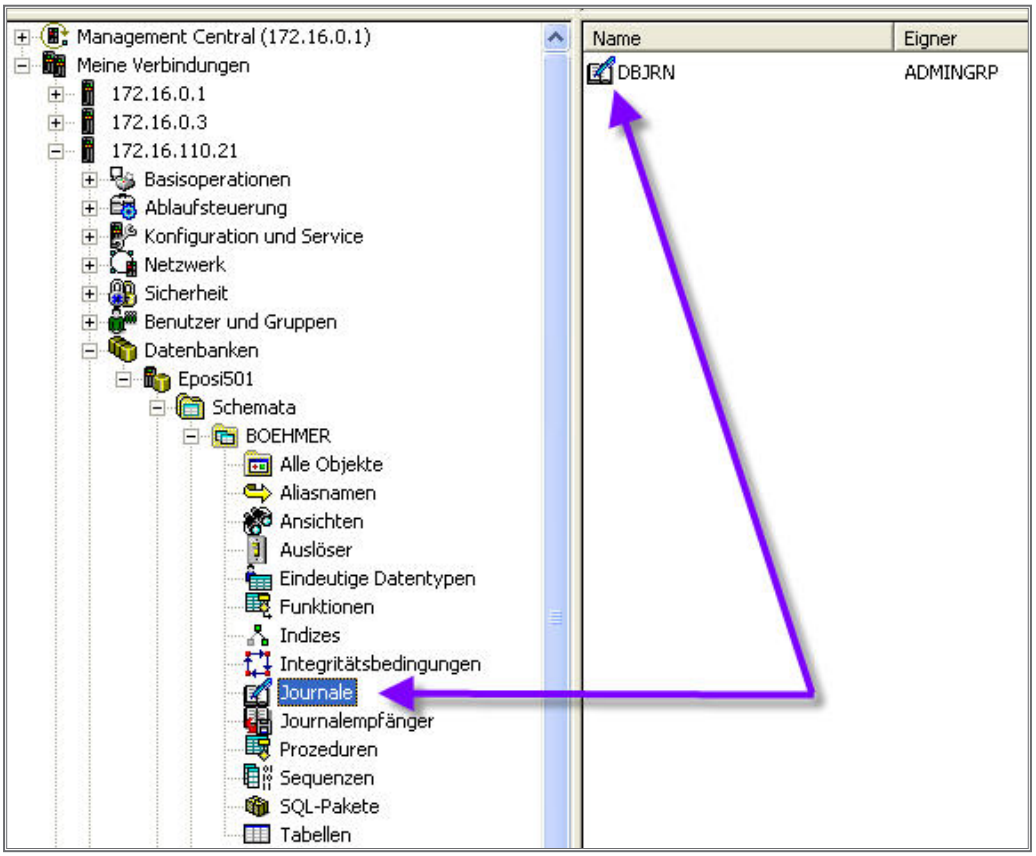
3.5.6
Seite 22



Anschließend navigieren Sie zu Ihrem System und dem Schema, dem Journale zugeordnet sind und öffnen auch diesen Unterpunkt.

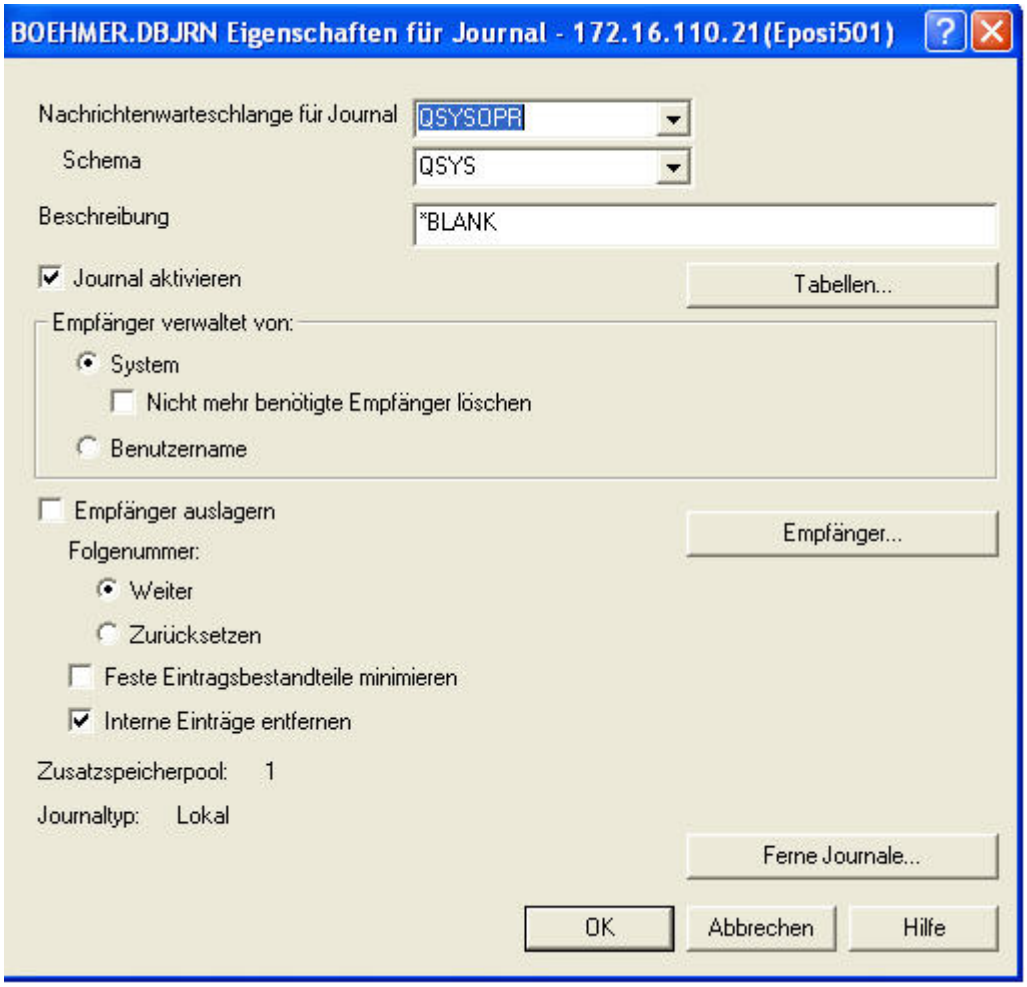


In der Auflistung sieht man die Einträge „Journal“ und „Journalempfänger“. Ich selektiere „Journale“ und im rechten Bereich taucht das Journal auf.



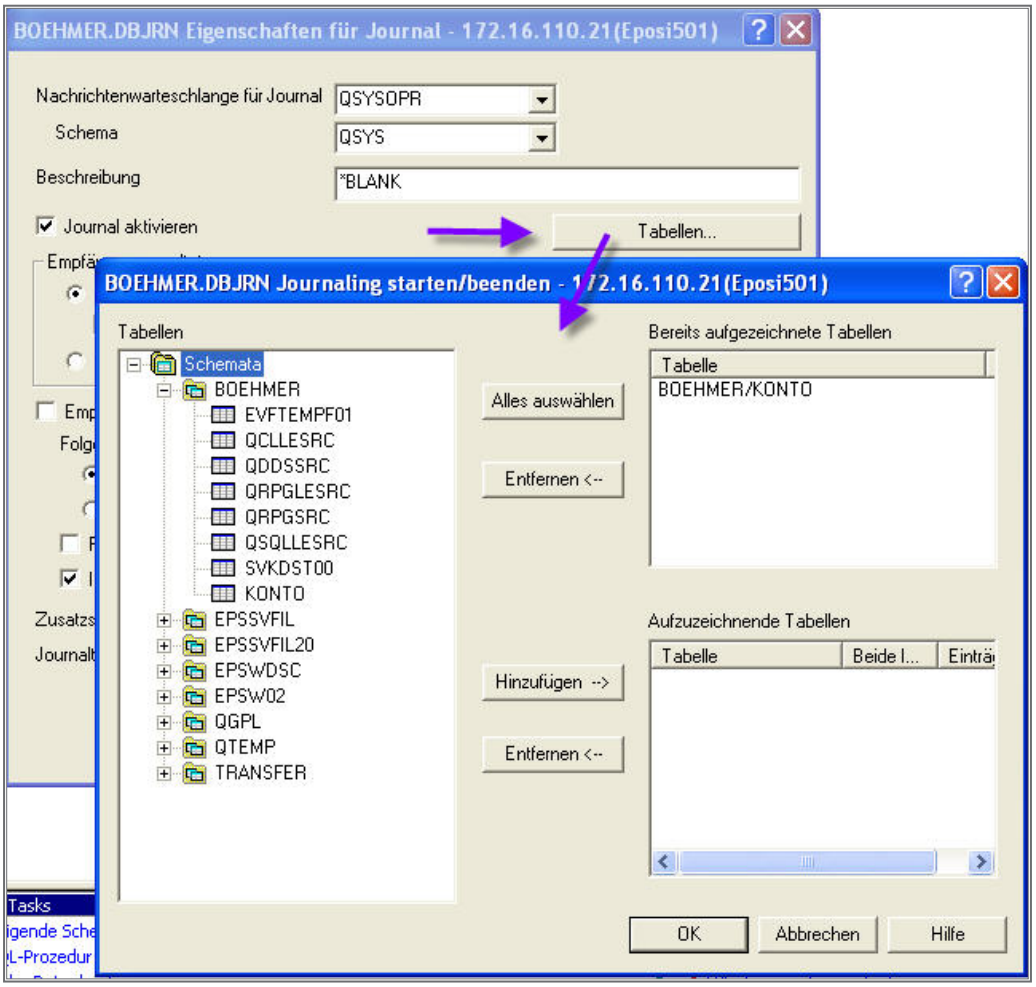
Über die Eigenschaften des Journals (rechte Maustaste) erhalte ich den folgenden Dialog.

3.5.6
Seite 24

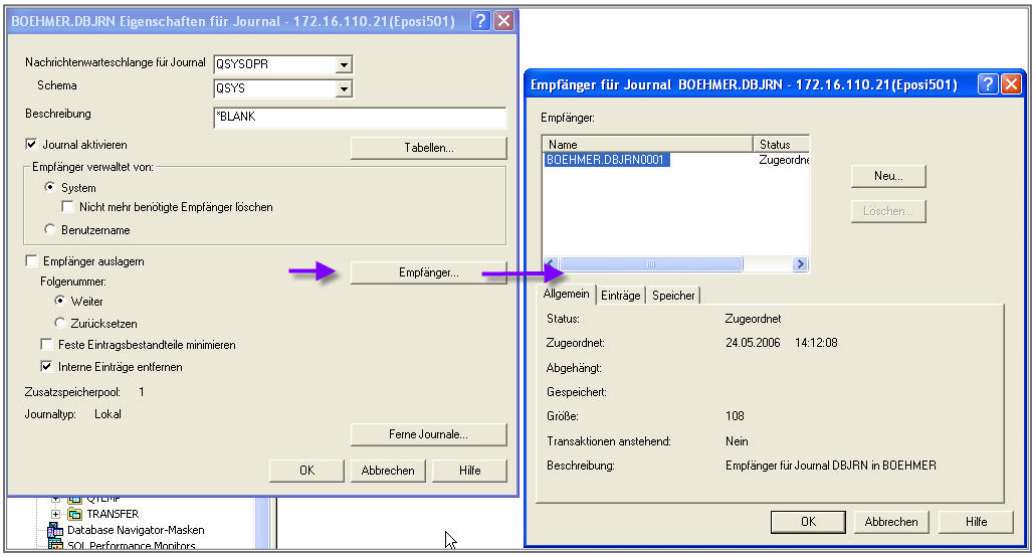


Für die Einstellungsmöglichkeiten sollten Sie bis zu diesem Punkt schon Verständnis entwickelt haben, denn ich spreche die einzelnen Parameter nicht noch einmal an.

Sie können aus diesem Bild heraus recht komfortabel mit den aufgezeichneten Tabellen und den zugeordneten Empfängern arbeiten. Ich gehe zunächst über die Schaltfläche „Tabellen…“. In dem folgenden Dialog können Sie die Aufzeichnung von Tabellen starten oder beenden.

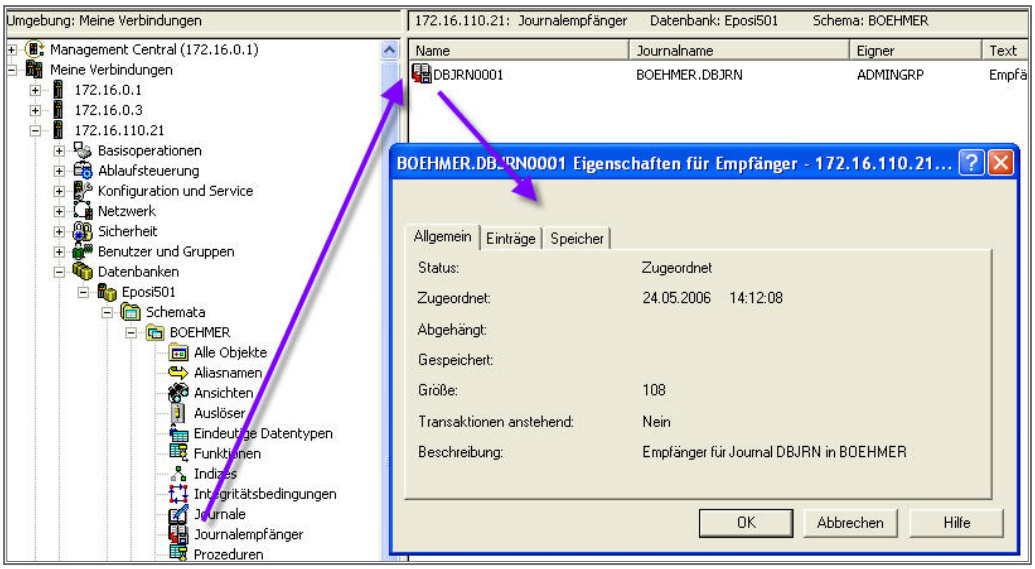


Über die Schaltfläche „Empfänger“ erreichen Sie folgenden Dialog. Auch dieser ist jetzt eigentlich selbsterklärend.



3.5.6
Seite 26

Ich werfe nun noch einen Blick auf die Empfänger selbst. Und auch in diesem Zusammenhang kann ich nur empfehlen: Machen Sie sich mit dem iSeries Navigator vertraut!



Statistikinformationen sammeln

Statistische Informationen über Datenbanken erfassen und auswerten, das kann man tun. Es stellt sich jedoch die Frage, warum und in welchen Umfeldern es sich empfiehlt. Es ist zwar kein großer Aufwand, aber so ganz nebenbei geht es auch nicht. Nehmen wir an, Sie haben eine detailliert modellierte Datenbank nach den Denkmustern des Entity-Relationship-Ansatzes im Einsatz: Eine Vielzahl von Tabellen, die mit komplexen SQL-Statements abgefragt werden. Dann können regelmäßig erfasste Statistikdaten Ihr System beim Datenzugriff ganz erheblich unterstützen.

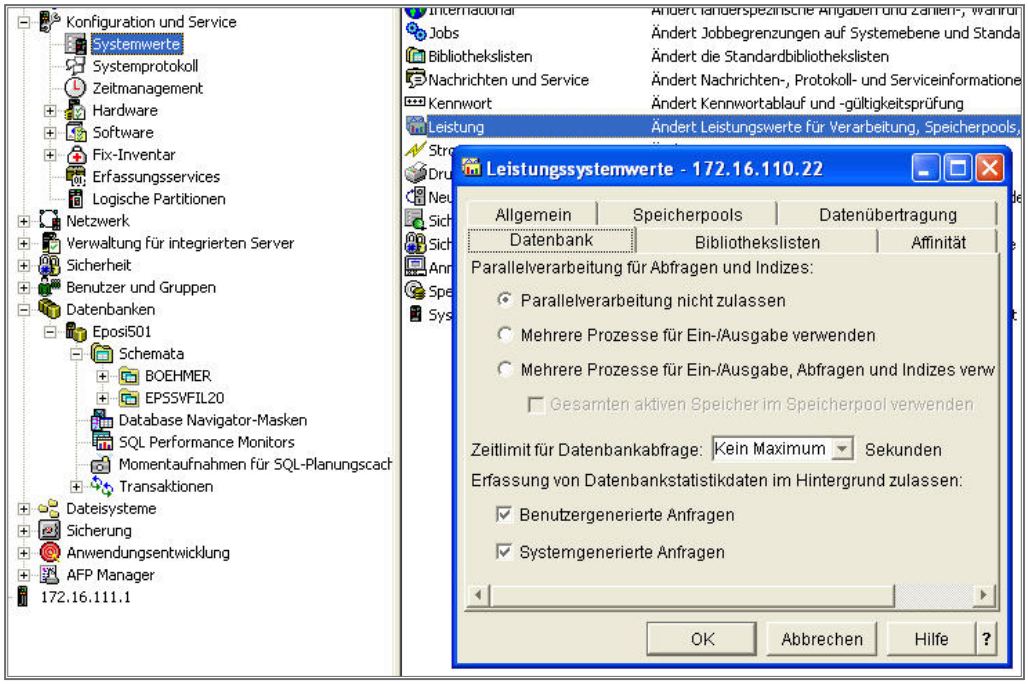
Bitte bedenken Sie, dass Ihr System automatisch Statistikdaten sammelt. Sie können nun aber auch eigene Erfassungen veranlassen. Verantwortlich ist dafür der Systemwert **QDBFSTCCOL**, der vier unterschiedliche Einstellungen erhalten kann. Beachten Sie die folgende Abbildung.

Der Systemwert QDBFSTCCOL

Die Bezeichnungen der einzelnen Einstellungen sind mehr oder weniger selbsterklärend. So bedeutet *ALL, dass sowohl das System als auch ein Benutzer die Sammlung von Statistikinformationen anfordern können.

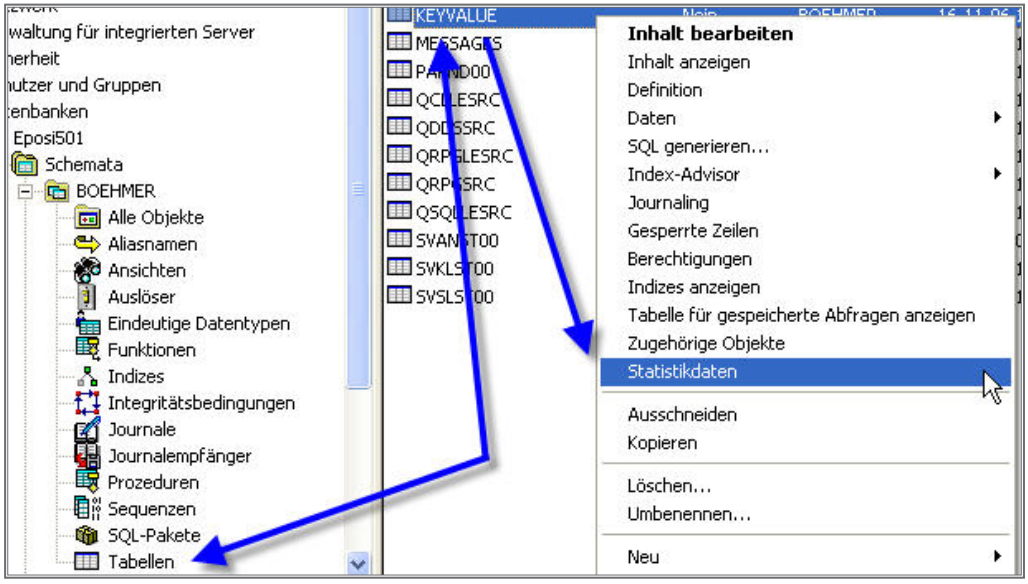
Wird *SYSTEM verwendet, so fragt nur das System an. Entsprechend kann nur eine benutzergesteuerte Anfrage zum Tragen kommen, wenn *USER eingesetzt wird.

*NONE verhindert jegliche Statistik.

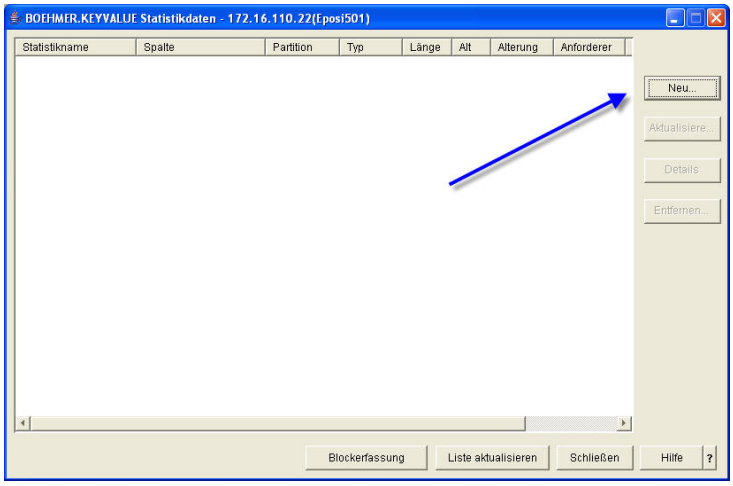


QDBFSTCCOL über den iSeries-Navigator

Navigieren Sie über den iSeries-Navigator zu einer Tabelle, für die Sie selbst eine Statistik initialisieren wollen und klicken Sie im Kontextmenü auf die Option „Statistikdaten“. Sie erhalten dann folgendes Bild.



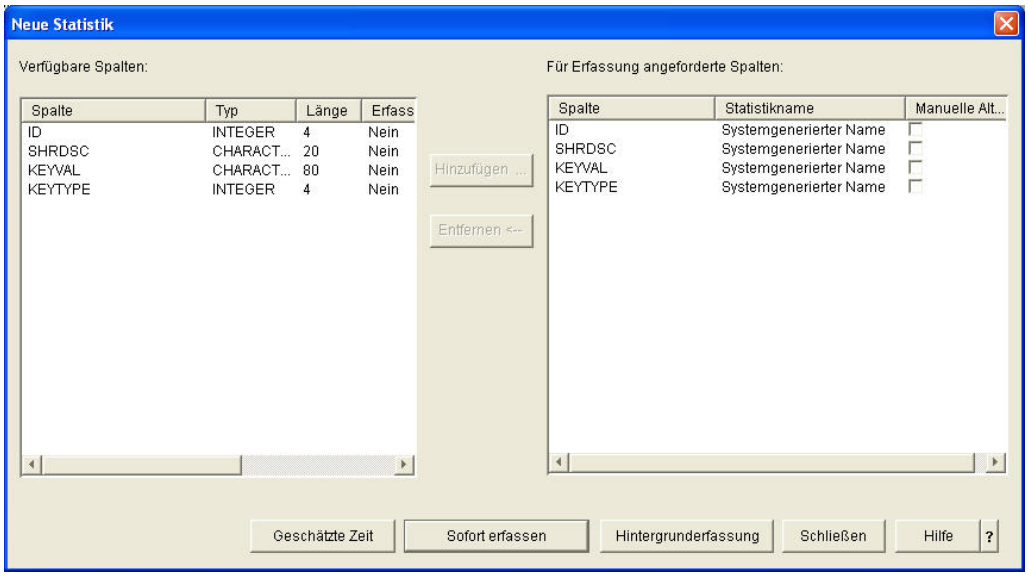
Neue Statistikdatenerfassung erstellen (1)



Neue Statistikdaten erfassen (2)

Dieser Dialog ist zurzeit noch ohne Inhalt und ich werde über den Pushbutton „Neu…“ eine neue Statistik anlegen. Unter Umständen ist bei Ihnen schon die eine oder andere Statistik sichtbar. Wenn Sie auf „Blockerfassung“ klicken, können Sie die automatische Erfassung durch das System starten. Wenn eine Blockerfassung aktiviert ist, lässt sie sich über diesen Button auch wieder abschalten.

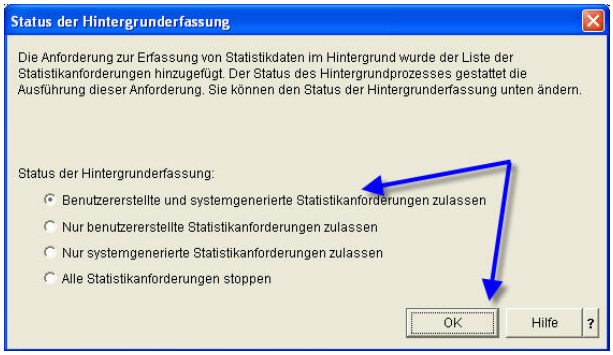
Im folgenden Dialog habe ich schon alle Spalten der Tabelle ausgewählt und für die Erfassung angefordert.



Spalten einer zu überwachenden Tabelle wählen

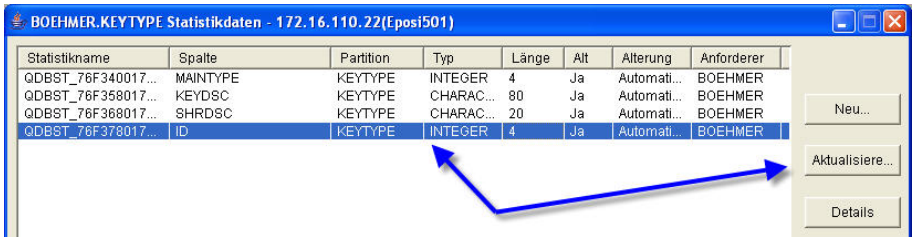
In diesem Fall möchte ich, dass das System die Statistikdaten im Hintergrund erfasst und klicke dazu auf den Button „Hintergrunderfassung“. Darauffolgend besteht noch die Möglichkeit, die Art der Erfassung zu ändern. Ich belasse es jedoch bei dem aktuell vorliegenden Wert.

3.5.7
Seite 4

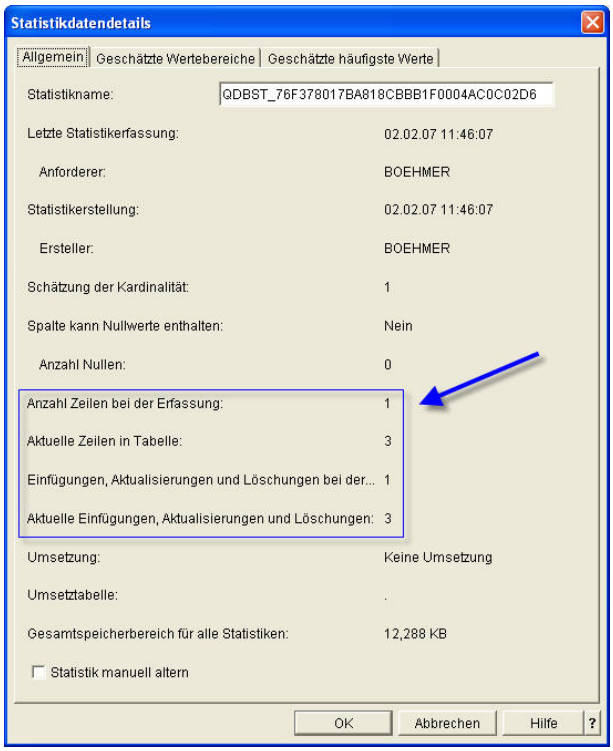


Aktueller Erfassungsstatus

Die Sammlung der Daten wurde nun dem System in Auftrag gegeben und wird auch im Ausgangsdialog (siehe Abb. 3.5.7 – 4) dargestellt. Ich habe einfach einige Operationen auf der überwachten Tabelle KEYTYPE durchgeführt und schaue mir die Statistik an.



Statistik nach durchgeführten Änderungen (1)



Statistik nach durchgeführten Änderungen (2)

Beachten Sie den kleinen Rahmen: Dort kann man sehen, dass Aktionen aufgezeichnet wurden. Die IBM hat in diesem Bereich keine grünen Bildschirme vorgesehen, Sie sind hier also auf den iSeries-Navigator angewiesen. Nun denn, auf diesen Umstand wurde schon mehrere Male deutlich hingewiesen.

Neben der Verwendung des iSeries-Navigators ist aber auch der Einsatz verschiedener APIs möglich.

List Statistics Collections (QDBSTLS, QdbstListStatistics) API

Required Parameter Group:

1	Qualified user space name	Input	Char(20)
2	Format of output	Input	Char(8)
3	Input data	Input	Char(*)
4	Length of input data	Input	Binary(4)
5	Format of input data	Input	Char(8)
6	Error code	I/O	Char(*)

Service Program Name: QDBSTMGR

Default Public Authority: *USE

Threadsafe: Yes

Auflistung bestehender Sammlungen für Tabelle

Request Statistics Collections (QDBSTRS, QdbstRequestStatistics) API

Required Parameter Group:

1	Input data	Input	Char(*)
2	Length of input data	Input	Binary(4)
3	Format of input data	Input	Char(8)
4	Feedback area	Output	Char(*)
5	Length of feedback area	Input	Binary(4)
6	Feedback keys	Input	Array(*) of Binary(4)
7	Number of feedback keys	Input	Binary(4)
8	Error code	I/O	Char(*)

Service Program Name: QDBSTMGR

Default Public Authority: *USE

Threadsafe: Yes

Sammlung für Tabelle anfordern

Update Statistics Collections (QDBSTUS, QdbstUpdateStatistics) API

Required Parameter Group:

1	Input data	Input	Char(*)
2	Length of input data	Input	Binary(4)
3	Format of input data	Input	Char(8)
4	Feedback area	Output	Char(*)
5	Length of feedback area	Input	Binary(4)
6	Feedback keys	Input	Array(*) of Binary(4)
7	Number of feedback keys	Input	Binary(4)
8	Error code	I/O	Char(*)

Service Program Name: QDBSTMGR

Default Public Authority: *USE

Threadsafe: Yes

Sammlung aktualisieren

Weitere APIs sind QDBSTSLRS (Liste von Spalten und deren Kombinationen sowie Teildatei-Namen mit angeforderten Statistiken), QDBSTLDS (Liste existierender detaillierter Informationen), QDBSTDS (Löschen abgeschlossener Sammlungen) und QDBSTCRS (Löschen nicht abgeschlossener Sammlungen). Die exakte Dokumentation zu APIs können Sie sich im Information Center beschaffen.
(http://publib.boulder.ibm.com/iseries/)

Warum aber ist die Erfassung dieser statistischen Daten von Vorteil? Wie eingangs bereits erwähnt, wird das System beim Zugriff auf Daten unterstützt. Das wird Performance-Vorteile zur Folge haben. Nun ist es nicht so, dass durch erfasste Statistiken die Abfrage an sich optimiert wird. Plant das System einen Datenbankzugriff, so wird es versuchen, bestehende Statistiken auf unterstützende Informationen hin zu untersuchen. Mit Hilfe der Resultate wird es dann beginnen, die gewünschte Abfrage zu optimieren. Wenn aber keine Statistiken in Bezug auf die betroffenen Tabellen vorliegen, werden natürlich Standardannahmen getroffen. Dass diese Werte nicht zwingend mit den tatsächlichen Gegebenheiten in Deckung zu bringen sind, dürfte einsehbar sein.

Eine andere Frage ist sicherlich, in wie weit das System durch das fortlaufende Sammeln von Daten belastet wird, da einerseits ständig Prozesse ablaufen und andererseits bei vielen bewegten Daten natürlich auch an den benötigten Speicherbedarf gedacht werden sollte. Sie können beruhigt sein, die Jobs laufen mit einer niedrigen Priorität, so dass Ihr System dadurch nicht über Gebühr belastet wird. Auch bei den Anforderungen an den Speicherplatz ist das System zurückhaltend. So werden Informationen im Kilobytebereich benötigt (pro Tabelle).

Datenbankstruktur darstellen

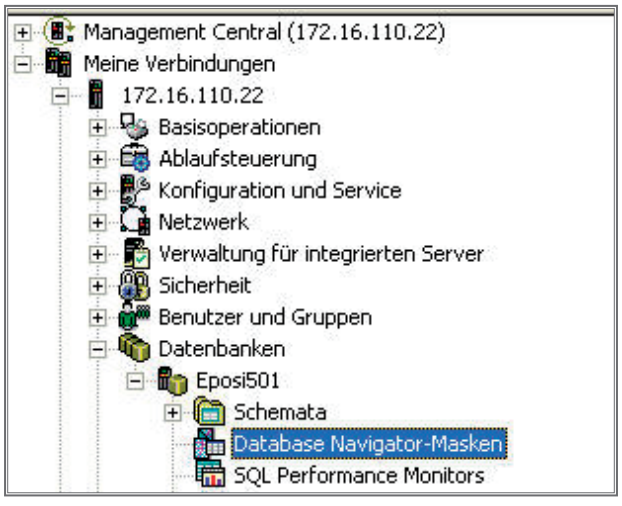
Wenn man die Fülle der Dateien, Tabellen, Indizes sowie die vielen weiteren Objekte einer Datenbank betrachtet, stellt man schnell fest, dass eine grafische Übersicht eine große praktische Hilfe sein kann. Daraus ergeben sich folgende Fragen: Welche logische Datei ist von welcher physikalischen Datei abhängig? Welche Indizes sind einer Tabelle zugeordnet worden? Bestehen Beziehungen in Form von referenziellen Integritätsbedingungen?

Für die Beschaffung von Informationen dieser Art steht Ihnen der CL-Befehl „Datenbankrelationen anzeigen“ zur Verfügung.

```
                              Befehl auswählen

Auswahl eingeben und Eingabetaste drücken.
  1=Auswählen

Ausw  Befehl      Bibliothek  Text
      DSPDBR      QSYS        Datenbankrelationen anzeigen
 =
```

Datenbankrelationen anzeigen

Sie bekommen für die Beantwortung dieser Fragen ein Visualisierungswerkzeug vom iSeries Navigator zur Verfügung gestellt. Dieses Tool erreichen Sie über den Unterpunkt „Datenbanken“ und den RDB-Namen Ihres Systems.



Datenbank-Navigation

Das Abrufen der gegenseitigen Abhängigkeiten ist auch über die Abfrage mehrerer Tabellen und Views des Systemkatalogs möglich. Die Vorgehensweise dazu werde ich Ihnen im Folgenden vorstellen. Beginnen wir mit den Möglichkeiten des DSPDBR-Befehls.

3.5.9
Seite 2

Der Befehl DSPDBR – Datenbankrelationen anzeigen

Sie können diesen Befehl verwenden, um die Beziehungen zwischen mehreren Dateien Ihres System zu ermitteln. Diese Informationen können angezeigt, gedruckt oder in eine Datei ausgegeben werden. Varianten der ausgelieferten Daten sind:

- die Abfrage von Dateien, die ein bestimmtes Satzformat nutzen,
- die Anzeige von Dateien, die von einer bestimmten Datenbankdatei abhängig sind (Datenabhängigkeit),
- Abhängigkeiten in Form von referenziellen Integritätsbedingungen.

Beispiel für die Verwendung von Satzformaten:

```
                    Datenbankrelationen anzeigen (DSPDBR)

Auswahl eingeben und Eingabetaste drücken.

Datei  . . . . . . . . . . . . . > *ALL        Name, generisch*, *ALL
  Bibliothek . . . . . . . . . . >   BOEHMER   Name, *LIBL, *CURLIB...
Ausgabe  . . . . . . . . . . . .   *          *, *PRINT, *OUTFILE

                Zusätzliche Parameter

Teildatei  . . . . . . . . . . .   *NONE       Name, *NONE
Satzformat . . . . . . . . . . . > PAKND00    Name, generisch*, *NONE, *ALL
```

Parametrisieren des Befehls

Drücken Sie die „Datenfreigabe“; anschließend werden die Daten auf dem Bildschirm angezeigt. Der Anzeigeausschnitt stellt allgemeine Informationen zur Abhängigkeit dar. Sie erhalten die Aufrufparameter des Befehls, der in diesem Beispiel nach allen Dateien sucht, die das Satzformat PAKND00 verwenden. Achten Sie bitte darauf, die Zahl der durchsuchten Bibliotheken niedrig zu halten, da einige Zeit für die Untersuchung beansprucht werden kann.

```
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+
  26.11.07           Datenbankrelationen anzeigen
 DSPDBR Befehlseingabe
   Datei . . . . . . . . . . . . . . . . . :           *ALL
     Bibliothek  . . . . . . . . . . . . . :           BOEHMER
   Teildatei . . . . . . . . . . . . . . . : MBR       *NONE
   Satzformat  . . . . . . . . . . . . . . : RCDFMT    PAKND00
   Ausgabe . . . . . . . . . . . . . . . . : OUTPUT    *
 Spezifikationen
   Dateiart  . . . . . . . . . . . . . . . :           Physisch
   Datei . . . . . . . . . . . . . . . . . :           PAKND00
     Bibliothek  . . . . . . . . . . . . . :           BOEHMER
     Teildatei . . . . . . . . . . . . . . :           *NONE
     Satzformat  . . . . . . . . . . . . . :           PAKND00
     Anzahl abhängiger Dateien . . . . . . :              9
```

Allgemeine Informationen

3.5.9.1
Seite 2

Das Satzformat wird in diesen Dateien verwendet:

```
Dateien, die das Satzformat der angegebenen Datei verwenden
   Abhängige Datei      Bibliothek
      PAKND00           BOEHMER
      PAKND01           BOEHMER
      PAKND00           LGA01
      PAKND00           LGA02
      PAKND00           LGA03
      PAKND00           LGA04
      PAKND00           LGA05
      PAKND00           LGA07
      PAKND00           LGA08
```

Anzeige der Dateien 1

Der letzte Teil der Liste ist in der nächsten Abbildung gezeigt.

```
Spezifikationen
  Dateiart . . . . . :          Logisch
  Datei . . . . . :             PAKND01
    Bibliothek . . . . . :      BOEHMER
    Teildatei . . . . . :       *NONE
    Satzformat . . . . . :      PAKND00
    Anzahl abhängiger Dateien . . . . . :          9
Dateien, die das Satzformat der angegebenen Datei verwenden
   Abhängige Datei      Bibliothek
      PAKND00           BOEHMER
      PAKND01           BOEHMER
      PAKND00           LGA01
      PAKND00           LGA02
      PAKND00           LGA03
      PAKND00           LGA04
      PAKND00           LGA05
      PAKND00           LGA07
      PAKND00           LGA08
```

Anzeige der Dateien 2

Lassen Sie sich eine einfache Darstellung der Beziehungen einer Datenbankdatei mit dem folgenden Aufruf von DSPDBR anzeigen.

```
                    Datenbankrelationen anzeigen (DSPDBR)

Auswahl eingeben und Eingabetaste drücken.

Datei  . . . . . > PAKND00     Name, generisch*, *ALL
  Bibliothek . . . . . >  BOEHMER    Name, *LIBL, *CURLIB...
Ausgabe  . . . . . *           *, *PRINT, *OUTFILE
```

Relationen in Bezug auf die Datei PAKND00

Die Wiedergabe der allgemeinen Befehlsparameter habe ich in der nächsten Abbildung entfallen lassen; es werden nur die Abhängigkeiten gezeigt. Die letzte Zeile stellt eine Relation (Rl) in Form einer Integritätsbedingung zwischen PAANS00 und PAKND00 dar. Es ist vorteilhaft, wenn sprechende Bezeichnungen für RIs verwendet werden. Dieser Relation kann ich aufgrund des Präfix „FK_“ entnehmen, das es sich um eine Fremdschlüsselbeziehung handelt (Foreign Key). Hierbei handelt es sich natürlich um meinen „Standard“; Sie sollten sich selbst auch einen überlegen.

```
Spezifikationen
  Dateiart . . . . . :          Physisch
  Datei . . . . . :             PAKND00
    Bibliothek . . . . . :      BOEHMER
    Teildatei . . . . . :       *NONE
    Satzformat . . . . . :      *NONE
    Anzahl abhängiger Dateien . . . . . :          2
Von angegebener Datei abhängige Dateien
  Abhängige Datei       Bibliothek    Abhängigkeit JREF    Integritätsbed.
     PAKND01            BOEHMER       Daten
     PAANS00            BOEHMER       Integritätsb.        FK_ANSPR_KUNDE
```

Übersicht der Relationen

Sie können auch eine Ausgabe der abgefragten Informationen in eine Datenbankdatei veranlassen. Geben Sie dazu folgende Parameter ein:

```
                    Datenbankrelationen anzeigen (DSPDBR)

Auswahl eingeben und Eingabetaste drücken.

Datei  . . . . . . . . . . . . > PAKND00      Name, generisch*, *ALL
  Bibliothek . . . . . . . . . >   BOEHMER    Name, *LIBL, *CURLIB...
Ausgabe  . . . . . . . . . . . > *OUTFILE     *, *PRINT, *OUTFILE
Datei für Ausgabe  . . . . . . > DSPDBR       Name
  Bibliothek . . . . . . . . . >   BOEHMER    Name, *LIBL, *CURLIB
Auswahl für Ausgabeteildatei:
  Teildatei für Ausgabe  . . . .   *FIRST      Name, *FIRST
  Sätze ersetzen od. hinzufügen    *REPLACE    *REPLACE, *ADD

                Zusätzliche Parameter

Teildatei  . . . . . . . . . . > *NONE        Name, *NONE
```

Parameter für Ausgabe in eine Datenbankdatei

Anschließend werfen Sie einen Blick auf den Inhalt der Datei. Wenn Sie den Befehl DSPPFM einsetzen, erhalten Sie untenstehendes Bild.

```
                    Physische Teildatei anzeigen
Datei  . . . . . :  DSPDBR            Bibliothek . . . :  BOEHMER
Teildatei  . . . :  DSPDBR            Satz . . . . . . :  1
Steuerung  . . . :  _________         Spalte . . . . . :  1
Suchen . . . . . :  ________________________________
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9....+....0....+....1....+....2....+....3
PPAKND00   BOEHMER   *NONE     *NONE     00002107112711264 6PAKND01   BOEHMER           D0000EPOSI501
PPAKND00   BOEHMER   *NONE     *NONE     00002107112711264 6PAANS00   BOEHMER           C0000EPOSI501BOEHMER     FK_ANSPR_KUNDE
```

Ausgabe mit DSPPFM

Sie haben auch die Möglichkeit, mit RUNQRY zu arbeiten; damit bekommen Sie auch die Spaltenüberschriften angezeigt.

```
Auswahl oder Befehl
===> RUNQRY QRYFILE((BOEHMER/DSPDBR))
```

RUNQRY aufrufen

In der ersten Abbildung ist der linke Teil des Berichts dargestellt; die Fortsetzung der Ausgabezeile erfolgt in der zweiten Grafik.

```
                                                Bericht anzeigen
                                                                                    Breite des Berichts  . :     475
Anfang auf Zeile . . . . . .  ____                                                Anfang in Spalte . . . . . .  ____
Zeile  ....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9....+...10....+...11....+...12....
       Dateiart  Datei      Bibliothek  Teildatei   Satzformat  Anzahl der      Abrufdatum-    Abhängige Datei  Abhängige Biblioth
                                                                Abhängigkeiten  und Uhrzeit
000001    P      PAKND00    BOEHMER     *NONE       *NONE              2      1071127112646    PAKND01            BOEHMER
000002    P      PAKND00    BOEHMER     *NONE       *NONE              2      1071127112646    PAANS00            BOEHMER
****** ********  Ende des Berichts  ********
```

Erste Hälfte des Berichts

```
                                                Bericht anzeigen
                                                                                    Breite des Berichts  . :     475
Anfang auf Zeile . . . . . .  ____                                                Anfang in Spalte . . . . . .  ____
Zeile  +...13....+...14....+...15....+...16....+...17....+...18....+...19....+...20....+...21....+...22....+...23....+...24....+...
       ek  Abhängige Teildatei  Art der Abhängigkeit  JFILE   JREF  System-   Bibliothek der        Name der
                                                      Anzahl        name     Integritätsbedingung  Integritätsbedingung
000001                                  D                0      0   EPOSI501
000002                                  C                0      0   EPOSI501       BOEHMER          FK_ANSPR_KUNDE
****** ********  Ende des Berichts  ********
```

Zweite Hälfte des Berichts

Die Art der Abhängigkeit ist bei der Ausgabe in Datenbankdateien kodiert worden. Sie erhalten zur Übersicht die unterschiedlichen Kodierungen in der nächsten Tabelle.

Type	Code	Description
Constraint	C	The physical file is dependent on the data in another physical file with which it is associated via a constraint.
Data	D	The file or member is dependent on the data in a member of another file.
Access path sharing	I	The file member is sharing an access path.
Access path owner	O	If an access path is shared, one of the file members is considered the owner. The owner of the access path is charged with the storage used for the access path. If the member displayed is designated the owner, one or more file members are designated with an I for access path sharing.
SQL View	V	The SQL view or member is dependent on another SQL view.

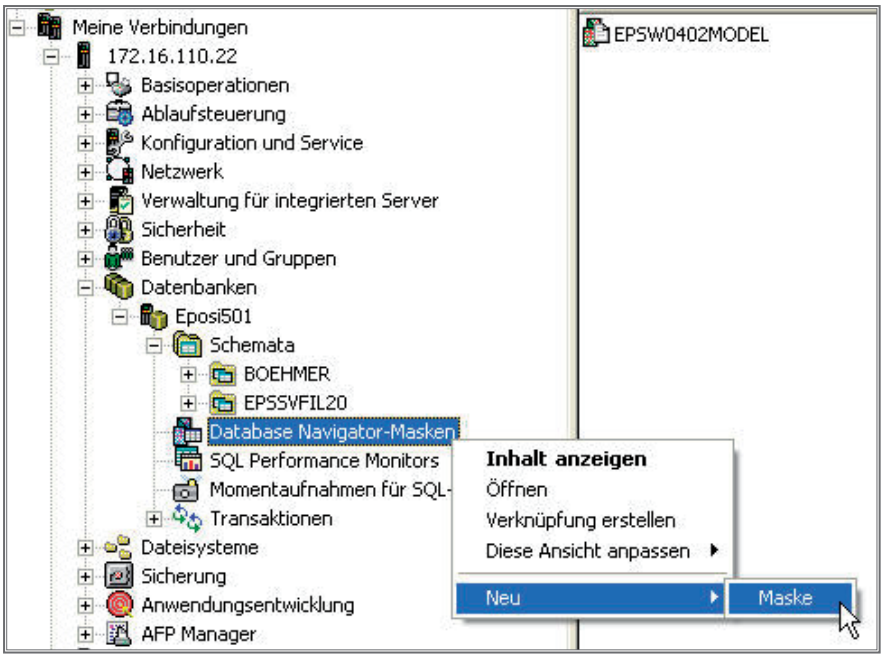
Kodierungen der Relationen

3.5.9.1

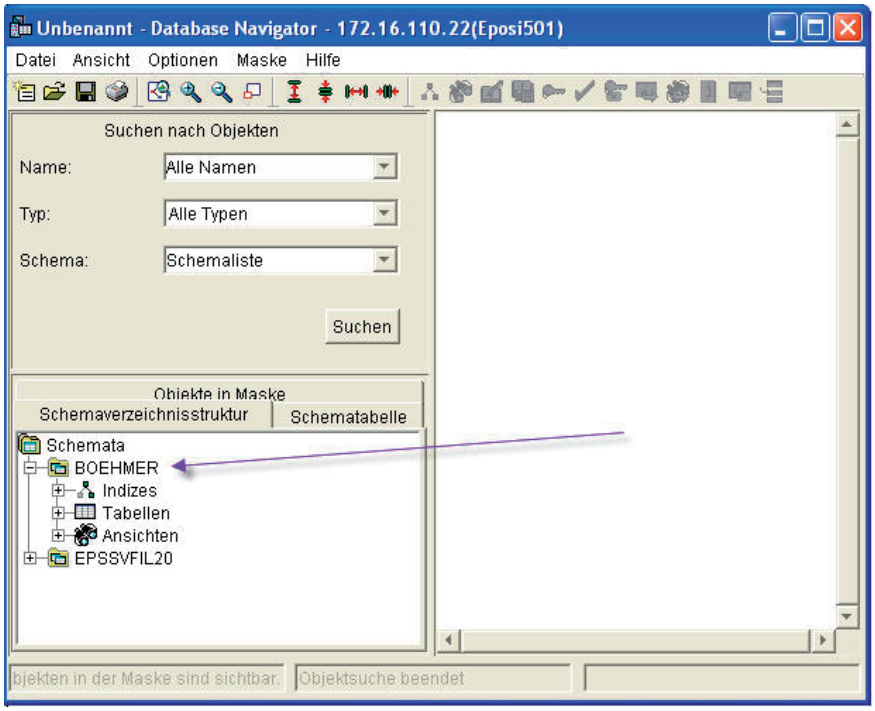
Seite 6

Datenbank-Navigator-Masken einsetzen

Alternativ zur Verwendung des Befehls DSPDBR unterstützt Sie der iSeries Navigator bei der Visualisierung der Dateirelationen. Den aufzusuchenden Eintrag im Navigator habe ich Ihnen bereits zu Beginn dieses Kapitels gezeigt. Über diesen Punkt werde ich nun eine neue Datenbankmaske erstellen.



Neue Maske erzeugen

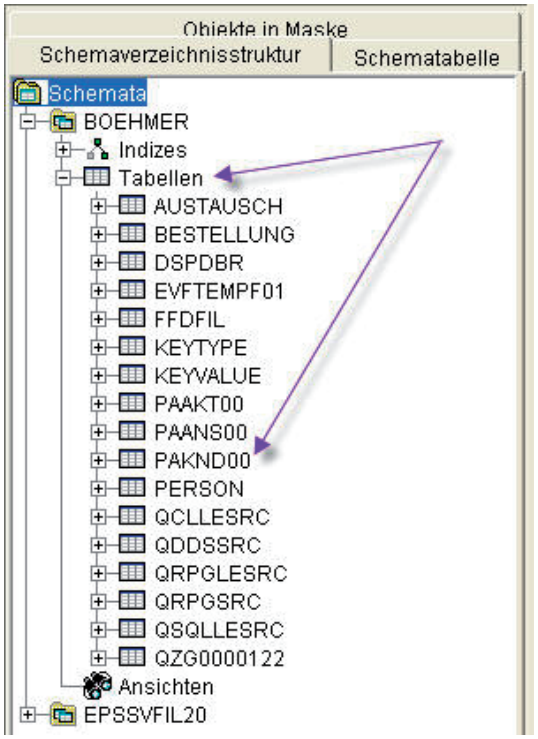


Ansicht nach Erstellung der Maske

3.5.9.2
Seite 2

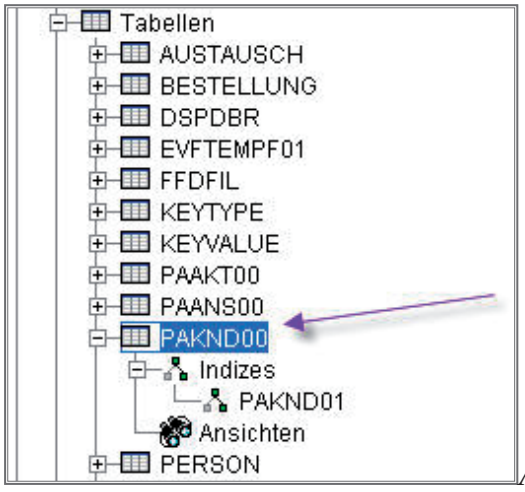
Im linken, unteren Teil der neuen Maske spiegelt sich das Bild der Schemaliste wieder. Die verschiedenen Unterpunkte, Indizes, Tabellen und Ansichten zeigen nach einer Expansion die einzelnen Objekte an.

Ich möchte die Abbildungen nicht überfrachten und öffne nur den Eintrag „Tabellen"; daraus zeige ich den Ausschnitt der Maske im betreffenden Bereich. Sie können das mit Hilfe Ihres Navigators entsprechend nachvollziehen.



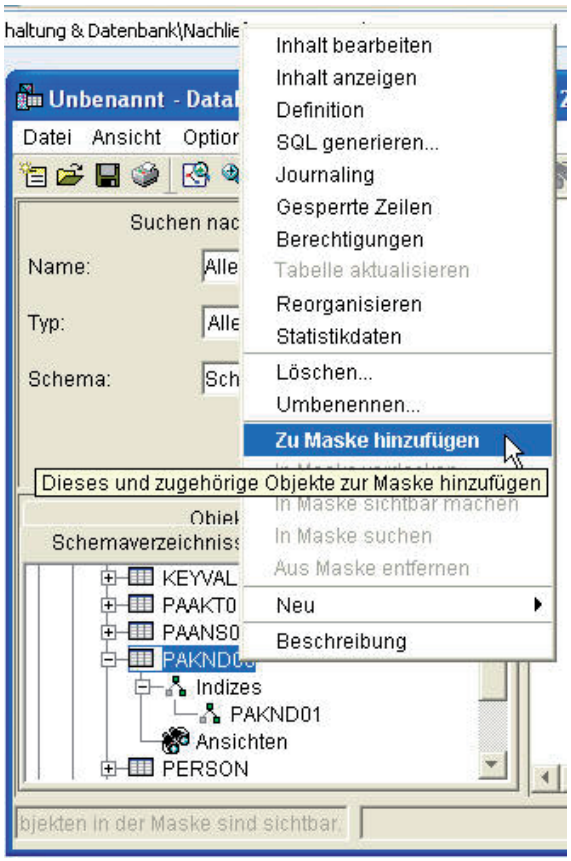
Ausschnitt der Navigator-Maske

Nach einem Klick auf das Plus-Zeichen (+) vor der Tabelle PAKND00 werden die zugeordneten Objekte aufgelistet. Verschaffen Sie sich einen Überblick über die Situation in Ihrer Datenbasis! Sie erhalten schon an dieser Stelle ähnliche Informationen wie zuvor mit dem DSPDBR-Befehl.



Ansicht der zugeordneten Objekte

Allerdings liegt jetzt noch nicht die grafische Darstellung vor, von der ich gesprochen hatte. Erst muss das Objekt, das interessiert, zur Maske hinzugefügt werden. Mit Hilfe eines Rechtsklicks auf die Tabelle „PAKND00“ und mit dem darauf erscheinenden Kontextmenü können Sie das erreichen.

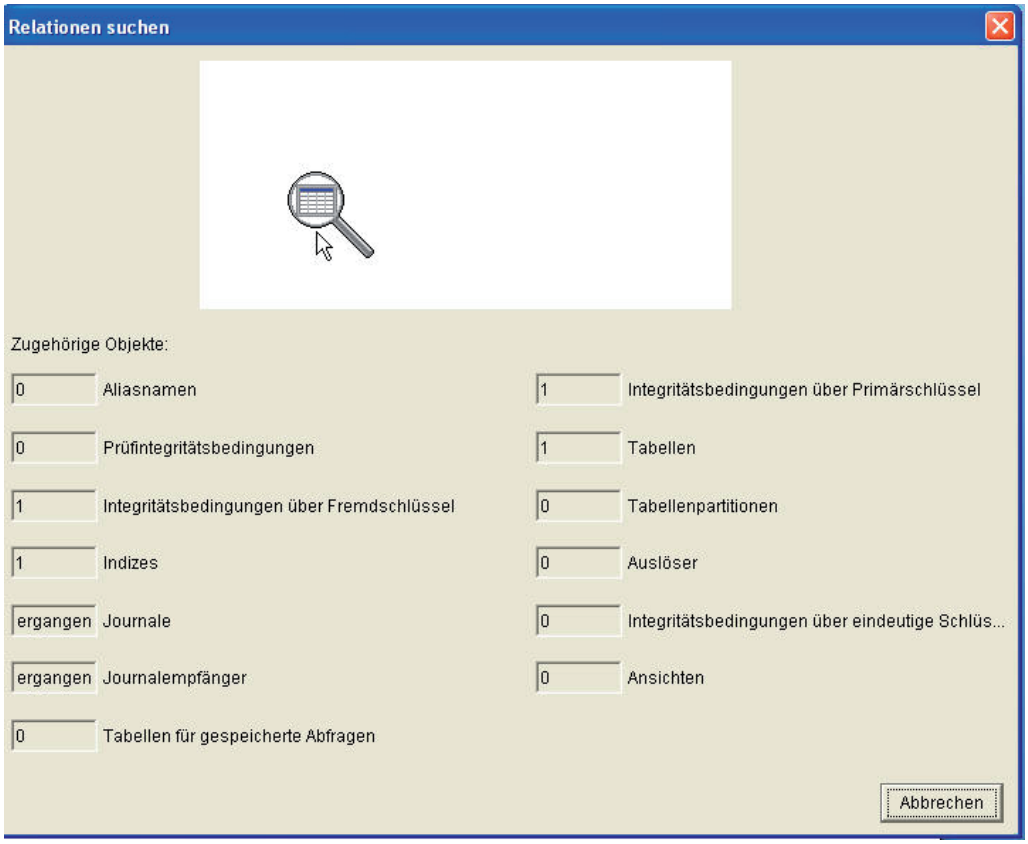


Objekt zur Maske hinzufügen

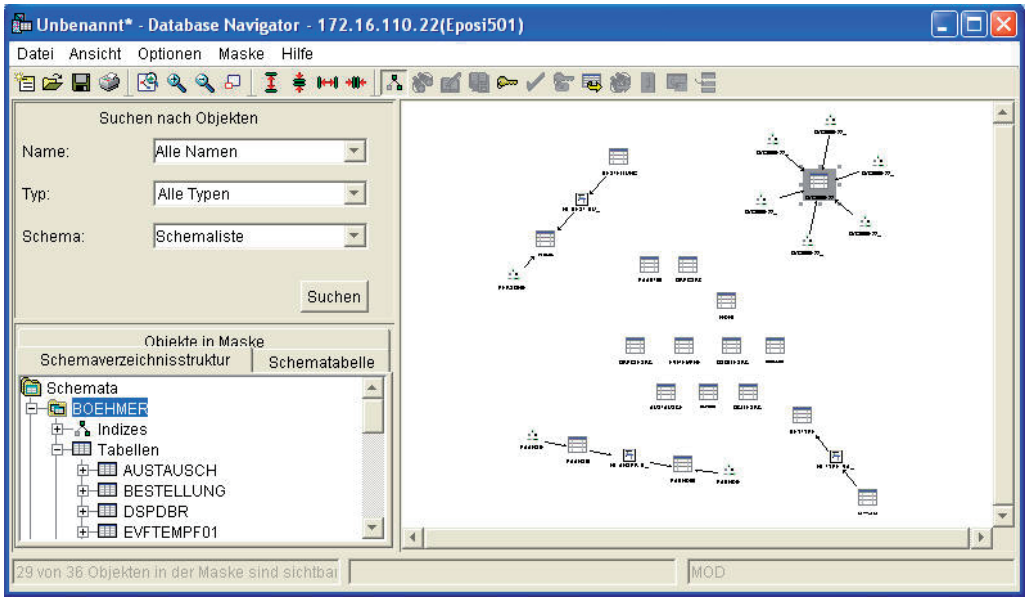
Das System wird nun die Datenbank auf Abhängigkeiten untersuchen; anschließend richtet es im rechten Bereich der Datenbankmaske eine grafische Darstellung ein. Wenn Ihre Datenbasis eine hohe Komplexität aufweist oder wenn Sie eine ganze Bibliothek zur Maske hinzufügen, dann kann etwas Zeit bis zur Fertigstellung der Anzeige vergehen.

In der folgenden Abbildung wird der Scan-Vorgang gezeigt. Ich habe die gesamte Bibliothek BOEHMER übernommen, damit ich Ihnen die Vielfalt der entstandenen Grafik verdeutlichen kann.

3.5.9.2
Seite 4

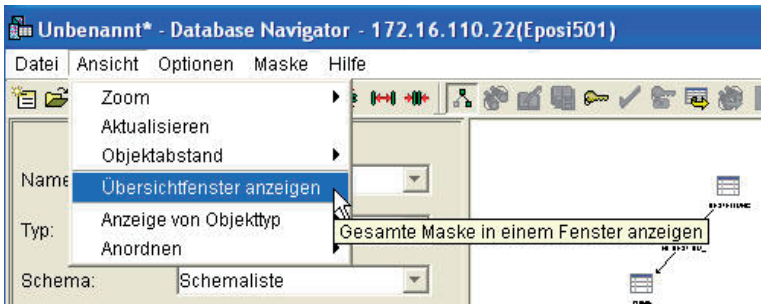


Scanvorgang



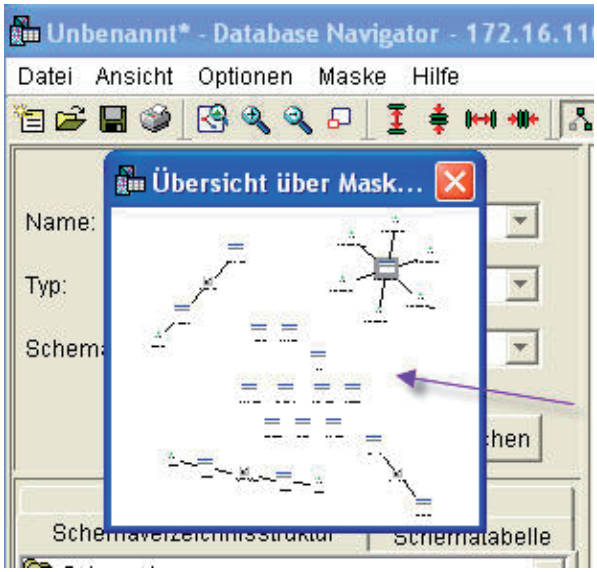
Die fertiggestellte Grafik

Je nach Anzahl der darzustellenden Objekte kann die Darstellung etwas klein ausfallen. Sie können mit Hilfe eines Übersichtfensters leicht einzelne Bereiche der Grafik ansteuern. Dazu öffnen Sie das Menü „Ansicht“ und klicken auf die Option „Übersichtfenster anzeigen“.



Übersichtfenster anzeigen

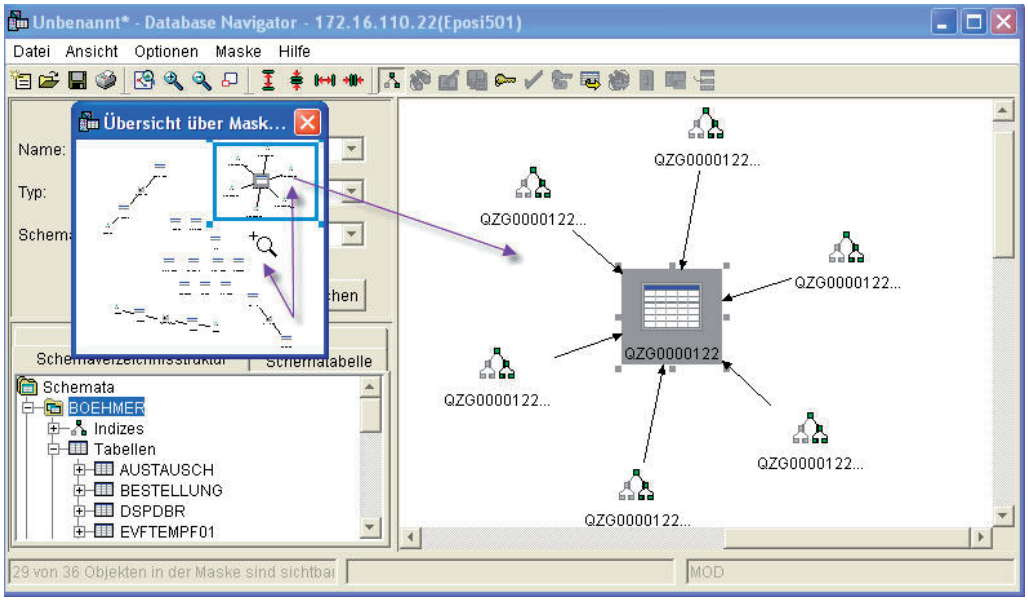
In einer separaten Anzeige erscheint dann folgendes Fenster:



Separate Anzeige der Übersicht

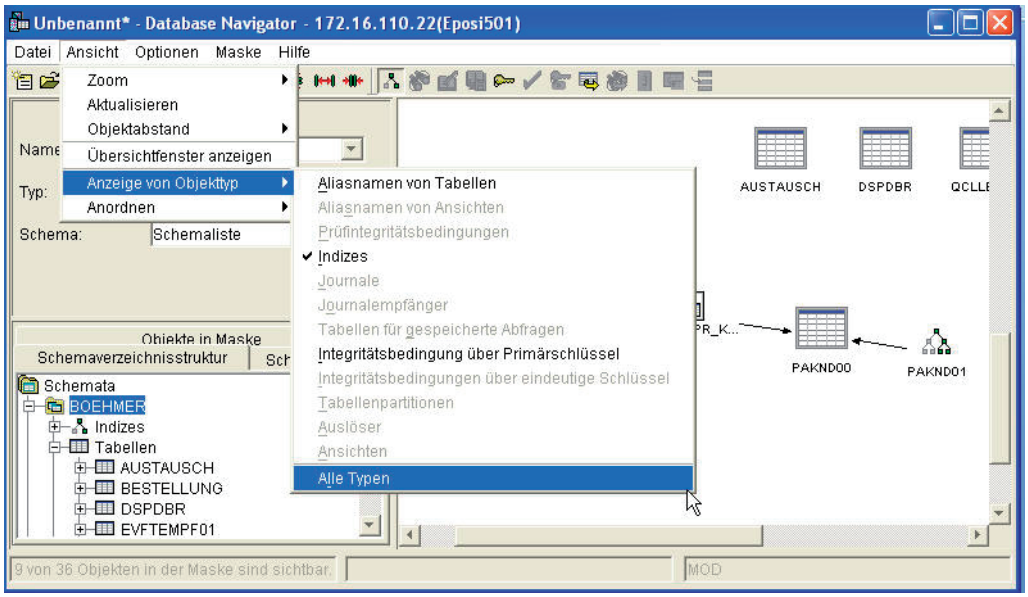
Bewegen Sie nun die Maus im Bereich dieses Fensters. Der Mauszeiger wird sich in eine kleine Lupe verwandeln; damit kann ein rechteckiger Bereich in der Übersicht aufgezogen werden. Die Inhalte des Bereichs werden im rechten Teil synchronisiert in die Datenbank-Maske eingepasst. So sind Sie in der Lage, recht komfortabel in den Diagrammen und an interessanten Positionen zu navigieren.

3.5.9.2
Seite 6



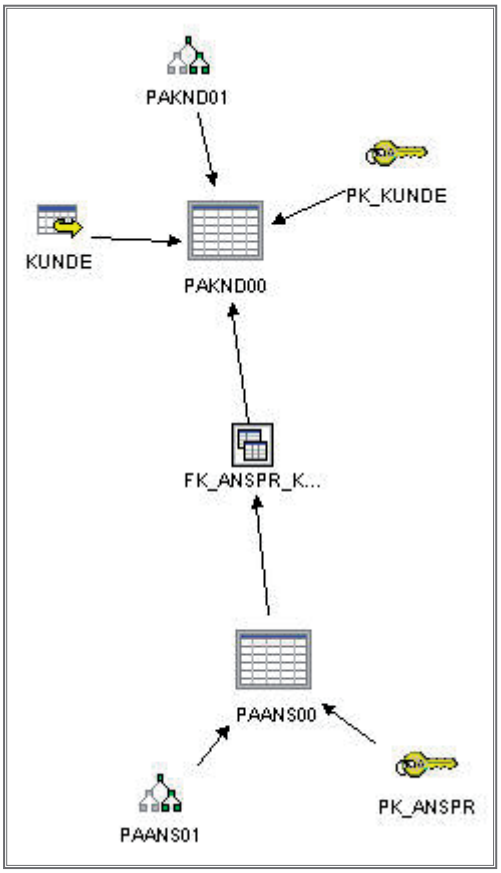
Navigation und Ansicht mit der Lupenfunktion

Wenn Sie Einfluss auf die Art der angezeigten Objekte nehmen möchten, dann öffnen Sie wieder das Menu „Ansicht“ und bewegen anschließend den Mauszeiger auf den Unterpunkt „Anzeige von Objekttyp“.



Anzeige von Objekttyp

Ich wähle „Alle Typen“ aus und erhalte eine neue Grafik, die als Ausschnitt in der nächsten Abbildung zu sehen ist.



Ansicht der neuen Grafik

Hinzugekommen sind Informationen von Aliasnamen und Primärschlüsseln. Wenn es vom Anwender gewünscht wird, ist die Darstellung sehr umfassend. Man kann sich auch nur die schmal gehaltenen Übersichtsinformationen anzeigen lassen.

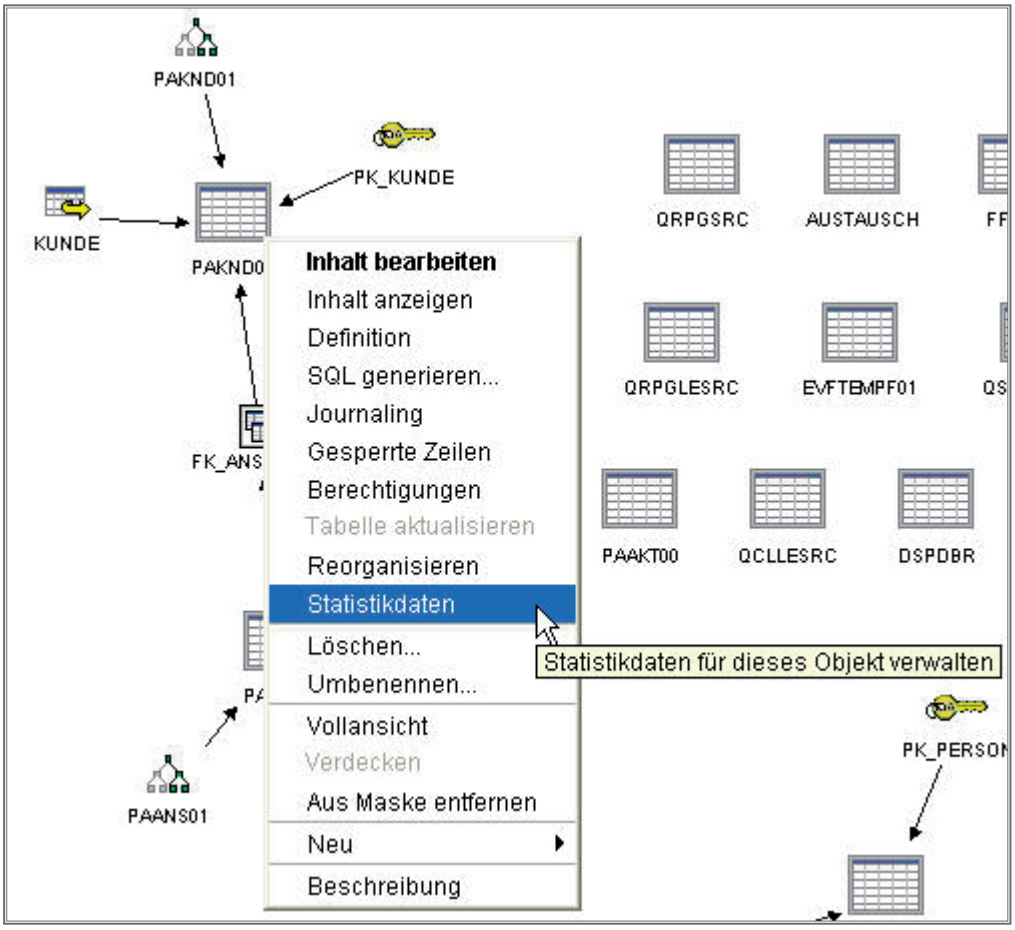
Wenn Sie den Mauszeiger auf eines der Symbole setzen, dann wird eine Art Tool-Tipptext aufgeklappt, der eine nähere Beschreibung des Objekts beinhaltet.

Ansicht der Objektbeschreibung

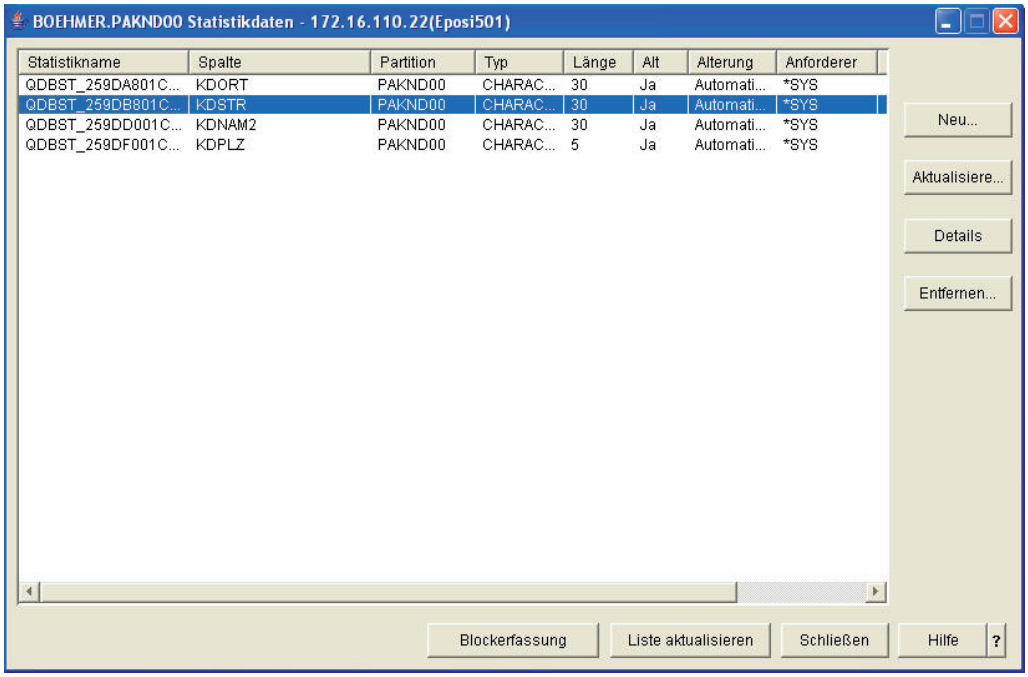
3.5.9.2
Seite 8

Hier habe ich die referentielle Integritätsbedingung „FK_ANSPR_KUNDE“ selektiert, die sich in der Mitte der Abbildung „Ansicht der neuen Grafik befindet.

Wenn Sie mit einem Klick der rechten Maustaste das Kontextmenu der gewählten Komponente aufrufen, werden Sie viele Optionen zur direkten Verwaltung finden – beispielsweise die Verwaltung der Statistikdaten einer Tabelle.



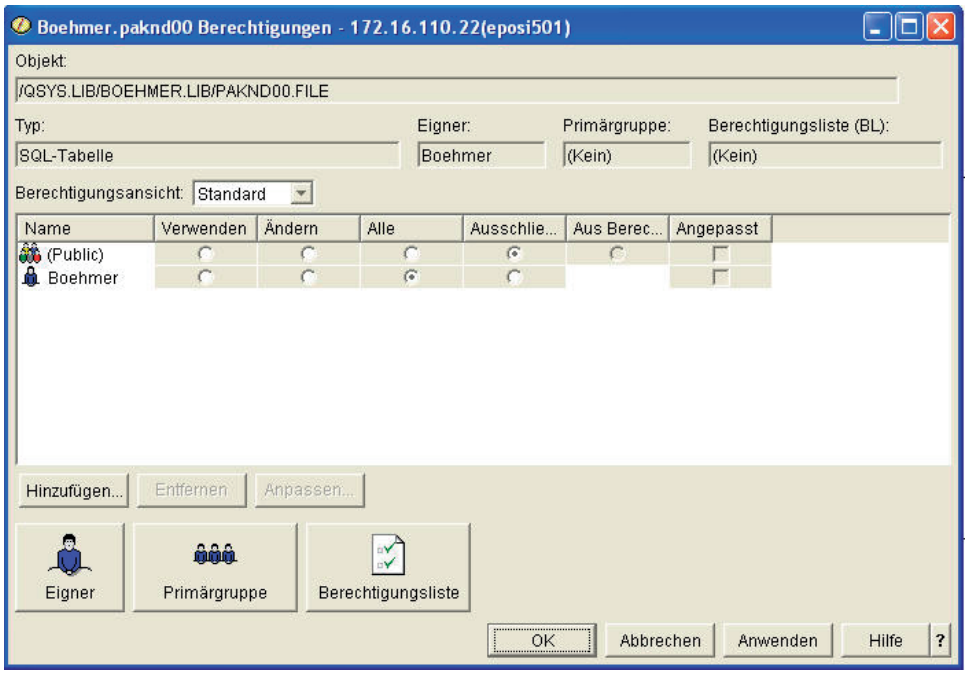
Optionen zur direkten Verwaltung



Statistikinformationen zu PAKND00

Auf eine weitere Erläuterung der Statistikdaten verzichte ich, da diese Dinge bereits im vorangegangenen Kapitel besprochen worden sind.

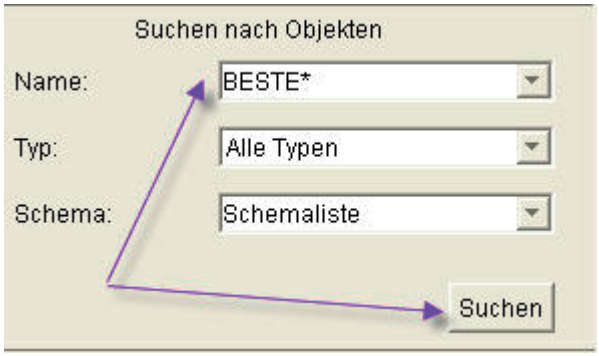
Ich werde später noch deutlich auf die Sicherheit im Zusammenhang mit der Datenbank-Verwaltung eingehen. Damit Sie im Vorfeld dazu schon einen ersten Eindruck gewinnen können, schauen Sie sich den nachfolgenden Punkt „Berechtigungen“ einmal an.



Einstieg in die Vergabe von Berechtigungen

Im oberen linken Bereich der Datenbank-Maske wird eine Suchfunktion bereitgestellt. Mit dieser Funktion ist eine gezielte Suche nach Objekten möglich.

Zunächst suchen wir generisch nach Objekten mit dem Namen „BESTE*“; wir berücksichtigen dabei alle Objekttypen und lassen die Suche in allen Bibliotheken der Schemaliste durchführen.

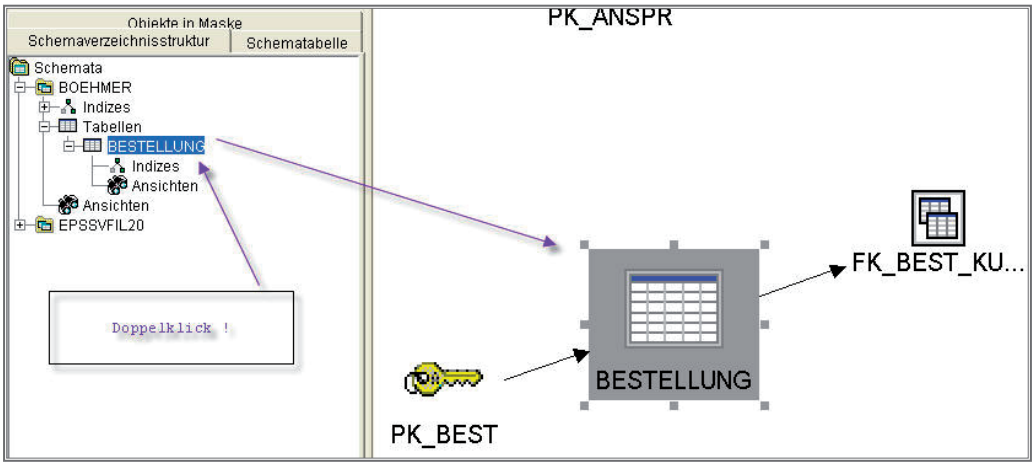


Generische Suche

Das Ergebnis der Filterung sieht wie folgt aus:

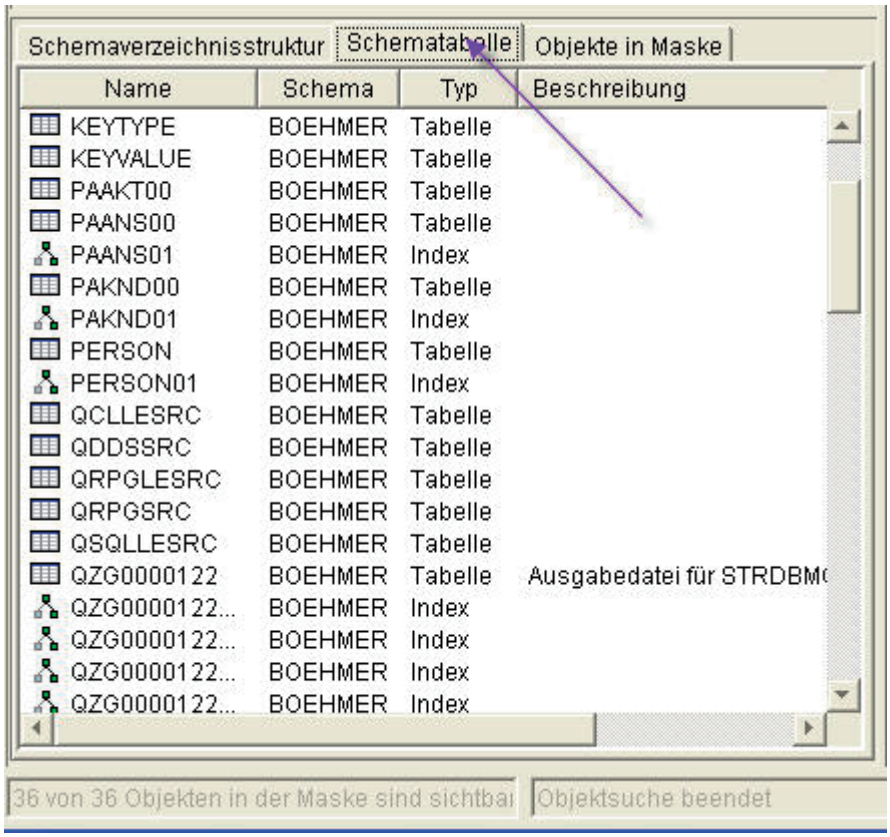
Resultat der Suche

Wenn man Objekte in einem sehr umfangreichen Diagramm auffinden möchte, kann das mit dem Übersichtfenster oder mit dem reinen Suchen in der Grafik sehr aufwändig und zeitraubend sein. Aber mit einem einfachen Doppelklick auf das Objekt erfolgt sofort die Selektion – auch im größten Diagramm!



Auffinden eines Objekts

Weitere Auflistungsvarianten befinden sich im unteren linken Bereich. Wenn man den Reiter „Schematabelle“ anklickt, werden die Komponenten der Maske in einer Liste dargestellt.

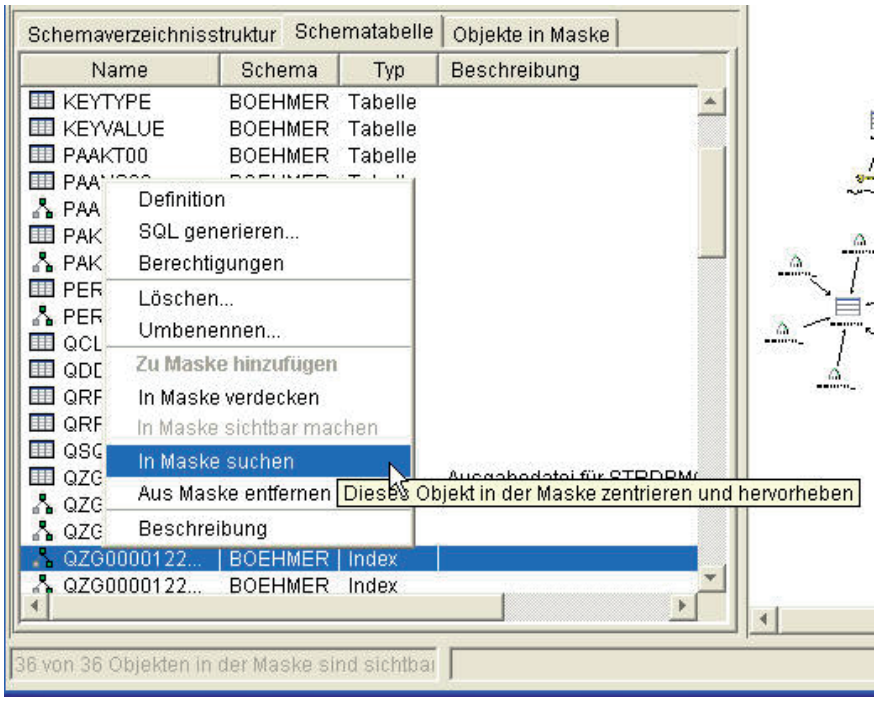


Komponenten der Maske

Sie können die Liste sortieren, indem Sie auf einen beliebigen Spaltenkopf klicken.

3.5.9.2
Seite 12

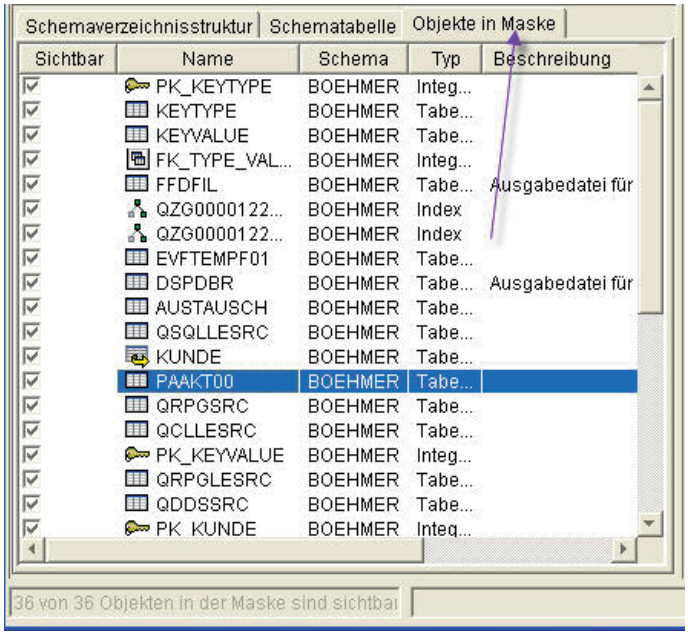
Wie man so schön sagt, führen viele Wege nach Rom, in unserem Fall führen die Wege zu den mannigfaltigen Aktionsmöglichkeiten mit Datenbank-Objekten. Setzen Sie auch hier intensiv die rechte Maustaste ein!



Weitere Aktionsmöglichkeiten

Die angezeigten Möglichkeiten sind selbsterklärend; probieren Sie sie einfach mal aus.

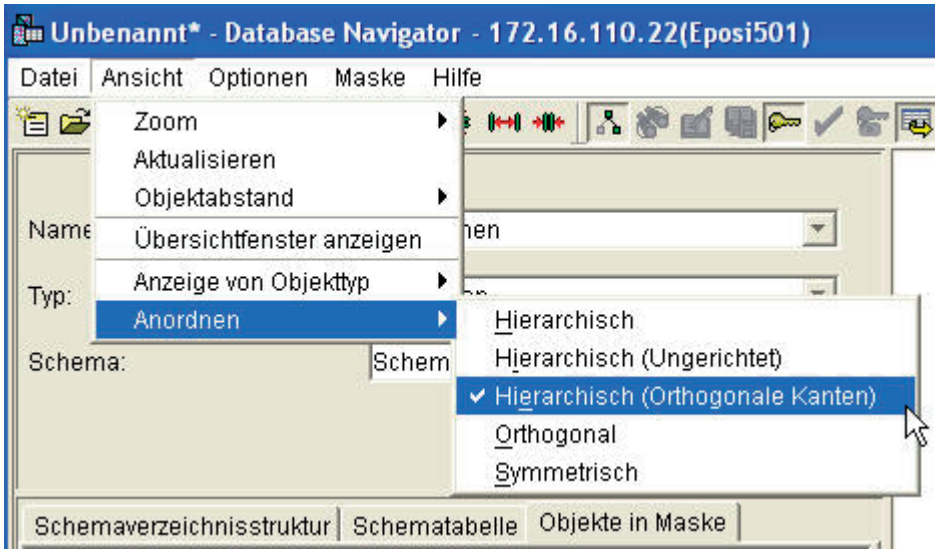
Zuletzt möchte ich noch einen Blick in den Reiter „Objekte in Maske“ werfen, der Ihnen die in der Maske befindlichen Komponenten anzeigt.



Objekte in Maske

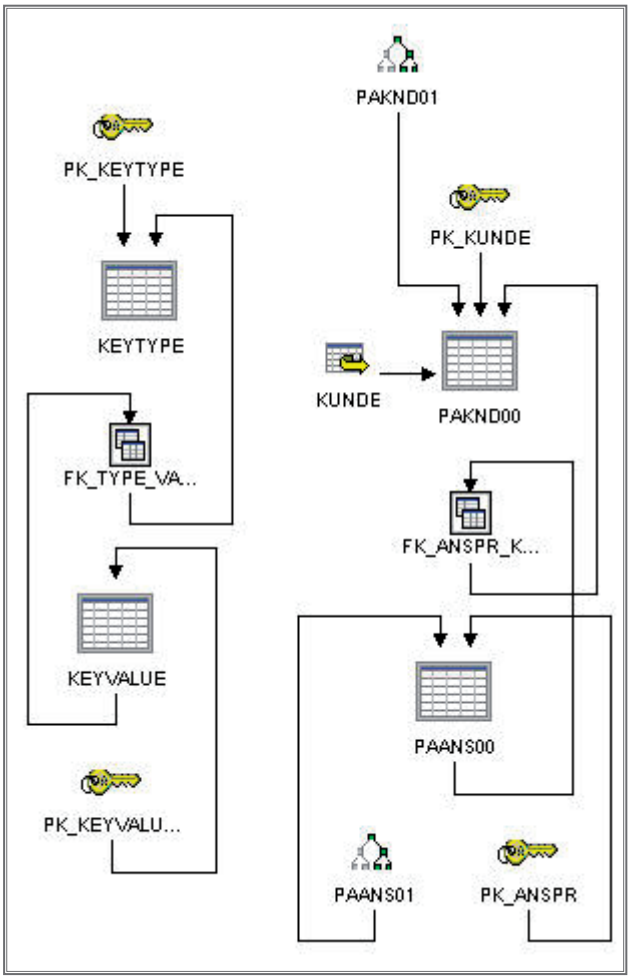
Zum Sortieren klicken Sie auf die Spaltenköpfe. Einzelne Objekte können im Diagramm ausgeblendet werden, wenn Sie mit den kleinen Auswahlboxen in der linken Spalte arbeiten.

Sollten Sie manchmal eine Reaktion vermissen, klicken Sie einfach auf den Spaltenkopf „Sichtbar“. In den meisten Fällen hilft das!

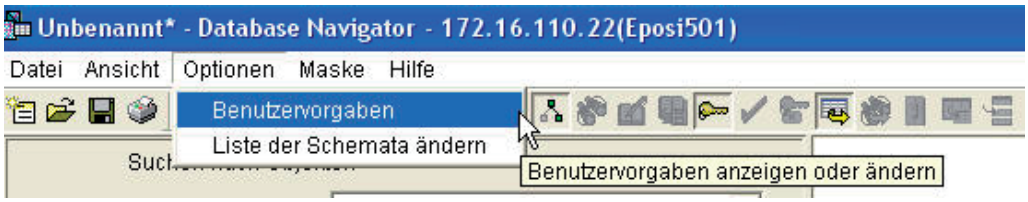


Optionen des Reiters „Ansicht“

Die Anordnung der Objekte wirkt zum Teil ein wenig unmotiviert. Dem können Sie abhelfen, indem Sie über den oben abgebildeten Menüpfad eine der angebotenen Voreinstellungen wählen. Bei der von mir selektierten Auswahl wird das Diagramm in der folgenden Weise umsortiert.

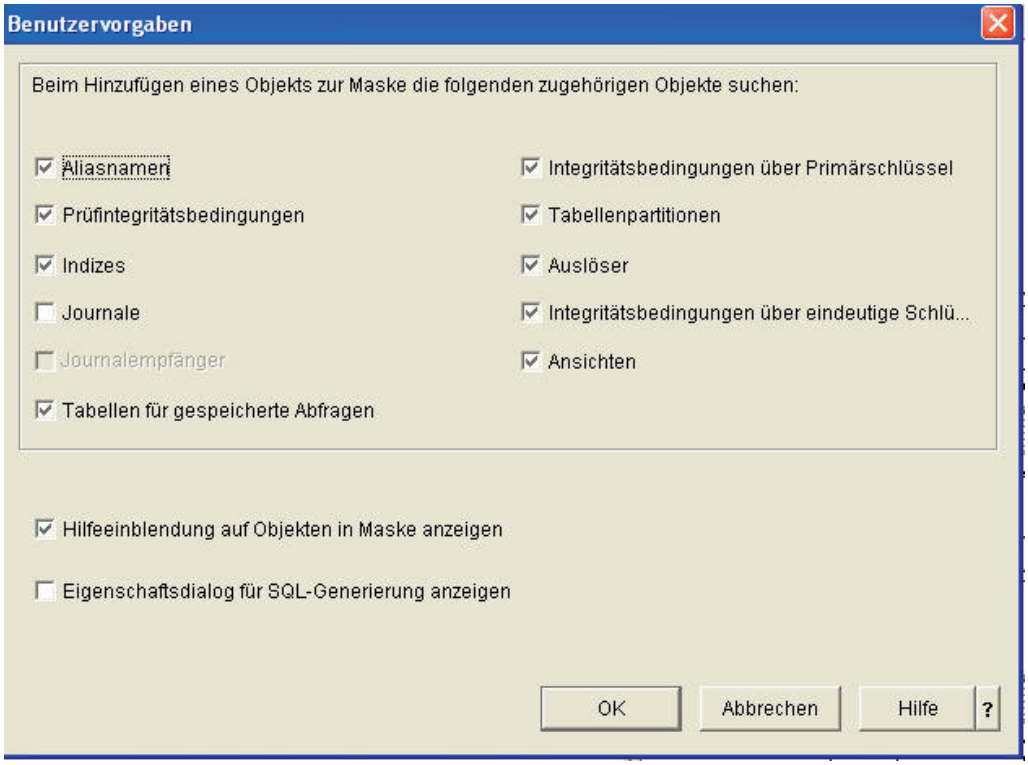


Hierarchische Anordnung mit senkrechten Kanten



Option „Benutzervorgaben“

Zum Abschluss des Kapitels zeige ich Ihnen noch eine Möglichkeit auf, wie Sie Einfluss auf die zu suchenden Objekte bei der Neuerstellung einer Datenbank-Maske nehmen können. Selektieren Sie dazu die Option „Benutzervorgaben“ im Menü „Optionen“.



Auswahl der Komponenten

In dieser sehr einfachen Maske wählen Sie einfach die für Ihre Belange in Frage kommenden Komponenten aus.

Noch ein kurzer Ausblick auf das folgende Kapitel: Dort werden wir uns mit der Sicherheit Ihrer Datenbank befassen, nachdem wir einen Blick auf die SQL-Abfrage der Datenbankkataloge geworfen haben. Das ist dann die dritte Variante, um Abhängigkeiten zwischen Dateien zu ermitteln.

3.5.9.2
Seite 16

Sicherheit und Berechtigung

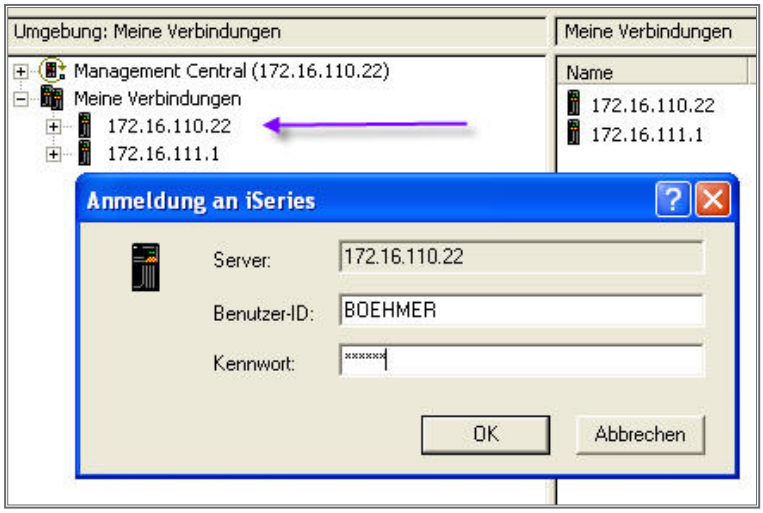
Die Absicherung Ihrer Daten gegen unberechtigte Zugriffe ist Thema der folgenden Abschnitte. Sie können mit verschiedenen Methoden Berechtigungen für Ihre Dateien vergeben. Derzeit kommt dafür in erster Linie der iSeries Navigator zum Einsatz. Sie können auch den klassischen Grant-Object-Authority-Befehl oder das SQL-Grant-Statement einsetzen. Ich werde in den folgenden Teilabschnitten die verschiedenen Varianten näher beschreiben und beginne mit dem iSeries Navigator.

14. Ergänzung 3/2008

3.5.10
Seite 2

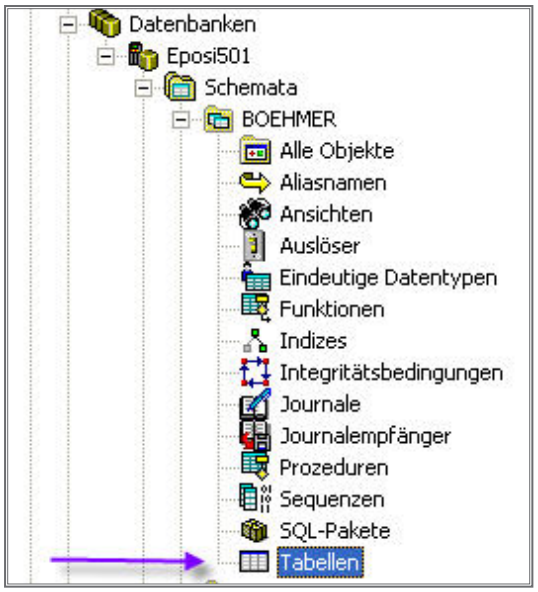
Berechtigungsvergabe mit dem iSeries Navigator

Möglicherweise benötigen bestimmte Benutzer oder Benutzergruppen spezifische Berechtigungen für Datenbankobjekte. Die Erteilung oder das Entziehen von Rechten für Profile und/oder Gruppen ist recht einfach über den iSeries Navigator möglich. Starten Sie dazu das Tool und öffnen Sie das System, mit dem Sie arbeiten möchten.



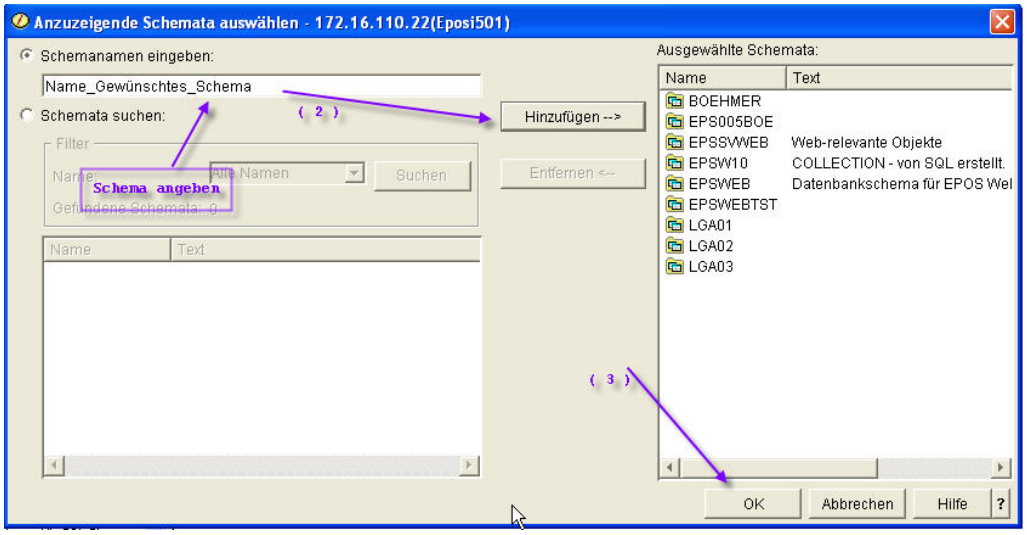
Anmeldung an iSeries

Navigieren Sie zum Schema, in dem sich das zu bearbeitende Objekt befindet. Sie finden es, indem Sie den Unterpunkt „Alle Objekte“ oder die spezifischen Filterkriterien anklicken.

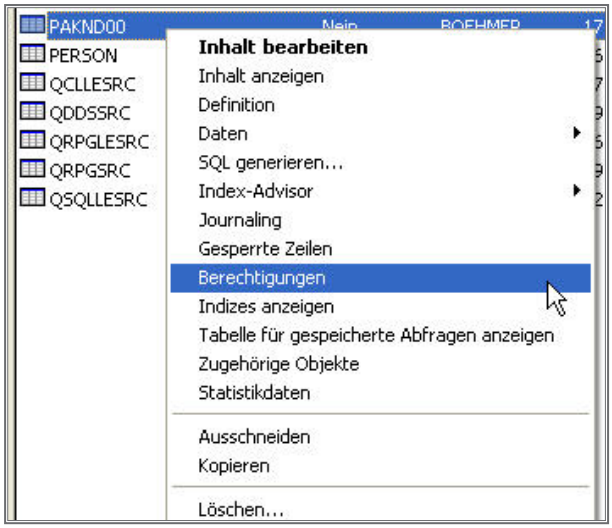


Möglicherweise ist das gewünschte Schema nicht im Ordner „Schemata“ aufgeführt. In diesem Fall können Sie es leicht hinzufügen. Markieren Sie „Schemata“ mit der rechten Maustaste und wählen Sie die in der Abbildung markierte Menüoption „Anzuzeigende Schemata auswählen“.

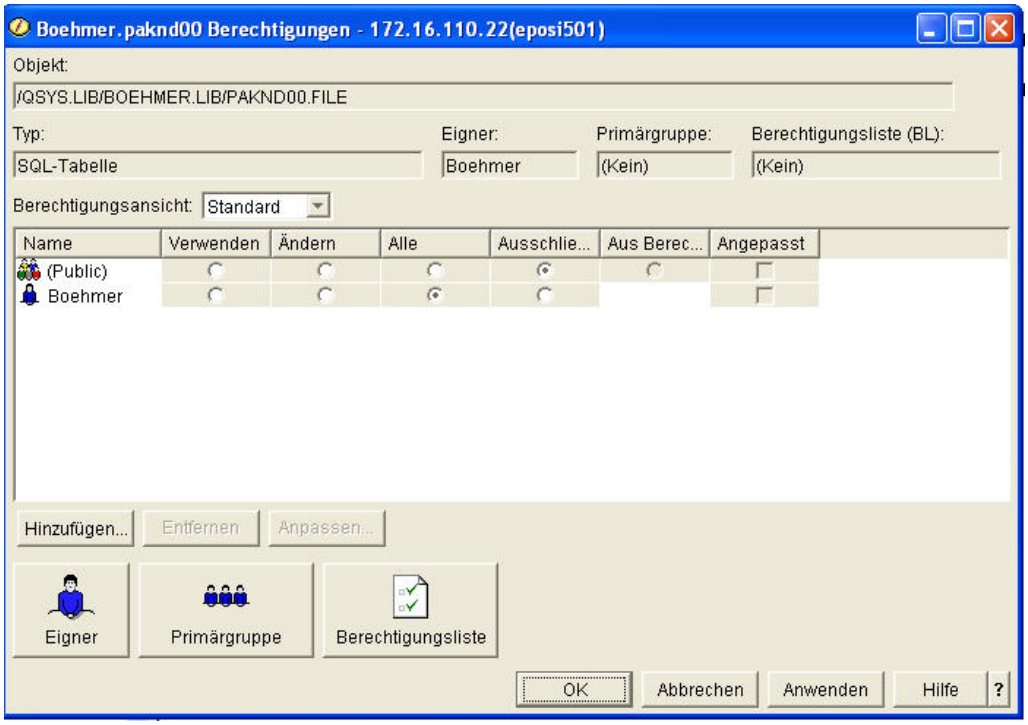
3.5.10.1
Seite 2



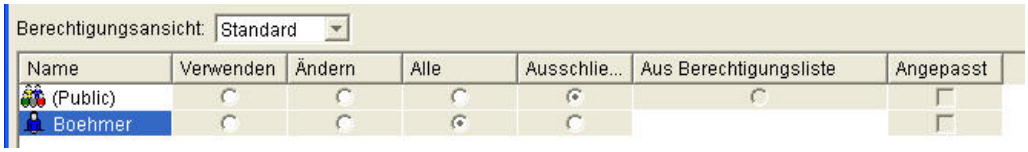
Im daraufhin angezeigten Formular tragen Sie den Namen des Schemas ein; Sie können aber auch den Radiobutton für die Schema-Suche selektieren. Mit der Schaltfläche „Hinzufügen“ nehmen Sie das Schema in den rechts dargestellten Listenteil auf. Mit „OK“ bestätigen Sie anschließend Ihre Wahl. Wenn Sie das Filterkriterium „Tabellen“ markiert haben, erhalten Sie eine Liste der vorhandenen Objekte im rechten Bereich. Dort rufen Sie mit der rechten Maustaste das Kontextmenü auf und wählen dann die Option „Berechtigungen“ aus.



Jetzt erscheint der Dialog zur Konfiguration der Berechtigungen des gewählten Objekts. Bitte denken Sie daran, dass nicht nur mit Tabellen gearbeitet werden kann, sondern auch mit Prozeduren und anderen Objekten in Ihrer Datenbank.



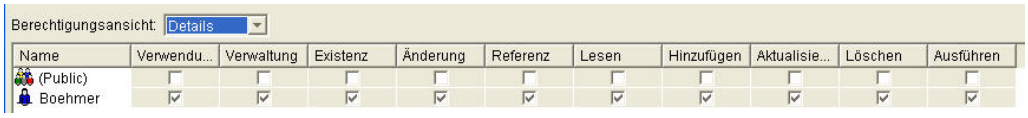
Im oberen (nicht änderbaren) Bereich sind Informationen über das betroffene Objekt abgebildet. Die Möglichkeiten im unteren Bereich werden wir nun besprechen.



Die Standardansicht der Berechtigungen stellt die Ihnen bekannte Objektberechtigung dar. In diesem Fall besitzt das Benutzerprofil „BOEHMER“ alle Berechtigungen für die Tabelle. Alle anderen Profile (PUBLIC) sind hier ausgeschlossen. Die Berechtigungen können einfach durch das Auswählen der entsprechenden Radiobuttons konfiguriert werden.

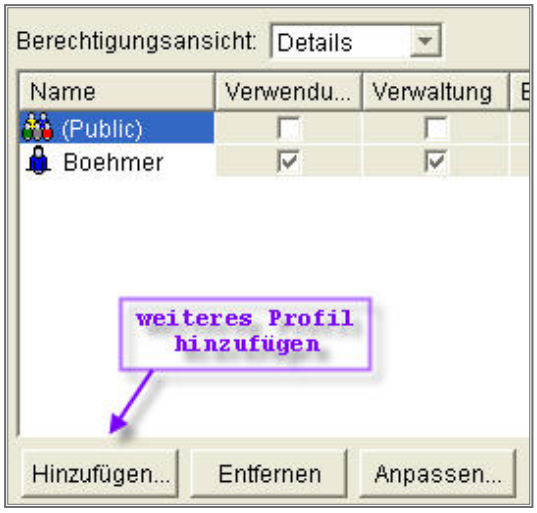
3.5.10.1
Seite 4

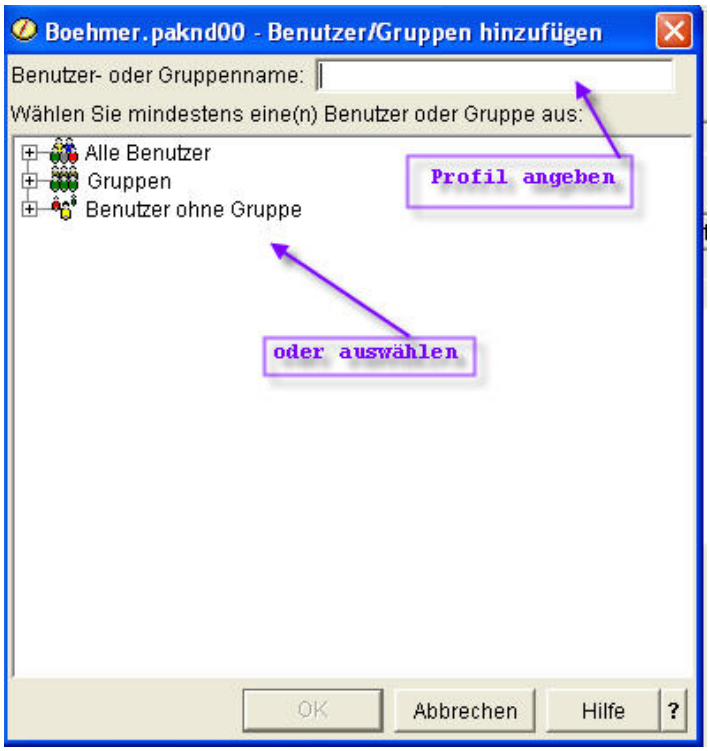
Die Combo-Box zur Berechtigungsansicht bietet mehrere Optionen. Ich wähle den Eintrag „Details“ aus; dabei verändert sich die Darstellung wie folgt:



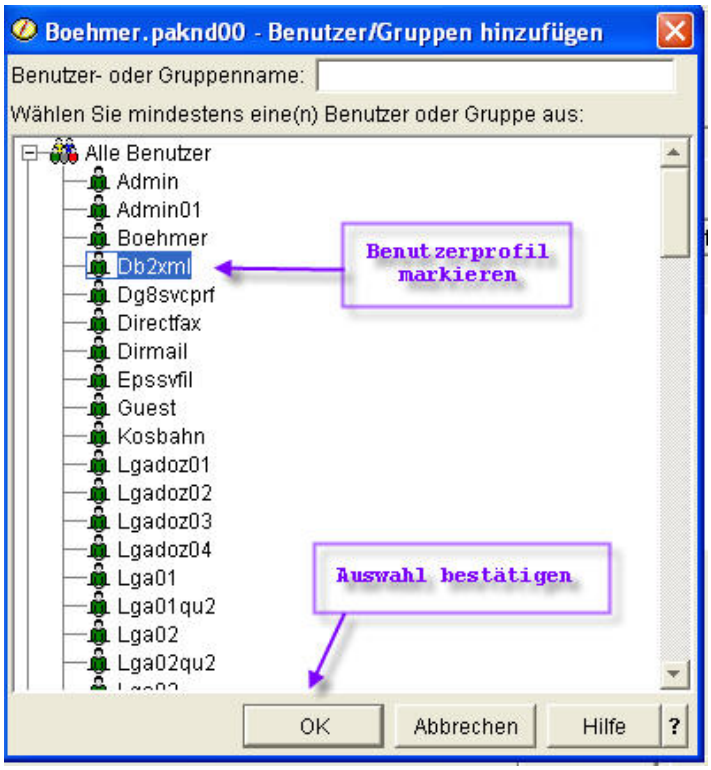
Diese Anzeige müsste Ihnen ebenfalls bekannt sein, es handelt sich hier um die Datenberechtigungen. In dieser Sicht können die Berechtigungen durch die Auswahl der verschiedenen Check-Boxen verändert werden.

Wenn ein weiteres Profil in die Berechtigungsverwaltung aufgenommen werden soll, dann klicken Sie auf die Schaltfläche „Hinzufügen…“.

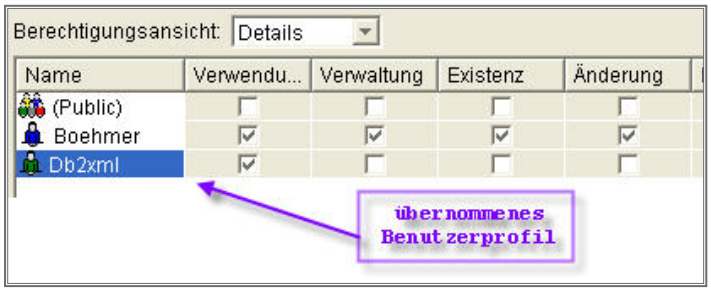




Sie können einfach das gewünschte Profil im oberen Eingabefeld eingeben; Sie können aber auch einen Listenpunkt im linken Bereich öffnen. Dort selektieren Sie einen Benutzer oder eine Benutzergruppe. Zur Demonstration öffne ich hier den Punkt „alle Benutzer“.

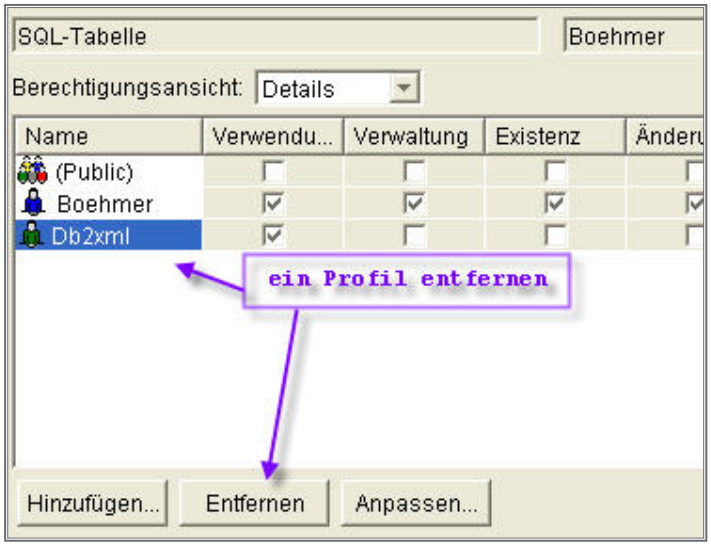


Das Benutzerprofil kann dann einfach mit „OK“ übernommen werden.

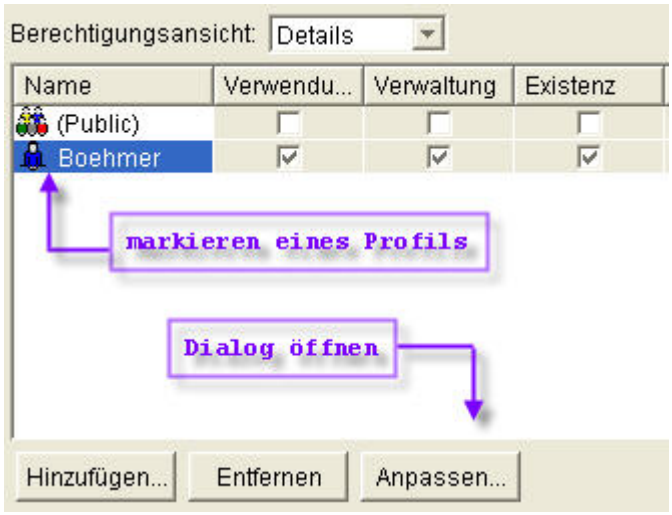


3.5.10.1
Seite 6

An dieser Stelle können Sie die Berechtigungen für das hinzugefügte Profil bearbeiten. Wenn Profile entfernt werden sollen, müssen sie lediglich markiert werden. Ein Klick auf „Entfernen“ nimmt das Benutzerprofil aus der Konfiguration heraus.



Sowohl in der Standard- als auch in der Detailansicht können Benutzerprofile angepasst werden. Auch hier wird ein Profil markiert und mit der Schaltfläche „Anpassen“ ein Dialog geöffnet.

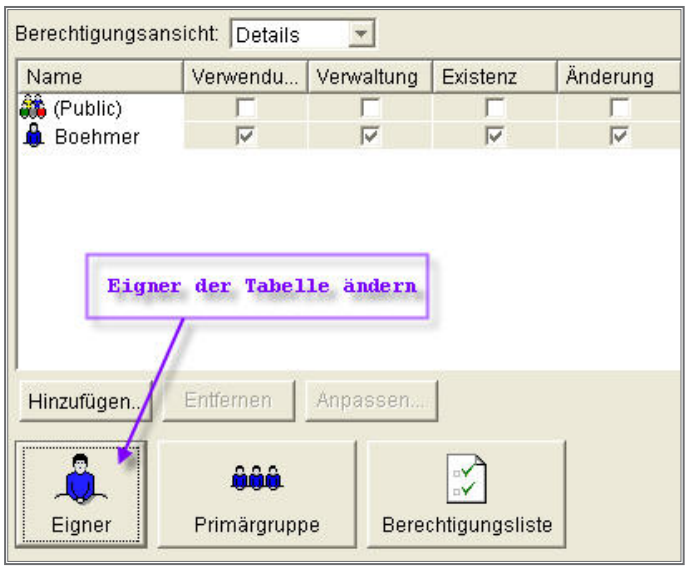


Die Einstellungen, die in diesem Dialog beim Öffnen vorgenommen worden sind, spiegeln die aktuellen Berechtigungen des Benutzers wieder – und zwar wie sie im Hauptdialog zu sehen sind. Die Radiobuttons im oberen Teil sind für die Übernahme vordefinierter Einstellungen vorgesehen. Wenn Sie einen Radiobutton selektieren, können Sie eine Synchronisierung der zugeordneten Eigenschaften im Bereich der angepassten Berechtigungen beobachten.

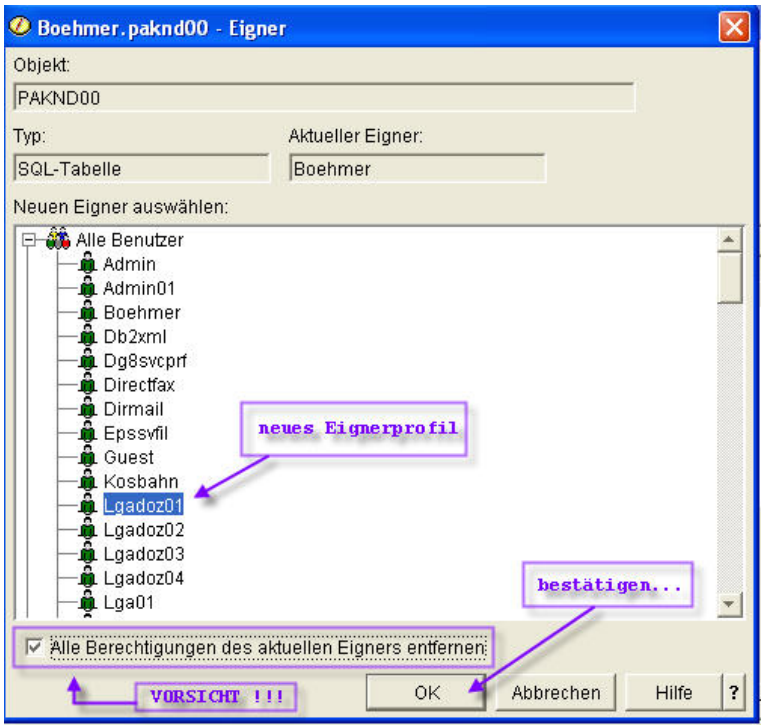
3.5.10.1
Seite 8

Sie können die angepassten Berechtigungen natürlich auch selbst vornehmen. Dazu selektieren Sie den Radiobutton „Angepasst“ und setzen anschließend die gewünschten Einstellungen im unteren Bereich.

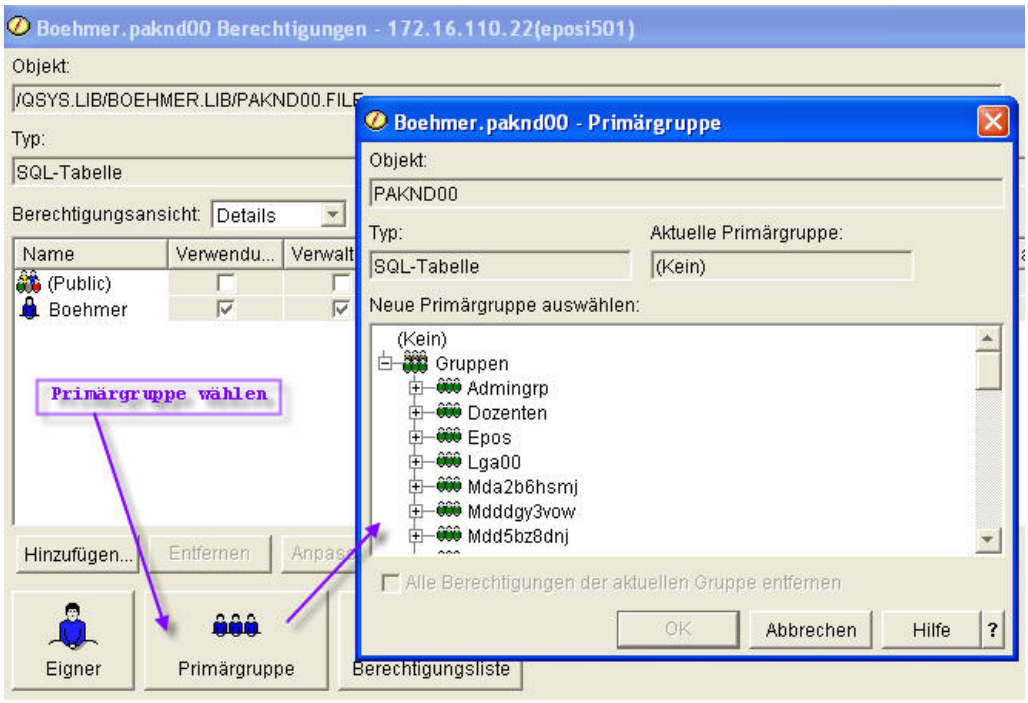
Soll der Besitzer der Tabelle geändert werden, ist auf den Button „Eigner“ zu klicken.



Im folgenden Dialog wählen Sie das neue Eigner- oder Gruppenprofil aus und bestätigen die Aktion dann mit „OK“.

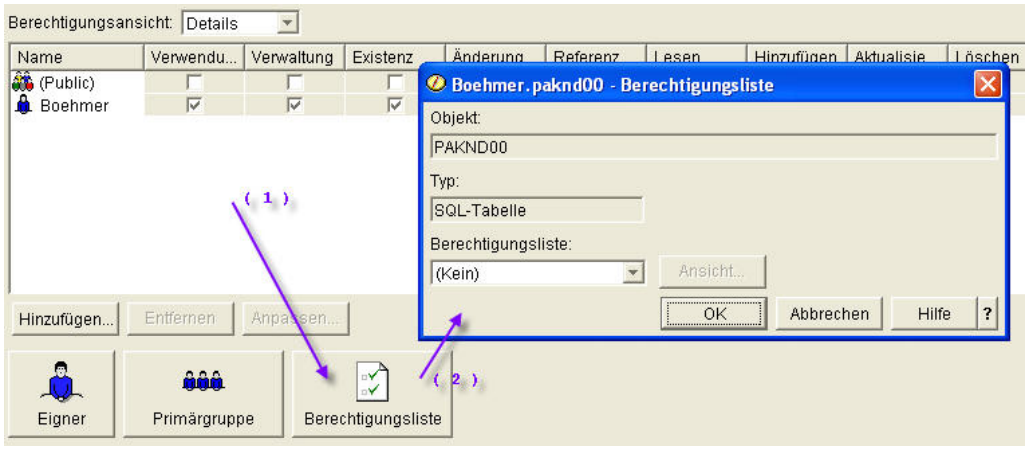


Bitte lassen Sie bei Entfernung der aktuellen Berechtigungen des aktuellen Eigners Vorsicht walten. Dieses Profil hat anschließend unter Umständen ***keine*** Möglichkeit mehr, auf das Objekt zuzugreifen.

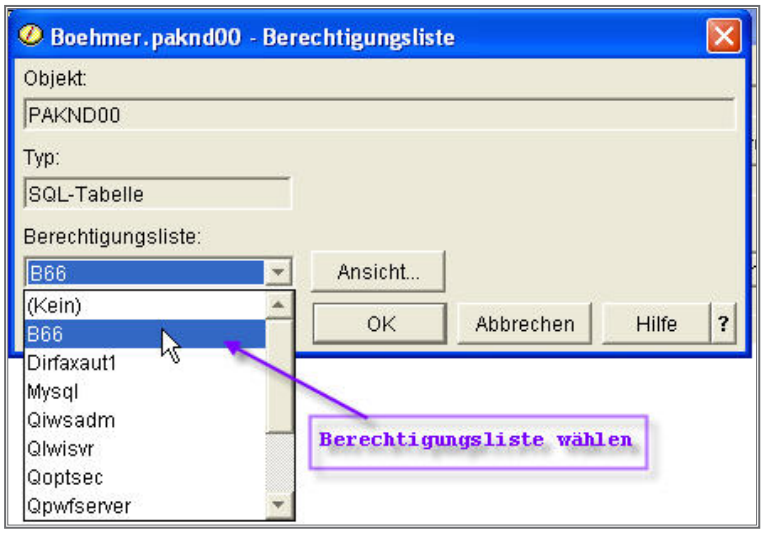


3.5.10.1
Seite 10

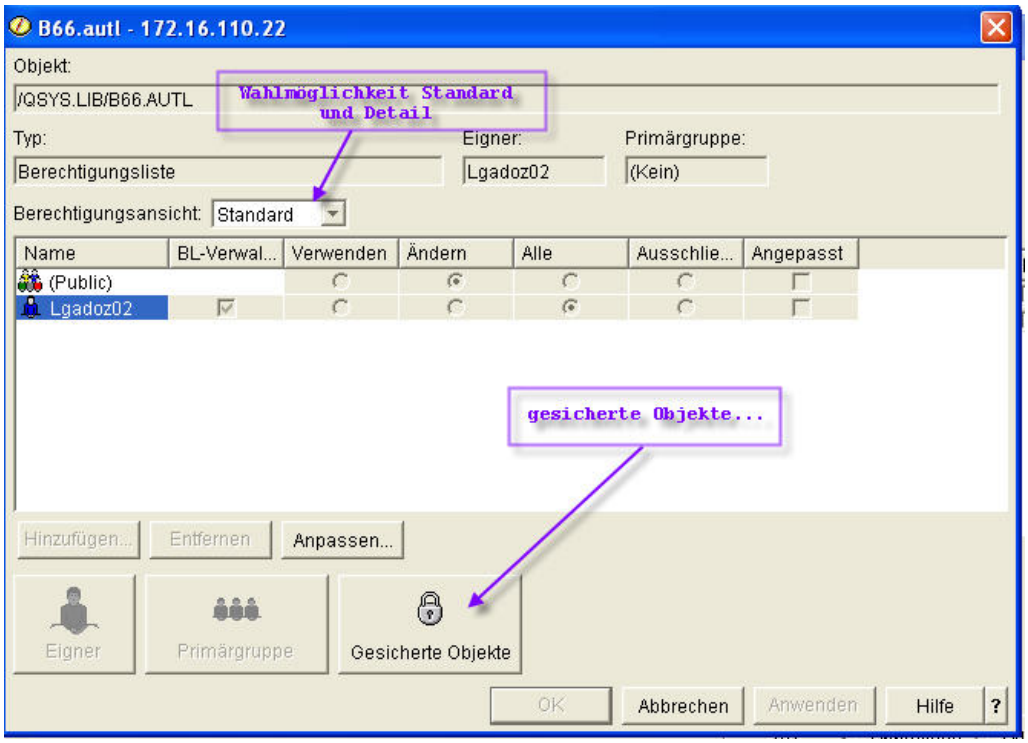
Genauso wie Sie die Eignerschaft bearbeiten können, so ist auch die Zuordnung eines Primärgruppenprofils herstellbar. Dazu öffnen Sie einfach über „Primärgruppe“ einen Dialog und wählen dort ein Gruppenprofil aus. Eine Absicherung des Datenbankobjekts ist auch über Berechtigungslisten möglich. Wenn Sie diese Art der Berechtigungszuteilung wählen, ist es möglich, über die Schaltfläche „Berechtigungsliste“ aus einem Fundus bestehender Listen auszuwählen.



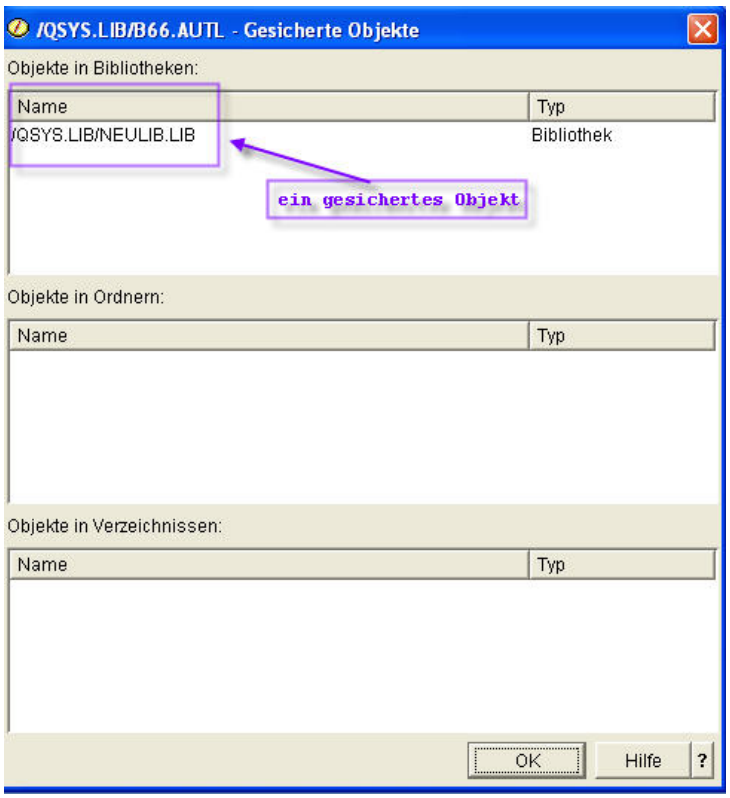
Über die Combo-Box „Berechtigungsliste“ ist nun eine vorhandene Liste wählbar.



Die Schaltfläche „Ansicht…“ zeigt einen Dialog, mit dem man über die Berechtigungsliste arbeiten kann.

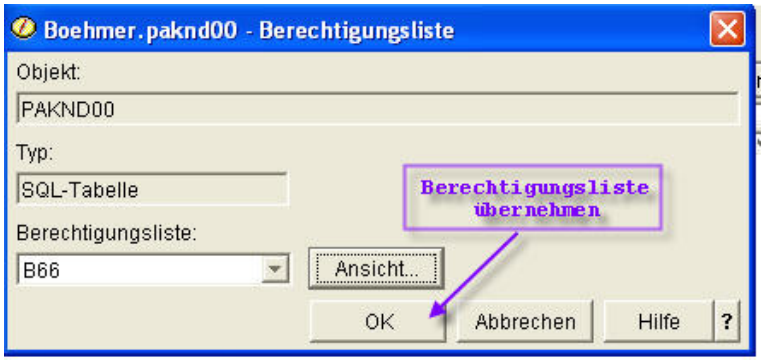


Im oberen Teil der Anzeige haben Sie die Auswahl zwischen Standard- und Detailansicht. Am interessantesten sind hier sicherlich die durch die Berechtigungsliste gesicherten Objekte.

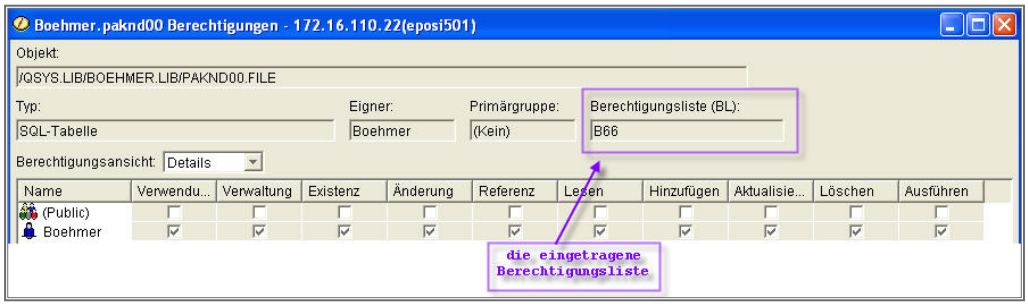


3.5.10.1
Seite 12

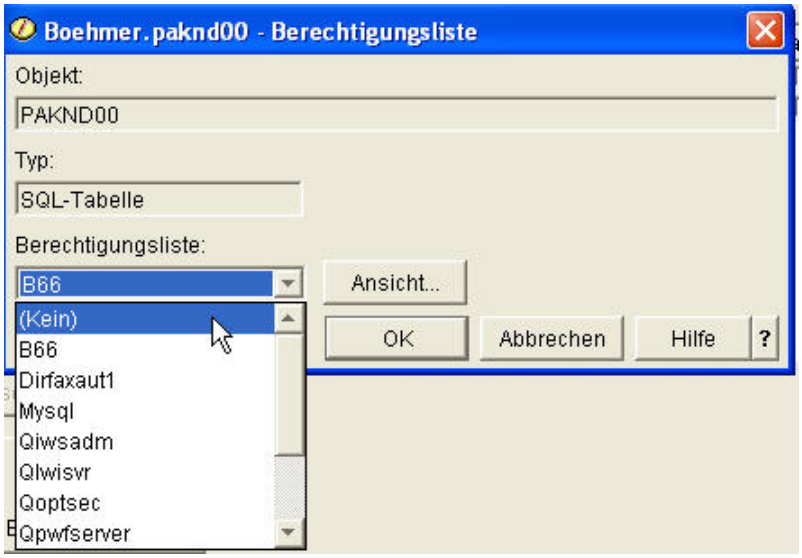
Wenn Sie die Berechtigungen bearbeiten, die in der Berechtigungsliste hinterlegt sind, sollten Sie einen Blick auf die bereits von ihr gesicherten Objekte werfen. Eine Änderung der Rechte hat Auswirkungen auf die Verwendbarkeit der betroffenen Datenbankobjekte. Sie können natürlich auch Ihre eigenen speziellen Berechtigungslisten erstellen. Klicken Sie auf „OK“, wenn Sie mit Ihren Änderungen fertig sind.

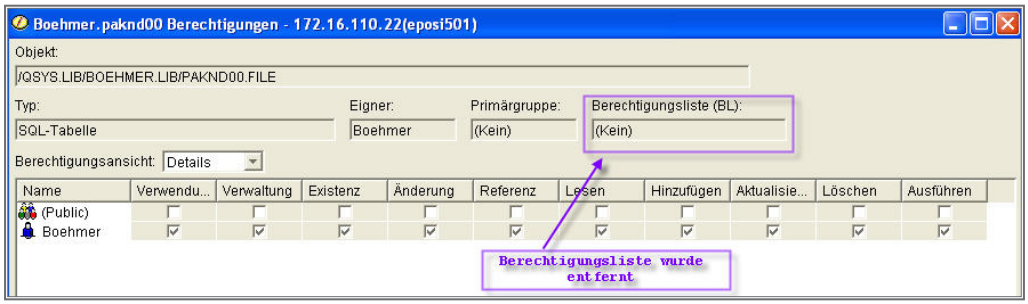


Sie erkennen am Hauptdialog, dass die Berechtigungsliste dem Datenbankobjekt zugeordnet wurde.



Wenn eine bestehende Berechtigungsliste entfernt werden soll, gehen Sie einfach in den Dialog für „Berechtigungslisten“ und wählen dort für die Berechtigungsliste „(Kein)“ aus.





Dem Hauptdialog können Sie entnehmen, dass keine Liste mehr zugeordnet ist.

Fast alle Optionen sind in diesem Dialog erläutert worden. Eine weitere Ansicht ist jedoch noch möglich. Schauen Sie noch einmal in die Combo-Box für „Berechtigungsansichten“. Dort ist die Einstellung „Spalten“ wählbar.

Nach dem Auswählen dieser Option zeigt sich folgendes Bild:

Dieser Dialog erlaubt Ihnen, die Berechtigungen sogar auf Spaltenebene dieser Tabelle vorzunehmen. Auch die Eignerschaft (natürlich nicht auf Spaltenebene) und andere Optionen der vorherigen Darstellungen sind hier möglich, wie sie bereits weiter oben beschrieben wurden.

3.5.10.1
Seite 14

An dieser Stelle sollte nun geklärt werden, was sich hinter den einzelnen Berechtigungsbezeichnungen verbirgt:

Objektberechtigungen

- Verwendungsberechtigung wird benötigt,
  - um die Tabelle für eine Verarbeitung zu öffnen. Mindestens eine Datenberechtigung muss noch vorliegen.
  - um ein Programm umzuwandeln, das die Dateibeschreibung benutzt.
  - um Informationen über aktive Teildateien anzuzeigen.
  - um die Tabelle für Abfrageoperationen zu öffnen. (Datenberechtigungen für die Zugriffe müssen natürlich auch vorliegen.)
- Existenzberechtigung wird benötigt,
  - wenn die Tabelle gelöscht werden soll.
  - wenn Sicherung, Wiederherstellung und Speicherplatzfreigabe zum Einsatz kommen sollen.
  - wenn Teildateien entfernt werden sollen.
  - wenn der Eigner geändert wird.
  - Abgesehen von der Sicherung und Wiederherstellung werden auch Verwendungsberechtigungen benötigt.
  - Wenn die Existenzberechtigung nicht speziell vergeben wurde, reicht die *SAVSYS-Benutzerberechtigung für diese Operationen aus.
- Verwaltungsberechtigung
  - Erstellen logischer Dateien mit Schlüsselzugriff (Berechtigung muss auch für die referenzierte, physische Datei vorliegen.)
  - Erteilen und Entziehen von Berechtigungen, wobei die Verwendungsberechtigung ebenfalls vorhanden sein muss.
  - Änderung der Datei
  - Teildateien hinzufügen
  - Teildateien ändern
  - Verschieben der Datei
  - Umbenennen der Datei oder einer Teildatei
  - Sätze aus einer Teildatei entfernen (Löschberechtigung muss auch vorliegen.)
  - Reorganisation der Tabelle
- Berechtigung für Änderungen
  - Für viele Aufgaben, für die die Verwaltungsberechtigung benötigt wird, ist eine Änderungsberechtigung nötig. Sie kann als Ersatz für die Verwaltungsberechtigung aufgefasst werden.
- Referenzberechtigung
  - Wird benötigt, wenn referenzielle Integritätsbedingungen eingerichtet werden sollen.

Werfen wir noch einen Blick auf die verschiedenen Datenberechtigungen:

3.5.10.1
Seite 15

Datenberechtigungen

- Leseberechtigung
  – Lesende Zugriffe auf Sätze einer Tabelle
- Berechtigung zum Hinzufügen
  – Neue Sätze in Tabelle schreiben.
- Aktualisierungsberechtigung
  – Sätze können inhaltlich geändert werden.
- Löschberechtigung
  – Sätze können aus einer Tabelle entfernt werden.
- Berechtigung zum Ausführen

14. Ergänzung 3/2008

3.5.10.1

Seite 16

Berechtigungsvergabe mit SQL Grant

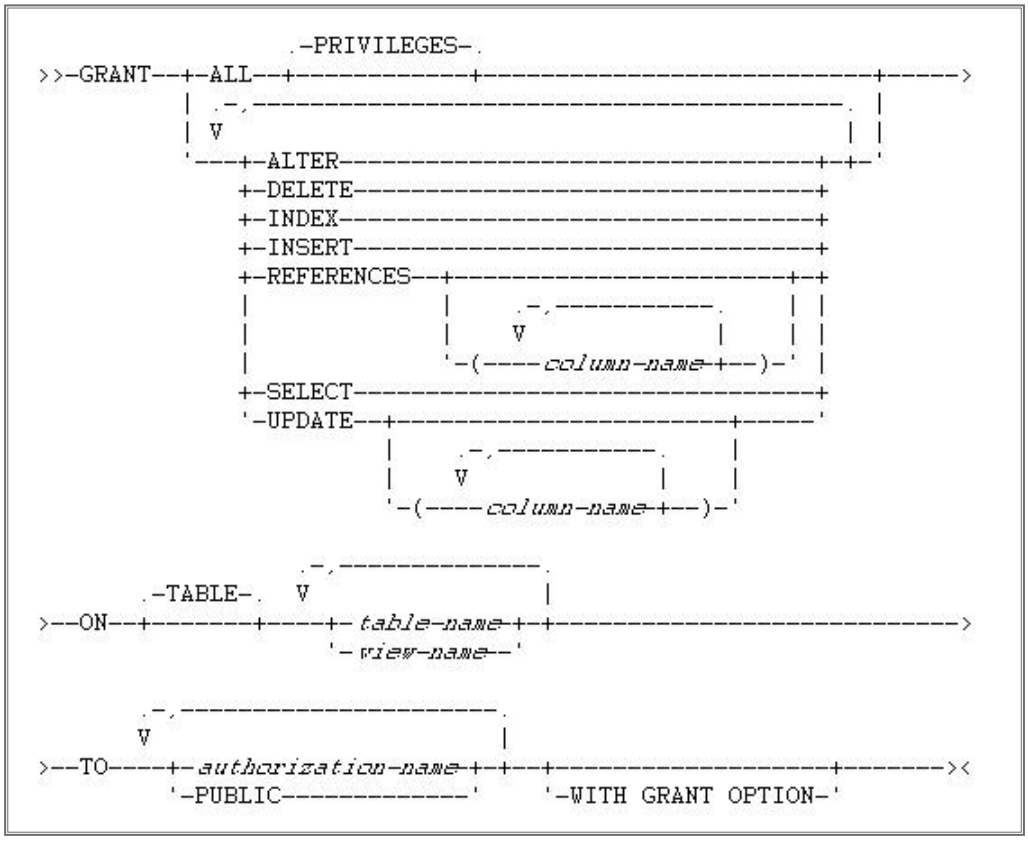
Die SQL-Statements GRANT (Berechtigung erteilen) und REVOKE (Berechtigung entziehen) sind auf folgende Datenbankobjekte anwendbar:

- SQL-Funktionen und Prozeduren
- SQL Packages
- Distinct-Typen
- Sequenz-Typen
- Tabellen und Sichten
- Einzelne Spalten von Tabellen und Sichten

Beginnen wir mit der Vergabe von Rechten auf Sichten und Tabellen. Der Aufruf kann interaktiv oder aus einem Programm heraus vorgenommen werden; er kann ein dynamisches (prepared) Statement sein. Welche Berechtigungen sind notwendig, damit ein solches Statement erfolgreich ausgeführt werden kann? Es muss natürlich mindestens jede Berechtigung vorhanden sein, die mit dem Statement erteilt werden soll. Ein Benutzer darf sich ja nicht selbst oder anderen höhere Rechte erteilen, als er selbst besitzt. Der Benutzer benötigt *OBJMGT für bezogene Tabellen und Views sowie *EXECUTE für die Bibliothek, in der sich die Datenbankobjekte befinden. Zudem wird die Verwaltungsberechtigung (administrative Berechtigung) verlangt. Mit dem GRANT-Statement können andere Profile ermächtigt werden, ebenfalls Berechtigungen zu vergeben (WITH GRANT OPTION).

3.5.10.2
Seite 2

Für diesen Fall sind entweder administrative Berechtigungen oder die Eignerschaft des betroffenen Datenbankobjekts notwendig. Diese Variante des GRANT-Statements erhält folgende Gestalt:



Betrachten wir die einzelnen Abschnitte der Syntax:

- ALL
  Hier können alle Rechte vergeben werden, die dem Benutzer selbst zur Verfügung stehen. Bitte verwechseln Sie diese Angabe nicht mit der *ALL-Berechtigung!
- ALTER
  Es wird die Berechtigung erteilt, die zugrundeliegende Tabelle oder View zu ändern. Auch Trigger können erstellt oder entfernt werden. Außerdem sind die LABEL- und COMMENT-Statements verwendbar.
- DELETE
  Sätze können aus der Tabelle gelöscht werden.
- INDEX
  Berechtigung, einen Index für die Tabelle zu erstellen. Diese Berechtigung kann nicht für eine Ansicht vergeben werden.
- INSERT
  Es können Sätze in eine Tabelle eingefügt werden. Soll diese Berechtigung für eine Ansicht vergeben werden, muss sie dies von ihrem Aufbau her gestatten.

- REFERENCES
  Erlaubt die Erstellung referenzieller Integritätsbedingungen für die bezogenen Tabellen im Sinne einer führenden Tabelle.
- REFERENCES (Spalte, Spalte…)
  Diese Berechtigung entspricht der vorhergehenden, nur das sie für die in der Liste aufgeführten Spalten gilt und nicht für alle Spalten der Tabelle. Jede hier aufgeführte Spalte muss in den Tabellen der ON-Klausel vorhanden sein. Qualifizierte Spaltennamen sind nicht zulässig.
- SELECT
  Berechtigung, Datensätze aus einer Tabelle oder View zu lesen. Es ist auch erlaubt, eine Ansicht für die Tabelle zu erstellen. Diese Berechtigung wird ebenfalls benötigt, wenn die Tabelle oder View in einer Abfrage angegeben wird.
- UPDATE
  Die Daten in einer Tabelle dürfen geändert werden.
- UPDATE (Spalte, Spalte…)
  Auch hier entspricht die Berechtigung dem normalen UPDATE, nur das hier wieder spezielle Spalten angegeben werden können.
- ON-Tabelle oder Ansicht
  In diesem Parameter geben Sie an, für welche Tabelle oder View Sie die Berechtigungen erteilen wollen.
- TO
  Bestimmen Sie mit dieser Angabe, wem Sie die Berechtigungen erteilen wollen.
- WITH GRANT OPTION
  Wenn die Empfänger der Rechte ihrerseits Berechtigungen vergeben sollen, dann setzen Sie diese Option ein.

Hier sehen Sie noch einige Beispiele für das GRANT-Statement:

GRANT INSERT, DELETE ON PAKND00 TO BOEHMER, LUTTKUS
Erteilt den Berechtigungs-IDs BOEHMER und LUTTKUS die Berechtigung, Sätze in der Tabelle PAKND00 einzufügen und zu löschen.

GRANT UPDATE(KDKDNR, KDKDNAM) ON PAKND00, PAKND01 TO BOEHMER, LUTTKUS
Erteilt den Berechtigungs-IDs BOEHMER und LUTTKUS die Möglichkeit, die Inhalte der Spalten KDKDNR und KDNAM sowohl in der Tabelle PAKND00 als auch in der Ansicht PAKND01 zu ändern.

GRANT ALL PRIVILEGES ON PAKND00 TO PUBLIC
Mit diesem Statement werden alle Berechtigungen für die Tabelle PAKND00 an PUBLIC vergeben.

3.5.10.2
Seite 4

Das GRANT-Statement erteilt im Grunde genommen Systemberechtigungen für SQL-Objekte (REVOKE entzieht sie). In der nachstehenden Tabelle ist ein Vergleich zwischen den Berechtigungen im GRANT-Befehl für Tabellen und für die entsprechenden Systemberechtigungen abgebildet.

SQL Privilege	Corresponding System Authorities when Granting to or Revoking from a Table
ALL (GRANT or revoke of ALL only grants or revokes those privileges the authorization ID of the statement has)	*OBJALTER *OBJMGT (Revoke only) *OBJOPR *OBJREF *ADD *DLT *READ *UPD
ALTER	*OBJALTER
DELETE	*OBJOPR *DLT
INDEX	*OBJALTER
INSERT	*OBJOPR *ADD
REFERENCES	*OBJREF
SELECT	*OBJOPR *READ
UPDATE	*OBJOPR *UPD
WITH GRANT OPTION	*OBJMGT

Die nächste Tabelle stellt einen analogen Vergleich her, wenn sich der GRANT-Befehl auf eine View bezieht. In diesem Fall können natürlich auch von der View referenzierte andere Ansichten und Tabellen betroffen sein.

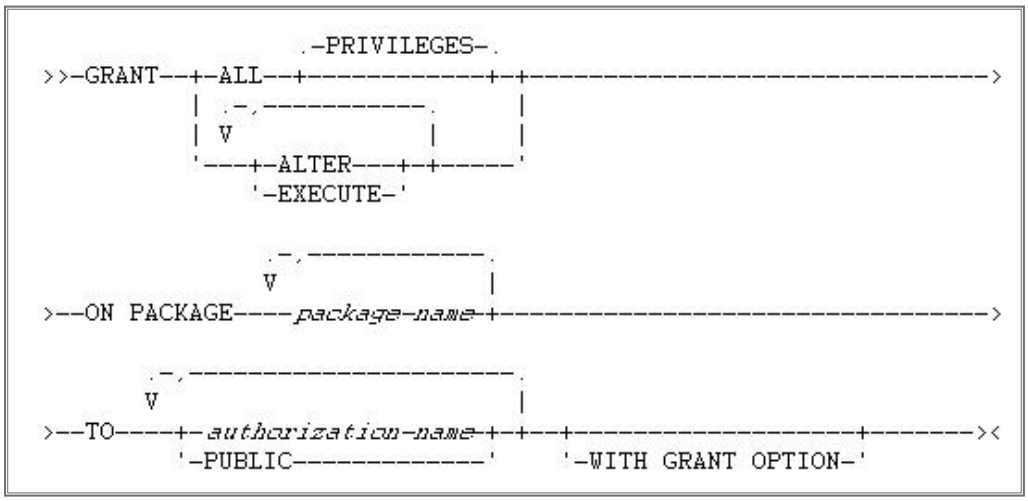
SQL Privilege	Corresponding System Authorities Granted to or Revoked from View	Corresponding System Authorities Granted to or Revoked from Referenced Tables and Views
ALL (GRANT or REVOKE of ALL only grants or revokes those privileges the authorization ID of the statement has)	*OBJALTER *OBJMGT (Revoke only) *OBJOPR *OBJREF *ADD *DLT *READ *UPD	*ADD *DLT *READ *UPD
ALTER	*OBJALTER	None
DELETE	*OBJOPR *DLT	*DLT
INDEX	Not Applicable	Not Applicable
INSERT	*OBJOPR *ADD	*ADD
REFERENCES	*OBJREF	None
SELECT	*OBJOPR *READ	*READ
UPDATE	*OBJOPR *UPD	*UPD
WITH GRANT OPTION	*OBJMGT	None

Wie ist der Zusammenhang, wenn die Berechtigungen bei Tabellen und Views geprüft werden – beispielsweise bei Zugriffen. Werfen wir doch einen Blick darauf:

SQL Privilege	Corresponding System Authorities when Checking Privileges to a Table
ALTER	*OBJALTER or *OBJMGT
DELETE	*OBJOPR and *DLT
INDEX	*OBJALTER or *OBJMGT
INSERT	*OBJOPR and *ADD
REFERENCES	*OBJREF or *OBJMGT
SELECT	*OBJOPR and *READ
UPDATE	*OBJOPR and *UPD

SQL Privilege	Corresponding System Authorities to the View	Corresponding System Authorities to the Referenced Tables and Views
ALTER	*OBJALTER and *OBJMGT	None
DELETE	*OBJOPR and *DLT	*DLT
INDEX	Not Applicable	Not Applicable
INSERT	*OBJOPR and *ADD	*ADD
REFERENCES	*OBJREF or *OBJMGT	None
SELECT	*OBJOPR and *READ	*READ
UPDATE-	*OBJOPR and *UPD	*UPD

Der folgende Teil beschäftigt uns mit der Berechtigungsvergabe für SQL Packages. Die Abbildung zeigt das Syntaxdiagramm des GRANT-Statements für diese Variante:



3.5.10.2
Seite 6

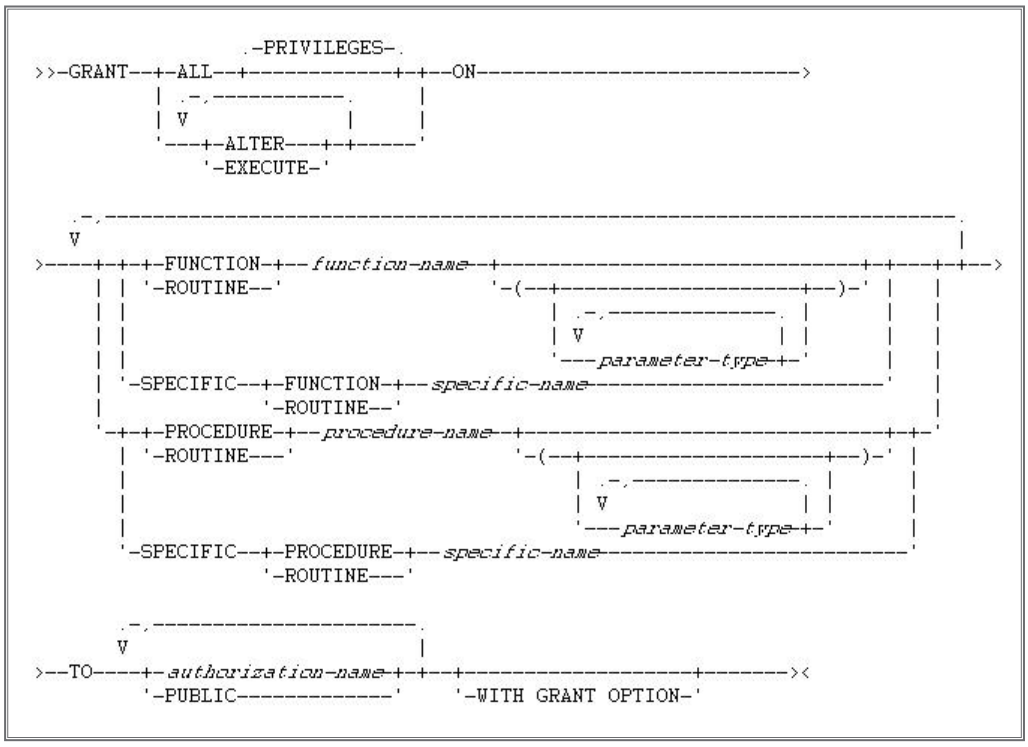
Da die grundlegenden Dinge schon im Abschnitt für Tabellen und Views erläutert wurden, werde ich mich hier nicht wiederholen. Von Bedeutung ist hier lediglich nur die Berechtigung EXECUTE. Es ist naheliegend, dass dieses Recht es erlaubt, Statements im bezogenen Package auszuführen. Auch hier sehen Sie im Folgenden zwei Abbildungen, die die Zusammenarbeit zwischen SQL-Privilegien und Systemberechtigungen verdeutlichen. Daran anschließend zeige ich noch ein kleines Beispiel.

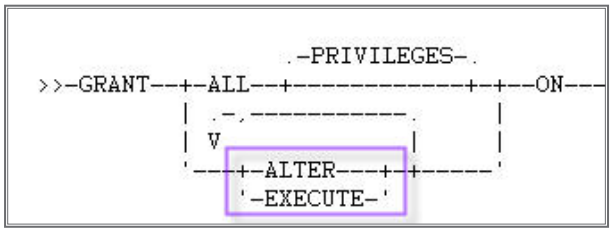
SQL Privilege	Corresponding System Authorities when Granting to or Revoking from a Package
ALL (Grant or revoke of ALL grants or revokes only those privileges the authorization ID of the statement has)	*OBJALTER *OBJOPR *EXECUTE *OBJMGT (Revoke only)
ALTER	*OBJALTER
EXECUTE	*EXECUTE *OBJOPR
WITH GRANT OPTION	*OBJMGT

SQL Privilege	Corresponding System Authorities When Checking Privileges to a Package
ALTER	*OBJALTER
EXECUTE	*EXECUTE and *OBJOPR

GRANT EXECUTE ON PACKAGE MYPACKAGE TO PUBLIC
Vergabe der Ausführungsberechtigung an PUBLIC.

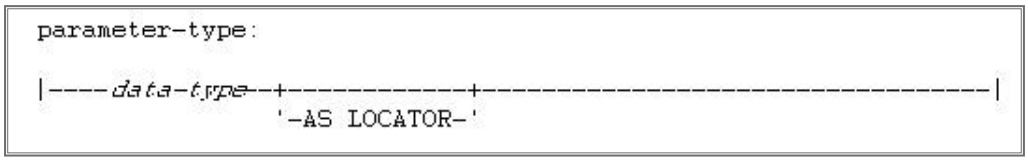
Bezüglich weiterer Ausprägungen des GRANT-Statements möchte ich an dieser Stelle noch auf die Behandlung von Prozeduren und Funktionen eingehen. Die Syntax ist hier allerdings erheblich komplexer als die bisher besprochenen. Grundsätzlich ist folgende Struktur aufzubauen:





Die Berechtigung ALTER erlaubt es, das ALTER PROCEDURE- und COMMENT-Statement einzusetzen. EXECUTE wird benötigt, um die Funktion oder Prozedur aufzurufen.

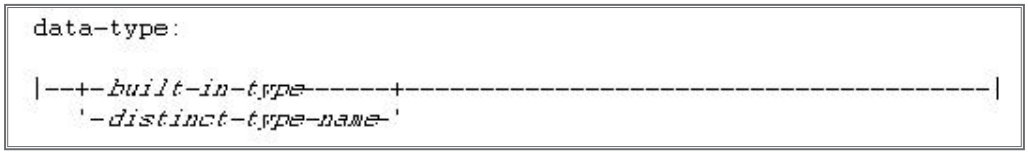
Mit FUNCTION oder SPECIFIC FUNCTION wird die betroffene Funktion angezogen. Dabei muss es sich um eine benutzerdefinierte Funktion handeln – nicht um eine der Standardfunktionen. Zudem liegt die betreffende Funktion auf dem lokalen Server. Die Parametersignatur dient ebenfalls zur eindeutigen Identifizierung einer Funktion. Die Angabe der Parametertypen ist selbst wieder zusammengesetzt.



Der Bestandteil *data-type* ist wiederum zusammengesetzt. Aber was steckt hinter der Angabe AS LOCATOR? Wenn dieser Wert angegeben ist, muss der übergebene Parameter entweder ein LOB sein oder zumindest ein Datentyp, der auf einem LOB basiert. Die Angabe SPECIFIC FUNCTION identifiziert eine benutzerdefinierte Funktion über ihren spezifischen Namen.

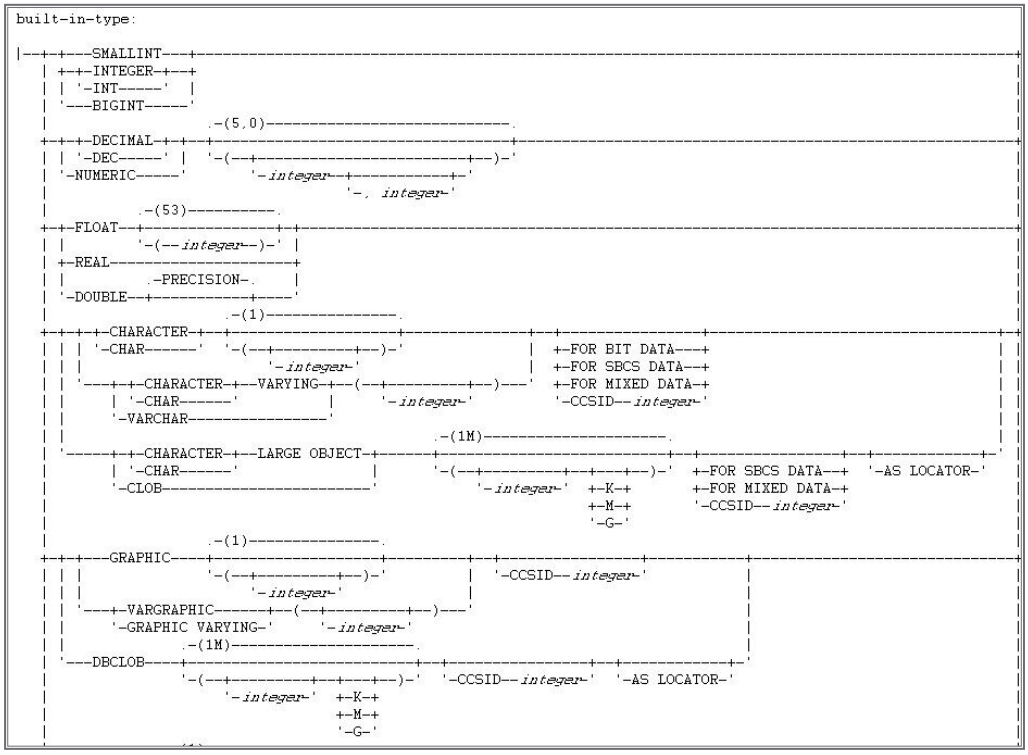
Die Angaben TO und WITH GRANT OPTION sind in den vorhergehenden Abschnitten besprochen worden und werden hier nicht noch einmal aufgeführt.

Die Teile der Syntax für die Rechtevergabe in Bezug auf Prozeduren sind aus den korrelierenden Bereichen für die Funktionen zu entnehmen. Ich stelle in der nachstehenden Abbildung die Struktur des *data-type* dar.



3.5.10.2
Seite 8

Damit ist der ausufernden Komplexität aber noch nicht Genüge getan, denn *built-in-type* ist ebenfalls aggregiert:



Ich möchte Sie nicht mit allen Details langweilen, daher verweise ich für nähere Betrachtungen dieses Abschnitts auf die entsprechenden Referenzinformationen der IBM.

Die Berechtigungen werden auf die zugrundeliegenden Programme (*PGM) oder Serviceprogramme (*SRVPGM) der Funktionen und Prozeduren gelegt.

ACHTUNG:

Wenn es sich um JAVA-Funktionen/Prozeduren handelt, laufen die Rechte gegen die entsprechenden Klassen- beziehungsweise JAR-Dateien!

Auch hier soll noch eine Gegenüberstellung von SQL-Privilegien mit Systemberechtigungen erfolgen.

Nicht-JAVA-Prozeduren und -Funktionen

SQL Privilege	Corresponding System Authorities when Granting to or Revoking from a Function or Procedure
ALL (Grant or revoke of ALL grants or revokes only those privileges the authorization ID of the statement has)	*OBJALTER *OBJOPR *EXECUTE *OBJMGT (Revoke only)
ALTER	*OBJALTER
EXECUTE	*EXECUTE *OBJOPR
WITH GRANT OPTION	*OBJMGT

JAVA-Prozeduren und -Funktionen

SQL Privilege	Corresponding Data Authorities when Granting to or Revoking from a Java Function or Procedure	Corresponding Object Authorities when Granting to or Revoking from a Java Function or Procedure
ALL (Grant or revoke of ALL grants or revokes only those privileges the authorization ID of the statement has)	*RWX	*OBJEXIST *OBJALTER *OBJMGT (Revoke only)
ALTER	*R	*OBJALTER
EXECUTE	*RX	*EXECUTE
WITH GRANT OPTION	*RWX	*OBJMGT

Berechtigungsprüfung Nicht-JAVA-Prozeduren und -Funktionen

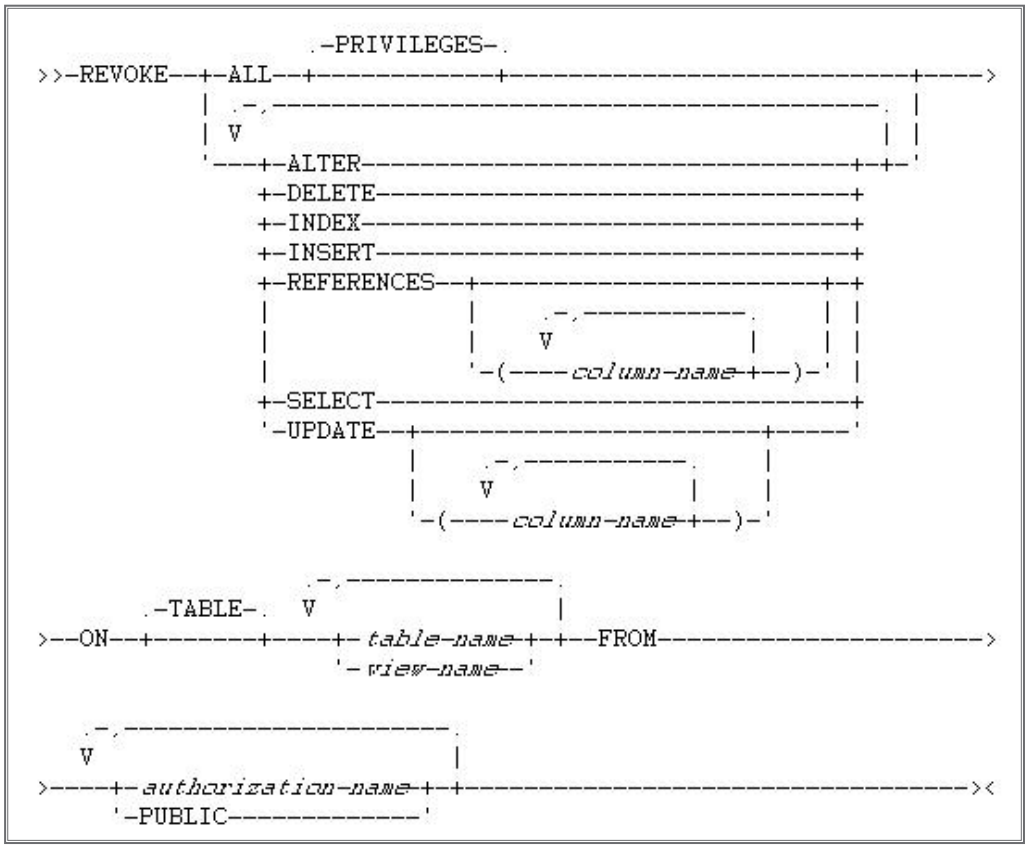
SQL Privilege	Corresponding System Authorities
ALTER	*OBJALTER
EXECUTE	*EXECUTE and *OBJOPR

Berechtigungsprüfung JAVA-Prozeduren und -Funktionen

SQL Privilege	Corresponding Data Authorities when Checking Privileges to a Java Function or Procedure	Corresponding Object Authorities when Checking Privileges to a Java Function or Procedure
ALTER	*R	*OBJALTER
EXECUTE	*RX	*EXECUTE

Zu Schluss betrachten Sie noch den Entzug von Berechtigungen mit dem REVOKE-Statement im Syntaxdiagramm. Die Bedeutung der einzelnen Einstellungen können Sie den entsprechenden Beschreibungen des GRANT-Statements entnehmen.

3.5.10.2
Seite 10



Berechtigungsvergabe mit GRTOBJAUT

Für die Vergabe von Objektberechtigungen steht Ihnen auch der Befehl GRTOBJAUT zur Verfügung, den ich nachstehend beschreiben werde.

Mit diesem Befehl können Sie die speziellen Berechtigungen, die in den Parametern des Befehls angegeben werden, an einzelne Benutzer oder auch an Gruppen von Benutzern vergeben. Die Vergabe ist möglich für:

- angegebene Benutzerprofile
- Benutzer, die keine speziellen Berechtigungen haben (*PUBLIC)
- Berechtigungsübernahme von Referenzobjekten
- und Berechtigungslisten

Abgesehen von Benutzerprofilen können Objekte durch Berechtigungslisten gesichert werden – also auch die uns interessierenden Datenbankobjekte. Dieser Befehl kann vom Besitzer eines Objekts verwendet werden oder von einem Benutzer, der die Objektmanagement-Berechtigung für das Zielobjekt besitzt.

Ein Profil mit dieser Berechtigung kann durch jede Berechtigung, die der Benutzer besitzt, an andere Benutzer vergeben – abgesehen von der Verwaltungsberechtigung. Objektmanagement-Rechte können vom Eigner oder von Benutzern mit *ALLOBJ-Berechtigung vergeben werden. Berechtigungslisten können von Benutzern mit *ALL-Rechten zugeordnet werden.

Über den REPLACE-Parameter können Sie festlegen, ob die zu erteilenden Rechte die bestehenden ersetzen oder ob sie einfach hinzugefügt werden sollen. Die Voreinstellung ist REPLACE(*NO), was dazu führt, dass einerseits die gewünschten Rechte erteilt werden. Andererseits werden keine Berechtigungen entfernt, die höher sind, als die von Ihnen vergebenen! Wenn Sie das erreichen möchten, können Sie REPLACE(*YES) verwenden. Bei dieser Einstellung werden die bisherigen Rechte entfernt und durch die von Ihnen gewünschten ersetzt.

Einige wenige Einschränkungen sind bei WRKOBJAUT zu beachten. So fordert der Befehl eine exklusive Sperre an, bevor Lese- oder Verwendungsrechte für eine Tabelle erteilt werden können. Der Aufrufer muss *ALL-Berechtigungen für die Datei besitzen oder ihr Eigner sein, wenn eine Berechtigungsliste verwendet werden soll (*AUTL-Parameter). Ebenfalls werden bei Tabellen Verwaltungs- und Verwendungsberechtigungen benötigt. Ich habe hier nur die Einschränkungen gezeigt, die in Bezug auf Tabellen in Frage kommen. Bei anderen Objektarten finden sich weitere Einschränkungen.

3.5.10.3
Seite 2

Die nächste Grafik zeigt den Befehl mit seinen Standard-Parameterbelegungen. Anschließend sehen Sie eine Tabelle, die die möglichen Verwendungsarten der verschiedenen Parameter zusammenfasst, wobei ich nicht die gesamte Kombinationsvielfalt besprechen werde, sondern einige wesentliche Punkte herausgreife.

Keyword	Description	Choices	Notes
OBJ	Object	*Qualified object name*	Required, Positional 1
	Qualifier 1: Object	*Generic name, name*, *ALL	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB, *ALL, *ALLUSR, *USRLIBL, *ALLAVL, *ALLUSRAVL	
OBJTYPE	Object type	*ALL, *ALRTBL, *BNDDIR, *CFGL, *CHTFMT, *CLD, *CLS, *CMD, *CNNL, *COSD, *CRG, *CRQD, *CSI, *CSPMAP, *CSPTBL, *CTLD, *DEVD, *DTAARA, *DTADCT, *DTAQ, *EDTD, *FCT, *FILE, *FNTRSC, *FNTTBL, *FORMDF, *FTR, *GSS, *IGCDCT, *IGCSRT, *IGCTBL, *IMGCLG, *IPXD, *JOBD, *JOBQ, *JOBSCD, *JRN, *JRNRCV, *LIB, *LIND, *LOCALE, *M36, *M36CFG, *MEDDFN, *MENU, *MGTCOL, *MODD, *MODULE, *MSGF, *MSGQ, *NODGRP, *NODL, *NTBD, *NWID, *NWSCFG, *NWSD, *OUTQ, *OVL, *PAGDFN, *PAGSEG, *PDFMAP, *PDG, *PGM, *PNLGRP, *PRDAVL, *PRDDFN, *PRDLOD, *PSFCFG, *QMFORM, *QMQRY, *QRYDFN, *RCT, *S36, *SBSD, *SCHIDX, *SPADCT, *SQLPKG, *SQLUDT, *SRVPGM, *SSND, *SVRSTG, *TBL, *TIMZON, *USRIDX, *USRPRF, *USRQ, *USRSPC, *VLDL, *WSCST	Required, Positional 2
ASPDEV	ASP device	*Name*, *, *SYSBAS	Optional
USER	Users	Single values: *PUBLIC Other values (up to 50 repetitions): *Name*	Optional, Positional 3
AUT	Authority	Single values: ***CHANGE**, *ALL, *USE, *EXCLUDE, *AUTL Other values (up to 10 repetitions): *OBJALTER, *OBJEXIST, *OBJMGT, *OBJOPR, *OBJREF, *ADD, *DLT, *READ, *UPD, *EXECUTE	Optional, Positional 4
AUTL	Authorization list	*Name*, *NONE	Optional
REFOBJ	Reference object	*Qualified object name*	Optional
	Qualifier 1: Reference object	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	

REFOBJTYPE	Reference object type	***OBJTYPE**, *ALRTBL, *BNDDIR, *AUTL, *CFGL, *CHTFMT, *CLD, *CLS, *CMD, *CNNL, *COSD, *CRG, *CRQD, *CSI, *CSPMAP, *CSPTBL, *CTLD, *DEVD, *DTAARA, *DTADCT, *DTAQ, *EDTD, *FCT, *FILE, *FNTRSC, *FNTTBL, *FORMDF, *FTR, *GSS, *IGCDCT, *IGCSRT, *IGCTBL, *IMGCLG, *IPXD, *JOBD, *JOBQ, *JOBSCD, *JRN, *JRNRCV, *LIB, *LIND, *LOCALE, *M36, *M36CFG, *MEDDFN, *MENU, *MGTCOL, *MODD, *MODULE, *MSGF, *MSGQ, *NODGRP, *NODL, *NTBD, *NWID, *NWSCFG, *NWSD, *OUTQ, *OVL, *PAGDFN, *PAGSEG, *PDFMAP, *PDG, *PGM, *PNLGRP, *PRDDFN, *PRDLOD, *PSFCFG, *QMFORM, *QMQRY, *QRYDFN, *RCT, *S36, *SBSD, *SCHIDX, *SPADCT, *SQLPKG, *SQLUDT, *SRVPGM, *SSND, *SVRSTG, *TBL, *TIMZON, *USRIDX, *USRPRF, *USRQ, *USRSPC, *VLDL, *WSCST	Optional
REFASPDEV	Reference ASP device	*Name*, *, *SYSBAS	Optional
REPLACE	Replace authority	***NO**, *YES	Optional

```
Auswahl eingeben und Eingabetaste drücken.

Objekt . . . . . . . . . . . . . . OBJ        >   PAK*
  Bibliothek . . . . . . . . . . .            >     BOEHMER
Objektart  . . . . . . . . . . . . OBJTYPE    >   *FILE
```

Im Parameter OBJ geben Sie den eindeutigen Namen des Zielobjekts an oder wahlweise auch einen generischen Namen. Auch die Eintragung *ALL ist denkbar. Sie führt dazu, dass alle Objekte der Objektart im Parameter OBJTYPE die Berechtigungen erhalten. In OBJTYPE geben Sie die gewünschte Objektart selbst an (hier *FILE). In diesem Parameter kann ebenfalls *ALL eingetragen werden. Man kann also für alle Objekte – unabhängig von der Objektart – Berechtigungen in einem Zug vergeben.

```
Benutzer . . . . . . . . . . . . . . USER
                + für weitere Werte
```

Hier tragen Sie den (oder die) Benutzer ein, dem Rechte erteilt werden sollen. Verwenden Sie Profilnamen oder den Eintrag *PUBLIC. Dieser Parameter muss angegeben werden, außer es wird ein Referenzobjekt oder eine Berechtigungsliste verwendet.

```
Berechtigung . . . . . . . . . . . AUT          *CHANGE
                + für weitere Werte
```

Dieser Parameter erhält die eigentlichen Berechtigungen. Hier sind einige vordefinierte Einstellungen verwendbar:

- *CHANGE
  Benutzer mit *CHANGE können alle Aktionen vornehmen, die keine Eigner-, Existenz- oder Verwaltungsberechtigung erfordern.
- *ALL
  Abgesehen von den Zugriffen, die Eignerschaft oder Verwaltungsrechte für Berechtigungslisten erfordern, kann der Besitzer dieser Berechtigung weitgehend arbeiten.
- *USE
  Erlaubt grundlegende Operationen – wie zum Beispiel das Lesen.

- *EXCLUDE
  Der Benutzer wird von der Verwendung des Objekts ausgeschlossen.
- *AUTL
  Sie können eine Berechtigungsliste für die Absicherung der Tabelle verwenden. Allerdings müssen Sie dann im Parameter USER *PUBLIC mit angegeben.

Darüber hinaus existieren noch weitere Eintragungen:

- *OBJALTER
  Diese Option wurde schon unter SQL GRANT beschrieben und wird zur Zeit auch nur für Datenbankdateien und SQL Packages verwendet.
- *OBJMGT
  Die Steuerung der Dateiabsicherung sowie das Verschieben, Umbenennen und Hinzufügen von Teildateien sind erlaubt.
- *OBJEXIST
  Die Existenz der Tabelle der betreffenden Aktionen ist zulässig.
- *OBJOPR
  Besagt, dass das Objekt entsprechend der vorhandenen Datenberechtigungen verwendet werden darf.
- *OBJREF
  Lässt die Referenzierung von Tabellen zu. Auch dieser Punkt wurde weiter oben bereits beschrieben (RIs!). Genau wie *OBJALTER wird dieser Eintrag aktuell nur für Datenbank-Files eingesetzt.

Im Bereich der Datenberechtigungen können folgende Werte verwendet werden:

- *ADD
- *DELETE
- *EXECUTE
- *READ
- *UPDATE

Ich denke, diese Angaben müssen nicht detailliert besprochen werden, da sie selbsterklärend sind. Wenn Sie sich bei *EXECUTE nicht ganz sicher sind, dann können Sie dazu auch Einiges unter GRANT nachlesen.

```
Berechtigungsliste . . . . . AUTL
```

Tragen Sie hier die Berechtigungsliste ein, die Ihr Objekt sichern soll. Dafür werden *AUTLMGT-Rechte benötigt. Wenn eine Berechtigungsliste angegeben ist, kann in AUT, REFOBJ und REFOBJTYPE keine Angabe gemacht werden.

```
Bezugsobjekt . . . . . REFOBJ
  Bibliothek . . . . . *LIBL
Bezugsobjektart  . . . . . REFOBJTYPE    *OBJTYPE
```

Geben Sie hier den Namen des Objekts an, das die Informationen über die zu erteilenden Berechtigungen besitzt. In den Parametern AUT und AUTL dürfen dann keine Angaben mehr gemacht werden. Der Name des Objekts ist anzugeben; als Bibliothek kann *LIBL, *CURLIB oder der Name einer vorhandenen Bibliothek eingesetzt werden. Falls es notwendig ist, kann die Objektart noch über REFOBJTYPE genau spezifiziert werden.

```
Berechtigung ersetzen  . . . . . REPLACE        *NO
```

Diese Einstellung habe ich ebenfalls schon beim GRANT-Statement erläutert.

3.5.10.3

Seite 6

Transaktionssteuerung

Was verbirgt sich hinter den Begriffen „Transaktion“ und „Transaktionssteuerung“? Über das Konzept einer Transaktion können Sie sicherstellen, dass Ihre Datenbestände konsistent bleiben. Dieses Konzept gilt durch die ausgiebig getestete Software als sicher. Es stellt sich jedoch die berechtigte Frage, was zu erwarten ist, wenn beispielsweise ein technisches Problem auftritt, während eine Anwendung mit der Änderung von Daten beschäftigt ist?

Schauen wir uns dazu ein kleines Beispiel an: Zwei Bankkunden besitzen jeweils ein Konto. Diese Konten sind in verschiedenen Tabellen gespeichert. Kunde A überweist einen bestimmten Betrag von seinem Konto auf das Konto von Kunde B. Die mit dieser Umbuchung beauftragte Anwendung bucht den gewünschten Betrag vom Konto A ab. Nachdem das Konto seinen neuen Saldo erhalten hat, führt ein technischer Umstand dazu, dass die Anwendung den Betrag nicht mehr auf Konto B gutschreiben kann (Stromausfall, Hardware-Schaden o.ä.). Kunde A befindet sich nun in der ärgerlichen Situation, dass er einerseits über weniger Geld verfügt und andererseits sich Kunde B bei ihm melden wird. Denn der wird sicher bald feststellen, dass der erwartete Betrag nicht auf sein Konto eingegangen ist.

Geld kann heutzutage auf viele Arten vernichtet werden. Die Variante aus unserem kleinen Beispiel lässt sich jedoch verhindern, wenn auf den Transaktionsmechanismus zurückgegriffen werden kann. Unter der Transaktionssteuerung kann sichergestellt werden, dass alle Änderungen, die innerhalb einer Transaktion stattgefunden haben, dauerhaft festgeschrieben werden. Auf der anderen Seite können auch all diese Änderungen rückgängig gemacht werden, falls die Abarbeitung eines Prozesses (in unserem Fall die Umbuchung eines Geldbetrags) unterbrochen wird. Damit können Probleme vermieden werden, sofern eine Anwendung unmotiviert „abstürzt“. Allerdings könnte die Anwendung auch selbst zu dem Schluss kommen, dass eine Verarbeitung – aus welchen Gründen auch immer – fehlerhaft ist. In diesem Fall kann die Anwendung selbst die Änderungen zurücknehmen. Die Software kann auch so entworfen werden, dass sie neu gestartet wird. Damit kann mithilfe der Transaktionssteuerung sichergestellt werden, dass keine unvollständigen Datenbank-Änderungen vorgenommen wurden.

16. Ergänzung 9/2008

3.5.11
Seite 2

Der Begriff der Transaktion

Stellen Sie sich vor, dass eine Reihe einzelner Schritte notwendig ist, um einen bestimmten Prozess abzubilden. Korrekt durchgeführt wurde der Prozess nur dann, wenn jeder einzelne Schritt richtig abgeschlossen wurde. Erinnern wir uns dazu noch einmal an unser kleines Beispiel mit der Geldumbuchung. Man kann also sagen, dass Bestandteile einer Transaktion unteilbar sind – „atomar" sozusagen.

Transaktionen können in mehreren Situationen zustande kommen: Das betrifft einfache Aktionen, die auf eine Tabelle zugreifen, oder auch umfassendere Transaktionen mit Zugriffen auf mehrere Tabellen. Es ist aber auch denkbar, dass eine Anzahl von Änderungen nur ein Teil einer größeren (logischen) Transaktionsgruppe ist.

Transaktionen können auf das lokale System beschränkt sein, müssen es aber nicht. Sie können aber auch im Zusammenhang mit einem oder mehreren fernen Systemen stehen. Außerdem können sie auch mehreren Journalen auf dem lokalen System zugeordnet sein.

Man sollte auch bedenken, dass eine Transaktion auf dem lokalen System durchaus Objekte betreffen kann, die ***keine*** Datenbanktabellen sind.

16. Ergänzung 9/2008

3.5.11.1

Seite 2

Commit- und RollBack-Befehle

Diese Befehle werden im Zusammenhang mit der Transaktionssteuerung eingesetzt, um Änderungen einer Transaktion entweder endgültig festzuschreiben (Commit) oder rückgängig zu machen (Rollback).

Language or API	Commit	Rollback
CL	COMMIT command	ROLLBACK command
IBM® Integrated Language Environment® (ILE) RPG	COMIT operation code	ROLBK operation code
ILE COBOL	COMMIT verb	ROLLBACK verb
ILE C	_Rcommit function	_Rrollbck function
PLI	PLICOMMIT subroutine	PLIROLLBACK subroutine
SQL	COMMIT statement	ROLLBACK statement
SQL Call Level Interface (CLI)	SQLTransact() function (It is used to commit and roll back a transaction)	
XA APIs	»xa_commit() and db2xa_commit() APIs«	»xa_rollback() and db2xa_rollback() APIs«

Die Tabelle zeigt die Sprachen an, aus denen heraus ein Commit-/Rollback-Auftrag abgesetzt werden kann. JAVA ist hier nicht explizit aufgeführt, damit die Transaktionssteuerung über den Methoden-Einsatz spezieller Klassen erfolgen kann und nicht als eigenständiger Befehl implementiert wurde. Das betrifft auch andere Programmiersprachen, bedeutet aber nicht, dass diese Sprachen nicht im Zusammenhang mit der Transaktionsteuerung eingesetzt werden können.

Mit einer Commit-Operation werden alle Änderungen seit der letzten Commit- oder Rollback-Operation festgeschrieben. Alle Sperren, die mit der Transaktion im Zusammenhang stehen, werden ebenfalls freigegeben. Wenn ein System eine Commit-Aufforderung erhält, speichert es eine sogenannte Commit-Identifikation für eine eventuelle spätere Wiederherstellungsaktion.

Wenn Datensätze zu einer Datenbankdatei unter Transaktionskontrolle hinzugefügt werden und wenn die Tabelle mit der Einstellung SEQONLY(*YES) geöffnet wird, dann schreibt das System diese Daten in die Tabelle, sobald die Commit-Anforderung eingeht. Im Fall einer Journalisierung werden C-CM-Journaleinträge weggeschrieben – und zwar in jedes Journal, das mit der Commitment-Definition assoziiert ist. Weiterhin werden Satz- und Objektsperren freigegeben, die aufgrund der Transaktionssteuerung angefordert (und gehalten) wurden. Damit sind diese Ressourcen wieder für andere Benutzer verfügbar.

3.5.11.2
Seite 2

Die Rollback-Aktion verläuft folgendermaßen: Datensätze, die aus der Datenbank entfernt wurden, werden wiederhergestellt und geänderte Sätze werden in den Originalzustand versetzt. Sollten jedoch neue Sätze geschrieben werden, dann bleiben sie in den betreffenden Tabellen als gelöschte Sätze. Die zugeordneten Journale erhalten C-RB-Einträge.

Achtung!

Beachten Sie, dass Dateien, die unter Transaktionskontrolle geöffnet wurden, positioniert werden. Dabei wird auf den Satz positioniert, der in der letzten Transaktion im Zugriff war, oder auf den Anfang, sofern bisher noch keine Commit-Aktion erfolgte. Diese Punkte werden speziell dann beachtet, wenn Tabellen sequenziell verarbeitet werden.

Wenn Dateien im Zuge der Transaktion geschlossen wurden, erfolgt kein erneutes Öffnen oder Positionieren. Beachten Sie bitte, dass Dateien, die mit CLEAR-Operationen geleert wurden, nicht wiederhergestellt werden können (Noncommittable Ressources). Die angeforderten Satz- und Objektsperren werden wie bei Commit freigegeben.

Um die Übersichtlichkeit zu wahren, habe ich die wesentlichen Punkte angesprochen und nicht alle Details des Commit-/Rollback-Zyklus dargestellt.

Der Begriff Commitment-Definition

Sogenannte Commitment-Definitionen beinhalten Informationen zu Objekten, die während einer Transaktion beeinflusst werden. Sie werden unter anderem durch die Ausführung des Befehls STRCMTCTL (Transaktionssteuerung starten) erstellt. In einer Commitment-Definition werden Informationen zu Änderungen innerhalb einer Transaktion vorgehalten – und zwar solange, bis die Transaktion beendet wurde. Die folgende Abbildung zeigt die einzelnen Parameter, die im Befehl STRCMTCTL angegeben werden können.

```
Auswahl oder Befehl
===> strcmtctl

F3=Verlassen    F4=Bedienerführung    F9=Auffinden    F12=Abbrechen
F13=Unterstützende Informationen      F23=Anfangsmenü festlegen
```

```
                    COMMIT-Steuerung starten (STRCMTCTL)

 Auswahl eingeben und Eingabetaste drücken.

 Sperrebene . . . . . .          *CHG, *CS, *ALL
 Mitteilungsobjekt:
   Objekt . . . . . .    *NONE     Name, *NONE
     Bibliothek . . . .            Name, *LIBL, *CURLIB
   Objektart:
   (*MSGQ *DTAARA oder *FILE) . .  *MSGQ, *DTAARA, *FILE
   Teildatei, wenn *FILE . . . .   Name, *FIRST
 COMMIT-Definitionsbereich . . .   *ACTGRP  *ACTGRP, *JOB
 Text 'Beschreibung' . . . . . .   *DFTTEXT
 Journal . . . . . .   *NONE      Name, *NONE
   Bibliothek . . . . .          Name, *LIBL, *CURLIB
 Wegzulassende Journaleinträge .   *NONE    *NONE, *LUWID

                                                              Ende
 F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
 F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Sie können die Gültigkeit der Definition so festlegen, dass sie entweder für den Job oder für die Aktivierungsgruppe (Standardeinstellung) gilt. Für Dateien, die unter der Transaktionssteuerung geöffnet werden sollen, muss sichergestellt sein, dass eine Journalisierung erfolgt. Über die Parametrisierung kann eine nähere Definition der Sperren (Sperrebene) vorgenommen und sogenannte Notify-Objekte zugeordnet werden. Weiter ist die Angabe eines Standardjournals möglich. Einträge, die Systeme und Journale betreffen, die im Zusammenhang mit einer Operation innerhalb einer Transaktion stehen, können in diesem Journal gespeichert werden.

16. Ergänzung 9/2008

Betrachten wir die einzelnen Parameter:

Sperrebene (LCKLVL)
Hier wird angegeben, auf welche Weise Sätze gesperrt werden sollen. Dabei sind folgende Angaben möglich:

**CHG*
Sätze, die für eine spätere Fortschreibung gelesen werden, sind zu sperren. Wenn ein Satz geändert, neu hinzugefügt oder gelöscht wurde, bleibt er gesperrt, bis die Transaktion beendet wurde.

**CS*
Bei dieser Sperrebene wird jeder Satz gesperrt, der in den Zugriff gelangt. Wenn ein Satz gelesen, anschließend aber nicht geändert oder gelöscht wurde, wird er freigegeben, sobald ein anderer Satz gelesen wird.

**ALL*
Jeder Satz, der im Zugriff liegt, wird gesperrt, bis die Transaktion beendet ist.

Mitteilungs-Objekt (NFYOBJ)
An Notify-Objekte können Benachrichtigungen gesendet werden. Dabei wird der Commitment-Identifier der letzten (erfolgreichen) Commit-Operation an das Objekt gesendet, wenn einer der folgenden Umstände bei einem jobbezogenen Gültigkeitsbereich eingetreten ist:

– Ein Systemfehler ist aufgetreten.

– Ein Job wurde beendet, ohne dass Änderungen festgeschrieben wurden.

– Der Job wurde mit einem Beendigungscode ungleich 0 beendet.

Wenn der Gültigkeitsbereich auf eine Aktivierungsgruppe festgelegt wurde, erfolgt eine Benachrichtigung bei unnormaler Beendigung der Gruppe. Eine Benachrichtigung erfolgt auch, wenn ein Commit für anstehende Änderungen festgestellt werden konnte. Die einzelnen Angabemöglichkeiten sind für diesen Parameter selbsterklärend.

```
Mitteilungsobjekt:
  Objekt . . . . . . . . . . . .   *NONE      Name, *NONE
    Bibliothek . . . . . . . . .     __________ Name, *LIBL, *CURLIB
  Objektart:
  (*MSGQ *DTAARA oder *FILE) . .   _______     *MSGQ, *DTAARA, *FILE
  Teildatei, wenn *FILE  . . . .   __________  Name, *FIRST
```

Commit-Definitionsbereich (CMTSCOPE)

**JOB*

Die Commitment-Definition gilt für den startenden Job. Andere Jobs können diese Definition nicht verwenden.

**ACTGRP*

Bei dieser Einstellung ist die Commit-Definition innerhalb der Aktivierungsgruppe sichtbar.

```
Journal . . . . . . . . . . . . .   *NONE        Name, *NONE
  Bibliothek . . . . . . . . . .     __________   Name, *LIBL, *CURLIB
Wegzulassende Journaleinträge  .   *NONE         *NONE, *LUWID
```

Sollte ein Journal angegeben werden, dann können Sie mit dem Eintrag *LUWID im Parameter OMTJRNE (wegzulassende Einträge) dafür sorgen, dass nur im Problemfall Einträge in das Journal geschrieben werden. Wenn die einzelnen Commit- und Rollback-Operationen fehlerfrei durchgeführt werden, entfallen die Einträge im Default-Journal.

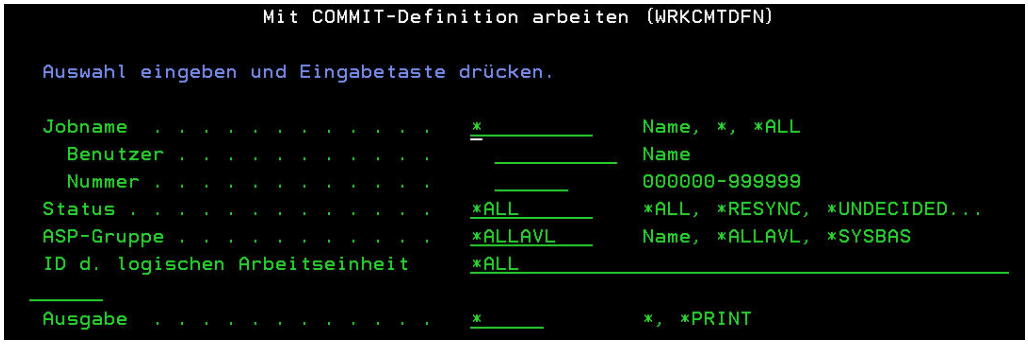
Ich rufe den Befehl STRCMTCTL einfach einmal von der Befehlszeile aus auf.

```
                    COMMIT-Steuerung starten (STRCMTCTL)

 Auswahl eingeben und Eingabetaste drücken.

 Sperrebene . . . . . . . . . . .   *chg          *CHG, *CS, *ALL
 Mitteilungsobjekt:
   Objekt . . . . . . . . . . . .   *NONE        Name, *NONE
     Bibliothek . . . . . . . . .     __________   Name, *LIBL, *CURLIB
   Objektart:
   (*MSGQ *DTAARA oder *FILE) . .   _______       *MSGQ, *DTAARA, *FILE
   Teildatei, wenn *FILE  . . . .   __________   Name, *FIRST
 COMMIT-Definitionsbereich  . . .   *job          *ACTGRP, *JOB
 Text 'Beschreibung'  . . . . . .   *DFTTEXT
 ______
 Journal  . . . . . . . . . . . .   *NONE        Name, *NONE
   Bibliothek . . . . . . . . . .     __________   Name, *LIBL, *CURLIB
 Wegzulassende Journaleinträge  .   *NONE         *NONE, *LUWID
```

Anschließend gebe ich in der Befehlszeile den Befehl WRKCMTDFN (Mit Commitment-Definitionen arbeiten) ein und lasse ihn mir mit „F4“ anzeigen.

```
Auswahl oder Befehl
===> WRKCMTDFN
```

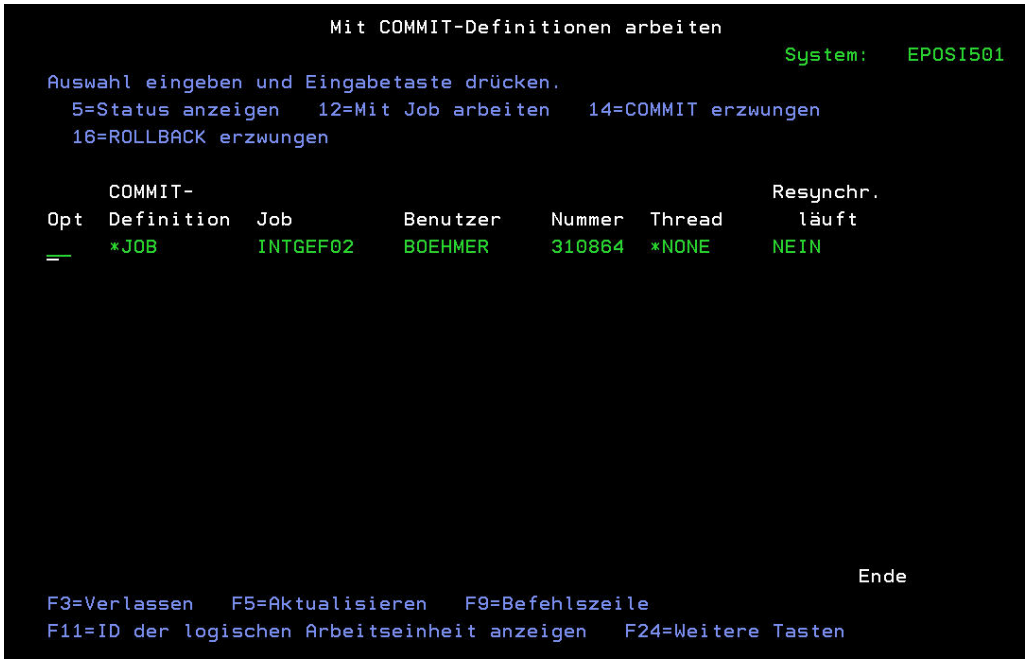
3.5.11.3
Seite 4



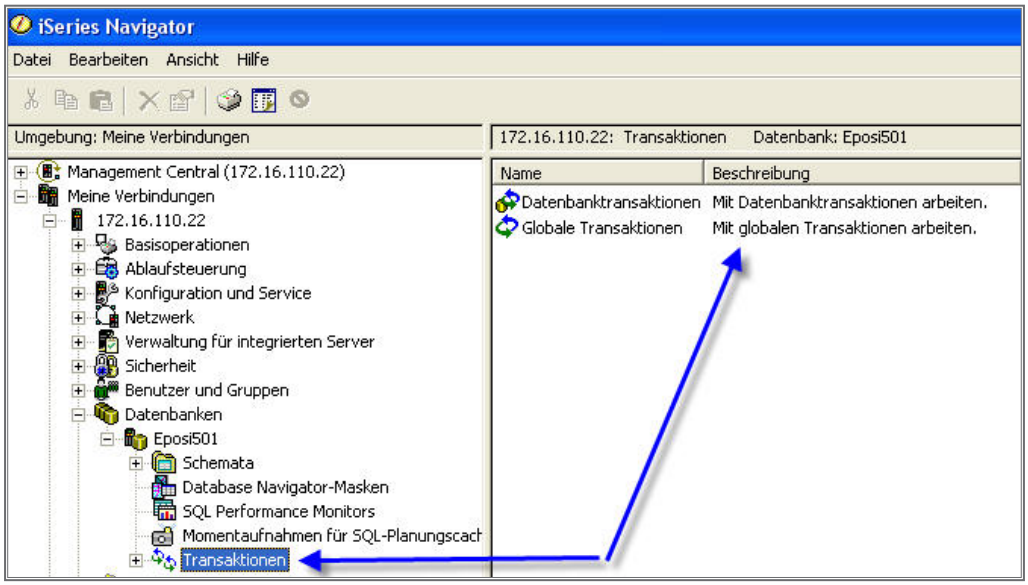
Über die jobbezogenen Angaben können Sie festlegen, welche Definitionen wichtig für Sie sind.

Die Verwendung von Sternchen (*) – wie es in der Standardbelegung vorgesehen ist – zeigt alle Commitment-Definitionen, die mit dem Job in Zusammenhang stehen. Setzt man dort *ALL ein, bekommt man die Definitionen aller Jobs auf dem System angezeigt. Ich belasse es vorerst bei der Einstellung Sternchen (*). Sollte es gewünscht sein, können Sie den Bereich durch spezifische Angaben – wie Jobnummer und Benutzer – noch weiter eingrenzen.

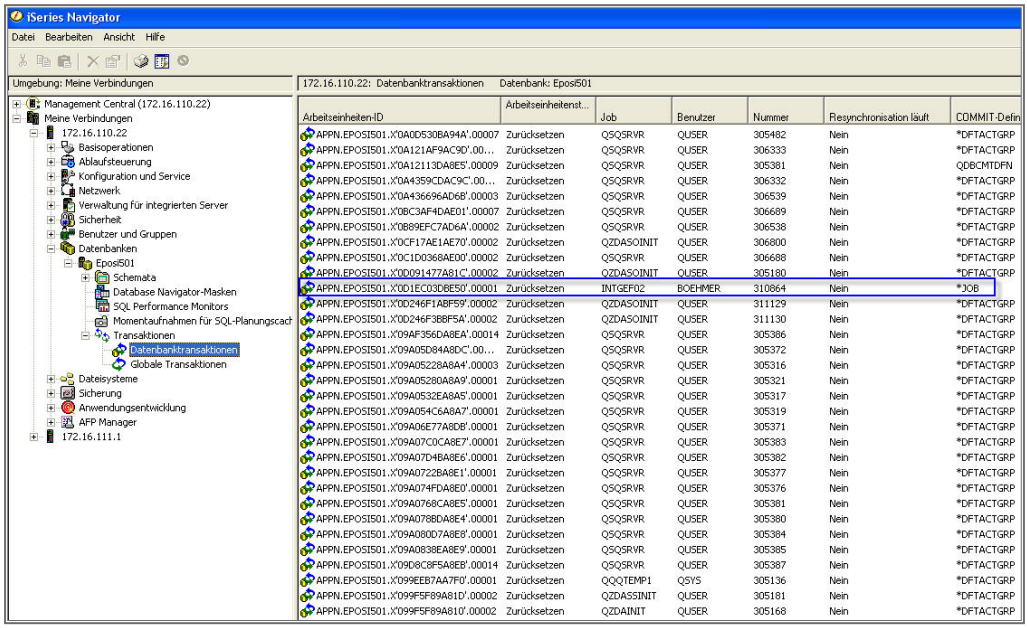
Nachdem das System den Befehl ausgeführt hat, erhalte ich folgende Anzeige:



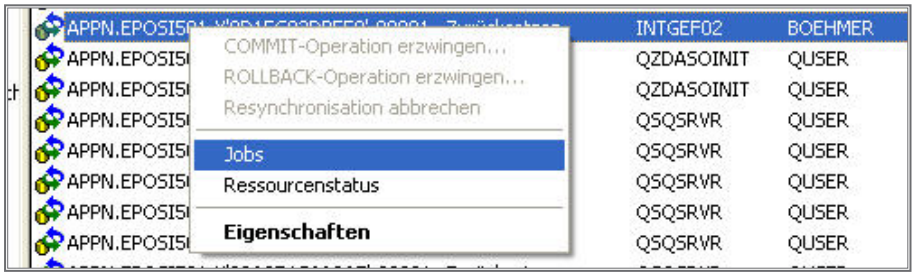
Auch über den iSeries Navigator besteht die Möglichkeit, mit Commit-Definitionen zu arbeiten. Dazu starten Sie den Navigator, expandieren den Punkt „Datenbanken“ und werfen einen Blick auf den Eintrag „Transaktionen“. Hier befinden sich die beiden Unterabteilungen „Datenbanktransaktionen“ und „Globale Transaktionen“. Dabei ist für uns die erste Unterabteilung interessant. Denn dort finden wir die verschiedenen Transaktionen, die sich zum aktuellen Zeitpunkt im System befinden.



Der Inhalt stellt sich wie folgt dar:

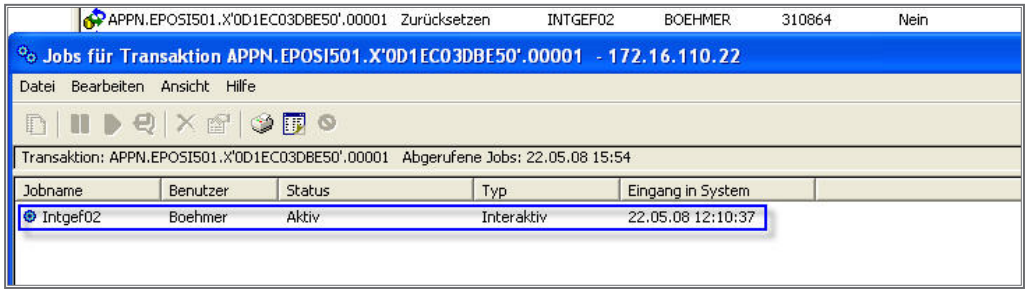


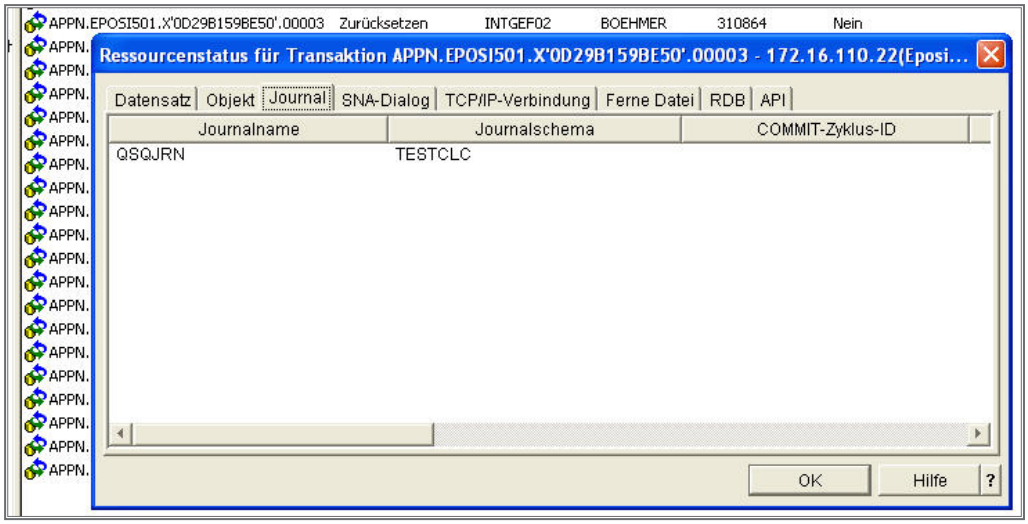
Die durch meinen Befehl generierte Commit-Definition wurde mit einem kleinen Rahmen hervorgehoben. Über das Kontextmenu dieses Eintrags lassen sich viele Informationen beschaffen – beispielsweise der zugeordnete Job.



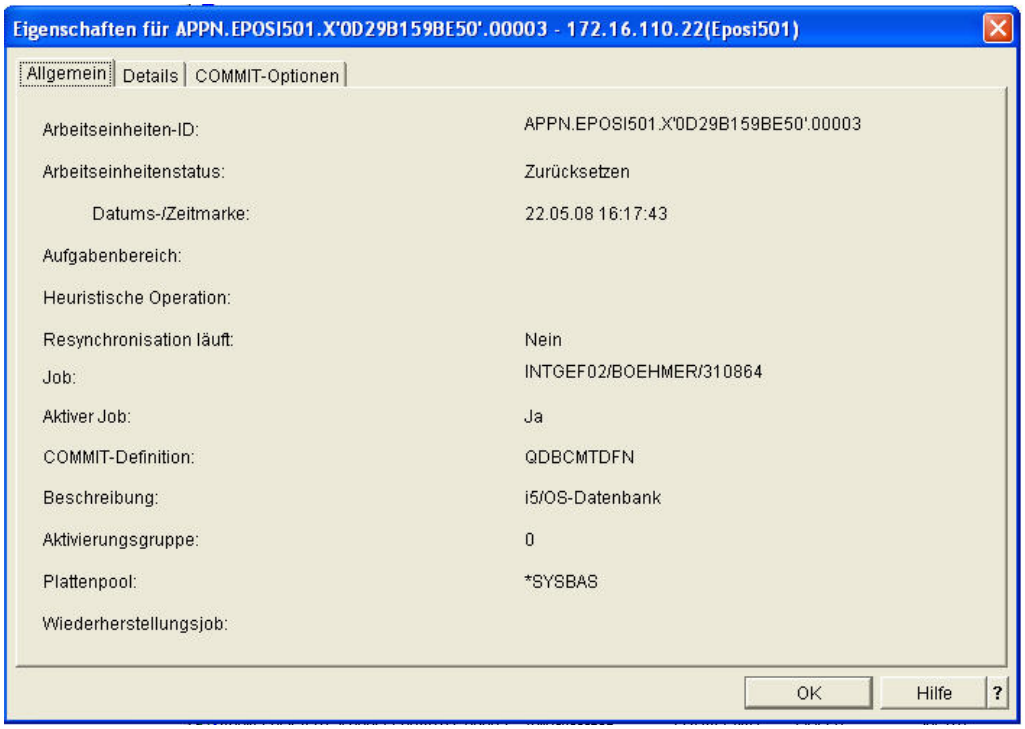
3.5.11.3
Seite 6

Das Kontextmenu bietet die Möglichkeit, mit den assoziierten Ressourcen zu arbeiten.

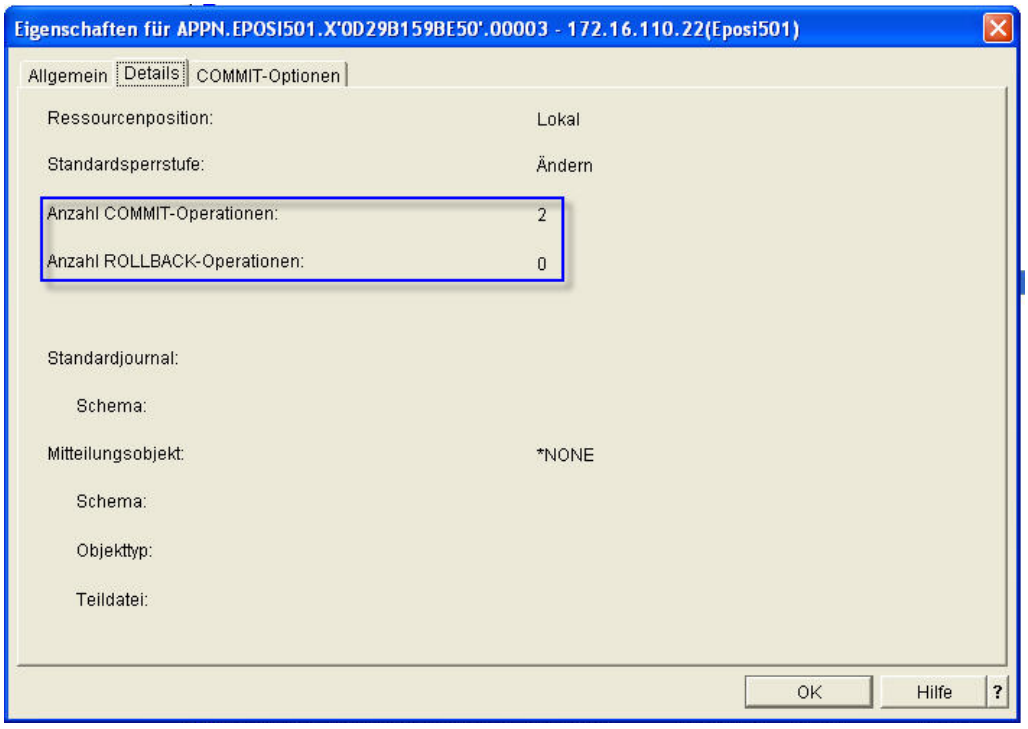




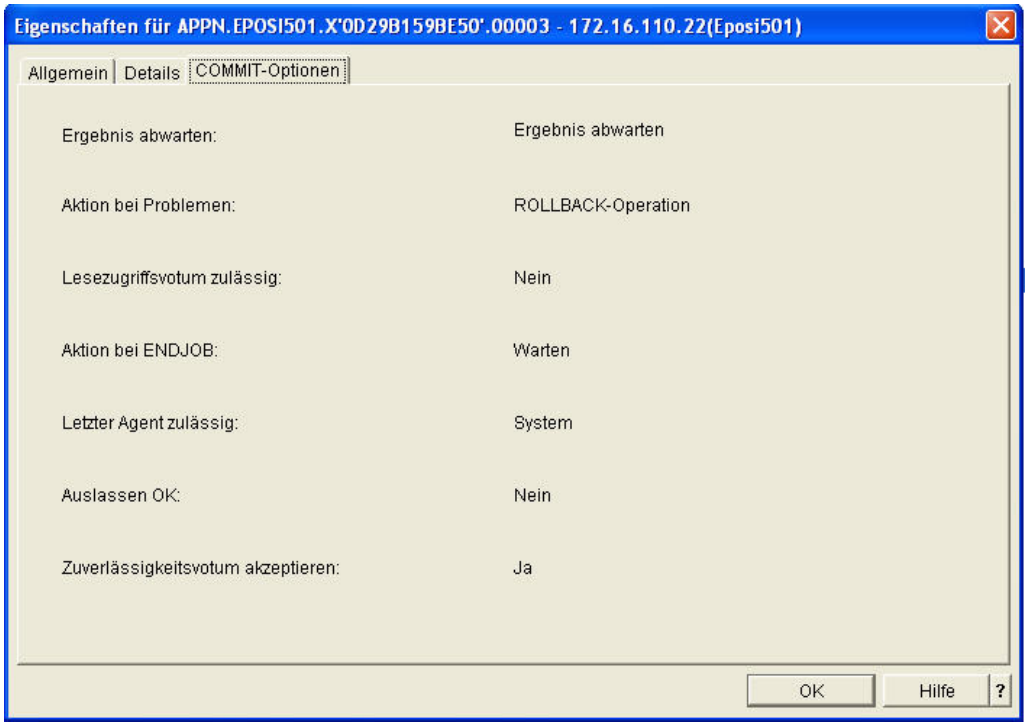
Schauen Sie noch auf die Eigenschaften, die ebenfalls über das Kontextmenu erreichbar sind:



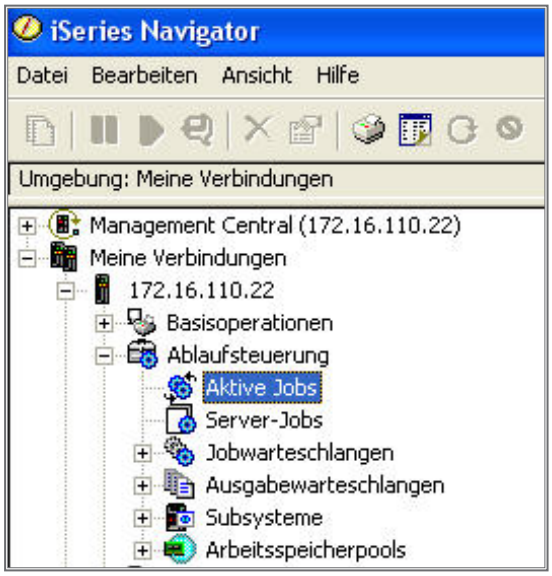
Zwischendurch habe ich via SQL in meiner 5250-Sitzung Daten geändert. Das spiegelt sich auch im Register „Details“ wieder, auf dem nun zwei Commit-Operationen gemeldet sind. Abschließend wird das Register „Commit-Optionen“ abgebildet.



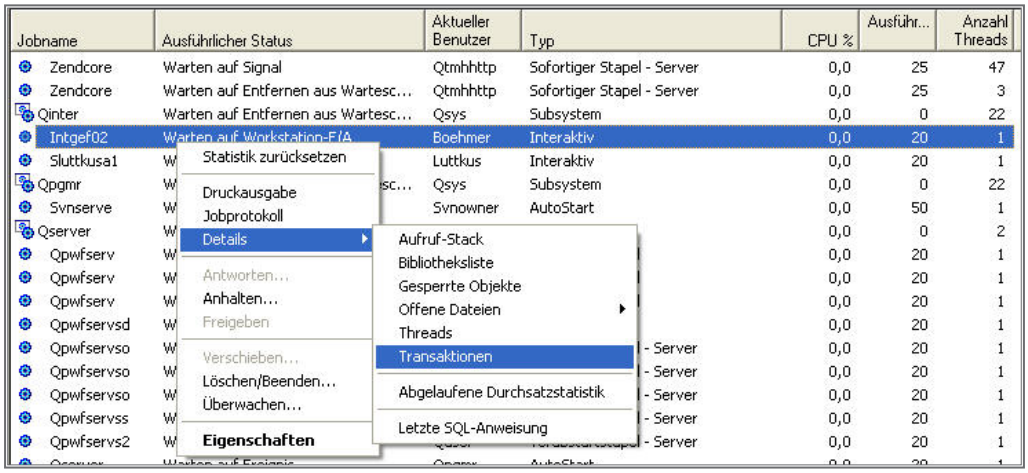
3.5.11.3
Seite 8



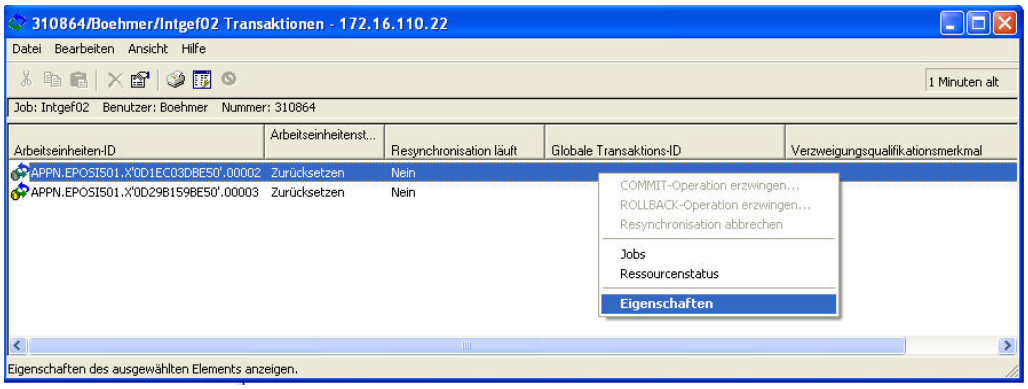
Welche Querverbindung hält der Navigator noch bereit, damit die Arbeit mit Transaktionen komfortabler wird? Um diese Frage zu beantworten, gehe ich in den Bereich der Ablaufsteuerung und schaue mir dort die aktiven Jobs an.



In der Auflistung der aktiven Jobs (siehe Detailanzeige auf der rechten Seite des Navigators) suche ich meinen interaktiven Job heraus und öffne das Kontextmenü.



Das Untermenü „Details“ bietet die Menüoption „Transaktionen“ an. In dem daraufhin angezeigten Fenster sehen wir zwei „Transaktionen“. Nach einem Klick mit der rechten Maustaste erhält man wieder ein Kontextmenü. Die vorhandenen Menüoptionen sind aus den bisherigen Erklärungen hinlänglich bekannt und werden nicht noch einmal beschrieben.



3.5.11.3
Seite 10

Die Funktionalität des iSeries Navigator ist hinreichend erklärt worden. Welche Optionen stellt aber die klassische „Grüne Oberfläche“ bereit? Nach meinen Aktionen in der Datenbank erscheint dort folgendes Bild:

```
                        Mit COMMIT-Definitionen arbeiten
                                                          System:   EPOSI501
Auswahl eingeben und Eingabetaste drücken.
  5=Status anzeigen   12=Mit Job arbeiten   14=COMMIT erzwungen
  16=ROLLBACK erzwungen

     COMMIT-                                                      Resynchr.
Opt  Definition  Job         Benutzer    Nummer  Thread     läuft
__   *JOB        INTGEF02    BOEHMER     310864  *NONE      NEIN
__   QDBCMTDFN   INTGEF02    BOEHMER     310864  *NONE      NEIN
                                                                    Ende
F3=Verlassen   F5=Aktualisieren   F9=Befehlszeile
F11=ID der logischen Arbeitseinheit anzeigen   F24=Weitere Tasten
```

Mit Funktionstaste „F11“ können Sie hier zwischen verschiedenen Sichten umschalten.

```
     COMMIT-
Opt  Definition  ID der logischen Arbeitseinheit          Status
__   *JOB        APPN.EPOSI501.X'0D1EC03DBE50'.00002     ZURÜCKSETZEN
__   QDBCMTDFN   APPN.EPOSI501.X'0D29B159BE50'.00003     ZURÜCKSETZEN
```

```
     COMMIT-
Opt  Definition  Text
__   *JOB        Jobebene
__   QDBCMTDFN   i5/OS-Datenbank
```

```
     COMMIT-     Globale              Verzweigungs-
Opt  Definition  Transaktions-ID      Quali.-Merkmal
__   *JOB
__   QDBCMTDFN
```

```
     COMMIT-
Opt  Definition  Status der Transaktionsverzweigung
__   *JOB
__   QDBCMTDFN
```

Betrachten wir nun die Auswahlmöglichkeiten der Maske. Dort besteht die Option, über Auswahl „12“ mit dem zugeordneten Job zu arbeiten.

```
Auswahl eingeben und Eingabetaste drücken.
  5=Status anzeigen   12=Mit Job arbeiten   14=COMMIT erzwungen
  16=ROLLBACK erzwungen

     COMMIT-                                                  Resynchr.
Opt  Definition  Job        Benutzer    Nummer  Thread       läuft
__   *JOB        INTGEF02   BOEHMER     310864  *NONE        NEIN
12   QDBCMTDFN   INTGEF02   BOEHMER     310864  *NONE        NEIN
```

Das im Anschluss gezeigte Jobmenü müsste bekannt sein!

```
Auswahlmöglichkeiten:

    1. Jobstatusattribute anzeigen
    2. Jobdefinitionsattribute anzeigen
    3. Jobausführungsattribute anzeigen, falls aktiv
    4. Mit Spool-Dateien arbeiten

   10. Jobprotokoll anzeigen, falls aktiv, in Jobwarteschlange oder anstehend
   11. Aufrufstapel anzeigen, falls aktiv
   12. Mit Sperren arbeiten, falls aktiv
   13. Bibliotheksliste anzeigen, falls aktiv
   14. Offene Dateien anzeigen, falls aktiv
   15. Dateiüberschreibungen anzeigen, falls aktiv
   16. COMMIT-Steuerungsstatus anzeigen, falls aktiv
                                                        Weitere ...
Auswahl oder Befehl
===> _
```

Über die Auswahlen „14“ und „16“ kann ein Commit- beziehungsweise Rollback erzwungen werden, wobei der iSeries Navigator diese Optionen bereits aktiviert oder deaktiviert hat – je nach Zustand. Letztendlich erhalten Sie über die Auswahl „5“ Informationen über den Ressourcenstatus.

```
     COMMIT-                                                  Resynchr.
Opt  Definition  Job        Benutzer    Nummer  Thread       läuft
__   *JOB        INTGEF02   BOEHMER     310864  *NONE        NEIN
5_   QDBCMTDFN   INTGEF02   BOEHMER     310864  *NONE        NEIN
```

3.5.11.3
Seite 12

```
                  Status der COMMIT-Definition anzeigen            EPOSI501
                                                        22.05.08  17:03:09
Job:   INTGEF02       Benutzer:   BOEHMER       Nummer:   310864

Thread . . . . . . . . . . . . . . :   *NONE
ID der logischen Arbeitseinheit . :   APPN.EPOSI501.X'0D29B159BE50'.00003
Sperrbereichs-ID . . . . . . . . . :   UDB_010000000007C05C
COMMIT-Definition . . . . . . . . . :   QDBCMTDFN
Aktivierungsgruppe . . . . . . . . :
ASP-Gruppe . . . . . . . . . . . . :   *SYSBAS

Ressourcenstandort . . . . . . . . :   KEINE
Standardsperrstufe . . . . . . . . :   *CHG
Funktion . . . . . . . . . . . . . :
Status . . . . . . . . . . . . . . :   ZURÜCKSETZEN
  Datums-/Zeitmarke . . . . . . . . :
Resynchronisation läuft . . . . . :   NEIN
Heuristische Operation . . . . . :
                                                                Weitere ...
Eingabetaste --> Weiter

F3=Verlassen      F5=Aktualisieren   F6=Ressourcenstatus anzeigen
F9=Befehlszeile   F12=Abbrechen
```

Hier kann geblättert werden, da nicht alle Daten auf eine Seite passen.

```
Aktiver Job . . . . . . . . . . . :   JA
Server-Job . . . . . . . . . . . . :
Sperrengeltungsbereich . . . . . :   *JOB
Zeitlimit für Transaktion . . . . :
Max. Wartezeit für Sperren . . . :
Anzahl COMMIT-Operationen . . . . :   2
Anzahl ROLLBACK-Operationen . . . :   0
                                                                Weitere ...
```

Über die Befehlstaste F6 gelangen Sie an den Ressourcenstatus.

```
                  Status der COMMIT-Definition anzeigen            EPOSI501
                                                        22.05.08  17:07:44
Job:   INTGEF02       Benutzer:   BOEHMER       Nummer:   310864

Thread . . . . . . . . . . . . . . :   *NONE
ID der logischen Arbeitseinheit . :   APPN.EPOSI501.X'0D29B159BE50'.00003
.....................................................................
:                        Ressourcenstatus anzeigen                  :
:                                                                   :
:  Auswahl eingeben und Eingabetaste drücken.                       :
:    1=Auswählen                                                    :
:                                                                   :
:  Ausw.     Ressource                                              :
:    =       Satzebene                                              :
:    _       Objektebene                                            :
:    _       Dialog                                                 :
:    _       TCP/IP-Verbindung                                      :
:    _       Ferne Datei                                            :
:    _       Relationale Datenbank (RDB)                            :
:    _       API                                                    :
:                                                          Weitere ...:
:  F5=Aktualisieren   F12=Abbrechen                                 :
:                                                                   :
:...................................................................:
```

Hier können Sie die Maske einmal mit den Möglichkeiten des Navigators vergleichen. Wenn Sie in dieser Anzeige weiter nach unten blättern, finden Sie den Eintrag „Journal“. Diesen Eintrag habe ich vor der folgenden Abbildung ausgewählt.

```
                          Ressourcenstatus anzeigen

  Auswahl eingeben und Eingabetaste drücken.
    1=Auswählen

  Ausw.      Ressource
    1        Journal
```

```
                            Journalstatus anzeigen
                                                                System:   EPOSI501
Job:    INTGEF02        Benutzer:    BOEHMER         Nummer:    310864

COMMIT-Definition  . . . . . . . . . . . :   QDBCMTDFN

Auswahl eingeben und Eingabetaste drücken.
  5=COMMIT-Zykluseinträge anzeigen   6=Alle Einträge anzeigen
  7=Mit Journalattributen arbeiten

                              ID
Opt  Journal      Bibliothek  COMMIT-Zykl.
 6   QSQJRN       TESTCLC     0

                                                                        Ende
F3=Verlassen      F5=Aktualisieren   F6=Ressourcenstatus anzeigen
F9=Befehlszeile   F11=ROLLBACK-Status anzeigen   F24=Weitere Tasten
```

3.5.11.3
Seite 14

Ich trage dort die Auswahl „6“ ein, damit ich mir die aufgelaufenen Journaleinträge anschauen kann.

Beachten Sie die letzten drei Einträge mit dem Code „C“; bei diesen handelt es sich um Einträge unter der Transaktionssteuerung.

Code BC bedeutet Beginn der Commit-Steuerung; unter SC wird der Commit-Zyklus gestartet und CM liefert den Hinweis, dass eine Änderung festgeschrieben wurde.

Werfen wir noch über die Auswahl „5“ einen Blick auf die Details eines solchen Eintrags:

```
                    Details der Journaleinträge anzeigen

 Journal . . . . . :   QSQJRN          Bibliothek . . . . . :   TESTCLC

 Folge . . . . . :   6
 Code . . . . . :   C  - COMMIT-Steuerungsoperation
 Art . . . . . :   CM - COMMIT-Operation

 Objekt . . . . . :
   Art . . . . . :
 Datum . . . . . :   22.05.08
 Uhrzeit . . . . . :   16:02:15
 Markierung . . . . . :   0
 Anzahl/RRN . . . . . :   0
 COMMIT-Zyklus-ID . . :   5
 Versch. COMMIT-Stufe :   0
 Job . . . . . :   310864/BOEHMER/INTGEF02
 Benutzerprofil . . . :   BOEHMER
 APY/RMV ignorieren . :   Nein
 Ref. Int. Beding. . :   Nein
                                                                Weitere ...
 F3=Verlassen   F10=Eintrag   F12=Abbrechen   F14=Vorheriger Eintrag
 F15=Eintragspezifische Daten
```

Zusätzlich können Sie auch durch die Informationen blättern. Das überlasse ich gerne Ihnen – für Übungszwecke. Möchten Sie Näheres über Verzweigungsoptionen oder über diverse Funktionstasten dieser Maske erfahren, dann verweise ich auf die Abschnitte über die „Journalisierung“.

3.5.11.3

Seite 16

XML und System i

IBM hat XML bereits vor einigen Jahren auf der iSeries bzw. im System i integriert. Seither besteht im Prinzip die Möglichkeit, XML-Funktionen mit den auf dem System i5 existierenden Datenbanken zu nutzen.

Neben den Unterstützungen für XML, die in der Datenbank DB2/UDB des System i integriert sind, steht uns bei Bedarf ein größerer Funktionsumfang durch den Einsatz von Lizenzprogrammen oder auch APIs zur Verfügung.

Bis zum Erscheinen der Version i7.1 war XML allerdings eher spärlich in System i-Bereiche integriert. In früheren Releases mussten System i-Betreiber in der Regel auf Zusatzprodukte zurückgreifen oder manuell XML-Funktionen erstellen, um den Datenaustausch mit anderen Plattformen oder beispielsweise die Anbindung von Internetanwendungen realisieren zu können.

Seit i7.1 stehen nun deutlich mehr XML-Funktionen zum Einsatz bereit und sie sind besser im Basisbetriebssystem bzw. der Datenbank integriert. Eine allumfassende Integration, wie sie andere DB2-Varianten bieten, ist damit zwar noch nicht umgesetzt, aber die Funktionalität ist nun doch sehr gut nutzbar.

Betreiber von Maschinen mit einem älteren Releasestand müssen aber nicht auf die Nutzung von XML verzichten. Zusatzprodukte – entweder in Form von IBM-Lizenzprogrammen oder aber von Drittanbietern – bieten ebenfalls Möglichkeiten, XML und AS/400, iSeries und System i mit XML zusammenzubringen. Ein wichtiges Lizenzprogramm ist in diesem Zusammenhang das sogenannte XML-Toolkit, das in Form des Lizenzprogramms 5722-XT1 angeboten wird. Dieses Lizenzprogramm besteht aus mehreren Optionen mit unterschiedlichen Funktionsumfängen und bietet in Releaseversionen vor 7.1 XML-Support für i.

Die Basisfunktionen, die uns bis zur Version i7.1 für die Integration von XML angeboten wurden, hatten klare Grenzen. Diese Grenzen wurden dann sichtbar, wenn es beispielsweise darum ging, den Datenaustausch mit anderen Systemen zu „standardisieren“ oder auch wenn die XML-Dokumente direkt auf dem System i gespeichert werden sollten. Diese Nachteile sind nunmehr in vielen Bereichen Geschichte.

Vergleicht man den Inhalt der XML-Dokumente mit Bestandteilen der DB2, dann entsprechen diese in etwa den bisherigen klassischen Ansätzen der Datenhaltung auf dem System i, also physischen und logischen Dateien. Aufgrund des Aufbaus der XML-Inhalte lassen sich XML-Dokumente so verarbeiten, dass ein zielgerichteter Zugriff auf die Inhalte möglich ist. Dabei werden die Formate der XML-Dokumente genutzt und beispielsweise für SQL-Abfragen verwendet.

3.6
Seite 2

IBM hat bei der Weiterentwicklung von RPG auch die Nutzung von XML nicht vernachlässigt. So finden wir in den aktuellen Versionen von RPG neue Funktionen, mit denen der Einsatz von XML stark vereinfacht wurde – ein weiterer Anreiz, XML in Verbindung mit bestehenden Anwendungen zu nutzen. Gerade wenn es um den Datenaustausch geht – denken wir z. B. an den elektronischen Auftragseingang oder die Übermittlung von Bestelldaten – dürften die Abläufe dazu nicht selten in RPG geschrieben sein. Durch die bessere Implementierung von XML direkt in RPG lassen sich die Anforderungen schneller und auch einfacher realisieren.

XML-Integration in DB2 (ab Version i7.1)

XML hat sich in den letzten Jahren als Standard immer weiter verbreitet. Nach anfänglich eher begrenzten Möglichkeiten, XML mit IBM i direkt verarbeiten zu können, hat IBM mit dem Erscheinen der Version 7.1 der DB2/UDB und des Betriebssystems IBM i einige Erweiterungen umgesetzt, die den Umgang mit XML deutlich vereinfachen.

XML ist mittlerweile eine feste Säule für den Einsatz von SOA (**Service Oriented Architecture**) -basierten Anwendungen und gewinnt schon aus diesem Grund nicht nur in Verbindung mit dem System i immer weiter an Bedeutung. Einer der wesentlichen Vorteile der XML-Architektur liegt somit in der Möglichkeit des vereinfachten Datenaustauschs über Plattformen hinweg.

Die Erweiterungen bestehen u. a. in der Bereitstellung eines neuen XML-Datentyps, mit dem das Speichern von XML deutlich erleichtert wird. Im Bereich SQL lassen sich nun (endlich) auch Funktionen einsetzen, um aus DB2-Datenbankobjekten XML-Werte zu erzeugen. IBM hat die neuen XML-Werkzeuge und -Funktionen basierend auf den ISO- und ANSI-Standards im System verankert.

Schauen wir uns die wesentlichen XML-Erweiterungen in Version 7 einmal genauer an.

Der bereits erwähnte XML-Datentyp ist sicher die Grundlage für die weiteren XML-Neuerungen.

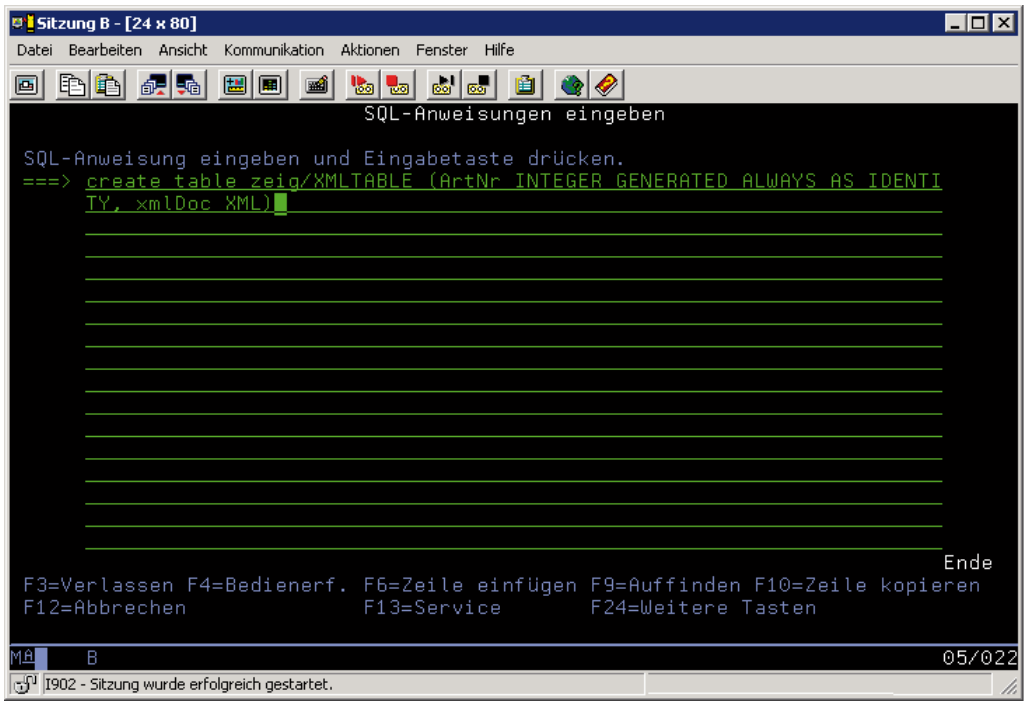
Anmerkung:

In einem weiteren Kapitel habe ich Ihnen den XML Extender vorgestellt. Dieses spezielle Lizenzprogramm der IBM bietet bereits seit einigen Jahren XML-Funktionen für den Einsatz mit den DB2-Versionen der früheren IBM i Releases an. Mit den neuen nativen XML-Funktionen ab Version 7.1 verfolgt IBM das Ziel, den Funktionsumfang von XML Extender mehr und mehr in Standardbereiche der Datenbank bzw. des Betriebssystems zu verlagern.

Eine der wesentlichen Neuerungen dabei ist der neue Datentyp XML. Dieser Datentyp kann sowohl in neuen als auch in bestehenden SQL-Tabellen zum Einsatz kommen.

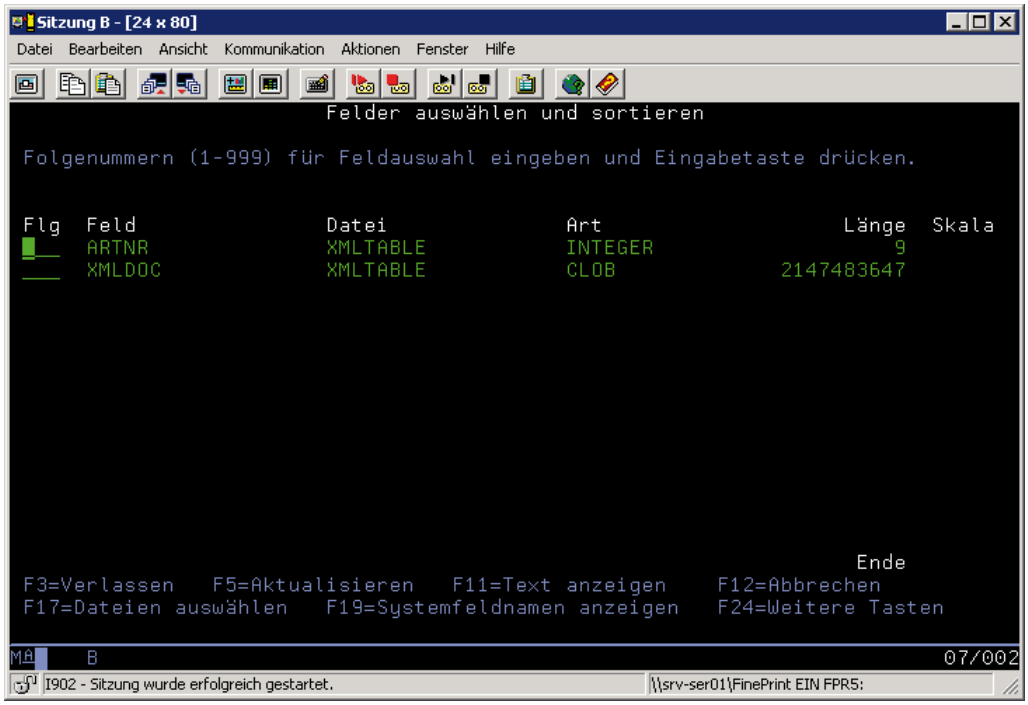
3.6.1
Seite 2

Die nachfolgende Abbildung zeigt exemplarisch die Definition eines XML-Datentyps.



XML-Spaltendefinition in SQL

Nach dem Erstellen einer SQL-Tabelle mit einem XML-Wert erhalten wir die folgenden Spaltendefinitionen:

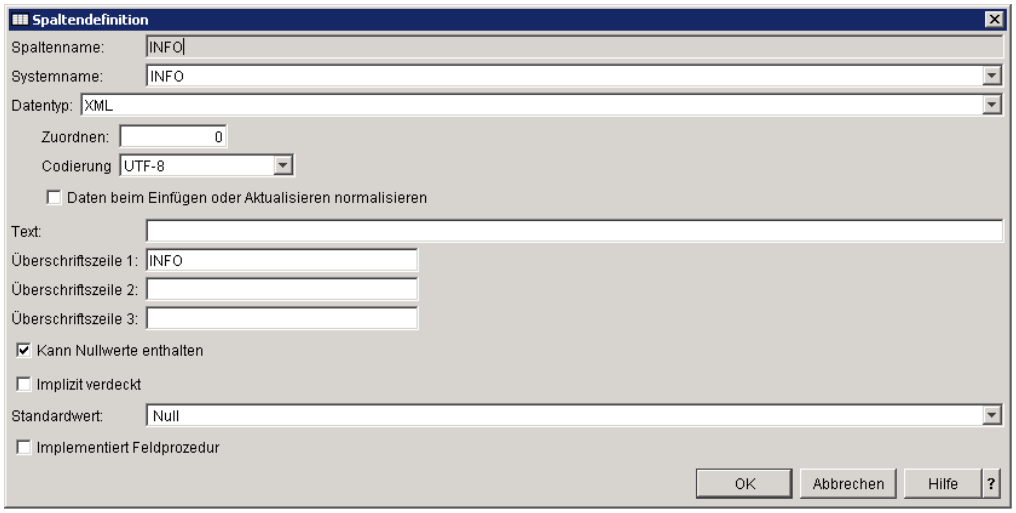


XML Spaltendefinitionen

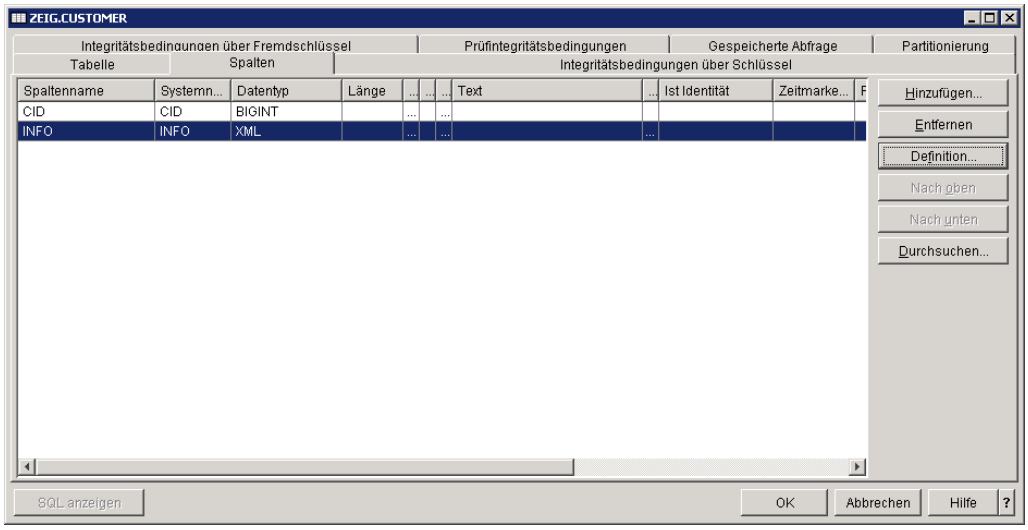
Bei der neuen Spalte „XMLDOC“ handelt es sich um den neuen Datentyp. Er wird in den SQL-Werkzeugen mit der Spaltenart CLOB dargestellt. Auch wenn in dieser Anzeige der Spalten die XML-Spalte „XMLDOC“ mit einer Länge dargestellt wird, so ist diese nicht beeinflussbar. Das heißt, dass wir anders als beispielsweise bei Zeichenspalten keine Länge angeben können. Die Spaltengröße für eine Spalte der Art XML wird stets mit 2 GB maximaler Länge definiert.

Die Zuordnung des benötigten Speichers richtet sich dann aber nach dem tatsächlichen Bedarf, denn die Speicherzuordnung erfolgt für XML-Felder dynamisch.

Natürlich können wir ab Version 7.1 auch den XML-Datentyp mit modernen Werkzeugen wie z. B. mit dem System i Navigator verarbeiten. Die folgende Abbildung zeigt die Darstellung einer Spalte mit XML-Definition im System i Navigator:



Spalteninformationen in System i Navigatior



Spaltendefinition im System i Navigator

Der neue Spaltentyp ‚XML" steht in SQL-basierten Datenbankkomponenten zur Verfügung und lässt sich nicht mit DDS-Mitteln erzeugen bzw. verarbeiten. SQL-Werkzeuge sind hier die erste Wahl (und sollten es, beachtet man die Empfehlungen der IBM, auch für andere Datenbankbereiche sein). Natürlich lassen sich die Informationen der XML-Spalten auch in allen herkömmlichen Sprachen wir RPG, COBOL und C einsetzen. Embedded SQL ist hier die Technik, die sich besonders für den Einsatz mit XML-Komponenten anbietet.

Jede Spalte des Datentyps XML beinhaltet für jede Zeile der Tabelle ein wohlgeformtes XML-Dokument. Für das Definieren der XML-Daten benötigen wir kein XML-Schema. Mit dem Einfügen der XML-Informationen in die betreffende Spalte wird das Parsing für die XML-Daten durchgeführt. Das automatische Parsen greift bei allen INSERT- oder UPDATE- Funktionen, für die nicht XMLPARSE angegeben wurde. Auf diese Weise wird sichergestellt, dass der XML-Inhalt wohlgeformt im XML-Spaltenbereich abgelegt wird.

Anmerkung:

Für Leser, die noch wenig Kenntnisse im Bereich XML haben: Unter Parsen versteht man die Umsetzung der XML-Daten vom String-Format in das Format des XML-Dokuments. Es handelt sich quasi um das Auslesen der Daten und das Umsetzen ins Zielformat.

IBM hat die XML-Funktionen der DB2/UDB mit zwei verschiedenen Parsing-Methoden ausgestattet: Dem expliziten und dem impliziten Parsen. Unter dem expliziten Parsen verstehen wir den Aufruf der Funktion XMLPARSE während des Einspielens der XML-Daten. Das folgende Codebeispiel zeigt exemplarisch die Verwendung des expliziten Parsens:

```
INSERT INTO KUNDE (KDNR, Info)
 VALUES (?, XMLPARSE(DOCUMENT CAST(? AS CLOB(1K))
PRESERVE WHITESPACE))
```

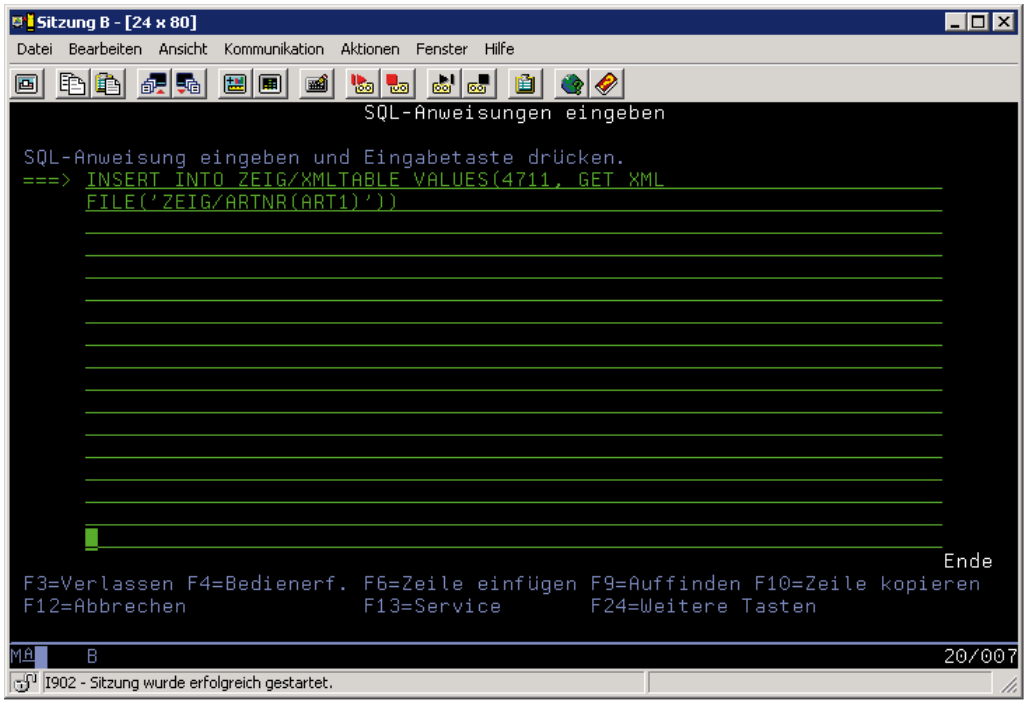
Dem gegenüber steht das implizite Parsen. Diese Form des Parsens wird dann ausgeführt, wenn XML-Daten ohne weitere Parse-Angaben übertragen werden sollen.

In beiden Fällen kann festgelegt werden, wie mit Whitespace umgegangen werden soll. Whitespace ist ein Überbegriff für alle Formen von Leerstellen bzw. auch für solche Elemente im einem Feldwert, die nicht dargestellt werden sollen. Beispiele sind Zeilenumbrüche, Tabulatoren etc. In der Regel müssen oder dürfen diese Steuerzeichen nicht in die nativen Daten der DB2 übernommen werden. Mit einem entsprechenden Vorgehen können wir sicherstellen, dass nur die reinen Daten ohne Steuerelemente in die DB2 übertragen und eingestellt werden.

Derzeit werden mit dem XML Parsing die folgenden Steuerzeichen behandelt:
- Leerstellen (U+0020)
- Carriage Return (U+000D)
- Linefeed (U+000A)
- Tabulatoren (U+0009)

Basis für das Parsing ist die CCSID 1208.

Die folgende Abbildung zeigt exemplarisch das implizite Parsen zum Einfügen neuer Informationen in unsere XML-Tabelle:



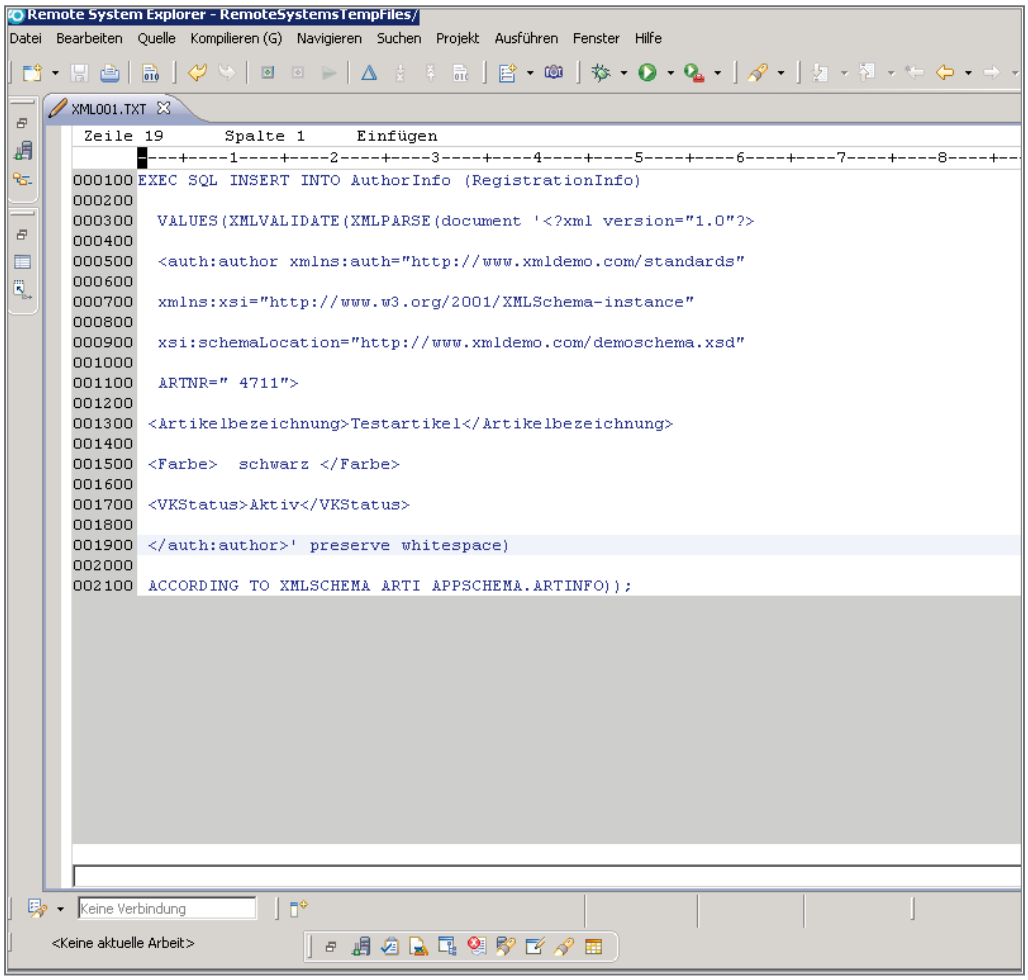
Implizites Parsing

3.6.1
Seite 6

Beim impliziten Parsen müssen wir uns nicht um die Prüfung der XML-Inhalte kümmern. Diese Prüfungen werden vom System durchgeführt. In unserem Beispiel haben wir die Funktion GET_XML_FILE verwendet. Diese Funktion liefert die Daten von einer Source Stream File oder von einer physischen Datei bzw. Tabelle. Mit der Funktion GET_XML_FILE wird neben dem Lesen des gewünschten Datensatzes auch die Umsetzung in UTF8 durchgeführt. Sollte in der Datei keine XML-Deklaration enthalten sein, dann übernimmt die Funktion auch diese Aufgabe und fügt die XML-Deklaration hinzu.

Grundsätzlich sollte beim Ausführen der SQL-Funktionen mit XML-Komponenten die COMMIT-Steuerung aktiviert werden. Nur dann vermeidet man unnötige Probleme!

Wem dieses Vertrauen in die XML-Standardfunktionen nicht genügt, dem steht es natürlich frei, die Prüfung der XML-Informationen durchzuführen.

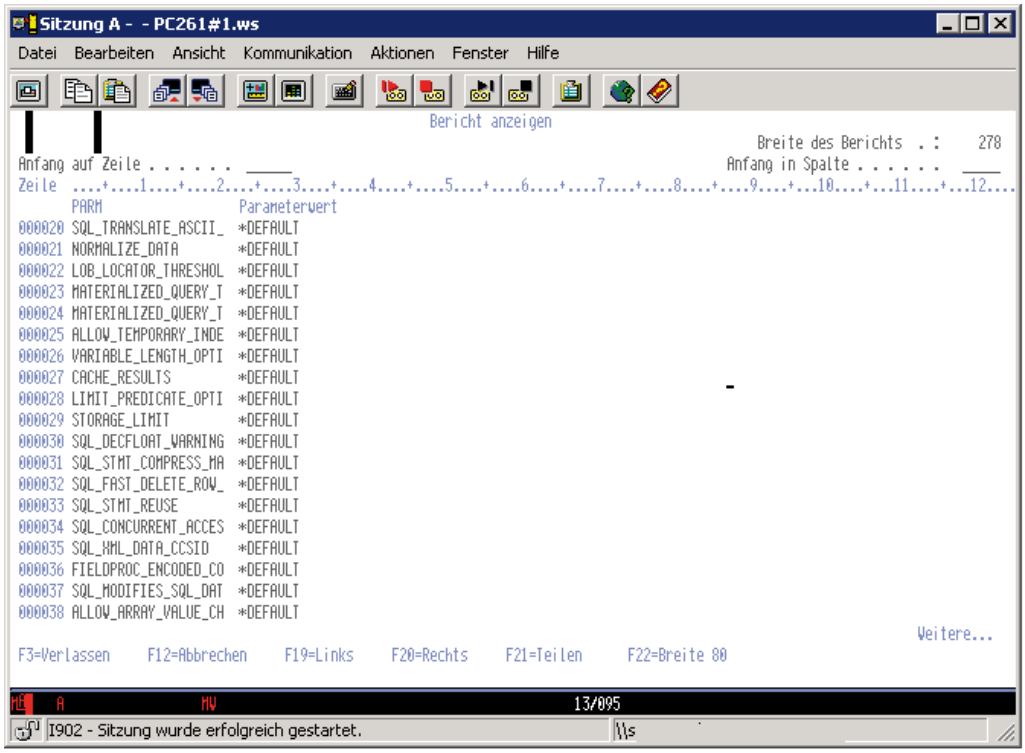


Schema Validierung

XML-Kenner sind es bereits gewohnt: Die Tatsache, dass auf Spaltenebene bei Bedarf die CCSID angegeben werden kann. Diese Angabe kann z. B. beim plattformübergreifenden Datenaustausch mit XML sinnvoll sein, sorgt sie doch für eine passende Umsetzung von Sonderzeichen. Wird auf der XML-Spaltenebene keine CCSID angegeben, so wird hier als Standardwert die CCSID 1208 verwendet. 1208 ist die Umsetzung in UTF8-Format. Dieses Format wird auch unabhängig von der Definition der Spalten CCSID beim Parsing verwendet. UTF8 (Unicode) ist hier u. a. in Bezug auf die Ausführungsgeschwindigkeit von wesentlicher Bedeutung!

Anmerkung!

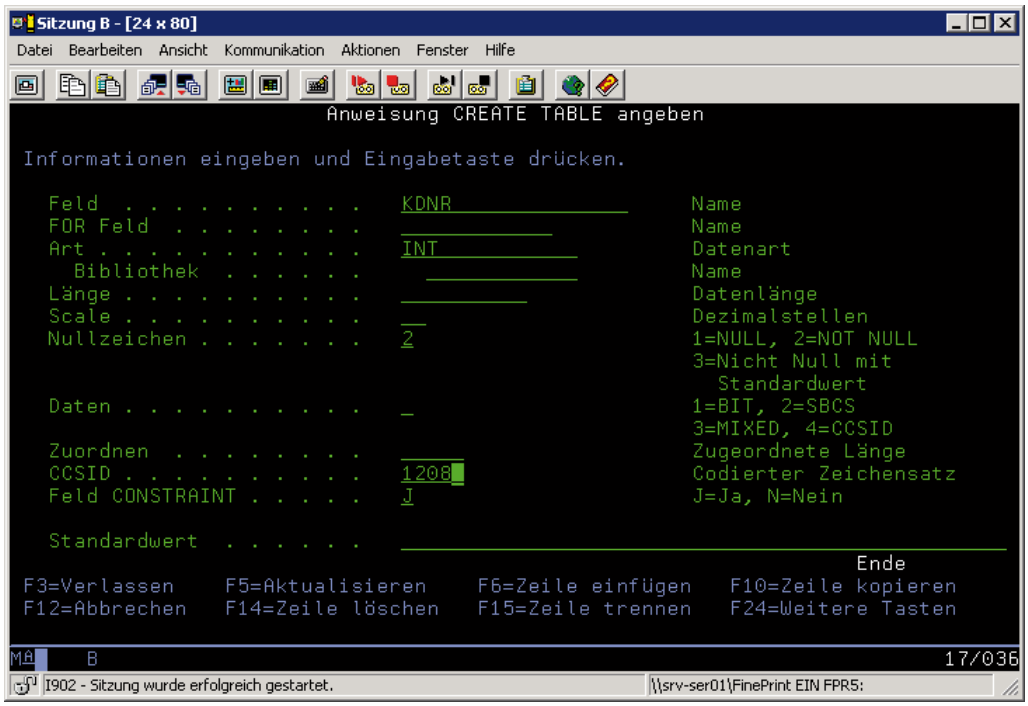
Die Standardeinstellung der CCSID richtet sich nach der QAQQINI-Einstellung SQL_XML_DATA_CCSID. Dieser Wert ist im Auslieferungsstand mit 1208 vorbelegt. Wurde der Wert auf Ihrem System geändert, dann wird diese Einstellung als Standard für die Anlage von XML-Spalten genutzt!



QAQQINI

3.6.1
Seite 8

Die folgende Abbildung zeigt die Angabe der CCSID auf Spaltenebene im Bereich der 5250-SQL-Eingabe:



CCSID auf Feldebene

Die Verwendung des neuen XML-Datentyps ist variabel gehalten. Das bedeutet, dass der XML-Datentyp durchaus auch mit anderen Datentypen in einer Tabelle existieren kann. Diese Mischung von XML- und Nicht-XML-Informationen stellt in der Regel auch für die Anwendungsprogramme kein Problem dar. Voraussetzung ist hier natürlich der korrekte Umgang mit den XML-Informationen – aber dazu später mehr.

Der neue XML-Datentyp ist in vielen Bereichen des Systems einsetzbar.

Die Basis für die Nutzung des XML-Datentyps bildet SQL. Die folgenden Programmiersprachen sind derzeit dafür ausgelegt, den neuen XML-Datentyp auf dem System i nutzen zu können:

- ILE RPG
- ILE COBOL
- C und C++
- Java

Wenn Sie mit dem neuen XML-Datentyp arbeiten wollen, dann sollten auch die Werkzeuge für die Erstellung von SQL, Quellcodes und Ähnlichem in der Lage sein, den neuen Datentyp verarbeiten zu können. RDp und System i Navigator sind Beispiele für solche Anwendungen, die in den aktuellen Versionen natürlich auch Unterstützung für den XML-Datentyp bieten.

Es gibt noch einige weitere Vorteile: Mit dem neuen Datentyp XML braucht der Betreiber der Datenbank sich keine zusätzlichen Gedanken darüber zu machen, ob und wie die XML-Daten gesichert werden. Mit der Integration der XML-Informationen in den Datenbankkomponenten werden diese bei allen gängigen Sicherungs- und Wiederherstellungsbefehlen mitberücksichtigt.

3.6.1
Seite 10

Das XML-Schema-Repository

Die XML-Schema-Beschreibungen sind seit Version i7.1 wichtige Komponenten beim Einsatz von XML. IBM hat für das XML-Schema-Repository einen neuen Objekttyp erzeugt: *SQLXSR. Parallel zu dem neuen SQLXSR-Objekttyp wurden neue SQL-Kataloge für die Nutzung mit den XML-Schemas bereitgestellt. Die Speicherung der XML-Schemas erfolgt in einem binären Format. Das hat den Vorteil, dass Validierungen mit XML-Funktionen performant ausgeführt werden.

Das XML-Schema wird auch für Umsetzungen verwendet, die von dem XML-Schema abgeleitet werden.

ACHTUNG!

Das XML-Schema ist NICHT zu verwechseln mit einem SQL-Schema!!!!

Bevor wir XML-Informationen verwenden können, muss ein XML-Schema definiert und für den Einsatz vorbereitet werden. Das Vorbereiten differenziert sich in zwei Hauptschritte:

- Erstellen eines XML-Schemas
  a. XSR_REGISTER
- Registrieren des erstellten XML-Schemas:
  b. XSR_COMPLETE
    i. Dieser Teilschritt sollte wirklich nur dann ausgeführt werden, wenn alle XML-Schemas registriert wurden.

Diese beiden Schritte können wir mit Hilfe von Stored Procedures umsetzen, die von IBM für die Registrierung von XML-Schemas bereitgestellt werden. Beide befinden sich in der Bibliothek SYSPROC, der Systembibliothek für DB2.

Hier die wichtigsten Stored Procedures, die im Zusammenhang mit XML und DB2 zum Einsatz kommen:

XSR_REGISTER
Mit dieser Prozedur wird das Registrieren eines XML-Schemas gestartet. Diese Prozedur wird beim initialen Hinzufügen eines XML-Schema-Dokuments zu einem XML-Schema aufgerufen.

XSR_ADDSCHEMADOC
Mit dieser Prozedur können weitere XML-Schema-Dokumente zu einem XML-Schema hinzugefügt werden. Dieses XML-Schema muss sich für diese Prozedur im Prozess der Registrierung befinden. XSR_ADDSCHEMADOC kann nur nach dem Aufruf von XSR_REGISTER und vor XSR_COMPLETE ausgeführt werden. Wird XSR_ADDSCHEMADOC nicht innerhalb dieser Befehlsreihenfolge ausgeführt, schlägt der Aufruf fehl.

3.6.1.1
Seite 2

XSR_REMOVE
Mit der Prozedur XSR_REMOVE lassen sich alle Komponenten eines XML-Schemas entfernen.

XSDBCOMPXML
XSDBCOMPXML wird für das Umsetzen von XML-Daten in DB2-Tabellen verwendet.

Das XML-Schema findet in unterschiedlichen Bereichen Verwendung und wird u. a. in der XSD-Definition angegeben.

Wenn ein XML-Schema nicht mehr benötigt wird, dann kann es mit dem Aufruf der Prozedur SYSPROC.XSR_REMOVE entfernt werden. Alternativ kann das Entfernen auch mit der SQL-Anweisung DROP XSROBJECT durchgeführt werden.

XML und DB2 – ein Beispiel

Der Prozess des Imports von XML-Daten in die Datenbank von IBM i ist seit Version 7.1 deutlich vereinfacht worden.

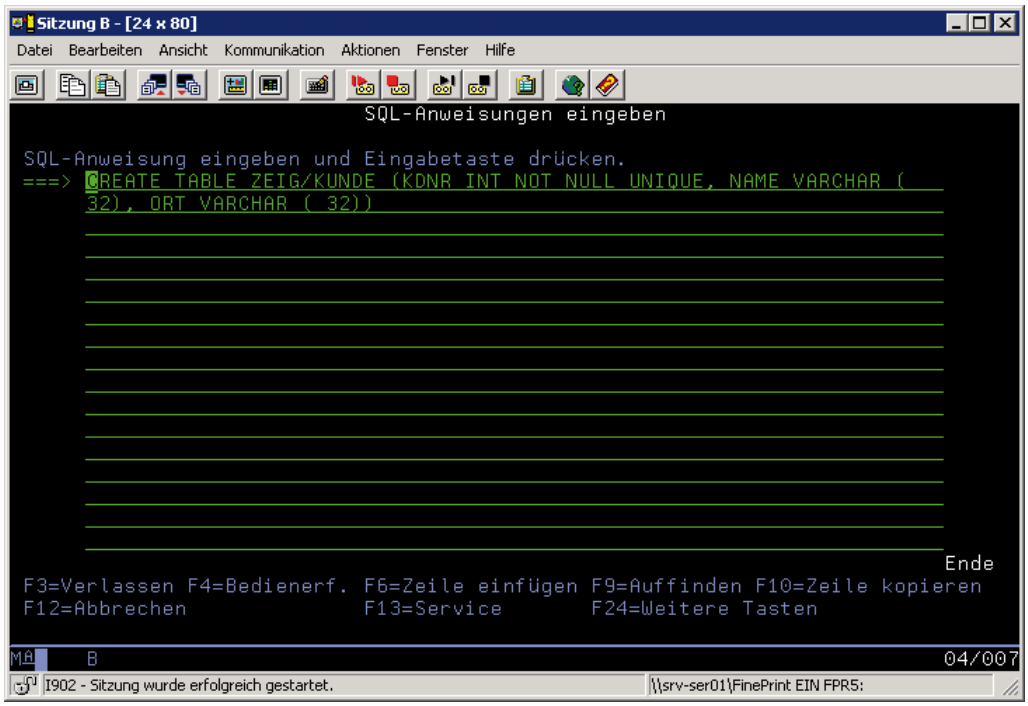
Im nachfolgenden Beispiel möchte ich Ihnen zeigen, wie XML-Informationen direkt in die DB2/UDB importiert werden können. Dabei wollen wir ein einfaches Beispiel anhand eines Kundenstamms durchspielen. Der Kundenstammsatz liegt dabei in Form einer XML-Datei vor. Der Inhalt ist in die zugehörige DB2-Tabelle zu importieren. Zur Veranschaulichung legen wir auch die erforderliche DB2-Tabelle „KUNDE“ an.

Die folgende Abbildung zeigt ein einfaches XML-Beispiel für einen Kunden:

```
<?xml version="1.0" encoding="UTF-8"?>
<kunde kdnr="4711">
 <name>Willi Winzig</name>
 <ort>Frankfurt</ort>
</kunde>
```

Diese einfachen XML-Informationen wollen wir nun in eine Tabelle auf dem System i einspielen.

Zunächst erstellen wir uns die passende Tabelle:



SQL-Tabelle KUNDE erstellen

3.6.1.2
Seite 2

Für das Importieren der XML-Daten bietet sich die Verwendung eines XML-Schemas an. Dieses XML-Schema erstellen und registrieren wir in einem Schema-Repository. Mit Hilfe dieses Schemas wird dann der Inhalt des XML-Dokuments geprüft. Hinter der Beschreibung verbirgt sich in XML-„Deutsch“ die XSD – also die Beschreibung des Aufbaus der XML-Informationen.

Nachfolgend finden Sie ein Beispiel für solch ein Schema:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
 xmlns:db2-xdb="http://www.ibm.com/xmlns/prod/db2/xdb1">

 <xs:annotation>
  <xs:appinfo>
   <db2-xdb:defaultSQLSchema>"ZEIG"</db2-xdb:defaultSQLSchema>
  </xs:appinfo>
 </xs:annotation>

 <xs:element name="kundenstamm">
  <xs:complexType>
   <xs:sequence>
    <xs:element name="kunde" maxOccurs="unbounded">
     <xs:complexType>
      <xs:sequence>
       <xs:element name="name" type="xs:string" maxOccurs="1"
        db2-xdb:rowSet="KUNDE" db2-xdb:column="NAME" />
       <xs:element name="Ort" type="xs:integer" maxOccurs="1"
        db2-xdb:rowSet="KUNDE" db2-xdb:column="ORT" />
      </xs:sequence>
      <xs:attribute name="id" type="xs:integer" use="required"
       db2-xdb:rowSet="KUNDE" db2-xdb:column="ID" />
     </xs:complexType>
    </xs:element>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
</xs:schema>
```

Das Schema stellen wir ins IFS. In diesem Beispiel stelle ich die XSD in das Home-Verzeichnis. Der Name der XSD lautet KUNDE.XSD.

KUNDE.XSD im IFS

Nachdem wir die XSD angelegt haben, muss sie registriert werden. Dazu verwenden wir die Prozedur XSR_REGISTER. Die IBM-Dokumentationen sind hier und da ein wenig verwirrend – wird doch auf einen CALL verwiesen. Allerdings sollte dieser in ein SQLRPGLE eingebunden werden, damit er auch erfolgreich ausgeführt werden kann. Bei XSR _REGISTER handelt es sich um eine von IBM bereitgestellte Stored Procedure!

Folglich können wir diese Stored Procedure durch das Erstellen einer Prozedur ausführen oder den Code beispielsweise in ein RPG-Programm einbinden.

Ein Beispiel für die Erstellung des SQLXSR-Objekts für unsere XML-Daten finden Sie hier:

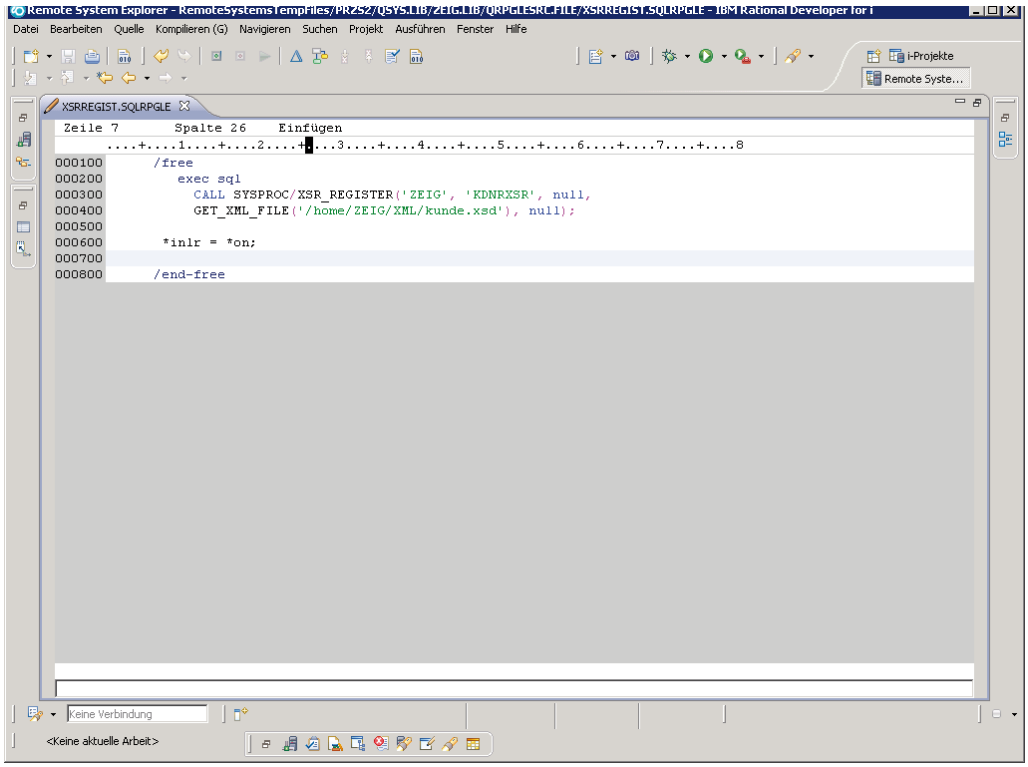
```
/free
   exec sql
     CALL SYSPROC/XSR_REGISTER('ZEIG', 'KDNRXSR', null,
     GET_XML_FILE('/home/ZEIG/XML/kunde.xsd'), null);

 *inlr = *on;

/end-free
```

Anmerkung:

Die Angabe des qualifizierten Objektnamens (im Beispiel QSYSPROC/XSR_REGISTER) kann in Abhängigkeit von den Systemeinstellungen unterschiedlich sein. Sollte bei der Eingabe bzw. dem Kompilieren ein Fehler auftreten, dann ersetzen Sie das Trennzeichen (/) durch einen Punkt (.).

3.6.1.2
Seite 4

Diesen Quellcode habe ich in der Source XSRREGIST gespeichert.



XSLREGIST

In dem Beispiel haben wir die Funktion GET_XML_FILE verwendet. Diese Funktion liefert die Daten von einer Source Stream File oder von einer physischen Datei bzw. Tabelle. Mit der Funktion GET_XML_FILE wird neben dem Lesen des gewünschten Datensatzes auch die Umsetzung in UTF8 durchgeführt. Sollte in der Datei keine XML-Deklaration enthalten sein, dann übernimmt die Funktion auch diese Aufgabe und fügt die XML-Deklaration hinzu.

```
CALL SYSPROC.XSR_REGISTER('ZEIG', 'KDNRXSR',
NULL, GET_XML_FILE('/home/ZEIG/XML/kunde.xsd'), NULL)
```

Bei den ersten beiden Parametern von XSR_REGISTER handelt es sich um die Bibliothek und den Namen des Objekts, in dem das XML-Schema gespeichert werden soll.

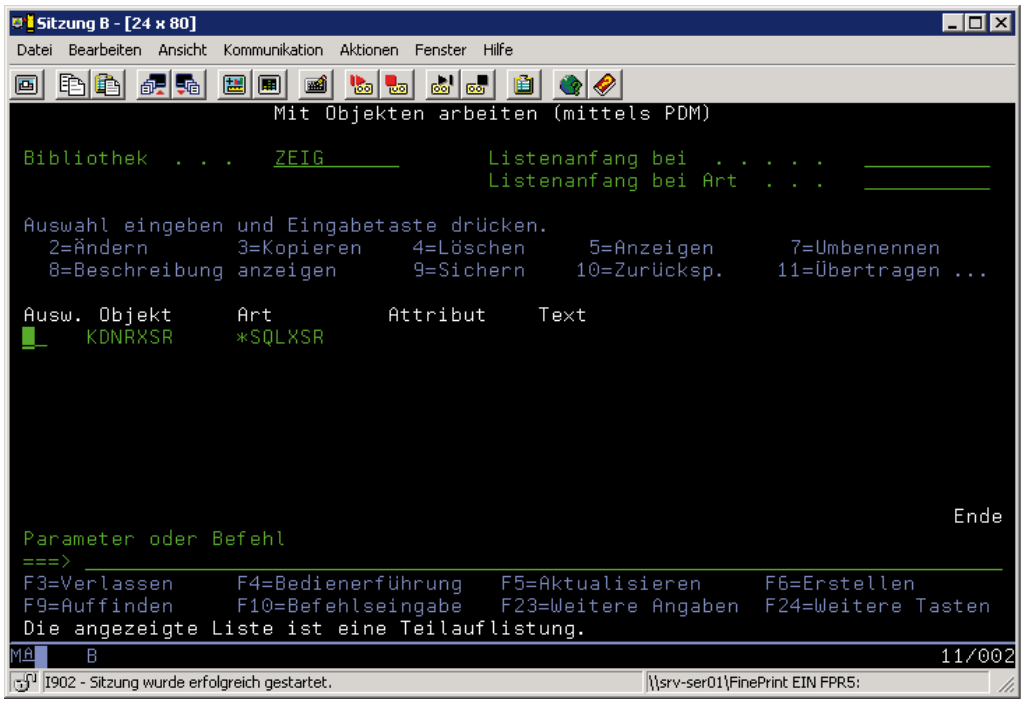
Schauen wir uns die Bestandteile der XSR-Registrierung einmal genauer an.

Wir verwenden das Programm XSR_REGISTER, das sich in der Bibliothek SYSPROC befindet. Dieses Programm benötigt die folgenden Parameter:

- Bibliothek für das Ablegen der SQLXSR-Informationen
- Objektnamen für die SQLXSR-Informationen
- Der vierte Parameter gibt den Ort an, in dem das XML-Schema-Dokument (die XSD) abgelegt ist.

Nachdem das Programm gewandelt wurde, rufen wir es auf, um die XML-Registrierung durchzuführen: CALL XSLREGIST

XML-Registrierungen werden auf dem System in dem XML-Schema-Repository (kurz: XSR) abgelegt.

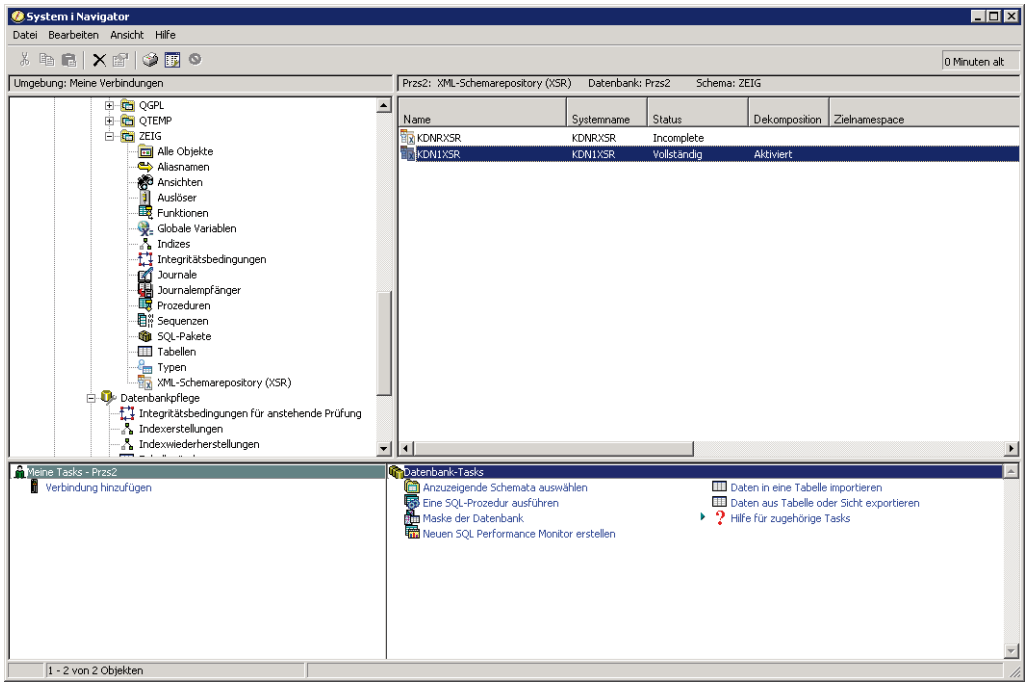


SQLXSR

Die vorhergehende Abbildung zeigt die neu angelegte KDNRXSR – ein Objekt vom Typ *SQLXSR.

3.6.1.2
Seite 6

Auch der System i Navigator verfügt über neue Funktionen, mit denen wir die XML-Komponenten einsehen und verwalten können. In der Navigationsübersicht befindet sich unter „Datenbank/Schema“ und dem jeweiligen Schema der Eintrag „XML-Schemarepository (XSR)“. Dieser Eintrag liefert eine Übersicht der XSR-Definitionen in dem ausgewählten Schema:



XSR-Anzeige im System i Navigator

Wie Sie in der Abbildung erkennen können, ist der Status der XSR „incomplete“ – es bedarf also noch weiterer Schritte, um die XSR verwenden zu können.

Um eine XSR verwenden zu können, muss sie den Status „vollständig“ haben und aktiviert sein. Ein Beispiel dazu zeigt der Eintrag „KDN1XSR“ in der vorhergehenden Abbildung.

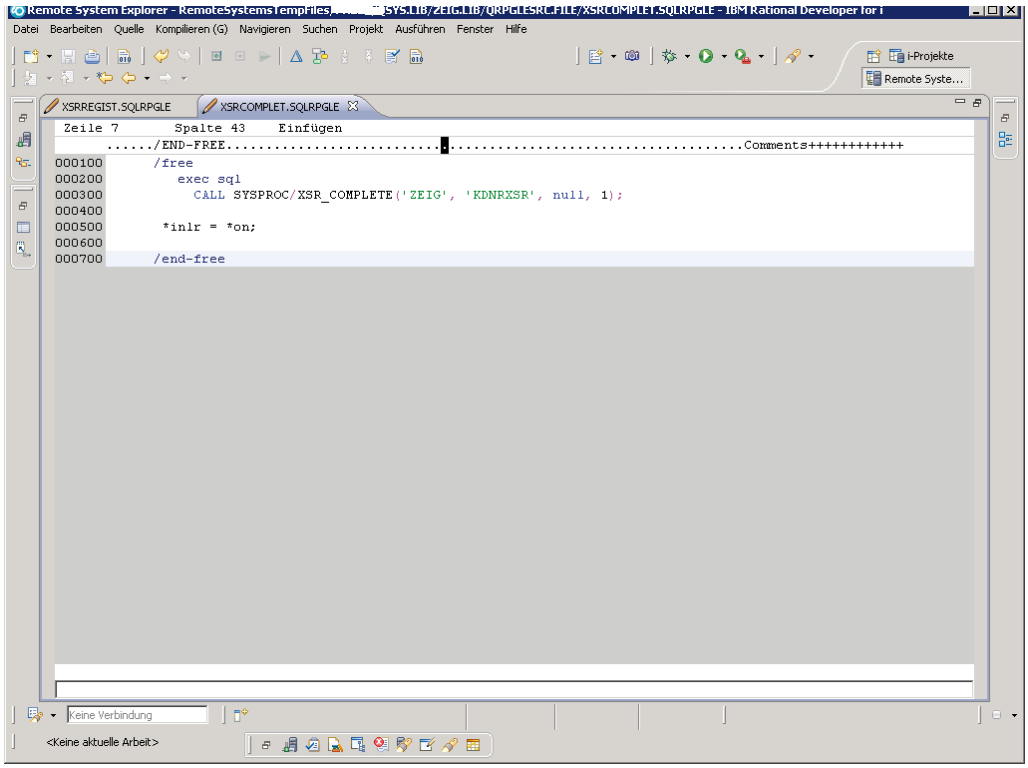
Damit ist die Registrierung leider noch nicht abgeschlossen. Als nächsten und finalen Schritt der Registrierung bedarf es noch eines weiteren Aufrufs.

```
CALL SYSPROC/XSR_COMPLETE(,ZEIG‘, ,KDNRXSR‘, null, 1)
```

Mit dem Aufruf der Stored Procedure XSR_COMPLETE und dem Parameter „issuedfordecomposition parameter“ mit dem Wert ‚1’ wird die Registrierung abgeschlossen.

XSR_COMPLETE ist deshalb erforderlich, weil bei Bedarf mehrere XML-Schemas in einer einzelnen XSD zusammengefasst werden können. Sollte das erwünscht sein, so müssen Sie dazu die Prozedur XSR_ADDSCHEMADOC einsetzen.

Diese Prozedur haben wir ebenfalls in einem SQLRPGLE eingebunden. Das folgende Beispiel zeigt den Quelltext der Source XSRCOMPLET:



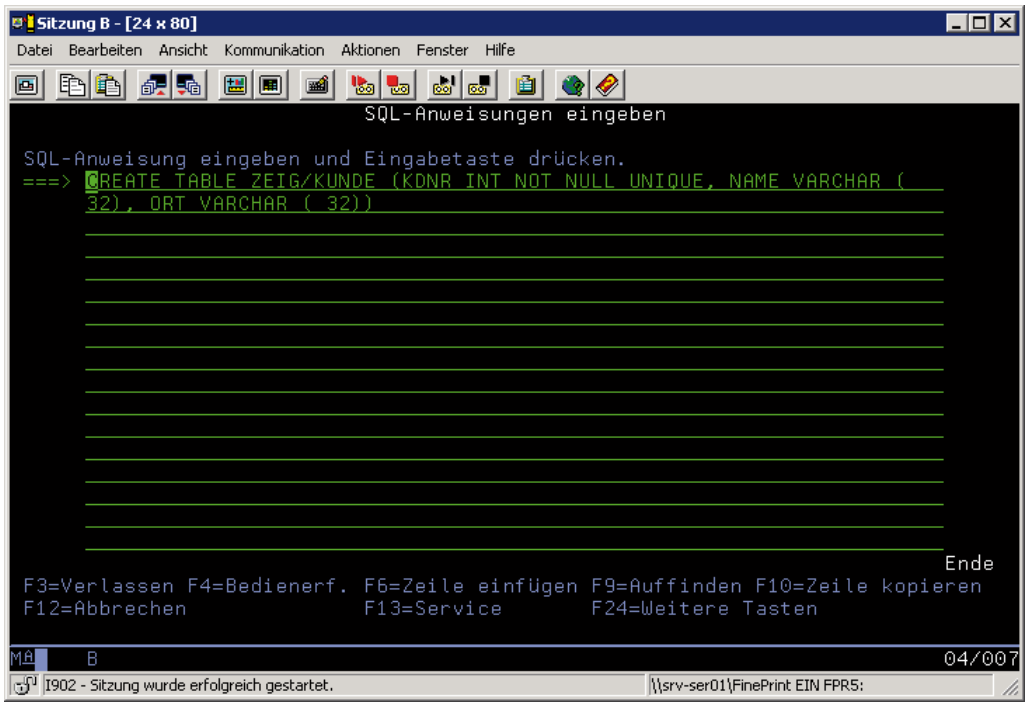
XSRCOMPLET

XSR_COMPLETE erwartet die folgenden Parameter:

- Schema (oder auch Bibliothek)
- Name des XML-Schemas
- Schema-Eigenschaften
  Wenn dieses Schema zum Umsetzen von XML verwendet werden soll, muss der Parameter mit „1“ angegeben werden!

3.6.1.2
Seite 8

Achten Sie nach der Ausführung dieser beiden Teilschritte auf die Ergebnisse, die im Jobprotokoll festgehalten werden. Mögliche Probleme, wie z. B. fehlerhafte Feldzuordnungen in Bezug auf den Inhalt der XML-Datei und der DB2-Tabelle werden hier protokolliert. Ein Beispiel finden Sie hier:



Jobprotokoll

Mit der Ausführung der beiden Prozeduren XSR_REGISTER und XSR_COMPLETE haben wir das Erstellen und Registrieren des XML-Schemas abgeschlossen.

Anmerkung:

Die Trennung des Aufrufs der beiden Prozeduren zum Registrieren des XML-Schemas wurde in diesem Beispiel nur zur Veranschaulichung vorgenommen. Natürlich können die beiden Prozeduren auch unmittelbar nacheinander in eine Programmquelle eingebunden und aufgerufen werden!

```
     /free
        exec sql
          CALL SYSPROC/XSR_REGISTER('ZEIG', 'KDNRXSR', null,
          GET_XML_FILE('/home/ZEIG/XML/kunde.xsd'), null);

        exec sql
          CALL SYSPROC/XSR_COMPLETE('ZEIG', 'KDNRXSR', null, 1);

       *inlr = *on;

     /end-free
```

Beide Prozeduren in einer Quelle

Damit steht einem Import von XML-Informationen in die DB2-Tabelle nichts mehr im Wege.

Die XML-Datei habe ich ebenfalls ins IFS des Home-Verzeichnisses gelegt.

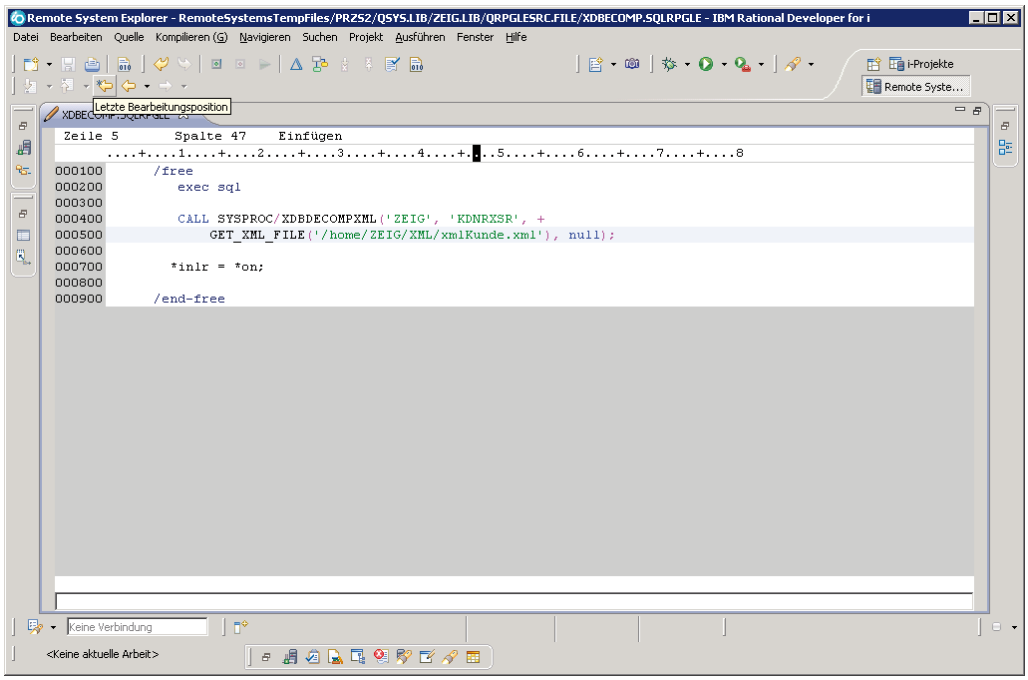
IFS-Inhalt

Für das Dekompilieren der XML-Informationen verwenden wir eine weitere Prozedur XDBDECOMPXML. Diese Prozedur verfügt über die folgenden Parameter:

- Schema
- Name
- XML-Dokument
- Dokument-ID

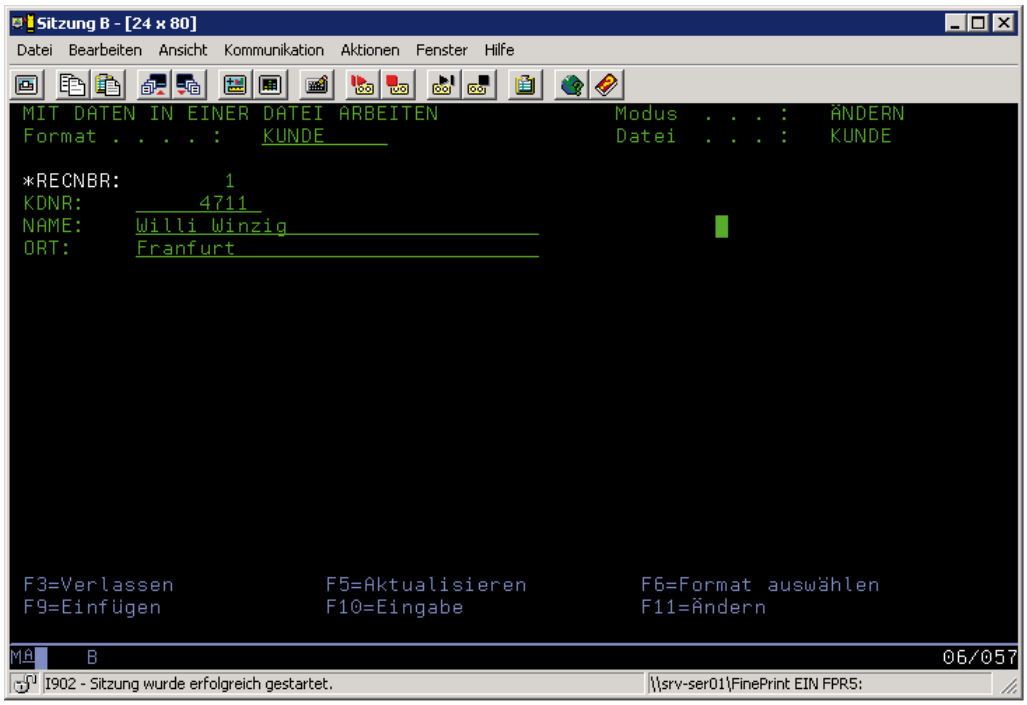
3.6.1.2
Seite 10

Die folgende Abbildung zeigt den Aufruf von XDBDECOMPXML in einem SQLRPGLE:



XDBDECOMPXML

Mit dem Aufruf des Programms werden die Datensätze aus dem definierten XML-Dokument in die zugeordnete DB2-Tabelle übertragen:



DB2-Datensatz

Zusammenfassend sind die folgenden Teilschritte notwendig, um XML-Daten in eine DB2-Tabelle zu importieren:

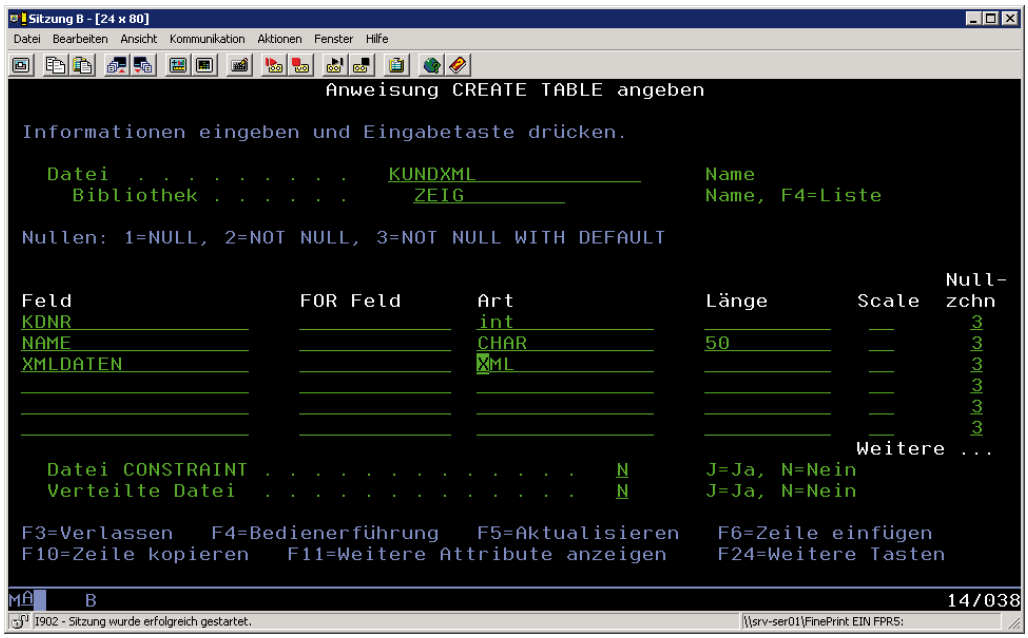
- XSD-File einer globalen Variable hinzufügen
  set zeig.var1 = GET_xml_FILE(,/home/ZEIG/XML/kunde.xsd,
- XSD in der XML-Repository registrieren
  CALL SYSPROC.XSR_REGISTER(,ZEIG', ,KDNRXSR', null, zeig.var1 , null)
- Registrierung vervollständigen
  call sysproc.xsr_complete (,ZEIG', ,KDNRXSR', null, 1)
- XML-File der globalen Variable hinzufügen
  set zeig.var1 = GET_xml_FILE(,/home/ZEIG/XML/kunde.xml')
- XML-Daten in die Datenbank übernehmen
  call sysproc.XDBDECOMPXML (,ZEIG', ,KDNRXSR', zeig.var1, null)

3.6.1.2
Seite 12

XML und SQL

Neuanlage von XML-Elementen in DB2

Bei der Neuanlage von DB2-Elementen können wir seit 7.1 bequem den Spaltentyp „XML“ angeben. Mit dieser Definition lassen sich in neuen Tabellen XML-Elemente schnell definieren.



CREATE TABLE

Die Angabe von XML-Elementen in einer Tabelle wird einfach über die ART „XML“ vorgenommen. Eine Längenangabe ist hier nicht notwendig/möglich.

Natürlich ist auch das Definierten von XML-Elementen in Anwendungen möglich. Ein Beispiel für die Verwendung von XML in einer SQL-Prozedur finden Sie hier:

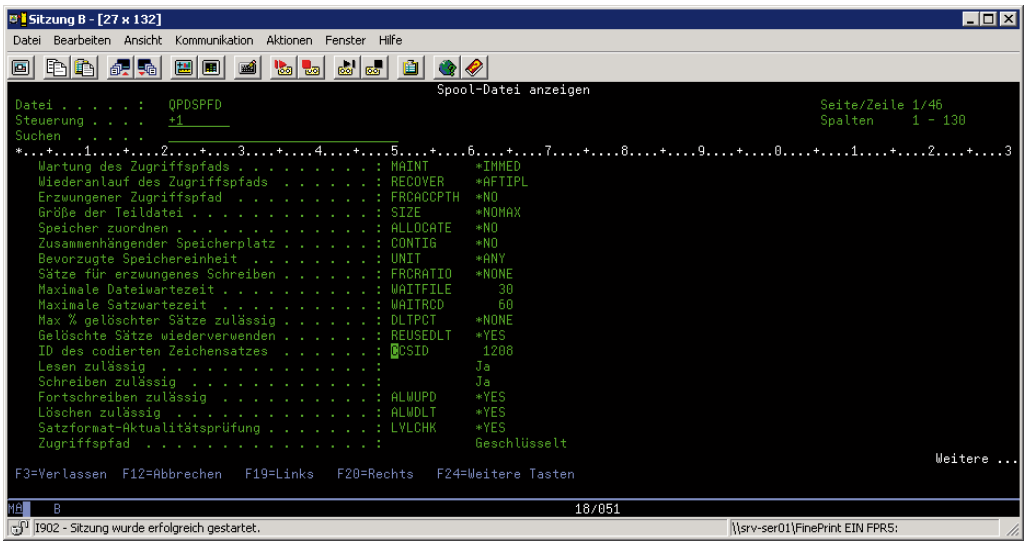
```
CREATE PROCEDURE XMLPROZ01 (IN p1 XML)
BEGIN
   DECLARE xmlv1 XML;
END
```

3.6.1.3
Seite 2

ACHTUNG!

Es ist ein immer wieder auftauchendes Phänomen: Das Zusammenspiel von Datenbank und CCSID ist in einigen Installationen „nicht optimal". Häufig findet man als Standardwert für die CCSID die Einstellung 65535 (keine Umsetzung). Ich möchte an dieser Stelle nicht zu sehr ins Detail gehen – aber die Nutzung der CCSID 65535 zusammen mit XML kann gewisse Probleme bereiten. Ein Problem kann sich z. B. darin äußern, dass eine Tabelle, die XML-Datentypen verwenden soll, mit der CCSID 65535 angelegt wurde, und eine Umsetzung der XML-Daten dann unter Umständen nicht oder nur fehlerhaft möglich ist.

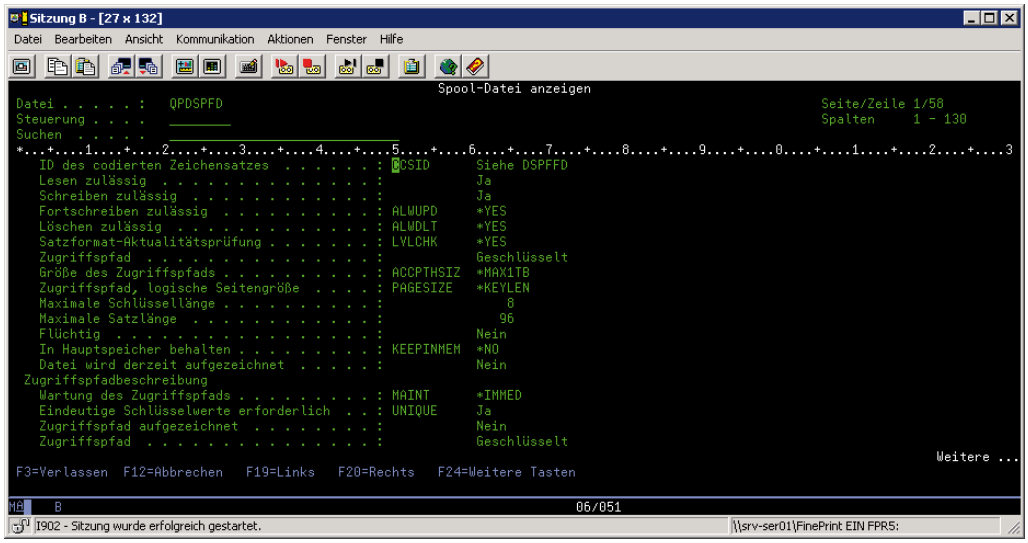
Schauen wir uns das an einem kurzen Beispiel an:
Jede Datei bzw. auch Tabelle verfügt auf dem System i über eine Zeichensatz-ID (CCSID). Diese ID kann z. B. mit dem Befehl DSPFD eingesehen werden:



DSPFD

In der Abbildung oben können Sie sehen, dass die Datei mit der CCSID 1208 (UTF-8) erstellt wurde. Eine solche Datei/Tabelle dürfte für den Einsatz mit XML-Daten keine Probleme bereiten.

Wenn wir eine Abweichung in der CCSID zwischen Datei und Felddefinition haben, dann wird die folgende CCSID in der Anzeige der Dateibeschreibung (DSPFFD) ausgegeben:



DSPFFD – abweichende CCSID

Kann eine Umsetzung der CCISD in UTF-8 nicht ordnungsgemäß vorgenommen werden, so erhalten wir in der Regel die folgende Fehlermeldung:

SQ20398

Nachdem eine Tabelle mit XML-Nutzung definiert wurde, können wir sie mit SQL-Funktionen nutzen.

3.6.1.3
Seite 4

Nachfolgend finden Sie ein Beispiel für das Hinzufügen von Daten zu einer Tabelle. Dieses Beispiel fügt der Datei „KUNDE“ einen neuen Datensatz hinzu. Dieser Datensatz besteht aus der Kundennummer (KDNR) und den XML-Daten, die über das Feld „XMLINFO“ eingestellt werden.

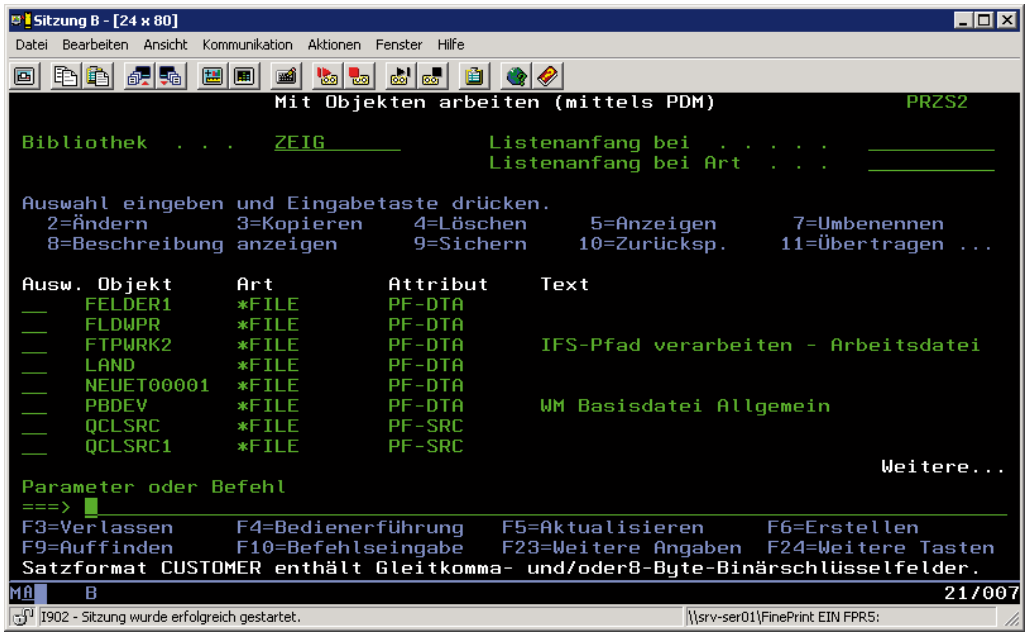
```
INSERT INTO KUNDE (KDNR, Info) VALUES (1000,
'<kundeninfo xmlns="http://posample.org" Cid="1000">
  <name>Willi Winzig</name>
  <addr
    <strasse>Hauptstrasse 47</strasse>
    <plz>60000</plz>
    <stadt>Frankfurt</stadt>
    <addr>
  <kundeninfo>');
```

Wie Sie sehen, wird der XML-Wert mit der betreffenden XML-Syntax eingegeben. Natürlich kann das Einfügen eines neuen Datensatzes in eine Tabelle mit allen anderen Technologien genutzt werden. Wichtig ist, dass die XML-Daten wohlgeformt sind.

Die Darstellung der XML-Spalten gestaltet sich hingegen durchaus schwieriger – da die Standardwerkzeuge, die das System i mit sich bringt, den XML-Inhalt nicht anzeigen können. Vielmehr erscheinen hier Hinweise, dass es sich um einen XML-Inhalt handelt. Einen solchen Hinweis finden wir in der Folgeabbildung, in der der Verweis auf den XML-Inhalt mit „*POINTER“ gegeben wird:

XML-Inhalt – ein Verweis

Einige Funktionen, wie z. B. der Befehl UPDDTA, können für die XML-Daten nicht mehr verwendet werden. Das äußert sich wie in der Folgeabbildung dargestellt:



UPDDTA

3.6.1.3
Seite 6

Erweitern von bestehenden Datenbankelementen

System i-Betreibern stellt sich durch die neuen Möglichkeiten rund um den Einsatz von XML mit der DB2/UDB die Frage, ob und wie man XML-Informationen zusammen mit bestehenden DB2-Elementen einsetzen und nutzen kann.

Grundsätzlich sei erwähnt, dass IBM die XML-Integration eng an die SQL-Funktionen der Datenbank angelehnt hat. Nicht umsonst empfiehlt IBM bereits seit vielen Jahren, auf die Verwendung von ISAM-Standards bei der Erstellung von Datenbankkomponenten zu Gunsten von SQL zu verzichten. Zur Erläuterung:

Das Erstellen von physischen Dateien oder logischen Dateien gehört mit zum ISAM-„Standard“, einer Technologie, die dem AS/400-Betreiber seit Anbeginn zur Verfügung steht. Durch die immer wichtiger gewordenen SQL-Funktionen hat IBM den Fokus auf SQL im Bereich der Datenbank ständig ausgebaut. Eine Unterstützung von XML mit DDS kann demzufolge nicht erwartet werden! Folglich bezieht sich die folgende Erläuterung ausschließlich auf die Datenbankelemente, die bereits mit SQL-Standards betrieben werden.

Die Möglichkeit, bestehende SQL-Tabellen um neue Spalten zu erweitern, ist nicht neu. Die Funktion ALTER TABLE dürfte den meisten Lesern bekannt sein. So ist es nicht verwunderlich, dass auch für die XML-Erweiterung bestehender Tabellen die Funktion ALTER TABLE eingesetzt werden kann. Das folgende Beispiel zeigt, wie man eine neue Spalte für die Aufnahme von XML-Informationen einer Tabelle hinzufügt:

```
ALTER TABLE KUNDE1 ADD COLUMN Preferences XML;
```

ACHTUNG!

Natürlich gilt auch für das Erweitern von bestehenden Datenbankelementen, dass die Problematik der CCSID hier ebenfalls zu beachten ist. Diese Problematik wurde im vorhergehenden Kapitel beschrieben.

3.6.1.3.1

Seite 2

Hinzufügen von XML-Inhalten in DB2-Tabellen

XML-Inhalte lassen sich mit Hilfe der SQL-Anweisung INSERT bequem den jeweiligen XML-Spalten zuweisen. Voraussetzung für erfolgreiches Hinzufügen ist es, sicherzustellen,, dass die XML-Daten wohlgeformt sind.

Das folgende Beispiel zeigt einen Ausschnitt aus XML-Daten, die auf das System i übertragen werden sollen:

```
<kunde xmlns="http://posample.org" Cid="4711">
 <name>Willi Winzig</name>
 <addr stadt="Franfurt">
   <strasse>Maingasse 1</strasse>
   <plz>60000</plz>
 </addr>
</kunde>
```

Mittels SQL-Anweisungen, die wir z. B. in RPG einbinden können, lassen sich die Daten leicht übertragen, wie der folgende Codeausschnitt zeigt:

```
EXEC SQL BEGIN DECLARE SECTION;
   sqlint64 cid;
   SQL TYPE IS XML AS BLOB (10K) xml_hostvar;
EXEC SQL END DECLARE SECTION;
…
cid=1015;
/* Read data from file kunde.xml into xml_hostvar */
…
EXEC SQL INSERT INTO Kunde (Cid,Info) VALUES (:cid, :xml_hostvar);
```

3.6.1.3.2

Seite 2

XML-Prozeduren

Hier die wichtigsten Stored Procedures, die im Zusammenhang mit XML und DB2 zum Einsatz kommen:

XSR_REGISTER
Mit dieser Prozedur wird das Registrieren eines XML-Schemas gestartet. Sie wird beim initialen Hinzufügen eines XML-Schema-Dokuments zu einem XML-Schema aufgerufen.

XSR_ADDSCHEMADOC
Mit dieser Prozedur können weitere XML-Schema-Dokumente einem XML-Schema hinzugefügt werden. Dieses XML-Schema muss sich für diese Prozedur im Prozess der Registrierung befinden. XSR_ADDSCHEMADOC kann nur nach dem Aufruf von XSR_REGISTER und vor XSR_COMPLETE ausgeführt werden. Wird XSR_ADDSCHEMADOC nicht innerhalb dieser Befehlsreihenfolge ausgeführt, schlägt der Aufruf fehl.

XSR_REMOVE
Mit der Prozedur XSR_REMOVE lassen sich alle Komponenten eines XML-Schemas entfernen.

XSDBCOMPXML
XSDBCOMPXML wird für das Umsetzen von XML-Daten in DB2-Tabellen verwendet.

3.6.1.4

Seite 2

XML-Funktionen

Nachfolgend finden Sie die wichtigsten Funktionen, die im Zusammenhang mit XML und DB2 zum Einsatz kommen:

XMLATTRIBUTES
Liefert als Rückgabewert eine XML-Sequenz mit dem Attribut für jedes Nicht-Null-Argument.

XMLCOMMENT
Liefert als Rückgabewert einen XML-Wert mit einem einzelnen Kommentar von dem angegebenen String-Ausdruck.

XMLCONCAT
Liefert als Rückgabewert einen XML-Wert, der aus einer Anzahl zusammengesetzter XML-Elemente besteht.

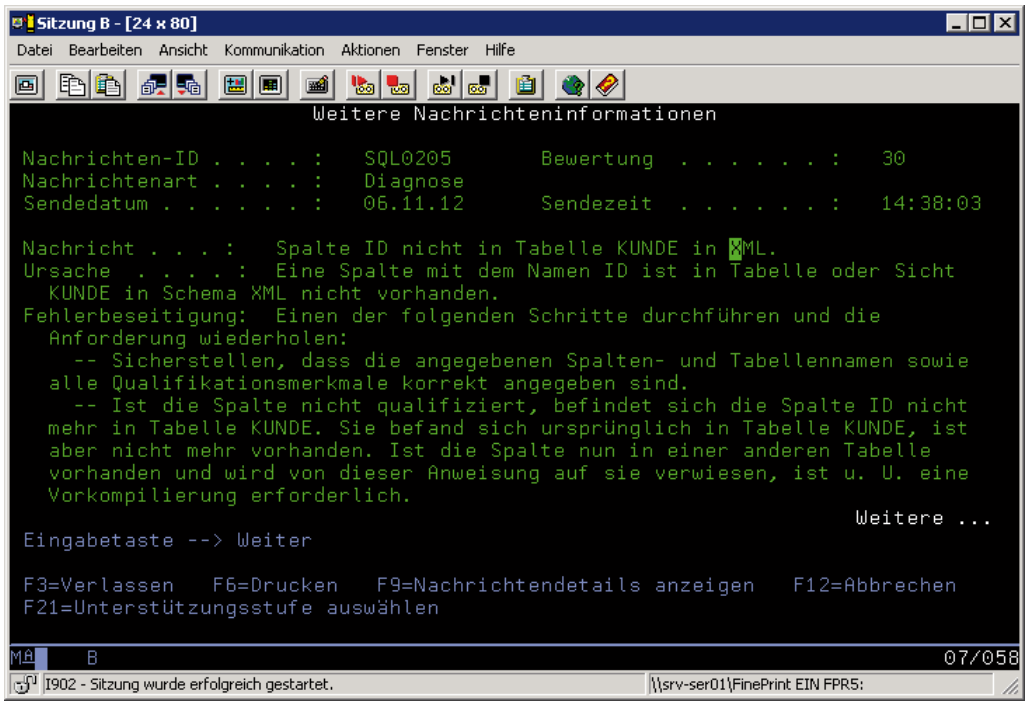
3.6.1.5

Seite 2

Problembehandlung

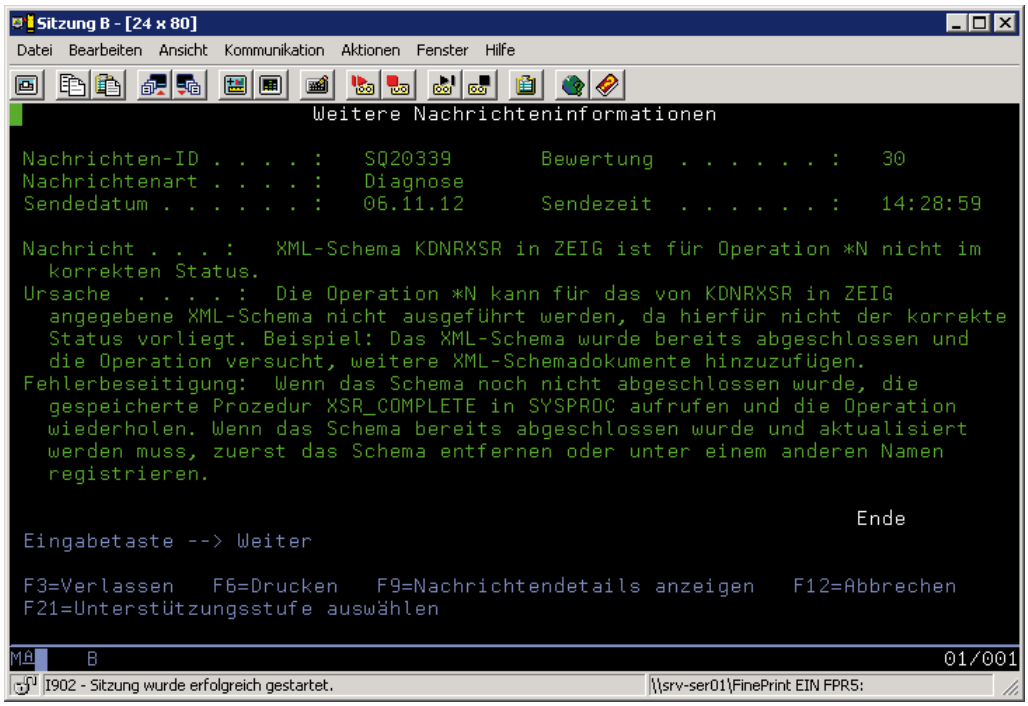
Auch beim Einsatz von XML sowie dem gemeinsamen Einsatz mit DB2-Elementen kann es zu Fehlern kommen. Es ist an dieser Stelle sicher nicht möglich, alle potenziell auftretenden Fehler zu benennen, trotzdem möchte ich Sie auf einige der am häufigsten auftretenden Fehler beim Einsatz von XML und DB2 hinweisen:

SQL0205:
Stimmt die Definition der XML-Struktur nicht mit der Zieltabelle in der DB2 überein, dann wird der Fehler SQL0205 ausgegeben. An dieser Stelle muss das Mapping zwischen XML und DB2-Tabelle erneut geprüft werden!



3.6.1.6
Seite 2

SQ20339:
In der Praxis kann es immer wieder einmal vorkommen, dass Umsetzungen nicht korrekt durchgeführt werden. Ein beliebter Fehler ist der Fehler mit der Nachrichten-ID SQ20339.



Fehler in der Umsetzung

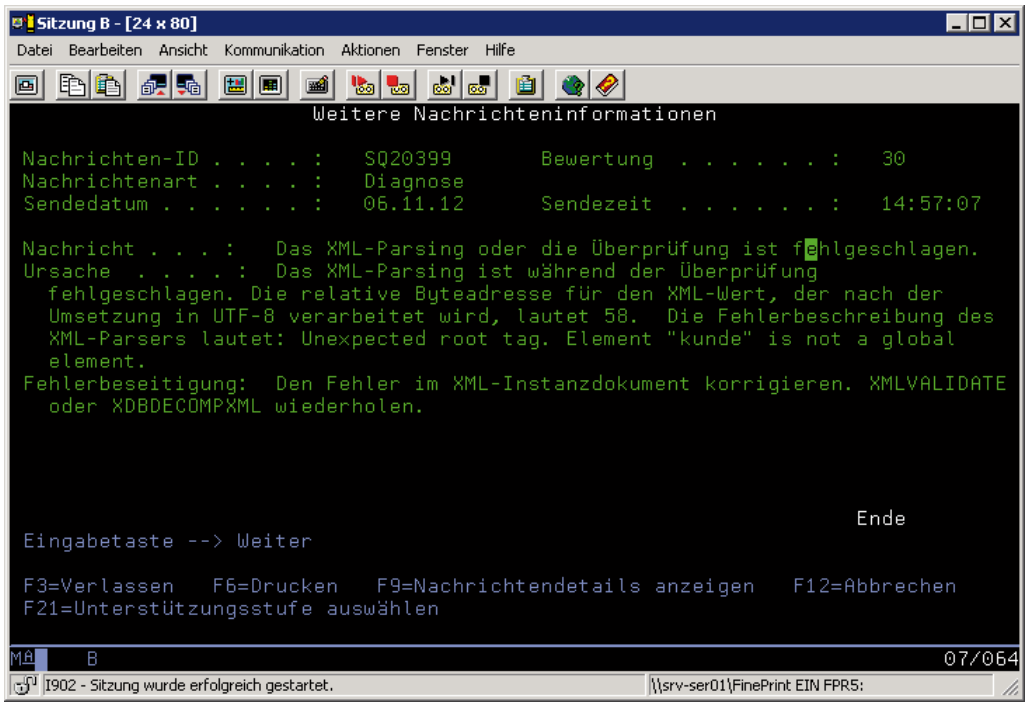
SQ20398:
Leider kommt es auch immer wieder vor, dass Systeme für eine Zeichenumsetzung „ungünstig“ eingestellt sind. Die Notwendigkeit, Zeichensätze umzusetzen, stellt sich in der Regel immer dann, wenn Daten mit anderen Systemen ausgetauscht werden müssen. Eine korrekte Einstellung der CCSID ist an dieser Stelle erforderlich. Gibt es Probleme bei der Umsetzung, dann wird der Fehler SQ20398 ausgegeben.

Eine mögliche Ursache für den Fehler SQ20398 ist das Erstellen einer SQL-Tabelle mit der falschen CCSID. Ändern Sie diese ab und wiederholen Sie die Anforderung.

SQ20398

3.6.1.6
Seite 4

SQ20399:
Parsing-Probleme werden mit dem Fehler SQ20399 ausgegeben. Hier ist ebenfalls eine Überprüfung der Zuordnung von XML zu einer DB2-Tabelle erforderlich.



SQ20399

4 Kapitel

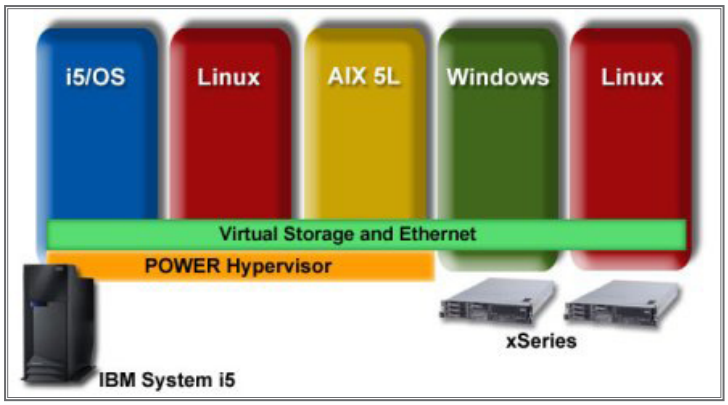
Betriebssysteme / Integration

4

Seite 2

System i Betriebssysteme

Auf dem Weg von der AS/400 zum System i (sprich: ssistem ei) rückten so genannte Virtualisierungstechniken immer weiter in den Vordergrund, was bis heute dazu führt, dass nicht nur das native i5/OS auf einem System i ausgeführt werden kann, sondern neben i5/OS auch LINUX, AIX und Windows (2000 und höher).



iSeries Virtualisierung

i5/OS, Linux und AIX werden als Betriebssystem in einem logischen System i-Partition installiert.

Windows wird mittels besonderer Hardware – heute Integrated X Server und Integrated X Adapter, in Zukunft via virtuellem SCSI – im System i installiert. Seit der 5xx Hardwareplattform kann Linux ebenfalls auf diesem Wege installiert werden.

Auf den folgenden Seiten werden Sie eine Reihe von Begriffen kennen lernen, die im Rahmen dieses Kapitels für Sie wichtig sind.

Danach erfolgt die Darstellung von Installations- und Konfigurationsgängen.

4.1
Seite 2

Linux on iSeries

Die eServer iSeries und i5 können Business-Applikationen und e-business-Lösungen mit Linux-Applikationen auf einem einzelnen Server kombinieren. Ein solcher Linux-Server kann schon mit lediglich 10% eines iSeries Prozessors (mittels Micro-Partitioning) aufgesetzt werden. Jede Partition unterstützt ihr eigenes, unabhängiges Betriebssystem und kann von anderen Partitionen isoliert werden, so dass Business-Applikationen sicher neben Internet-Lösungen ablaufen können. Mit einer OS/400 V5R2 oder i5/OS V5R3 Primärpartition können Prozessorressourcen dynamisch zwischen Partitionen verschoben werden, um sich ändernde Geschäftsanforderungen zu stützen.

Die Linux-Implementierung bei iSeries und i5 nutzt die Virtualisierungsservices von OS/400, wie z. B. das dynamische Verschieben von Verarbeitungseinheiten zwischen Partitionen und Virtual Storage Management. 64-bit Umgebungen können aufgrund größerer Speicher- und Adressräume mehr Skalierbarkeit bieten als traditionelle 32-bit Linux Umgebungen. Pro Prozessor werden bis zu 10 Linux Partitionen unterstützt, mit einem System-Maximum von 253 Partitionen, die i5/OS V5R3 auf POWER5 Technologie-basierten Servern betreiben, und 31 Partitionen mit V5R3 auf POWER4 und SStar Prozessoren.

Hardware Features, die bei Linux eingesetzt werden, können für die Modelle 520, 550, 570 und 595 bestellt werden. Bei diesen Features ist eine POWER Linux Distribution erforderlich. Weiter Informationen über Linux Distributionen, die für die iSeries verfügbar sind, finden Sie unter:

http://www.ibm.com/servers/eServer/iseries/linux/dist.html

4.1.1
Seite 2

Weitere Merkmale der Linux-Unterstützung:

- Mit der POWER5 Hardware können maximal 4064 virtuelle Netzwerke mittels IEEE 802.1Q VLAN konfiguriert werden.
- Der iSeries Navigator Support erzeugt und handhabt Partitionen, die eine planmäßige Verschiebung von Quellen umfasst, bei denen HMC nicht verfügbar ist. Eingeschlossen sind:
  - Support für die Konfigurierung von Linux mit dem Create Partition Wizard
  - Die Fähigkeit, Partitions-Konfigurationsdaten zu speichern
  - Die Fähigkeit, Konfigurationsdaten in ein HTML-Format für Hardcopy Prints zu exportieren
  - Partition Numbers sind mit den Namen sichtbar
  - Die Fähigkeit, bei der Verschiebung von Verarbeitungseinheiten virtuelle Prozessoren zu wechseln
  - Partitions-Profile für gemeinsame und festgelegte Prozessoren updaten
  - Die Fähigkeit, Views über die Selektion von Spalten auszuwählen, um diese anzuzeigen

LPAR Partition Support

Supported processors				Secondary partition					
	Primary partition	Shared processor	Partitions per processor	V4R5	V5R1	V5R2	V5R3	Linux	AIX 5L
Model 270 (SStar Uni) #2431, #2432, #2452	V5R1	Yes	4		Yes	Yes		Yes	No
	V5R2	Yes	4		Yes	Yes	Yes	Yes	No
	V5R3	Yes	4			Yes	Yes	Yes	No
Model 270 (SStar 2-way) #2434 and #2454	V5R1	Yes	4		Yes	Yes		Yes	No
	V5R2	Yes	4		Yes	Yes	Yes	Yes	No
	V5R3	Yes	4			Yes	Yes	Yes	No
Model 520 (POWER5) #8950	V5R3	Yes	2				Yes	Yes	Yes
Model 520 (POWER5) #8951, #8952	V5R3	Yes	4				Yes	Yes	Yes
Model 520 (POWER5) #8953, #8954	V5R3	Yes	10				Yes	Yes	Yes
Model 520 (POWER5 2-way) #8955	V5R3	Yes	10				Yes	Yes	Yes
Model 570 (POWER5 1/2-way) #8961	V5R3	Yes	10				Yes	Yes	Yes
Model 570 (POWER5 2/4-way) #8961	V5R3	Yes	10				Yes	Yes	Yes
Model 800 (SStar Uni) #2463 and #2464	V5R2	Yes	4			Yes	Yes	Yes	No
	V5R3	Yes	4			Yes	Yes	Yes	No
Model 810 (SStar Uni & 2-way) #2465, #2466, #2467, #2469 (2-way)	V5R2	Yes	4			Yes	Yes	Yes	No
	V5R3	Yes	4			Yes	Yes	Yes	No
Model 820 (Pulsar Uni) #2395, #2396, #2425	V5R1	Yes	4		Yes	Yes		No	No
	V5R2	Yes	4		Yes	Yes	Yes	No	No
	V5R3	Yes	4			Yes	Yes	No	No
Model 820 #2397, #2398, #2426, #2427	V5R1	Yes2	4		Yes	Yes		Ded	No
	V5R2	Yes2	4		Yes	Yes	Yes	Ded	No
	V5R3	Yes	4			Yes	Yes	Ded	No
Model 820 (SStar Uni) #0150, #2435, #2436, #2456	V5R1	Yes	4		Yes	Yes		Yes	No
	V5R2	Yes	4		Yes	Yes	Yes	Yes	No
	V5R3	Yes	4			Yes	Yes	Yes	No
Model 820 (SStar n-way) #0151, #0152, #2437, #2438, #2457, #2458	V5R1	Yes	4		Yes	Yes		Yes	No
	V5R2	Yes	4		Yes	Yes	Yes	Yes	No
	V5R3	Yes	4			Yes	Yes	Yes	No
Model 825 (POWER4 n-way) #2473	V5R2	Yes	10			Yes	Yes	Yes	No
	V5R3	Yes	10			Yes	Yes	Yes	No
Model 830 (IStar n-way) #0153 and #2349	V5R1	Yes	4		Yes	Yes		Ded	No
	V5R2	Yes	4		Yes	Yes	Yes	Ded	No
	V5R3	Yes	4			Yes	Yes	Ded	No
Model 830 (IStar n-way) #2400, #2402, #2403	V5R1	Yes2	4		Yes	Yes		Ded	No
	V5R2	Yes2	4		Yes	Yes	Yes	Ded	No
	V5R3	Yes2	4			Yes	Yes	Ded	No
Model 840 (IStar n-way) #2416, #2417,#2418, #2419, #2420	V5R1	Yes2	4		Yes	Yes		Ded	No
	V5R2	Yes2	4		Yes	Yes	Yes	Ded	No
	V5R3	Yes2	4			Yes	Yes	Ded	No
Model 840 (SStar n-way) #0158 and #0159	V5R1	Yes	4		Yes	Yes		Yes	No
	V5R2	Yes	4		Yes	Yes	Yes	Yes	No
	V5R3	Yes	4			Yes	Yes	Yes	No
Model 840 (SStar n-way) #2461, #2460,#2352, #2353, #2354	V5R1	Yes	4		Yes	Yes		Yes	No
	V5R2	Yes	4		Yes	Yes	Yes	Yes	No
	V5R3	Yes	4			Yes	Yes	Yes	No
Model 870 (POWER4 n-way) #2486	V5R2	Yes	10			Yes	Yes	Yes	No
	V5R3	Yes	10			Yes	Yes	Yes	No
Model 890 (POWER4 n-way) #0197, #0198, #2487, #2488, #2497, #2498	V5R2	Yes	10			Yes	Yes	Yes	No
	V5R3	Yes	10			Yes	Yes	Yes	No

Notes:

- *All partitions must be at V5R1 or later to supportprocessor sharing and dynamic movement of processor resources.*
- *Linux supported on Dedicated Processors only, other partitions support processor sharing.*

4.1.1
Seite 4

POWER Linux läuft auf den neuen System i- und ausgewählten iSeries-Maschinen in einer Logical Partition. Die nachfolgende Übersicht zeigt, welche System i- und iSeries-Server Linux unterstützen.

Model 520				
CPW	# of CPUs	LPAR micro-partitions	Linux	Max # of partitions
500, 600	1	Yes*	Yes	2
1000	1	Yes*	Yes	4
1200	1	Yes*	Yes	3
2400	1	Yes	Yes	10
2800	1	Yes*	Yes	7
3100, 3300, 3800	1	Yes	Yes	10
6000, 7100	2	Yes	Yes	20

* These systems do not support un-capped partitions

Model 550				
Feature #	# of CPUs	LPAR micro-partitions	Linux	Max # of partitions
All	1 to 4	Yes	Yes	10 to 40

Model 570				
Feature #	# of CPUs	LPAR micro-partitions	Linux	Max # of partitions
All	1 to 16	Yes	Yes	10 to 160

Model 595				
Feature #	# of CPUs	LPAR micro-partitions	Linux	Max # of partitions
All	8-64	Yes	Yes	80-254

Linux für iSeries

Linux ermöglicht eine Vielzahl von e-Business Applikationen, mit denen die Stärken der eServer iSeries und i5 als eine integrierte Lösung für das Kerngeschäft vervollständigt werden können. Linux übernimmt bedeutende Stärken und Hochverfügbarkeits-Features der iSeries-Architektur.

OS/400 wurde dahingehend verbessert, um Linux zu unterstützen, das in einer sekundären, logischen Partition läuft. Die primäre Partition muss auf OS/400 V5R1 (oder neuer) laufen. Abhängig vom iSeries-Modell werden dabei bis zu 32 Linux-Partitionen unterstützt.

Prozessor-Features für iSeries Modell 270, 800, 810, 820 und 840, die SStar verwenden, Modell 825, 870 und 890 mit POWER4 Prozessoren und 520 bis 595 Prozessoren mit POWER5 ermöglichen, dass Linux in einem gemeinsamen Prozessor-Pool läuft, in dem sich vier OS/400 und Linux Partitionen einen Prozessor teilen können. Auf n-Way Prozessor-Features für die iSeries Modelle 820, 830 und 840 Server mit IStar-Prozessoren ist für Linux mindestens ein Prozessor pro Linux-Umfeld erforderlich.

Neu bei V5R3

- Linux Partition Management wurde mit dem iSeries Navigator aufgewertet. Die umfasst
  - das Starten und Beenden eines Servers
  - das Erzeugen und Löschen einer Festplatte
  - das An- und Abhängen von Laufwerken mit dem Server
- Aufgewertete Grid-Partizipierung über IBM Grid Toolbox V3 für Linux auf iSeries (5733-GD1)

Es gibt zwei Möglichkeiten, Linux auf der iSeries zu implementieren:

- hosted
- non-hosted

In einem hosted Umfeld verwendet Linux virtuelle Quellen (wie z. B. Festplatte, Tape, CD-ROM und LAN), die einer OS/400-Partition gehören und mit dieser OS/400-Partition geteilt werden. Linux wird von der Hosting-OS/400-Partition durch das Anhängen einer Network Server Discription (NWSD) gestartet.

Die NWSD wird verwendet, um die LPAR-Funktionen zu kontrollieren. LPAR kann nur aktiv sein, wenn der Server aktiv ist und sich nicht in einem eingeschränkten Status befindet. Wenn OS/400 sich in einem eingeschränkten Status befindet, werden alle NWSDs automatisch abgeändert.

Eine OS/400-Partition kann der Host für mehrere LPARs sein, die Gegenstand einer entsprechenden Kapazitätsplanung sind.

Für beliebige Input/Output-Ressourcen ist die Linux Partition in einem non-hosted Umfeld nicht von einer Hosting-OS/400-Partition abhängig. Die LPAR hat eigene Festplatteneinheiten oder die Partition verwendet Netzwerk-Support, um einen Netzwerk-Start durchzuführen.

Eine non-hosted LPAR kann gestartet werden, auch wenn die primäre Partition nicht vollständig aktiv ist. Sie können sie vom „Work with Partitions Status"-Display aus starten. Die non-hosted LPAR hat ihre eigenen Quellen, die auf sie definiert sind.

Anmerkung:

Linux ist nicht Teil von OS/400. Folglich muss es seine eigene Partition der Systemprozessorressourcen haben, die von OS/400 getrennt sind.

Virtuelles Input/Output in einer Linux Partition

Virtuelle Input/Output-Quellen sind Vorrichtungen, die der Hosting-OS/400-Partition gehören, welche der LPAR Input/Output-Funktionen zur Verfügung stellt. Der iSeries- und i5-Linux-Kernel und OS/400 unterstützen verschiedene unterschiedliche Arten von virtuellen Input/Output-Quellen:

- **Virtual Console**
  Die Virtual Console stellt über eine OS/400-Partition Konsolenfunktionen für die LPAR zur Verfügung. Durch den Einsatz der Virtual Console kann das Installationsprogramm mit dem User kommunizieren, bevor die Netzwerk-Ressourcen konfiguriert werden.
- **Virtuelle Festplatten-Einheit**
  Eine virtuelle Festplatten-Einheit ermöglicht den Zugriff auf virtuelle Festplatten für Linux. Der „Create Network Server Storage Space (CRTNWSSTG)"-Befehl erzeugt einen Speicherplatz, der von einem Netzwerk-Server zu verwenden ist, in diesem Fall die Linux Partition. Das Linux-Installationsprogramm formatiert die Festplatte für Linux neu, oder Sie können Linux-Befehle verwenden, wie z. B. fdisk und mke2fe, um die Festplatte für Linux zu formatieren.
- **Virtual CD**
  Eine Virtual CD wird benötigt, um die Installation von Linux zu stützen. Standardmäßig kann eine Linux-Partition alle CD-Laufwerke auf der Host LPAR sehen. Den Zugriff von Linux auf einige oder alle CD-Laufwerke können Sie einschränken.

- **Virtual Tape**
  Ein Virtual Tape ermöglicht den Zugriff von einer LPAR auf die OS/400 Tape-Laufwerke. Standardmäßig kann eine LPAR alle Tape-Laufwerke auf der Host Partition sehen. Den Zugriff von Linux auf einige oder alle Tape-Laufwerke können Sie einschränken.
- **Virtual Ethernet**
  Ein Virtual Ethernet bietet die gleiche Funktion wie der Einsatz eines 1 GB Ethernet-Adapters. Eine LPAR kann ein Virtual Ethernet einsetzen, um mehrere Hochgeschwindigkeitsverbindungen zwischen Partitionen herzustellen. OS/400 und Linux Partitionen können mittels TCP/IP über die Virtual Ethernet-Kommunikationsportale miteinander kommunizieren.

Direkt angebundenes Input/Output in einer Linux Partition

Mit direkt angebundenem Input/Output ist bei Linux eine direkte Handhabung der Hardware-Ressourcen gegeben. Alle Input/Output-Ressourcen befinden sich unter der ständigen Kontrolle des Linux-Betriebssystems.

Festplatten-Einheiten, Tape Devices und LAN-Adapter können alle einer LPAR zugeteilt werden, die Linux betreibt. Eine NWSD ist notwendig, um Linux in einer Partition zu installieren. Nachdem Linux installiert ist, kann die Partition so konfiguriert werden, dass sie unabhängig startet. Für die direkt angebundene Hardware werden alle Fehler- und Diagnosemeldungen in der LPAR angezeigt.

Mit direktem Input/Output gehören die Vorrichtungen zu Linux. OS/400 sieht die Vorrichtungen nicht und kann sie nicht direkt benutzen. Da Linux keinen direkten Support für IOPs bietet, werden sie nicht mit direktem Input/Output verwendet.

Bei der Linux Konsole handelt es sich um einen PC, der über ein TCP/IP LAN mit der primären Partition oder der Hosting Partition der iSeries verbunden ist. Die Operations Console mit LAN Connectivity (eine Funktion von iSeries Access) ist erforderlich, um eine sichere Verbindung zu OS/400 herzustellen. Dann wird ein Telnet-Umfeld verwendet, um sich über das Virtual LAN mit dem Linux-Umfeld zu verbinden. Die Konsole wird für Maßnahmen zur Installation und Problemermittlung eingesetzt. In der Linux Partition gibt es keinen Console Adapter.

Damit Linux auf der iSeries laufen kann, setzt IBM die Open Source Kernel Version 2.6 für PowerPC ein. Neben Produkt-Upgrades, -Support und -Wartung ist auch Linux für iSeries direkt über Linux-Distributoren erhältlich. IBM arbeitet z. Z. mit:

- Red Hat, Inc. (Red Hat Enterprise Linux AS for POWER Version 3)
- SUSE LINUX AG (SUSE LINUX Enterprise Server 9 for POWER)

4.1.1
Seite 8

Linux Integration mit OS/400

Die folgenden Produkte helfen, Linux und die iSeries zu integrieren:

- iSeries Open Database Connection (ODBC) Treiber für Linux

Linux-Programme, die auf das ODBC-Interface geschrieben werden, können über einen ODBC Treiber für Linux auf die DB2 UDB für iSeries Datenbanken zugreifen. Der ODBC Treiber wird abgerufen, um Datenbank-Anfragen durchzuführen. Diese Anfragen werden über den Datenbank-Server per IP-Verbindung zur Verarbeitung an die iSeries gesendet.

Weitere Informationen und das Produkt-Download finden Sie unter:

http://www-1.ibm.com/severs/eServer/iseries/linux/odbc/

- **iSeries Access für Linux**

iSeries Access für Linux bietet einen Linux-basierten Zugriff auf iSeries Server. Damit können Sie auf die DB2 UDB für iSeries mit deren ODBC Treiber zugreifen und eine 5250 Session zu einem iSeries Server von einem Linux-Client aus herstellen.

Weitere Informationen und das Download finden Sie unter:

http://www-1.ibm.com/servers/eServer/iseries/access/linux/

- **Grid Toolbox V3 für Linux auf iSeries (5733-GD1)**

Grid Toolbox V3 für Linux auf iSeries unterstützt Unternehmen, die Grid Computing einsetzen, handhaben und kontrollieren. Es unterstützt auch Entwickler von Produkten, die bei der Handhabung und beim Einsatz von Grids unterstützend wirken.

Weitere Informationen finden Sie unter:

http://www.alphaworks.ibm.com.tech/gridtoolbox

- **DB2 UDB Workgroup Server Edition für Linux (5733-LD1)**

Die DB2 UDB Workgroup Server Edition für Linux ist eine Datenbank, die für kleine Unternehmen oder Abteilungen entwickelt wurde, in denen eine geringe Anzahl von internen Usern vorhanden ist. Sie steht nur auf den 520, 550, 570, 595, 810, 825, 870 und 890 Enterprise Editionen zur Verfügung (andernfalls über Passport Advantage zu bestellen).

Weitere Informationen finden Sie unter:

http://www-306.ibm.com/software/data/db2/udb/edition-wse.html

4.1.1
Seite 9

- **WebSphere Application Server Express für Linux (5733-WL1)**

Der WebSphere Application Server Express für Linux ist ein fest integriertes Entwicklungstool und ein Applikationsserver, der Unternehmen, die dynamische Web Sites generieren, einen erschwinglichen Zugang zum e-Business bietet. Er ist auf den 570, 595, 810, 825, 870 und 890 Enterprise Editionen verfügbar (andernfalls über Passport Advantage zu bestellen).

Empfohlene Lektüre

Zusätzliche Informationen über Linux finden Sie auf folgender Web Site:

http://www.ibm.com/eServer/iseries/linux

4.1.1
Seite 10

System i Integration mit dem Windows-Server

iSeries und i5 Server können über den Integrated xSeries Server (IXS) oder den Integrated xSeries Adapter (IXA) Intel-basierte Windows-Server handhaben. Der Integrated xSeries Server unterstützt auf ausgewählten eServer- und i5-Modellen bis zu 60 Windows-Server. Der Anschluss von externen n-Way IBM eServer xSeries Servern wird von iSeries- und i5-Servern über den High-Speed Link (HSL) unterstützt.

Mit dem IXA können ausgewählte xSeries Server, die einen Windows 2000 Server betreiben, dazu verwendet werden, die Skalierbarkeit von Windows-Applikationen zu erweitern. Gleichzeitig können sie die Speicherkonsolidierung und Systemmanagement-Vorteile des Integrated xSeries Servers beibehalten.

Aufgrund des virtuellen Speicher-Managements kann ein Administrator einem Windows-Server, der in Betrieb ist, Speicherungen dynamisch hinzufügen, ohne einen Reboot durchführen zu müssen.

Mit den User-Administrationsfeatures ist es möglich, User Accounts, User Profile und Passwörter zwischen OS/400 und Windows zu synchronisieren.

Virtual Ethernet kann ohne ein externes LAN einen sicheren, leistungsstarken Bus Interconnect zwischen Windows, Linux und OS/400 herstellen.

Flexible Server-Entwicklung und Testhilfen (Testing Features) bieten die Möglichkeit, multiple Windows-Server Images auf der iSeries zu speichern (z. B. verschiedene Service Packs für Applikationen), und dann nur den erforderlichen Server auf einem Integrated xSeries Server zu booten. Eine einzige Backup-Methodologie für alle Windows-Server und OS/400 gewährleistet auch eine robuste Disaster-Recovery-Lösung. Microsoft Cluster Service unterstützt das dynamische Wechseln von virtuellen Speicherplätzen (Disks) zwischen Windows-Servern.

Der Integrated xSeries Server hat einen 2.0 GHz Intel Xeon Prozessor mit einem 512 KB L2 Cache, einen 400 MHz Front Side Bus (FSB) und einen onboard 10/100 Mbps Ethernet Controller und vier USB-Ports.

Der Integrated xSeries Adapter ist ein PCI-Adapter, der xSeries Server mit dem iSeries HSL-Bus verbindet und dem Server Virtual Storage sowie Ethernet zur Verfügung stellt. Der Integrated xSeries Adapter wird von den xSeries Servern unterstützt.

IBM unterstützt Windows-Server sowohl auf dem Integrated xSeries Adapter als auch auf dem Integrated xSeries Server und plant den Support von Windows .NET und Enterprise Servern.

Mit den neuen Systemen i5 Power5+ führt IBM eine neue offene Technologie ein, um Windows Server in das System i zu integrieren. iSCSI ist ein Storage-

over IP-Verfahren für Speichernetzwerke, das die Art und Weise spezifiziert, wie direkte Speicherprotokolle nativ über IP betrieben werden können. Bei diesem Verfahren werden SCSI-Daten in TCP/IP-Pakete verpackt und über IP-Netze transportiert. Bei der Kommunikation werden die verpackten SCSI-Kommandos zu einem SCSI-Router geschickt, der auf Basis vorhandener Mapping-Tabellen das entsprechende Zielsystem auswählt. iSCSI wird eingesetzt, um über eine virtuelle Ende-zu-Ende-Verbindung den Zugriff auf das Speichernetz zu ermöglichen, ohne dass eigene Speichergeräte aufgestellt werden müssen. Vorhandene Netzwerkkomponenten (Switch) können genutzt werden, da keine neue Hardware für die Knotenverbindungen nötig ist.

Die Spezifikation des iSCSI-Standard wurde durch die Storage Networking Industry Association (SNIA) erstellt. Die SNIA ist ein Zusammenschluss von mehr als 300 im Storage-Bereich aktiver Firmen.

Derzeit wird noch keine entsprechende Hardware für die Windows-Integration angeboten. Es ist aber zu erwarten, dass iSCSI die bisherigen PCI- und Opticonnect HSL-basierten Techniken ablösen wird.

Zuverlässige Managed Availability

Der iSeries Server hat eine zuverlässige Historie im Design von Schlüsselfunktionen für Hard- und Software. Hochverfügbarkeit ist ein Grund, weshalb ein Managed Availability-Ansatz gewählt werden sollte. Andere Gründe sind minimale Unterbrechungen bei Backup-Lösungen und die Tatsache, dass man in der Lage ist, ohne Unterbrechungen neue Versionen, Releases oder Software Fixes installieren und Pre-Tests durchführen zu können, um alle Unternehmens- und Systemressourcen optimal zu nutzen.

Die Kennzeichen der iSeries-Verfügbarkeit umfassen redundante, interne Hardware-Features, wie z. B. RAID-5 und Mirroring. Die Widerstandsfähigkeit und Stabilität von OS/400 dehnt sich auf seinen multiplen Subsystem-Support aus (Batch, Interaktivität, Multilingualität und Applikationen). Der iSeries Server bietet außerdem Managed Availability.

Die iSeries Managed Availability-Software wird auch Cross-System Mirroring genannt. Diese Software ermöglicht:

- dass ein System als Hot-Backup-System für ein oder mehrere Primärsysteme fungiert. Primär- und Sekundärsysteme müssen nicht die gleiche Größe haben oder das gleiche Modell sein.
- ein schnelles Umschalten zur Sekundärmaschine im Falle eines Notfalls.
- Neue Software-Versionen und -Releases (OS/400 und dazugehörige Software) oder Fixes können beim Sekundärsystem angewendet werden, während das Primärsystem weiterläuft. Das Testen kann auf dem Sekundärsystem durchgeführt werden, bevor das Primärsystem ohne Unterbrechung aktualisiert wird.

iSeries Server bieten herausragende Technologie, Service und Support für jede der fünf kritischen Availability-Komponenten:

- Single-System-Reliabilität: Architektur und Baseline Design machen den iSeries Server zu einem der zuverlässigsten Server weltweit. Seit ihren Anfängen weist die iSeries-Architektur ein Design auf, bei dem Reliabilität und Verfügbarkeit ihre Entsprechung in Features, wie z. B. Prozessorgeschwindigkeit, Speicherkapazität des Hauptspeichers, der Festplatten und Anzahl der Festplatten bei der Verfügbarkeitsplanung, haben.
- Das iSeries-Design und die Entwicklungsressourcen, die hohe Availability Levels in einem einzelnen Systemumfeld ermöglichen, können genutzt werden, um ungeplante Ausfallzeiten zu verhindern. Die Single-System-iSeries bleibt der zentrale Baustein, um Funktionen in anderen Geschäftsbereichen zu wiederholen und auf sie auszudehnen.
- Availability Management beim Single-System: iSeries Server haben Hochverfügbarkeitsfunktionen, die nicht nur schnell und automatisiert, sondern auch bedienerfreundlich sind. Geplante und ungeplante Ausfallzeiten werden mit Hochverfügbarkeitsfunktionen reduziert. Dazu gehören:
  - automatisiertes Journal-Management
  - Schutz des Zugriffspfads
  - Batch Journal Caching
  - Speichern-während-aktiv
  - Parallele Speicherung und Wiederherstellung
  - Backup Recovery und Media Services (BRMS) für iSeries
  - Teil- und Vollsicherung von RAID-5-Systemen
  - Festplatten-Mirroring-Schutz
- Clusters: Cluster-Technologie wird implementiert, um die Stillstandszeit, die durch geplante Ausfallzeiten und Site-Disaster verursacht wird, zu verringern. Die Systemverfügbarkeit während geplanter Ausfallzeit trägt zu einer Wachstumsabdeckung bei ungeplanter Ausfallzeit bei.
- Applikationen – Cluster-fähig: Eine Hochverfügbarkeitslösung für den iSeries Server umfasst eine aktive Partizipation von Cluster Middleware Providern. IBM Business Partner stellen fortschrittliches Cluster-Management und Data Resiliency Tools zur Verfügung. Solution Developers entwerfen Applikationen, um den Zustand einer Applikation während einer Ausfallzeit zu erhalten.
- Availability Service und Support: Als ein weltweit führender Lieferant von Unternehmenscomputern verfügt IBM über eine Palette von Produkten und Dienstleistungen, die dem Kunden helfen, ein Hochverfügbarkeitsumfeld zu entwickeln und zu pflegen. Die On Demand-Fähigkeiten

der Modelle 825, 870 und 890 umfassen Angebote für die Kontinuität der Geschäftsprozesse durch High Availability und Capacity BackUp.

Clustering mit umschaltbarem DASD und IASP

OS/400 mit HSL OptiConnect liefern eine umschaltbare Festplatte, die zwischen zwei Servern wechseln kann. V5R2 lässt auf einer HSL-Schleife drei Systeme zu. Mit unabhängigen, zusätzlichen Speicherpools (independent auxiliary storage pools, IASPs) und umgeschalteten Festplatten-Clusters kann man von einem zweiten System aus auf den Inhalt eines Sets von Festplatten-Einheiten zugreifen. Sowohl für geplante als auch für ungeplante Ausfallzeiten wird der Support verbessert, wenn bei dem System, das zu diesem Zeitpunkt eine umschaltbare Festplatte einsetzt, eine Ausfallzeit auftritt. Befehle werden auf einem System weiter ausgeführt, selbst wenn ein geschützter Controller eine geschützte Festplatteneinheit ausfällt.

Die Datenverfügbarkeit wird mit IASPs oder Switched Disk Clustering verbessert. Bei einer Ausfallzeit innerhalb eines Clusters können User auf einen alternierenden Knoten im Cluster (einem anderen iSeries Server) umgeschaltet werden. Integrated File System (IFS)-Daten und Bibliotheksobjekte des Betriebssystems, die sich in einem IASP befinden, können ohne ein IPL zu einem anderen iSeries Server geschaltet werden. So ist eine iSeries in der Lage, Daten und einen Input/Output-Regler von einer anderen iSeries in einen Festplatten-Tower zu übernehmen.

Die Primärfunktion in den frühen Stadien des Clusterings besteht darin, Abdeckung für geplante Upgrades und die Pflege des Produktionssystems zu bieten, ohne die User beim Zugriff auf Daten des umgeschalteten Festplatten-Towers zu beeinflussen und dies ausschließlich für das User-definierte Dateisystem (user-defined file system, UDFS). Cluster Management Middleware, die als Teil von OS/400 Option 41 (HA Switchable Resources) geliefert wird, regelt das Umschalten. Zum Zweck der Hochverfügbarkeit stellt sie sicher, dass zwei Systeme nicht gleichzeitig Zugriff auf die Festplatten (Daten) haben.

Ein gut entworfener umgeschalteter Festplatten-Cluster kann gegenüber einem Daten-Replication-Cluster Vorteile bieten. Da ein umgeschalteter Festplatten-Cluster keine Datenreplikation verwendet, tritt bei den Systemen weniger Overhead auf. Aufgrund dessen stehen mehr Ressourcen für die Bearbeitung von Transaktionen zur Verfügung. Es kann einfacher sein, einen umgeschalteten Festplatten-Cluster zu betreiben. Die Applikation ist entscheidend für das Design eines tatsächlich ununterbrochen verfügbaren Umfelds.

Domino nutzt diesen Support und setzt die umgeschaltete Festplatten-Architektur ein, um ClusterProven Spezifikationen zu erlangen. Umgeschaltete Festplatten beseitigen nicht die Erfordernis, über Application Resiliency zu verfügen.

Cross-Site Mirroring

Cross-Site Mirroring (XSM), auch geographisches Mirroring genannt, ermöglicht Ihnen, Daten auf Festplatten an unterschiedlichen Aufstellungsorten wiederzuspiegeln, die durch einen signifikanten geographischen Abstand getrennt werden können. Sie verwenden diese Technologie, um die Funktionalität einer Gerät-Cluster Resource Group (CRG) über die Einschränkungen einer physikalischen Komponentenverbindung hinaus auszudehnen.

Das geographische Mirroring bietet die Möglichkeit, Änderungen, die bei der Produktionskopie eines unabhängigen Festplatten-Pools (independent auxiliary storage pools, IASPs) vorgenommen wurden, auf eine Spiegelkopie dieses unabhängigen Festplatten-Pools zu replizieren. Wenn Daten in die Produktionskopie eines unabhängigen Festplatten-Pools geschrieben werden, spiegelt das Betriebssystem diese Daten über ein anderes System in eine zweite Kopie des unabhängigen Festplatten-Pools. Dieser Prozess behält mehrere, identische Kopien der Daten.

Sollte ein Failover oder ein Switchover auftreten, kann über die Gerät-CRG der Backup-Knoten nahtlos die Rolle des Primärknotens übernehmen. Der Server oder die Server, die als Backups dienen, werden in der Recovery Domain definiert. Die Backup-Knoten können sich an der gleichen oder an einer anderen physikalischen Position wie der Primärknoten befinden.

Wenn eine Ausfallzeit auf dem Server auftritt, der in der Recovery Domain als Primärknoten definiert ist und ein Switchover oder Failover initiiert wird, dann wird der in der Recovery Domain als Backup festgelegte Knoten der primäre Zugriffspunkt für die Quelle und enthält dann die Produktionskopie des unabhängigen Festplatten-Pools. So können Sie sich vor der einzigen Stelle, an der Fehler auftreten können, die im Zusammenhang mit umschaltbaren Ressourcen stehen, schützen.

Dynamische, logische Partitionierung

Die logische Partitionierung (LPAR) verbessert die Rolle der iSeries als konsolidierter Server. Mit LPAR verfügen Unternehmen sowohl über das Potential als auch über die Flexibilität, um sich multiplen Systemanforderungen in einer einzelnen Maschine zu widmen.

Server-Virtualisierung, eine Bezeichnung, die oft im Zusammenhang mit Partitionierung verwendet wird, erreicht man auf der iSeries über den Einsatz von Hypervisor-Technologie. Hypervisor umfasst eine Kombination aus Hardware-Features und Control Code. eServer i5 Server setzen POWER Hypervisor ein, um diesen Virtualisierungssupport zu liefern.

LPAR, so wie es auf der iSeries implementiert ist, weitet das ursprüngliche architektonische Designkonzept der Applikationsdurchführung aus, indem zugelassen wird, dass in jeder beliebigen Partition entweder OS/400, Linux

oder AIX laufen kann. Das extensiv dynamisch und granulierte Teilen von Ressourcen bei Prozessoren (SMP Konfigurationen), Speicher, Festplatte, Tape und andere Geräte, einschließlich virtuelle LANs erlaubt. Multiple Partitionen werden bei ausgewählten iSeries Uni-Prozessor-Modellen gestützt.

1519-100 Integrated xSeries Adapter für iSeries (direkte Anbindung)

Der IBM Integrated xSeries Adapter für iSeries bietet einen direkten High-Speed Link (HSL) von einem xSeries Server zu einem iSeries Server. Er ist in ausgewählten xSeries Servern installiert. Der Integrated xSeries Adapter weitet die iSeries-Integration mit Microsoft Windows-Servern auf xSeries High Performance n-Way Intel-Architekturserver aus. Mit dem Integrated xSeries Adapter können mehr Windows-User und komplexere Windows-Applikationen mit iSeries Servern integriert werden.

Der direkte Anbindungsserver besteht aus einem xSeries Server Tower, der einen Integrated xSeries Adapter für iSeries (1519-100) enthält. Der Integrated xSeries Adapter berichtet dem iSeries Hardware Service Manager.

Der externe xSeries Server hat SPCN Steuerung. SPCN, das den externen xSeries Server verbindet, folgt den gleichen Richtlinien wie SPCN, das für bestehende HSL-angebundene Towers verbindet.

Die Zahl der Integrated xSeries Servers und Integrated xSeries Adapters, die angebunden werden können, variiert je nach Modell, HSL-Schleifen und der Anzahl von Input/Output-Tower. Das Modell 810 unterstützt beispielsweise bis zu 13 Integrated xSeries Server und sieben Integrated xSeries Adapter. Das Modell 890 unterstützt bis zu 48 Integrated xSeries Server und 60 Integrated xSeries Adapter.

iSeries Windows Integration und Microsoft Cluster-Support

Die Funktionen des iSeries Integrated xSeries Servers und des Integrated xSeries Adapters unterstützen Virtual Ethernet LAN, Microsoft Cluster Services und Automatic Cartridge Loader-Support. Mit V5R2 werden jetzt mittels Integrated xSeries Adapter neue xSeries-Modelle unterstützt.

Virtuelles Ethernet LAN

Das Virtual LAN, das bei V5R1 eingeführt wurde, um Hochgeschwindigkeitskommunikation zwischen OS/400- und Linux-Partitionen innerhalb des iSeries Servers zu ermöglichen, wird ausgeweitet, um Integrated xSeries Server und xSeries Server, die an Integrated xSeries Adapter angebunden sind, zu unterstützen. Mit diesem Support können Windows-Server miteinander und mit OS/400- und Linux-Partitionen über die schnellen, sichereren und zuverlässigeren Virtual Ethernet LANs kommunizieren.

Microsoft Cluster Services

Mit dem Clustering Support, das z.B. bei Windows 2000 Advanced Servern zur Verfügung steht, können zwei Integrated xSeries Server oder zwei xSeries Server, die mit Integrated xSeries Adaptern verbunden sind, einen Cluster bilden und die 16 neuen, gemeinsamen Speicherplätze, die bei V5R2 zur Verfügung stehen, nutzen. Im Cluster-Umfeld können im Fall einer Ausfallzeit bei einem Windows-Server die Speicherplätze auf den zweiten Windows-Server umgeschaltet werden. Die Applikationen können automatisch neu gestartet werden, um die Dauer des Systemausfalls zu reduzieren.

Auto Cartridge Loader

Kunden, die Systeme mit großen Datenmengen haben, setzen häufig Auto Cartridge Loader (ACL) Tape Devices (3570, 3580 und 3590) ein, die in der Lage sind, eine andere Tape Cartridge automatisch zu laden. Der Windows Integration Support wird jetzt durch einen Support für die Handhabung von Befehlen für ACLs ergänzt. Auf mehrere Tape Cartridges kann während Backup/Wiederherstellungsoperationen, die von dem Windows-Server initiiert wurden, zugegriffen werden.

Empfohlene Lektüre

Zusätzliche Informationen über Windows Integration finden Sie auf folgender Web-Seite:

http://www-1.ibm.com/servers/eServer/iseries/windowsintegration

Integration mit Windows-Servern

Die iSeries Server bieten eine Integration mit Windows, um größere und komplexere Windows-Applikationen sowie zusätzliche Tools zu unterstützen und damit einen Beitrag zur Senkung der Kosten für die Handhabung eines Windows-Server-Umfelds zu leisten.

Anbindung von N-Way xSeries Servern

Die iSeries stützt die Anbindung von N-Way xSeries Servern über den High-Speed Link. Mit dem Integrated xSeries Adapter können ausgewählte xSeries Server, die Windows-Server ausführen, eingesetzt werden, um die Skalierbarkeit von Windows-Applikationen zu erweitern und gleichzeitig die Speicherkonsolidierung und die Vorteile des Systemmanagements des Integrated xSeries Servers beizubehalten.

Verbesserter Hardware-Support

4.1.2
Seite 8

Verbesserungen für den Integrated xSeries Server umfassen den Support für:

- bist zu 32 Server auf ausgewählten iSeries Modellen
- 1 Gb Ethernet LAN-Adapter
- iSeries DVD Device

Die Hardware Management Console (HMC)

Um die nativen Betriebssysteme i5/OS, Linux und AIX auf dem System i installieren zu können, müssen für die jeweiligen Betriebssysteme logische Partitionen (LPAR) eingerichtet werden. Seit der Baureihe 5xx kann die Partitionierung eines System i durch zwei verschiedene Werkzeuge realisiert werden:

- Die Hardware Management Console (im folgenden HMC), ersetzt die für das Management der Partitionen als praktisch angesehene und die für die vorherigen Modelle notwendige Primary-Partition. Die HMC bietet darüber hinaus den Vorteil, dass sie gleichzeitig Konsole für alle Partitionen eines Systems sein kann. Weiterhin kann die HMC mehr als ein System i oder System p. Sie kann ebenfalls Remote über ein Intranet oder Extranet erreicht werden, so dass diese Technik, auch wenn sie komplexer ist als die „alte“ Primary Partition-basierte Technik, insgesamt eine sehr komfortable Partitionsverwaltung ermöglicht.
- Virtual Partition Manager. Nachdem die ersten 5xx Modelle für die Partitionierung eine HMC benötigten, die HMC jedoch je nach Modell den Preis eines Systems i erhöhten, hat IBM den Partition Manager eingeführt. Diese Software ist Bestandteil der System Service Tools (SST) und ermöglicht ebenfalls die Partitionierung von Systemen. Die Partitionierungsaufgaben können jedoch nur für System i Hardware ausgeführt werden.

Im Folgenden finden Sie eine kleine Tabelle der IBM, die Ihnen die Unterschiede beider Technologien aufzeigt:

	Virtual Partition Manager	Hardware Management Console
Operating Systems supported	i5/OS and Linux	i5/OS, Linux and AIX 5L
Maximum number of partitions	5 (1 i5/OS + 4 Linux)	254
Uncapped partition support	Yes	Yes
Dynamic resource movement	No	Yes
I/O support for Linux partitions	Virtual	Virtual and Direct
Maximum # of Virtual Ethernet connections	4	4096
Maximum Virtual Disk per partition	64TB	64TB

Quelle: *http://www-03.ibm.com/servers/eserver/iseries/linux/vpm.html*

4.2
Seite 2

Die Hardware

Was ist eine HMC? Sie ist ein IBM-PC, den es entweder als Desktop-Variante (aktuell:7310-C04 vorher: 7310-C03, C02, C01) oder als Rack-mounted (aktuell: 7310-CR3 vorher 7310-CR2, CR1) gibt. Auf diesem PC finden wir ein vorinstalliertes RedHat Linux, auf das wir jedoch nur mit Hilfe des IBM-Service zugreifen können, für uns also insgesamt eine Black Box darstellt.

In Linux wird dann, ebenfalls vorinstalliert, die HMC-Software aufgerufen, die uns das Administrieren eines Systems und seiner Partitionen erlaubt.

4.2.1 Seite 2

HMC verkabeln

Achtung, Achtung!

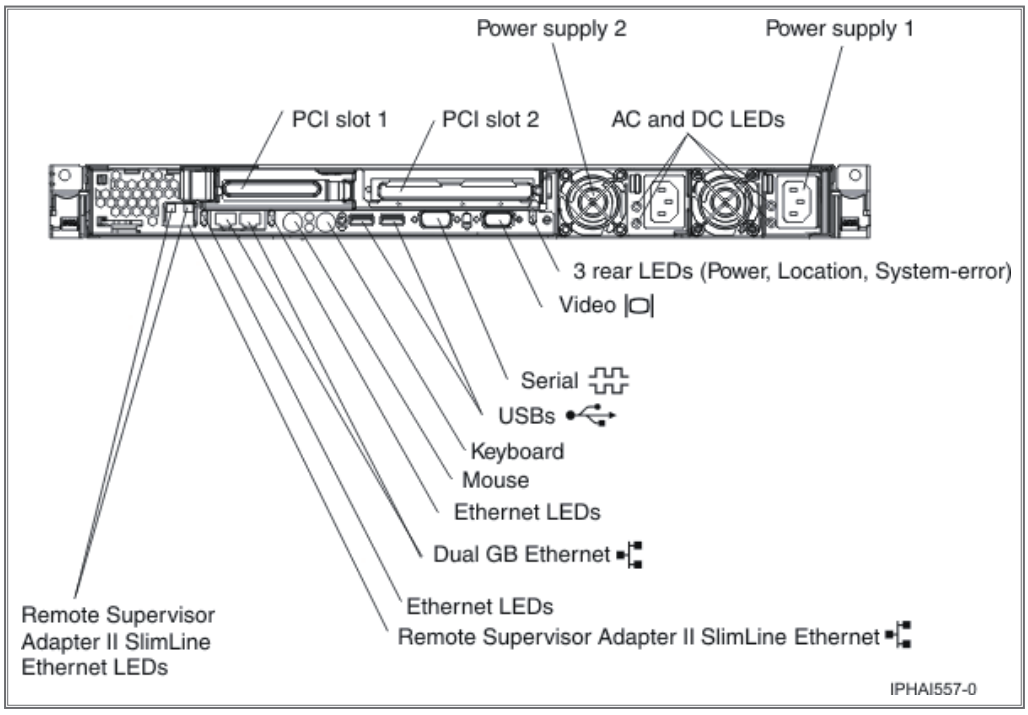
Beabsichtigen Sie eine iSeries mit HMC-Konsole in Betrieb zu nehmen, so machen Sie sich das Leben deutlich einfacher, wenn Sie die iSeries **nicht** an das Stromnetz anschließen. Setzen Sie Ihre iSeries erst dann unter Strom, wenn die HMC konfiguriert worden ist und betriebsbereit ist!

Nachdem Sie Ihre HMC ausgepackt und die Vollständigkeit der Lieferung überprüft haben, werden Sie vor die Aufgabe gestellt, die HMC mit Ihrem System i und dem lokalen Netzwerk zu verbinden.

Dabei ist es üblich die HMC über ein so genanntes privates Netzwerk direkt mittels eines Ethernet-Kabels mit einer oder mehreren Systemen i / iSeries zu verbinden. So gehen Sie dabei vor:

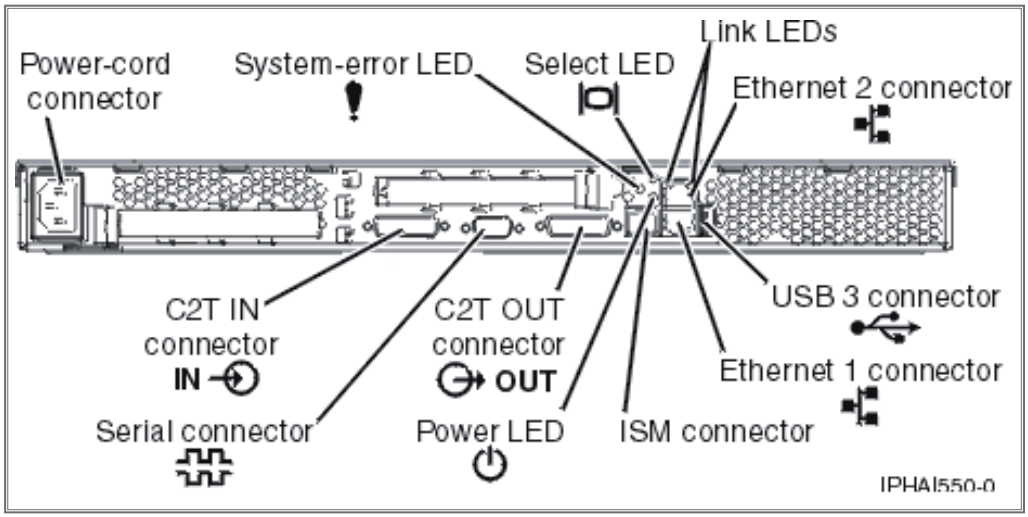
Rack-Installation

Verwenden Sie die folgenden Abbildungen, um die Position der in diesen Anweisungen beschriebenen Anschlüsse zu ermitteln:



Rückansicht einer in ein Rack eingebauten HMC (7310-CR3)

4.2.2
Seite 2



Rückansicht einer in ein Rack eingebauten HMC (7310-CR2)

Installieren Sie die HMC in einem Rack.

Schließen Sie Monitor, Tastatur und Maus an:

Für den Anschluss an eine HMC, Modell 7310-CR2, schließen Sie Tastatur und Bildschirm an das C2T-zu-KVM-Adapter-Verteilerkabel (Tastatur, Video, Maus) an, das Sie zuvor an die HMC angeschlossen haben. Die Maus ist in die Tastatur integriert.

Wenn Sie einen eigenständigen Monitor, Tastatur und Maus verwenden:

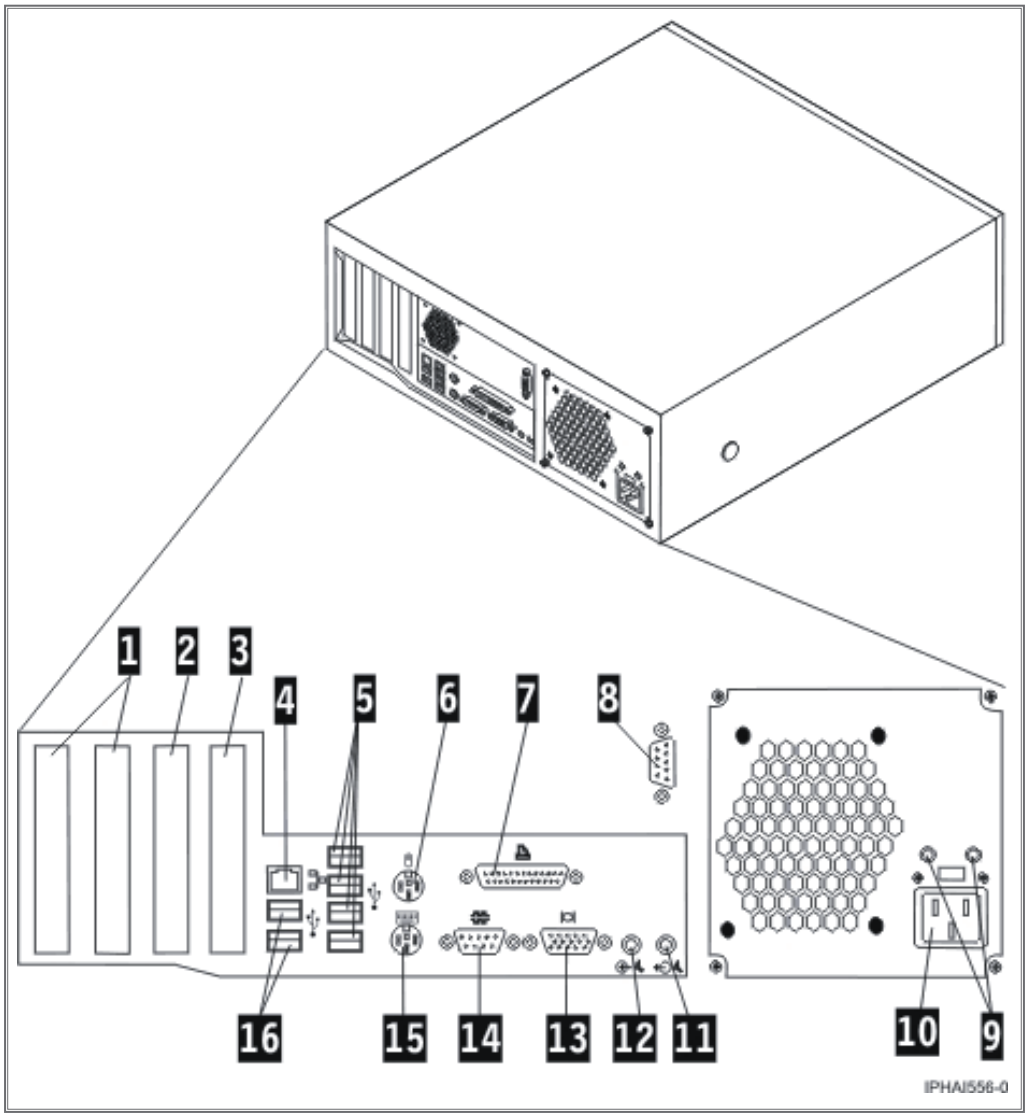
Für den Anschluss an eine HMC, Modell 7310-CR2, schließen Sie Tastatur und Bildschirm an das C2T-zu-KVM-Adapter-Verteilerkabel (Tastatur, Video, Maus) an, das Sie zuvor an die HMC angeschlossen haben. Wenn für Ihre Tastatur und Maus USB-Anschlüsse erforderlich sind, können Sie sie auch an die USB-Anschlüsse an der Frontverkleidung der HMC anschließen.

Für den Anschluss an eine HMC, Modell 7310-CR3, schließen Sie Tastatur, Bildschirm und Maus mit dem USB-Konvertierungskabel an.

Installation Stand-alone HMC

Führen Sie die folgenden Schritte aus, wenn Sie eine eigenständige HMC installieren:

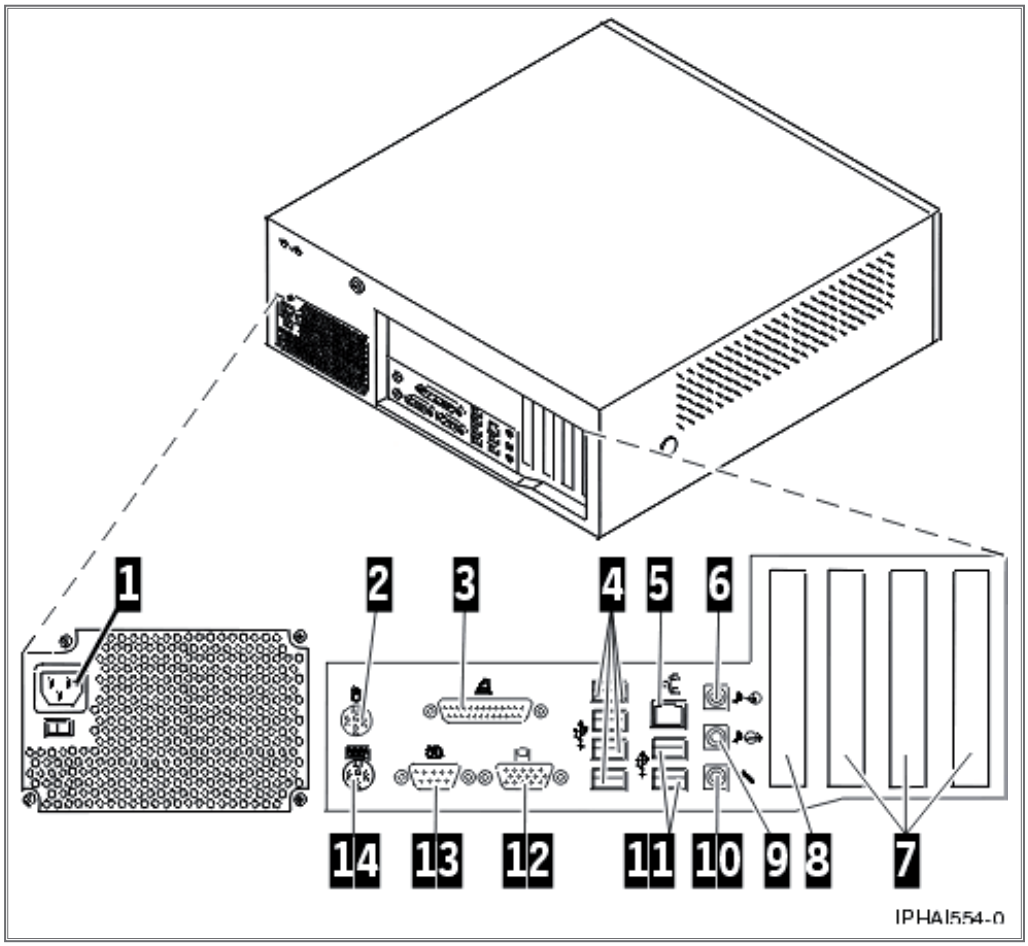
Verwenden Sie die folgenden Abbildungen, um die Position der in diesen Anweisungen beschriebenen Anschlüsse zu ermitteln:



Rückansicht einer eigenständigen HMC (7310-C04)

1 PCI-Anschlüsse (Steckplatz 1 links)	9 Diagnose-LEDs
2 PCI Express-Anschluss (x1)	10 Netzteilanschluss
3 PCI Express-Grafikanschluss (x16)	11 Audioausgang
4 Ethernet-Anschluss	12 Audioeingang
5 USB-Anschlüsse	13 VGA-Monitoranschluss
6 Mausanschluss	14 Systemanschluss
7 Parallelanschluss	15 Tastaturanschluss
8 Systemanschluss	16 USB-Anschlüsse

4.2.2
Seite 4



Rückansicht einer eigenständigen HMC (7310-C03)

1 Netzteilanschluss	8 AGP-Steckplatz
2 Mausanschluss	9 Audioausgang
3 Parallelanschluss	10 Mikrofonanschluss
4 USB-Anschlüsse	11 USB-Anschlüsse
5 Ethernet-Anschluss	12 VGA-Monitoranschluss
6 Audioeingang	13 Systemanschluss
7 PCI-Steckplätze (Steckplatz 1 rechts)	14 Tastaturanschluss

Verbinden Sie das Bildschirmkabel mit dem Bildschirmanschluss und ziehen Sie die Schrauben fest.

Schließen Sie das Netzkabel am Bildschirm an.

Vergewissern Sie sich, dass der Spannungsauswahlschalter der HMC auf die Spannung eingestellt ist, die in Ihrer Region verwendet wird. Der Spannungsauswahlschalter ist rot und befindet sich in der Nähe des Netzteilanschlusses. Bewegen Sie den Schalter, so dass die in Ihrer Region verwendete Spannung angezeigt wird.

Schließen Sie das Netzkabel an die HMC an.

Schließen Sie Tastatur und Maus an:

USB-Anschlüsse: Schließen Sie Tastatur und Maus an die USB-Anschlüsse (USB = Universal Serial Bus) der HMC an. Sie können Tastatur und Maus an die USB-Anschlüsse an der Vorder- oder Rückseite anschließen.

Anmerkung:

Wenn Sie mit einer eigenständigen HMC, Modell 7310-C01 oder 7310-C02 arbeiten, schließen Sie Tastatur und Maus nur an die USB-Anschlüsse an der Vorderseite an.

PS/2-Anschlüsse: Schließen Sie Maus und Tastatur an den jeweiligen Anschluss an der Rückseite der HMC an.

4.2.2
Seite 6

Netzwerkverbindung erstellen

Um das System i mit der HMC physikalisch zu verbinden, finden Sie auf der Rückseite des Systems i zwei Ethernet-Ports die mit HMC1 und HMC2 bezeichnet sind.

Stecken Sie das Ethernet-Kabel in den Port HMC1. Das andere Ende des Ethernet-Kabels muss jetzt in einen Ethernet-Port der HMC gesteckt werden. Je nach Modell und Konfiguration, werden Sie ein bis drei Ethernet-Ports finden. Netzwerk- und Linux-technisch gesprochen müssen Sie das Ethernet-Kabel in den HMC-Port eth0 stecken.

Das ist nicht so leicht wie es sich liest, da IBM die physikalischen Ethernet-Ports nicht im Rahmen dieser Linux-Nomenklatur bezeichnet.

So finden Sie heraus, welches in Ihrer HMC der Port eth0 ist:

Wenn Sie keine zusätzlichen Ethernet-Adapter in den PCI-Steckplätzen Ihrer HMC installiert haben, ist der primäre integrierte Ethernet-Anschluss immer als eth0 definiert.

Wenn Sie zusätzliche Ethernet-Adapter in den PCI-Steckplätzen installiert haben, richtet sich der Anschluss, der als eth0 definiert ist, nach der Position und dem Typ des installierten Ethernet-Adapters:

HMC-Typ	Regeln für Ethernet-Position
In Rack eingebaut	Die HMC unterstützt nur einen zusätzlichen Ethernet-Adapter. Wenn ein zusätzlicher Ethernet-Adapter installiert ist, ist dieser Anschluss als eth0 definiert. In diesem Fall ist der primäre integrierte Ethernet-Anschluss dann als eth1 definiert, und der sekundäre integrierte Ethernet-Anschluss ist als eth2 definiert. Wenn keine Adapterkarten installiert sind, ist der primäre integrierte Ethernet-Anschluss als eth0 definiert.
Eigenständiges Modell 7310-C04	Die Definitionen richten sich nach dem Typ des installierten Ethernet-Adapters: Wenn nur ein Ethernet-Adapter installiert ist (unabhängig davon, ob es sich um einen 1-Gigabit-Ethernet-Adapter oder einen 10/100-Ethernet-Adapter handelt), ist dieser Adapter als eth0 definiert. Wenn sowohl ein 10/100-Ethernet-Adapter als auch ein 1-Gigabit-Ethernet-Adapter installiert ist, wird der 1-Gigabit-Adapter immer als eth0 definiert. Wenn zwei 10/100-Ethernet-Adapter installiert sind, ist der Adapter in Steckplatz 1 als eth0 definiert. Wenn zwei 1-Gigabit-Ethernet-Adapter installiert sind, ist der Adapter in Steckplatz 1 als eth0 definiert.

4.2.3
Seite 2

HMC-Typ	Regeln für Ethernet-Position
Eigenständiges Modell 7310-C03	Die Definitionen richten sich nach dem Typ des installierten Ethernet-Adapters: Wenn vorhanden, ist der 1-Gigabit-Ethernet-Adapter gewöhnlich als Position eth0 definiert. Eine Ausnahme von dieser Regel besteht dann, wenn Sie in Steckplatz 1 (der am weitesten rechts liegende PCI-Steckplatz, wenn die HMC von der Rückseite betrachtet wird) platziert ist. Diese Position wird allerdings nicht empfohlen. Wenn mehrere 1-Gigabit-Ethernet-Adapter installiert sind, wird die Konfiguration in der folgenden Reihenfolge definiert: Steckplatz 2 ist eth0, Steckplatz 3 ist eth1 und der integrierte Ethernet-Anschluss ist eth2. Wenn andere Adapter als der 1-Gigabit-Ethernet-Adapter installiert sind, wird der integrierte Ethernet-Anschluss immer als eth0 definiert.

Sie werden sich fragen, warum es im Bereich der Ethernet-Schnittstellen so viele Schnittstellen gibt. Die Antwort liegt in einem Wort: Flexibilität.

Sie sollten daran denken, dass die HMC nicht einfach ein Ersatz für die DSP01-Systemkonsole ist, sondern das sie ein Administrations-Tool ist, um eine oder mehrere Partitionen auf einem oder mehreren Systemen (sowohl System i als auch System p) verwalten zu können.

System i Ports – Service Prozessor – DHCP-Client

An Ihrem System i finden Sie zwei Ethernet-Schnittstellen, die mit dem Namen HMC1 und HMC2 gekennzeichnet werden.

Diese beiden Ethernet Ports sind die Zugänge zum so genannten Service Prozessor (SP). Im SP wird die System i-seitige Logik der HMC ausgeführt. Betrachten Sie den SP einfach als Eingang in das durch die HMC zu verwaltende System.

Der SP wird mittels Ethernet mit der HMC verbunden. Der SP arbeitet als DHCP-Client der HMC und dieser wiederum tritt als DHCP-Server für den SP auf.

Achtung!

Deshalb ist es so wichtig, dass Sie vor Inbetriebnahme des Systems i zuerst die HMC vollständig in Betrieb nehmen und konfigurieren.

Um jetzt beide Systeme, HMC und System i, miteinander verbinden zu können, müssen wir noch über verschiedene Verbindungsvarianten sprechen.

4.2.3.1

Seite 2

Private und offene Netze in der HMC-Umgebung

Die Verbindung zwischen der HMC und ihren verwalteten Systemen kann entweder als privates oder als offenes Netz implementiert werden. Der Terminus „offen“ bezieht sich auf ein allgemeines, öffentliches Netz, das andere Elemente als HMCs und Serviceprozessoren enthält und nicht hinter einer HMC isoliert ist. Die anderen Netzverbindungen der HMC werden als offen betrachtet. Das bedeutet, dass sie auf eine Art konfiguriert sind, die Sie beim Anschluss einer Standardnetzeinheit an ein offenes Netz erwarten würden.

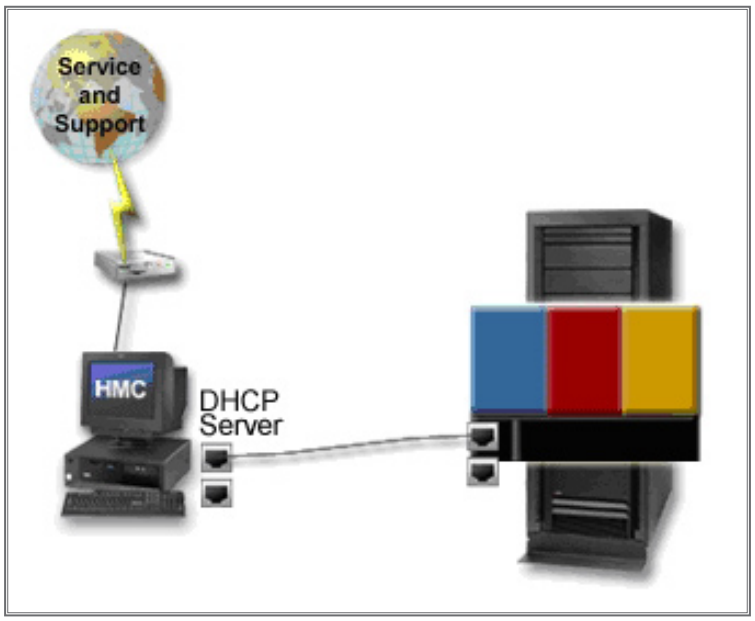
In einem privaten Service-Netz sind die HMC und die Serviceprozessoren der verwalteten Systeme die einzigen Elemente im physischen Netz. Außerdem stellt die HMC DHCP-Services (Dynamic Host Configuration Protocol) auf diesem Netz bereit, mit denen sie automatisch IP-Konfigurationsparameter erkennen und diesen Serviceprozessoren zuweisen kann. Sie können bei der Konfiguration der HMC einen von mehreren Adressbereichen für diesen DHCP-Service auswählen, so dass die Adressen, die den Serviceprozessoren bereitgestellt werden, nicht mit Adressen anderer Netze in Konflikt geraten, mit denen die HMC verbunden ist. Durch die DHCP-Services können die Elemente im privaten Netz automatisch von der HMC konfiguriert und erkannt werden, während gleichzeitig Adresskonflikte im Netz vermieden werden.

In einem privaten Netz werden daher alle Elemente von der HMC gesteuert und verwaltet. Die HMC fungiert auch als funktionale Firewall, die das private Netz von den anderen offenen Netzen isoliert, an die die HMC ebenfalls angeschlossen ist. Die HMC lässt keine IP-Weiterleitung zu; Clients auf einer Netzschnittstelle der HMC können nicht direkt auf Elemente einer anderen Netzschnittstelle zugreifen.

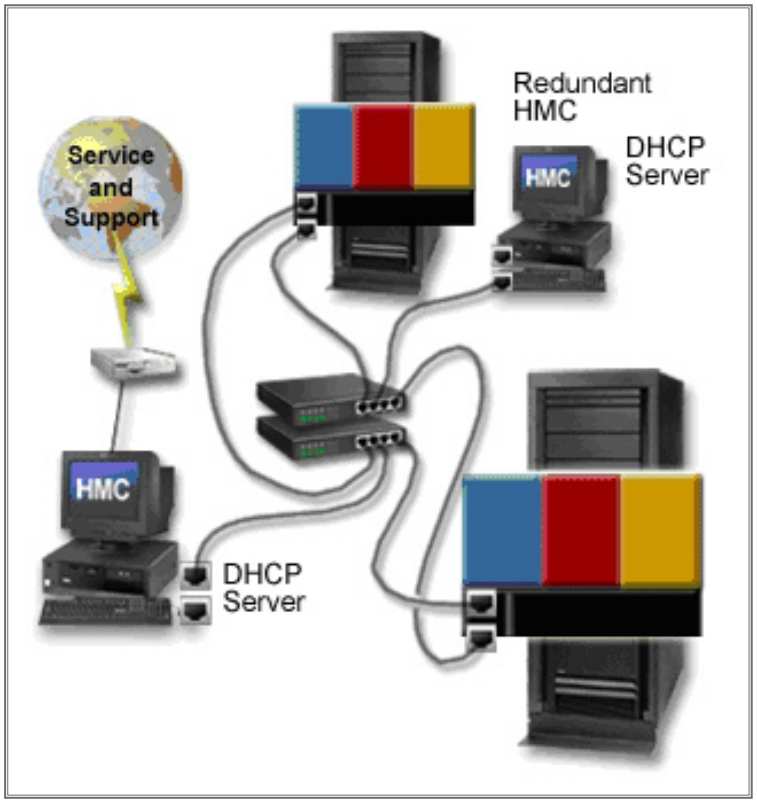
Um die Vorteile der zusätzlichen Sicherheit und der einfachen Konfiguration nutzen zu können, implementieren Sie die Service-Netzkommunikation über ein privates Netz. In manchen Umgebungen ist dies auf Grund von Überlegungen hinsichtlich der physischen Verkabelung, des Aufstellungsplans oder der Steuerzentrale jedoch nicht machbar. In diesem Fall kann die Service-Netzkommunikation über ein offenes Netz implementiert werden. In beiden Netztypen ist dieselbe Funktionalität verfügbar, obwohl bei der Erstkonfiguration in einem offenen Netz mehr manuelle Schritte erforderlich sind.

4.2.3.2
Seite 2

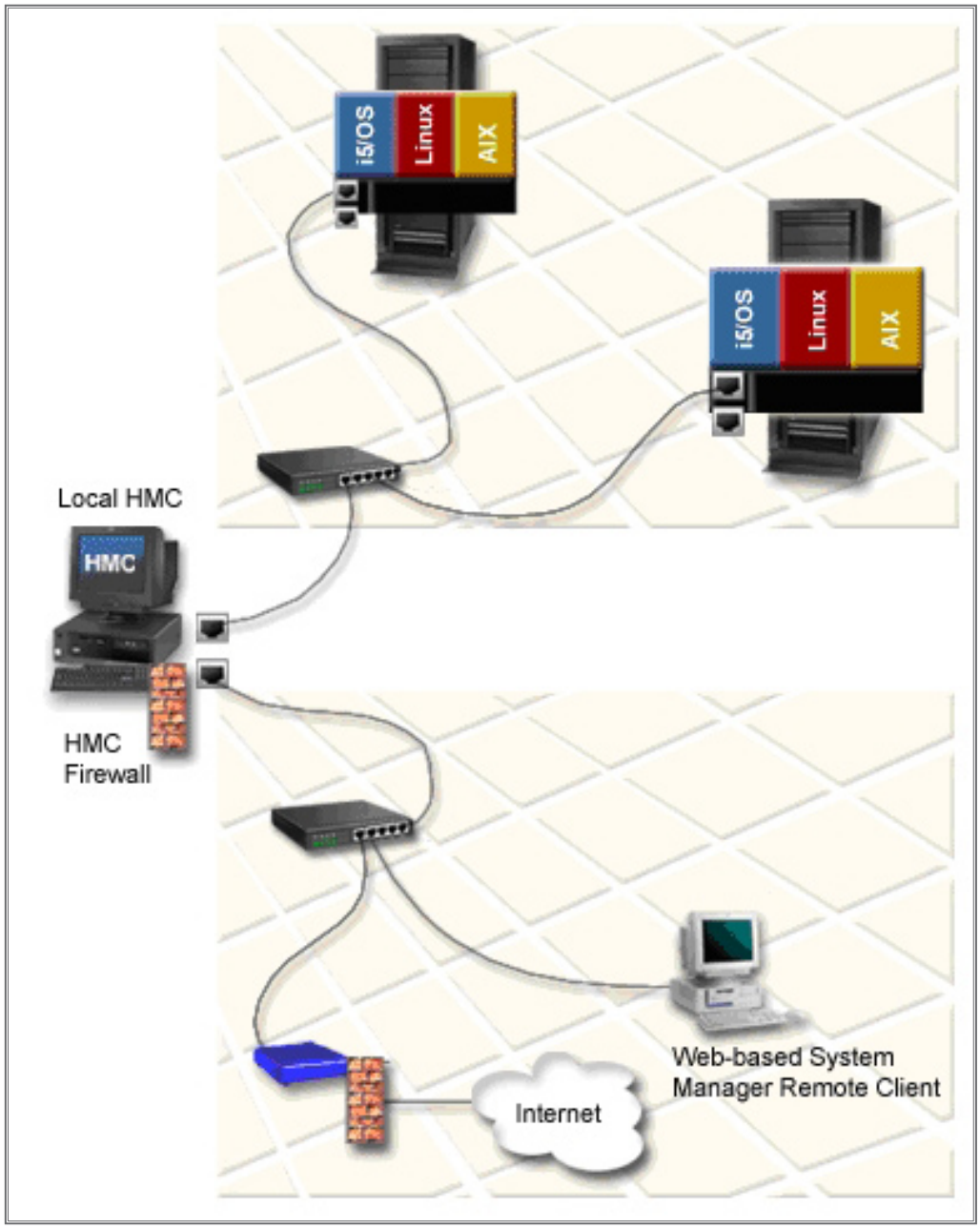
Die folgenden Abbildungen zeigen verschiedene private und offene Netze:



Privates Netz: Direktverbindung



Privates Netz: indirekte Verbindung



Offenes Netz

4.2.3.2

Seite 4

HMC Anfangskonfiguration erstellen

Nachdem Sie die HMC aufgebaut haben, Sie diese mit Tastatur, Maus und Bildschirm verbunden haben und Sie sodann angeschaltet haben, wird das „on Board Linux“ gestartet.

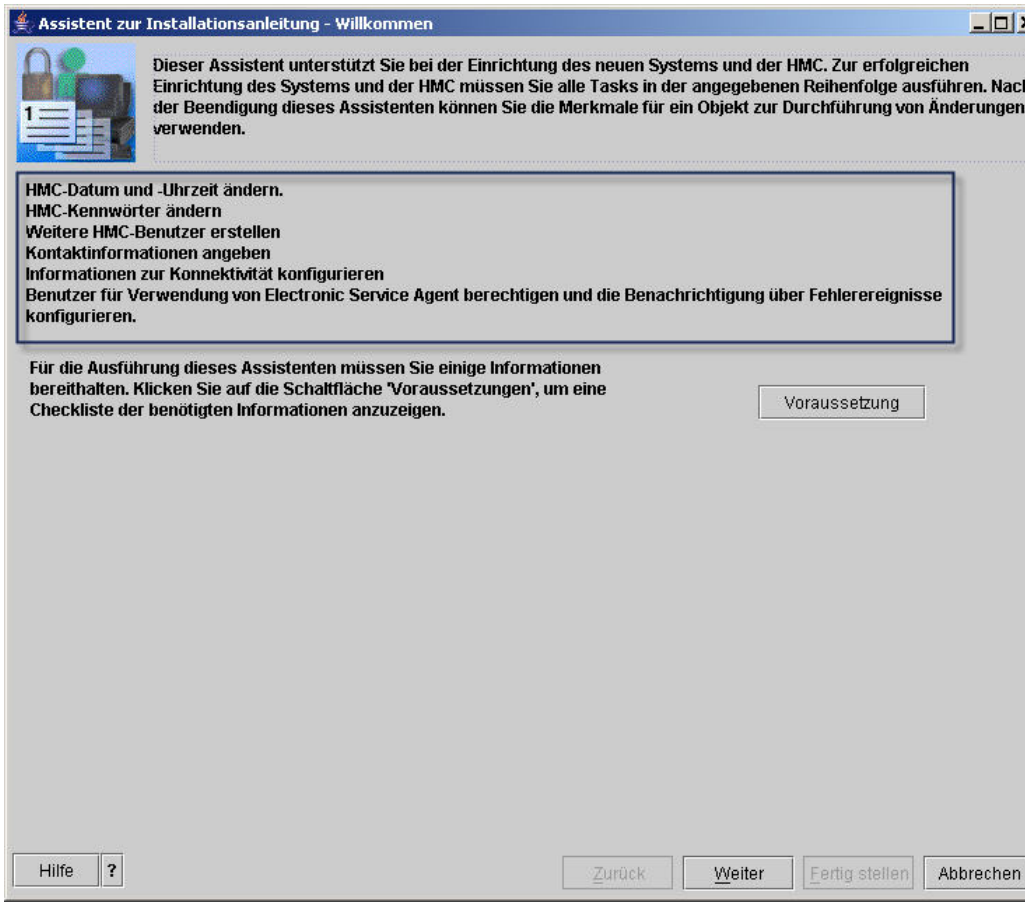
Nach einem erfolgreichen Start meldet sich die HMC mit einem grafischen Anmeldungsdialog.

Melden Sie sich hier mit folgendem Benutzer an:

Benutzer: hscroot
Kennwort: abc123

Achten Sie auf Kleinschreibung, Sie befinden sich im UNIX/LINUX-Umfeld. Achten Sie weiter darauf, dass der Benutzer „hscroot“, der mächtigste Benutzer der HMC ist. Mit ihm können alle nur denkbaren und wahrscheinlich später auch alle undenkbaren Konfigurationen durchgeführt werden. Mit dem Benutzer hscroot sind Sie sozusagen der QSECOFR der HMC!

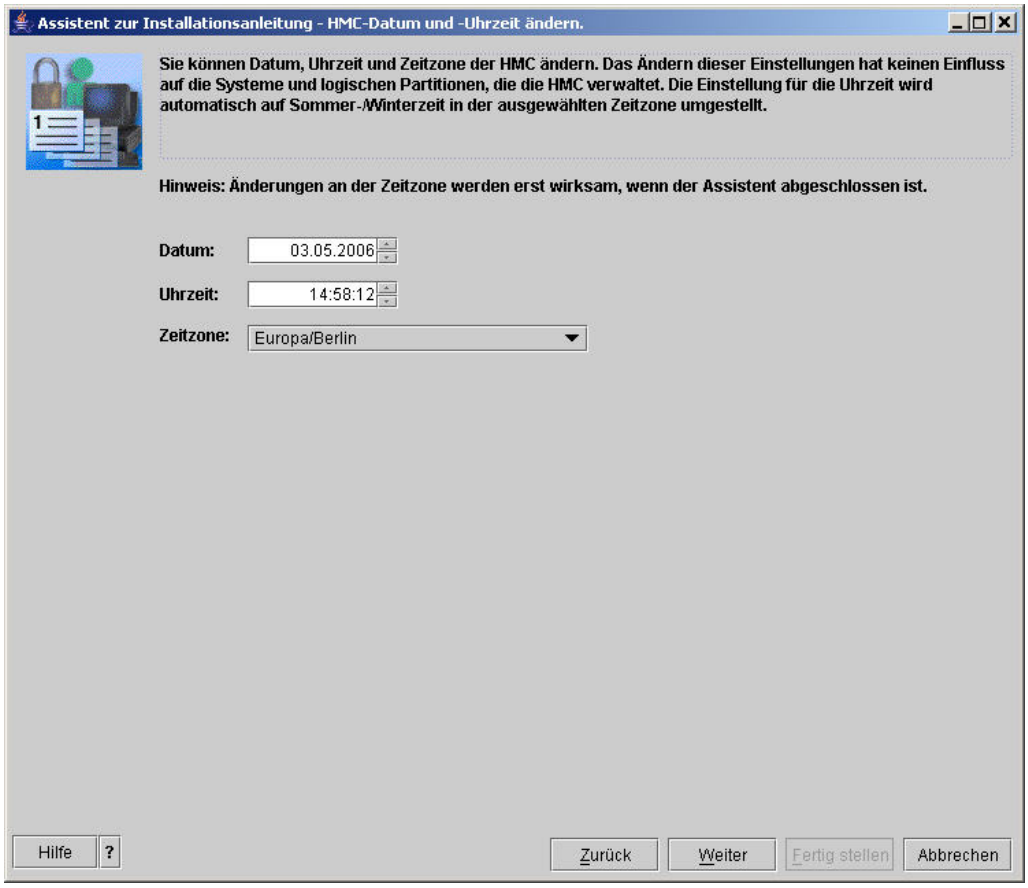
Nach der Anmeldung wird sofort der Assistent zur Installationsanleitung (Guided Setup Wizzard) gestartet.



Assistent zur Installationsanleitung

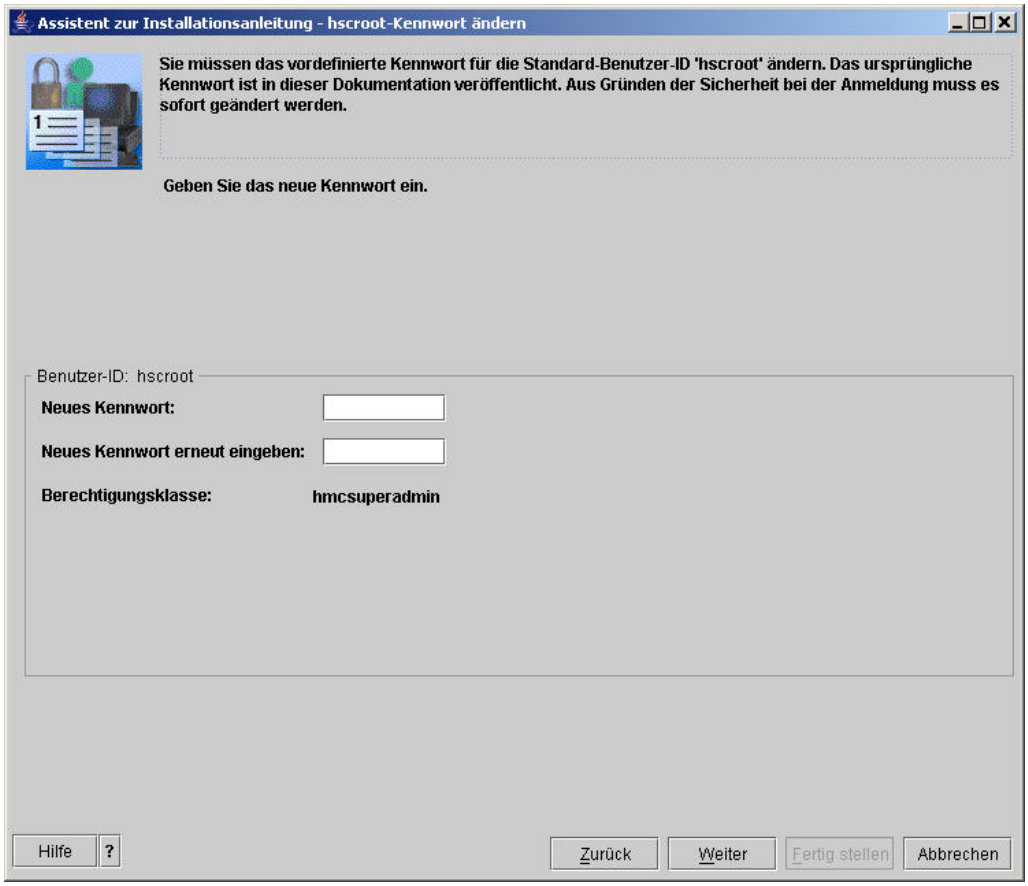
4.2.4
Seite 2

Dieser Dialog gibt Aufschluss darüber, welche anfänglichen Konfigurationen von Ihnen zu erledigen sind.



Uhrzeit, Datum einstellen

Prüfen und ändern Sie ggf. die Einstellungen für Uhrzeit, Datum und Zeitzone. Klicken Sie auf „Weiter“.



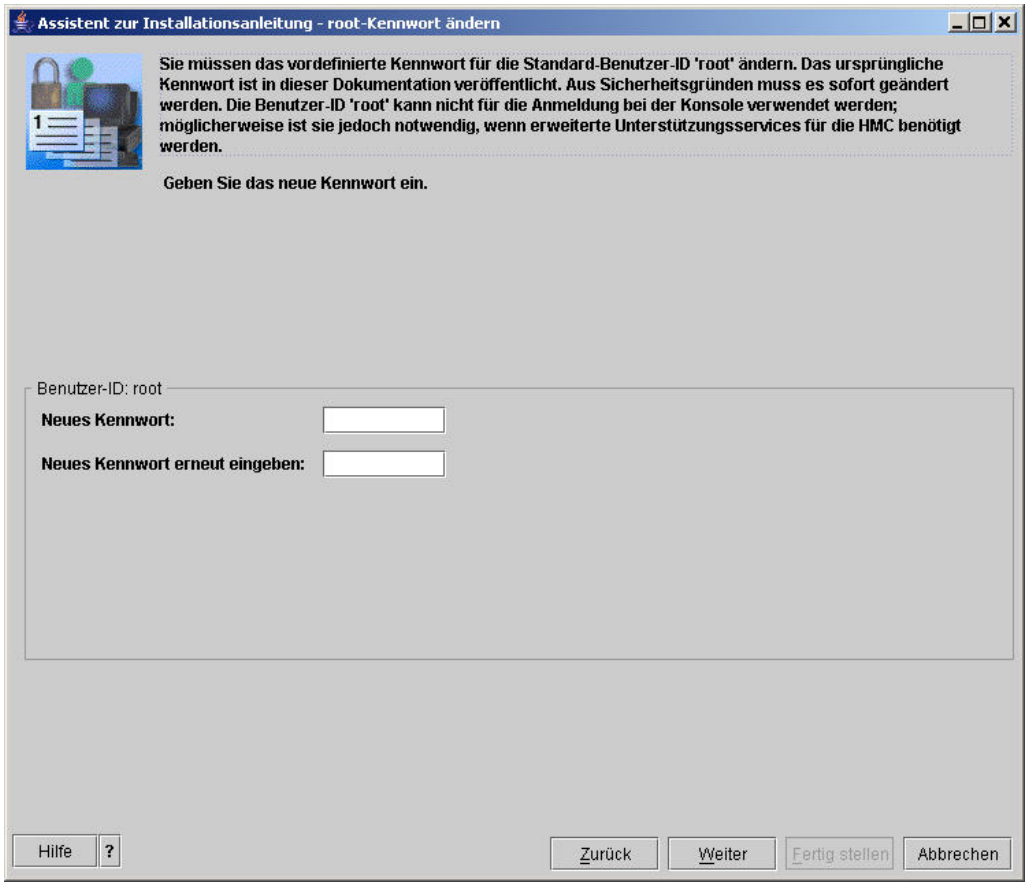
hscroot-Kennwort ändern

Ändern Sie das Kennwort für hscroot:

Geben Sie das neue Kennwort zwei Mal ein. Das Kennwort muss mindestens sieben Zeichen lang sein.

Klicken Sie auf „Weiter“.

4.2.4
Seite 4



Root-Kennwort ändern

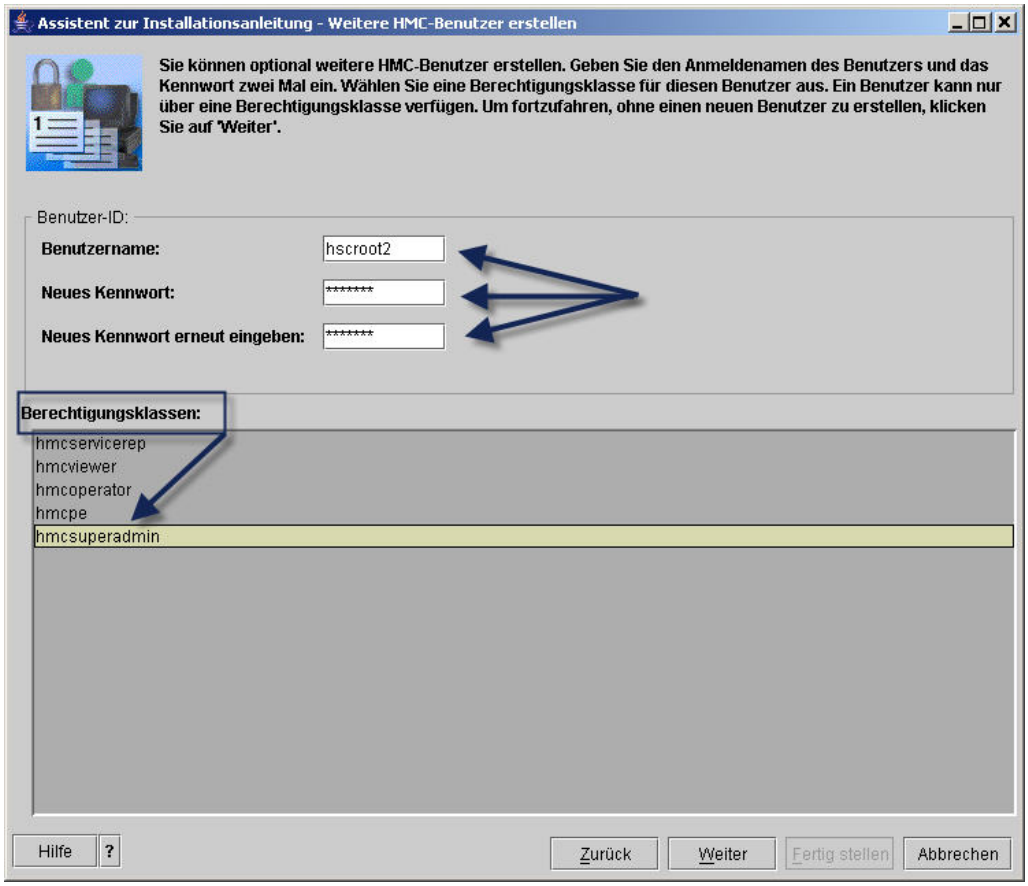
Ändern Sie das Kennwort für „root“:

Geben Sie das neue Kennwort zwei Mal ein. Das Kennwort muss mindestens sieben Zeichen lang sein.

Klicken Sie auf „Weiter“.

Klicken Sie noch einmal auf „Weiter“.

Erzeugen Sie sodann weitere HMC-Benutzer, wie z.B. einen Stellvertreter für hscroot.



hscroot-Stellvertreter einrichten

Ich vergebe hier folgende Werte:

Benutzername: hscroot2
Neues Kennwort: *******

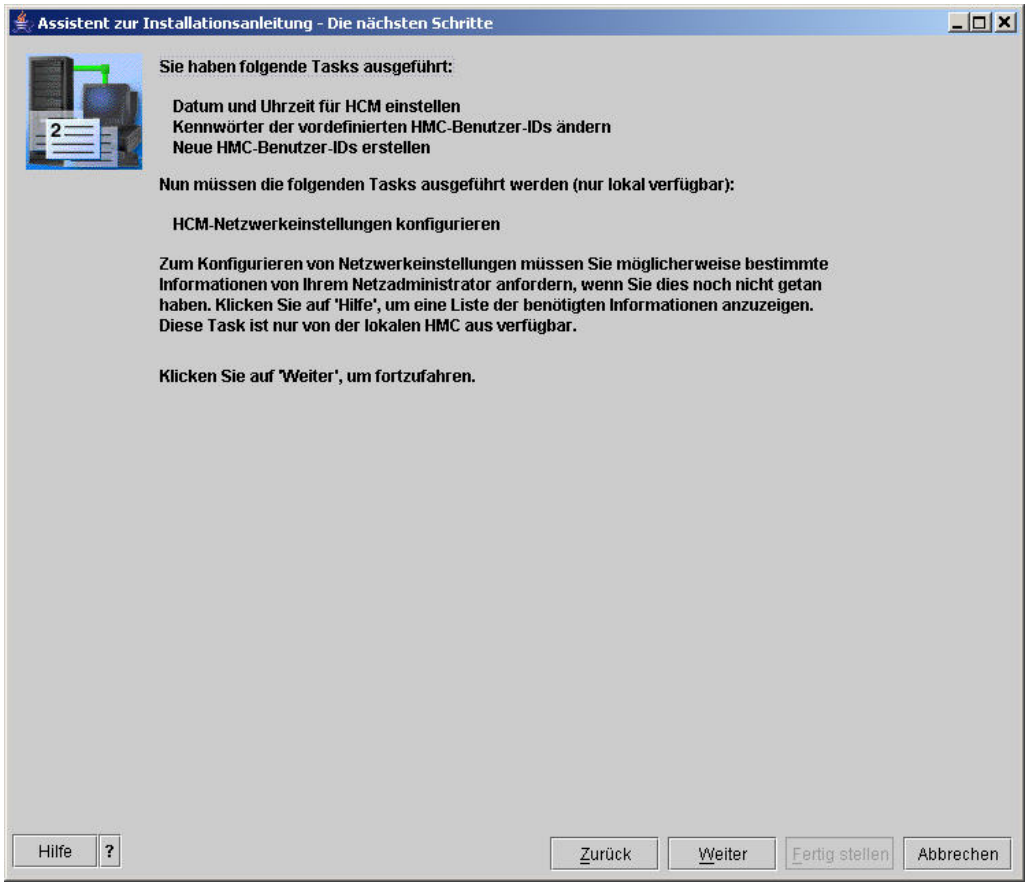
Achtung! Das Kennwort muss mindestens sieben Zeichen lang sein.

Neues Kennwort erneut eingeben: *******

Ich ordne durch Markieren mit der Maus dem Benutzer hscroot2 die Berechtigungsklasse „hmcsuperadmin“ zu.

Danach wird eine Zusammenfassung der bisher getätigten Konfigurationen angezeigt.

4.2.4
Seite 6



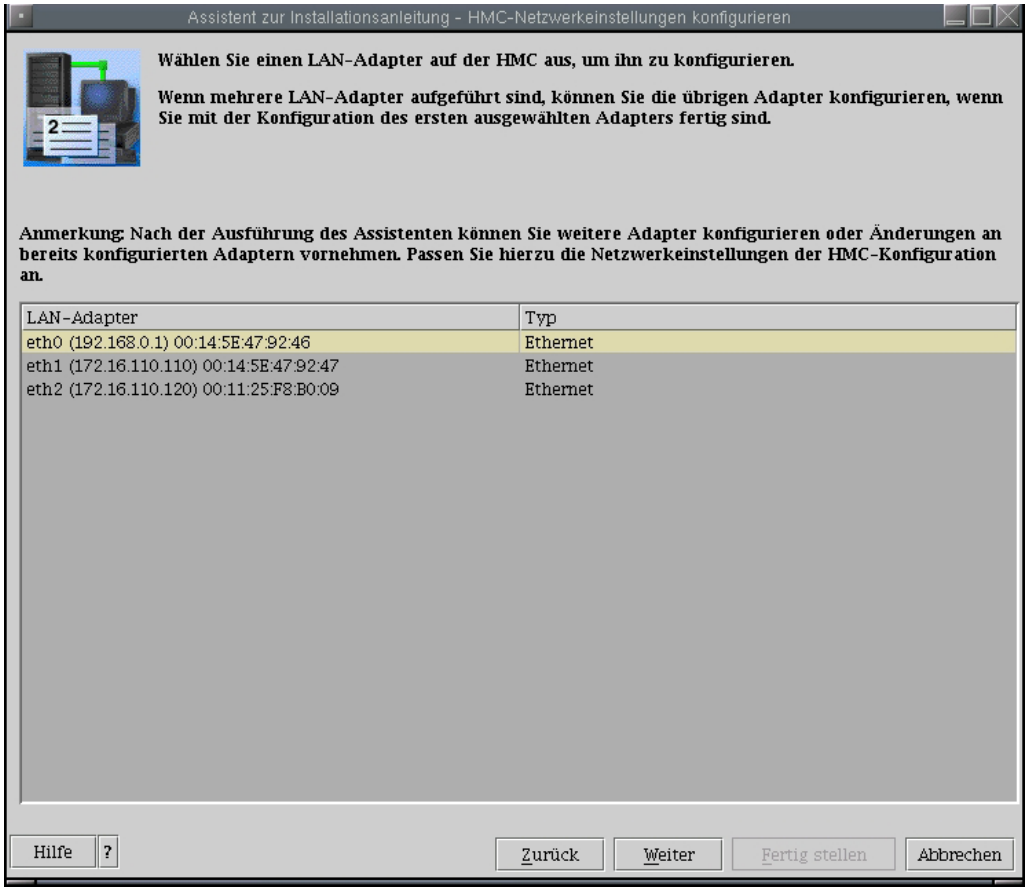
Zusammenfassung der ersten Konfigurationsschritte

Jetzt müssen die Netzwerkschnittstellen der HMC konfiguriert werden. Spätestens an dieser Stelle sollten Sie wissen, wie Sie die HMC mit dem System i verbinden wollen. Meine Verbindungskonfiguration soll folgendes Bild darstellen.

Die HMC wird via eth0 mit dem ersten HMC-Port des Systems i in einem privaten Netzwerk verbunden. In dieser Umgebung fungiert die HMC als DHCP-Server für den System i Service-Prozessor. Das private Netzwerk wird in den Netzbereich 192.168.0.0 gelegt.

Die HMC wird via eth1 mit dem Firmen-internen Netzwerk verbunden, da ich später über einen remote HMC-Client von meinem Arbeitsplatz auf die HMC zugreifen möchte. Das offene Netz wird in das Netzwerk 172.16.0.0 gelegt.

Wählen Sie im Fenster „Netzwerkeinstellungen konfigurieren“ den LAN-Adapter mit der Bezeichnung „eth0“ aus.

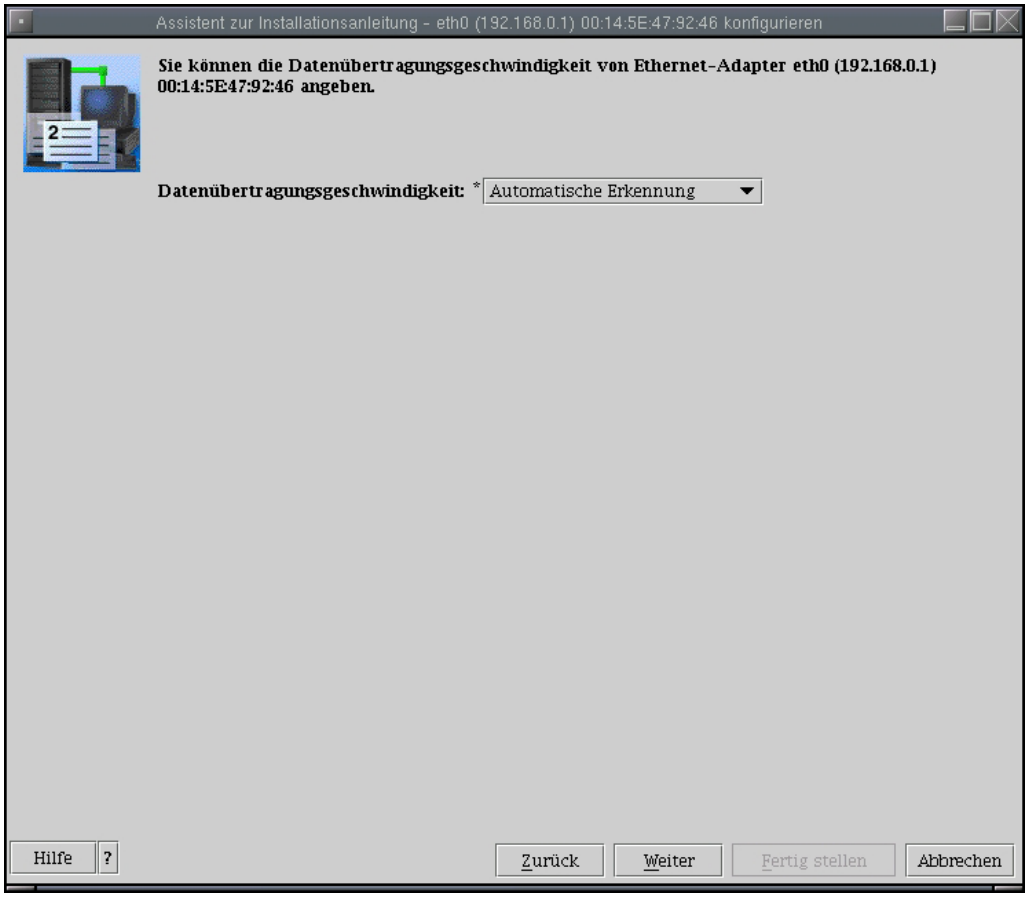


Auswahl der Ethernet-Schnittstelle

Klicken Sie auf „Weiter“.

Klicken Sie im Fenster für LAN-Adapter-Geschwindigkeit (Automatische Geschwindigkeitserkennung) auf „Weiter“.

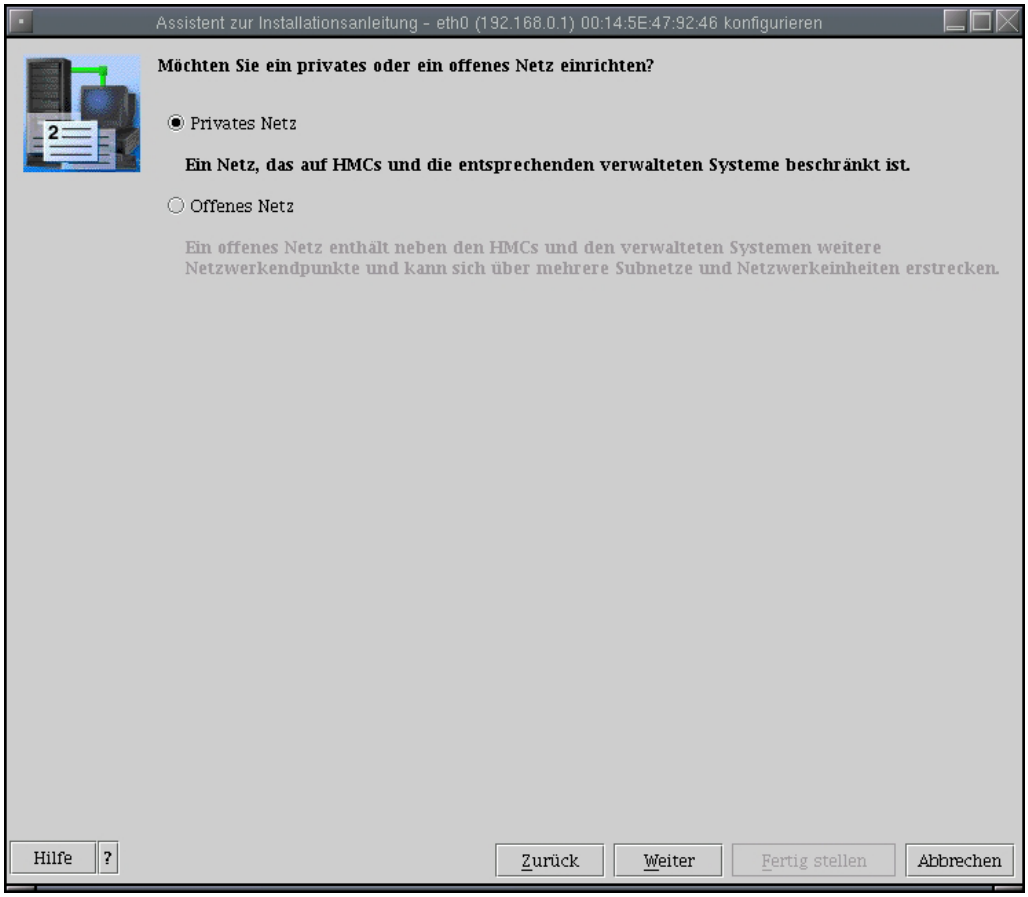
4.2.4
Seite 8



Datenübertragungsgeschwindigkeit einstellen

Klicken Sie im Fenster „(LAN-Adaptername) konfigurieren“ auf „Weiter“.

Wählen Sie im zweiten Fenster „(LAN-Adaptername) konfigurieren“ die Option für „Privates Service-Netz“ aus.

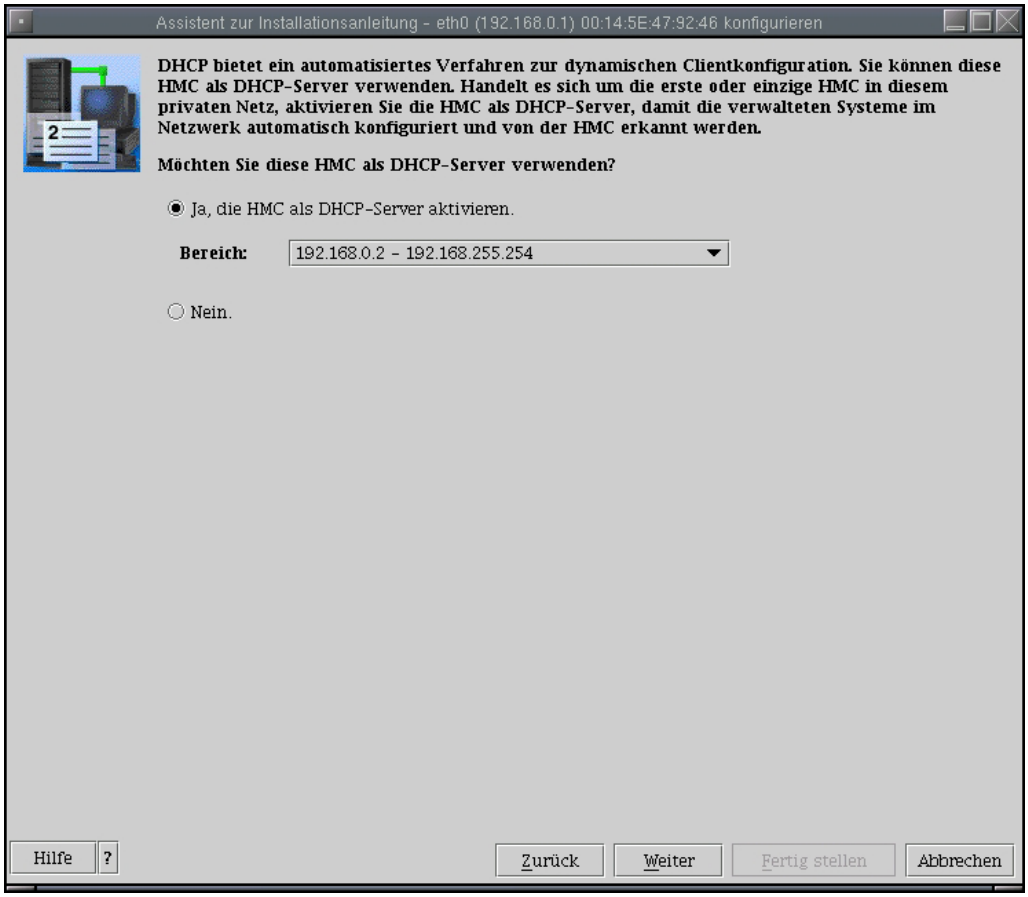


Privates Netz anwählen

Klicken Sie auf „Weiter“.

Wählen Sie im dritten Fenster „(LAN-Adaptername) konfigurieren“ die Option „Ja, die HMC als DHCP-Server aktivieren“ aus.

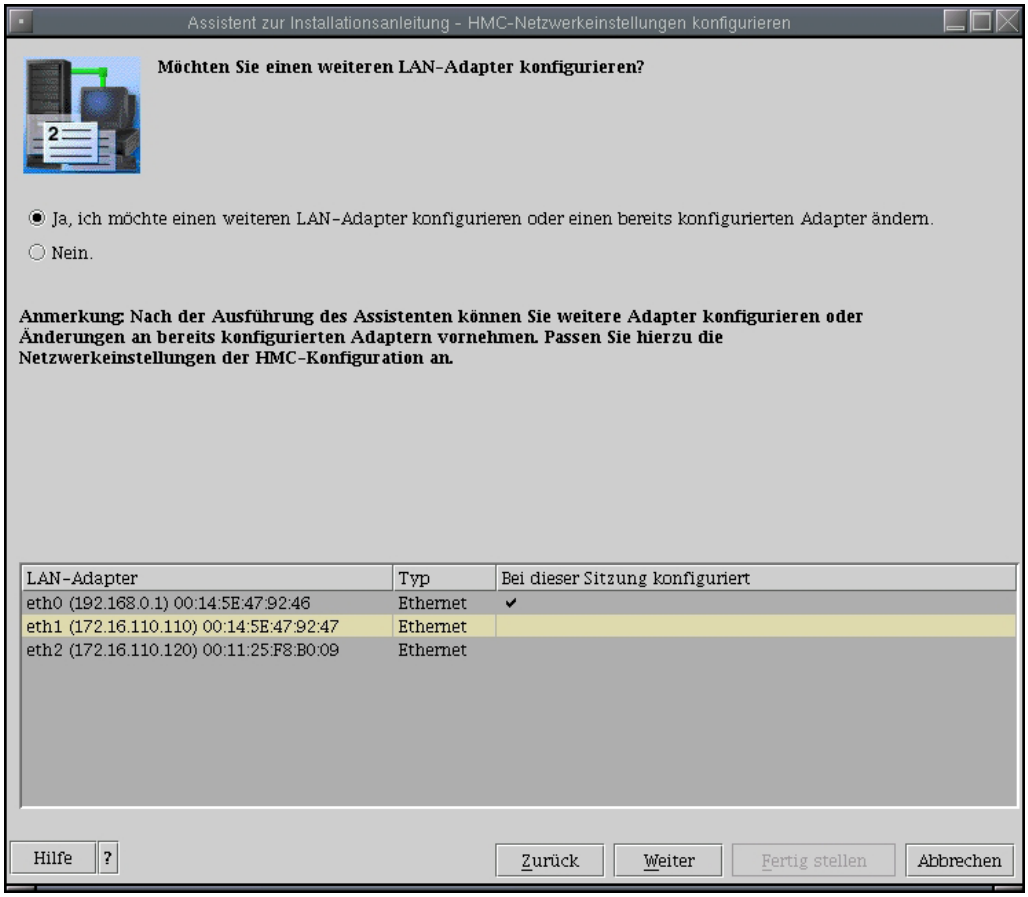
4.2.4
Seite 10



DHCP konfigurieren

Klicken Sie auf „Weiter“.

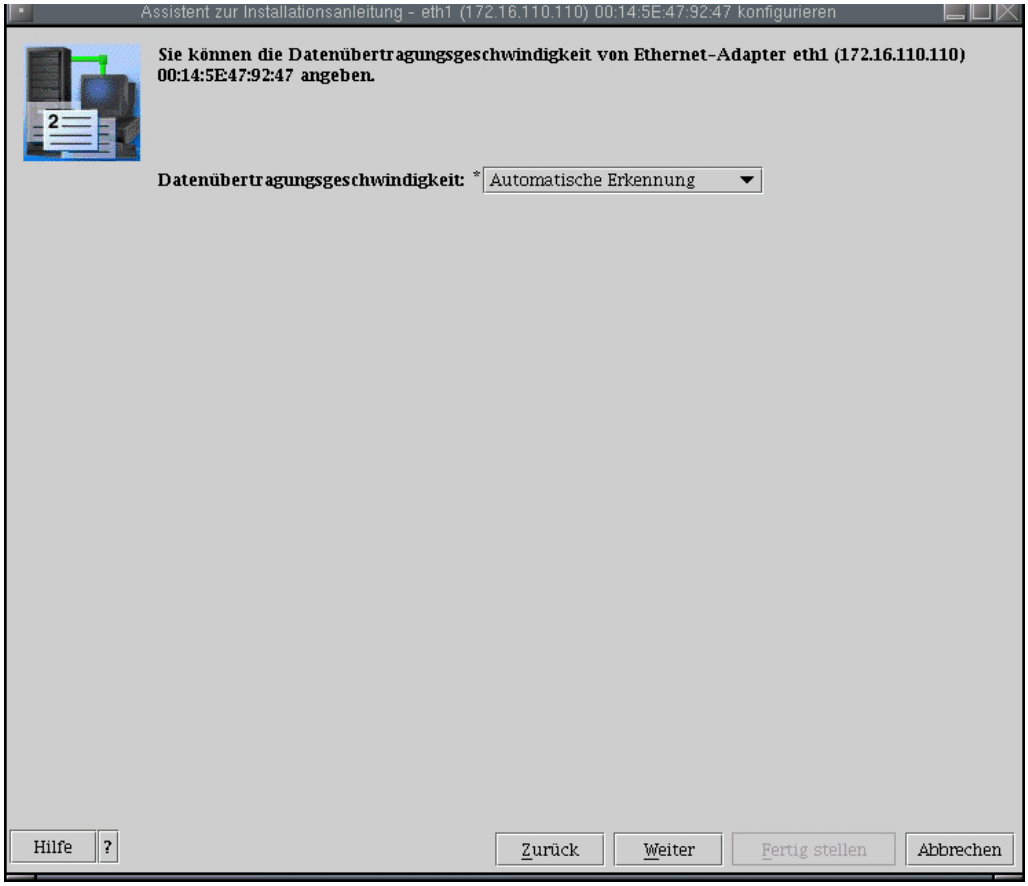
Wählen Sie im Fenster „Netzwerkeinstellungen konfigurieren“ die Option „Ja“ aus, um weitere Schnittstellen für das offene Netz zu konfigurieren.



Weitere Schnittstellen konfigurieren

Klicken Sie auf „Weiter“.

4.2.4
Seite 12

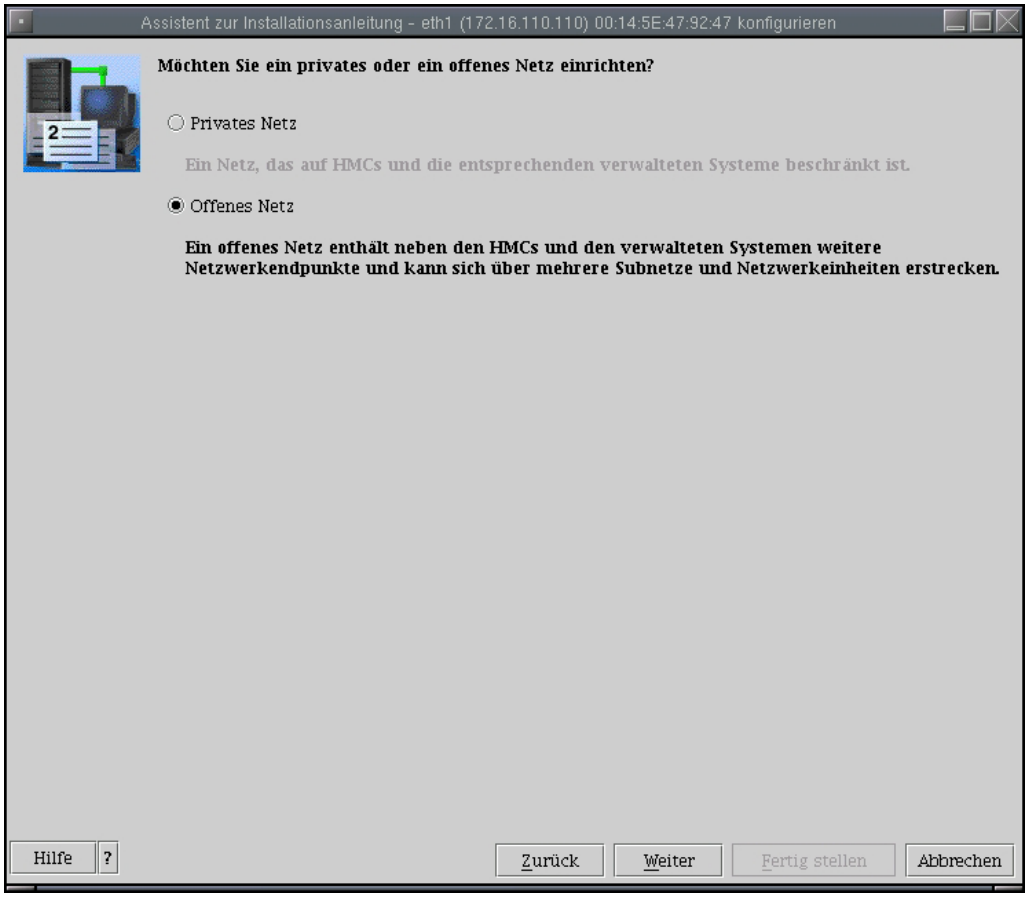


Datenübertragungsgeschwindigkeit einstellen

Belassen Sie es hierbei auch bei der automatischen Geschwindigkeitserkennung.

Klicken Sie auf „Weiter“.

Wählen Sie im folgenden Dialog die Option „Offenes Netz“ an.

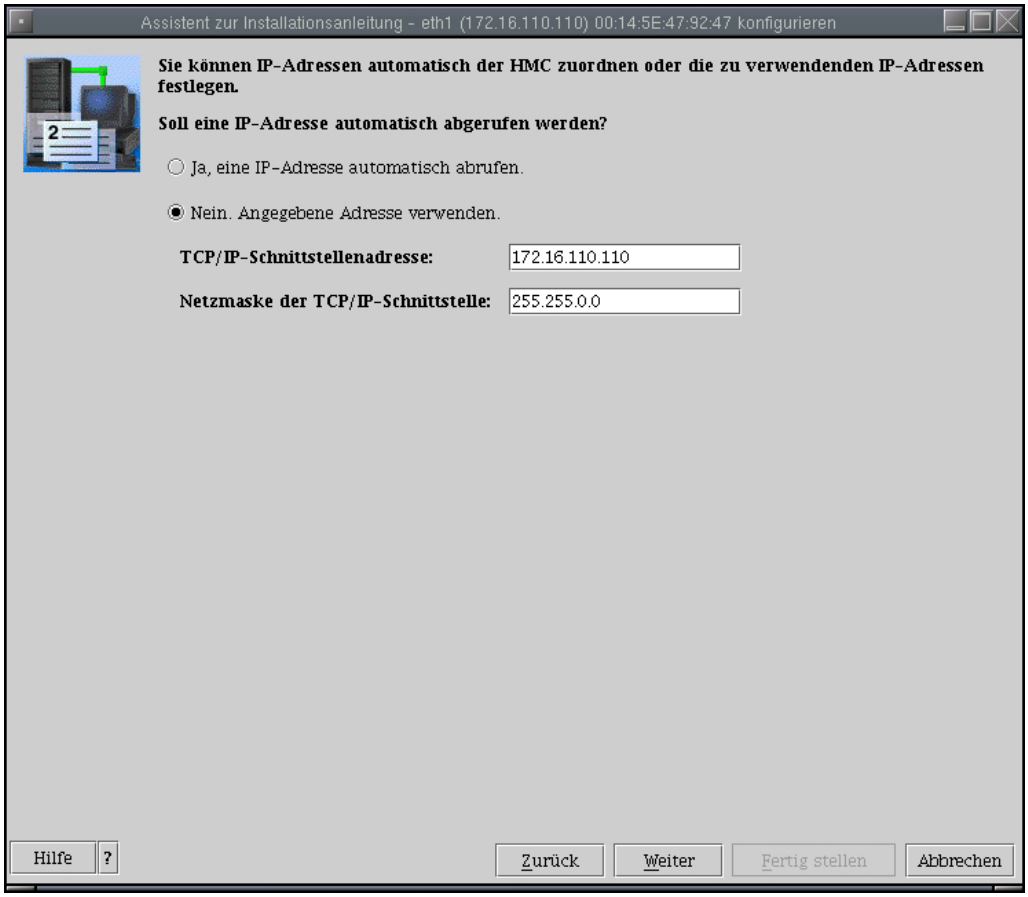


Offenes Netz

Klicken Sie auf „Weiter“.

Betreiben Sie in Ihrem Firmennetz einen DHCP-Server, so können Sie für eth1 automatisch eine Adresse abrufen lassen.

4.2.4
Seite 14

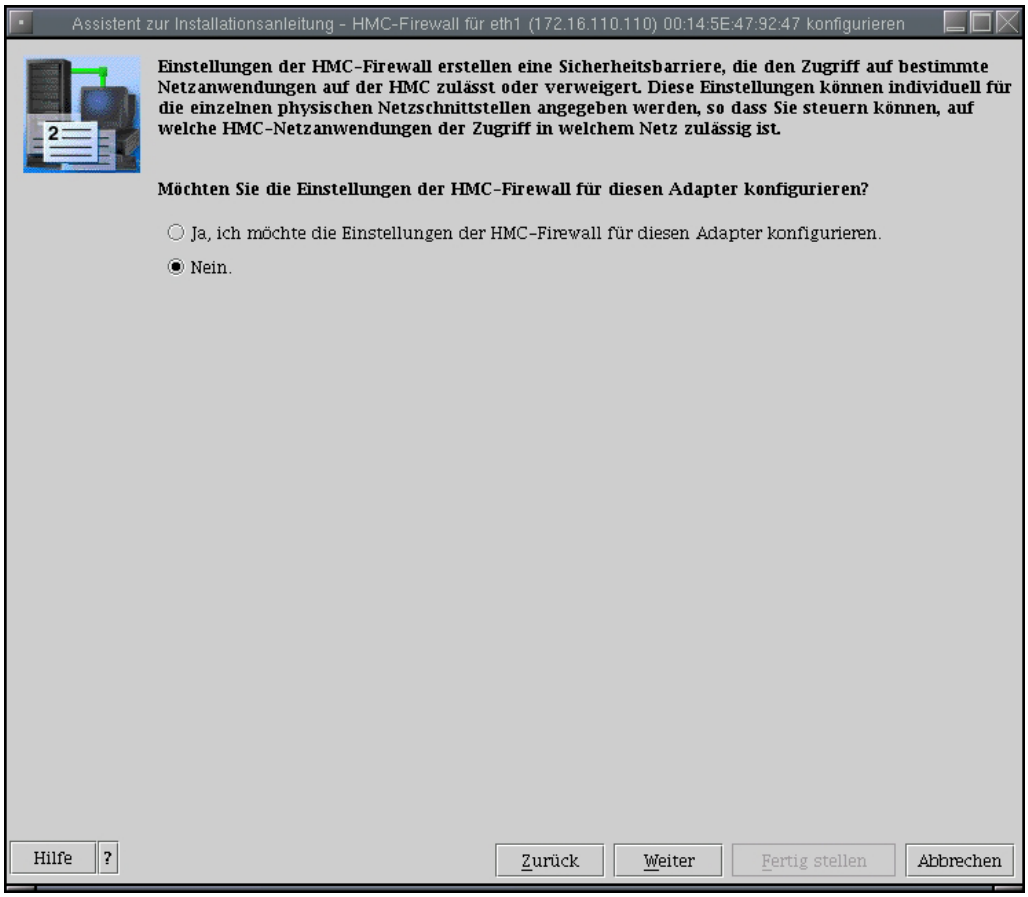


Adresse vergeben

Ich vergebe hier jedoch eine statische Adresse: 172.16.110.110 mit der Subnetzmaske 255.255.0.0

Klicken Sie auf „Weiter“.

Sie können sodann Firewall-Einstellungen definieren, die regeln, welche HMC-Dienste aus Ihrem Firmennetz angesprochen werden können.



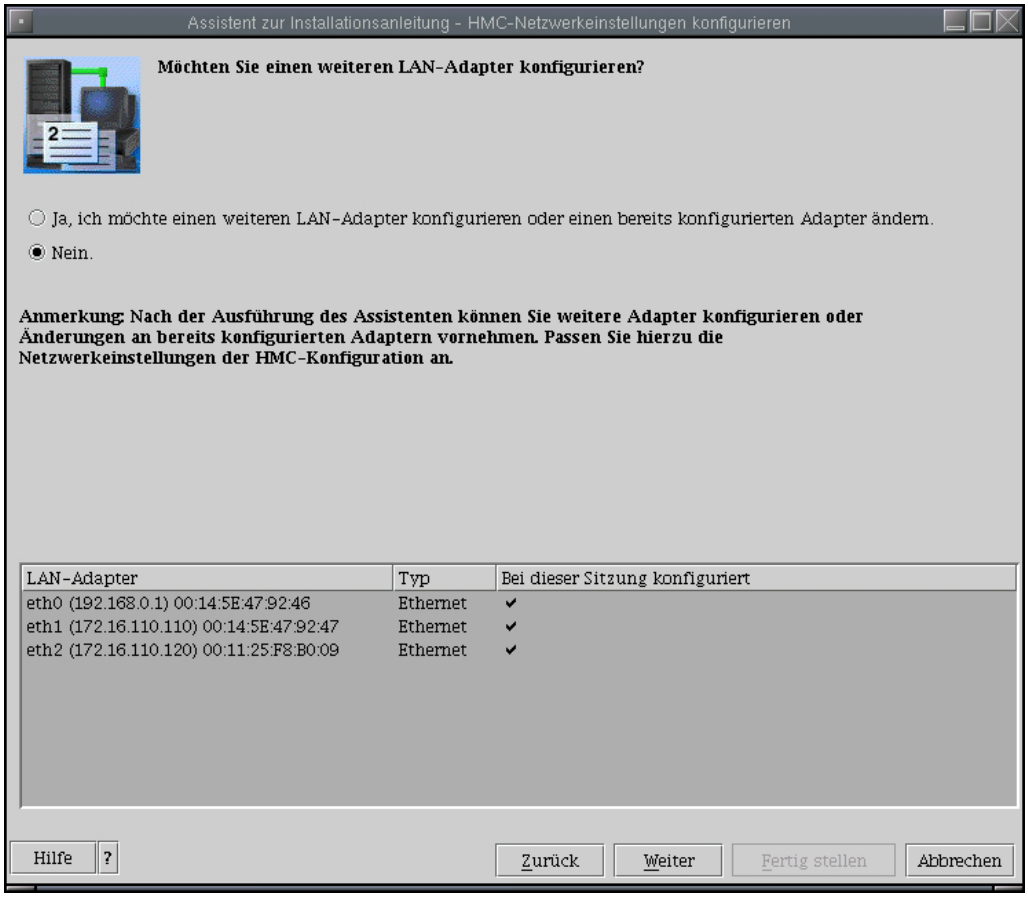
Firewall-Einstellungen

Ich wähle diese Möglichkeit zuerst einmal ab. Später, wenn ich Zugriffe aus dem Firmennetz auf die HMC konfigurieren will, werde ich jedoch die Firewall-Einstellungen bearbeiten müssen.

Klicken Sie auf „Weiter“.

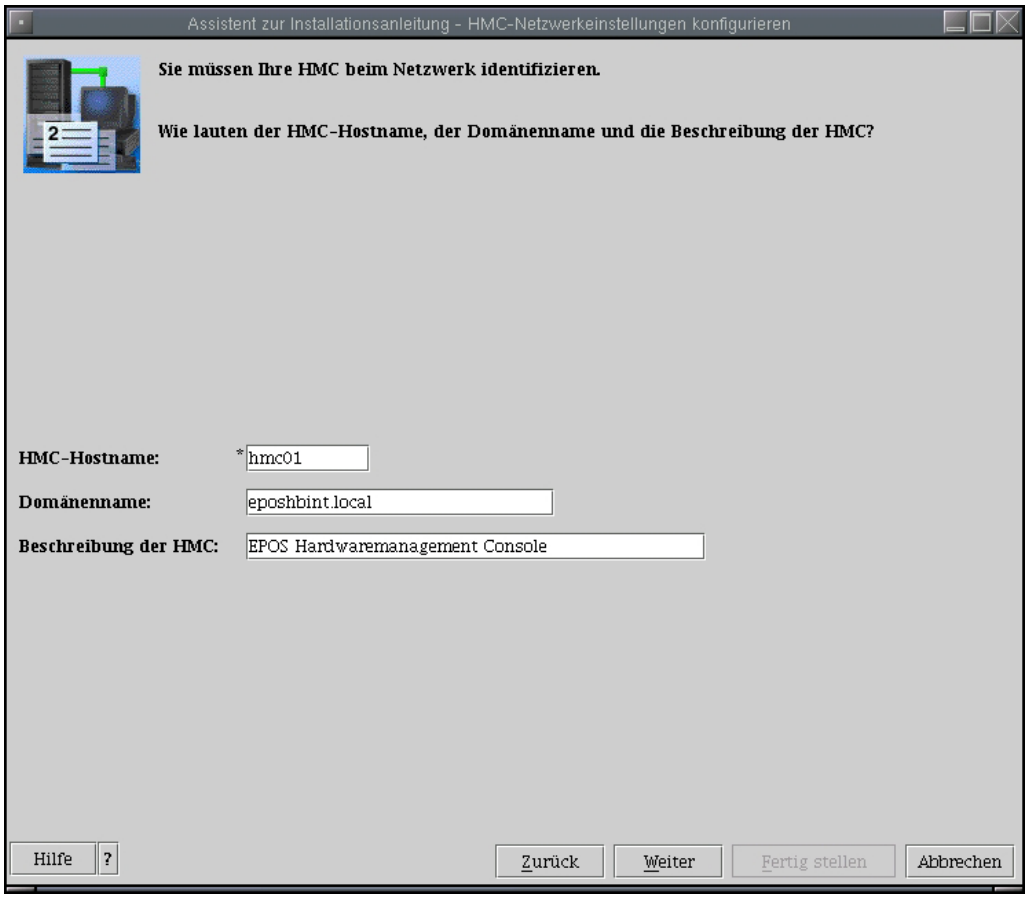
Gleichermaßen verfahre ich mit der dritten und letzten Schnittstelle in der HMC.

4.2.4
Seite 16



Alle Ethernet-Schnittstellen sind konfiguriert

Im Weiteren ordnen Sie Ihrer HMC einen Hostnamen und Domänennamen zu, konfigurieren wenn gewünscht ein Standard-Gateway, um z.B. Zugriffe ins Internet ausführen zu können.



Host- und Domänennamen vergeben

Im oben gezeigten Dialog vergebe ich folgende Werte:

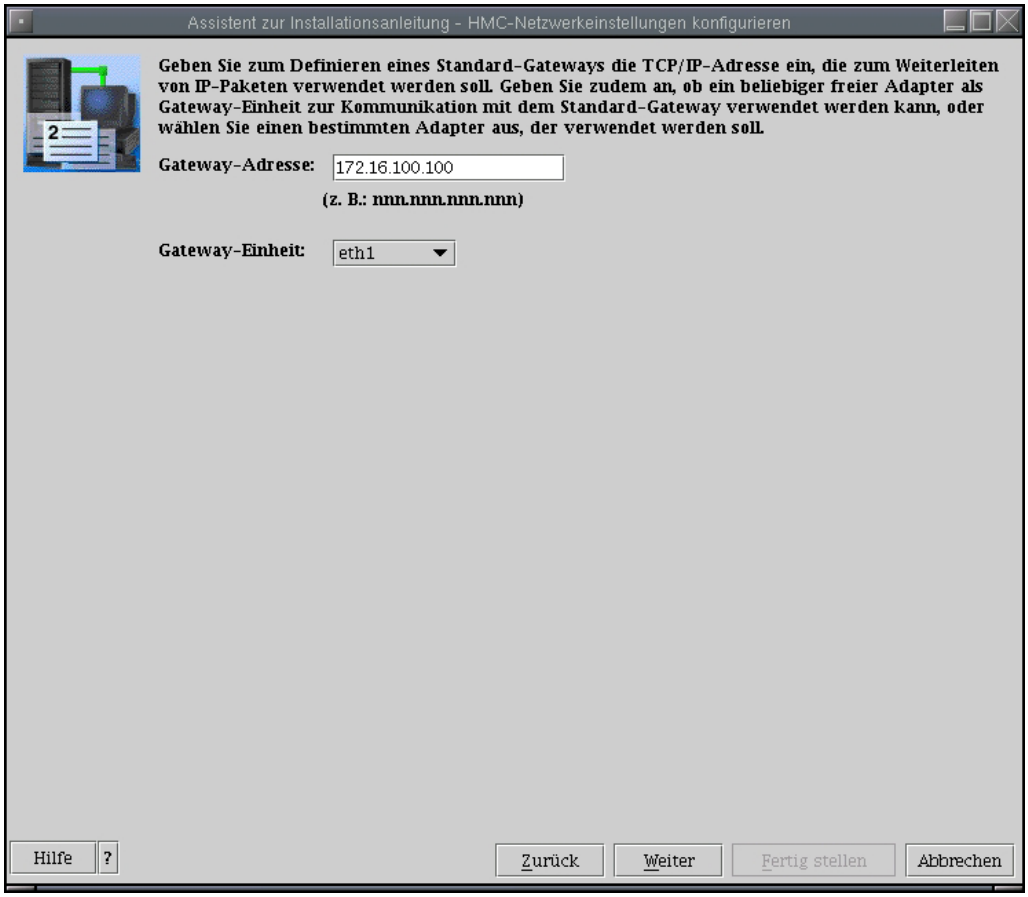
HMC-Hostname: Dieser Parameter muss gefüllt werden. Beabsichtigen Sie die HMC mit diesem Hostnamen anzusprechen, muss dieser Name in den DNS Ihres Netzwerks eingetragen werden, oder alternativ in die Hosttabellen aller Rechner, die auf die HMC zugreifen können sollen. Hier: hmc01

Domänenname: Der Name der TCP/IP-Domäne Ihres Hauses. Hier: eposgmbhint.local

Beschreibung….: Ein sinnvoller Erläuterungsbegriff

Tragen Sie jetzt, sofern Sie es wollen, die Adresse des Standard-Gateways ein.

4.2.4
Seite 18

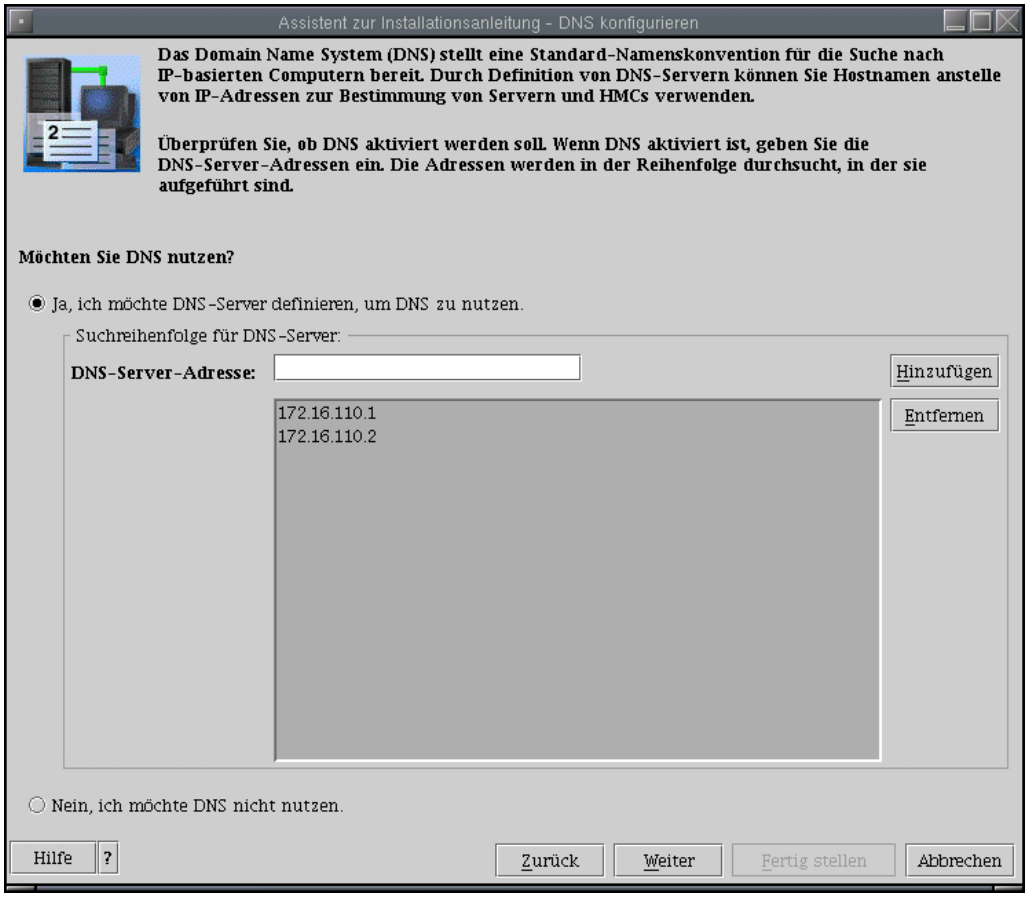


Standard-Gateway benennen

Parameter Gateway-Einheit: Hier tragen Sie ein, über welches physikalische Interface Sie auf das Standard-Gateway zugreifen. Da die Schnittstelle eth0 vom privaten Netzwerk der Verbindung von der HMC zu Ihrer iSeries belegt ist, kommen jetzt nur die Schnittstellen eth1 und eth2 (sofern vorhanden) in Betracht.

Hier: eth1

Ordnen Sie für die Namensauflösung, sofern vorhanden, einen oder mehrere DNS-Server für die Namensauflösung zu.



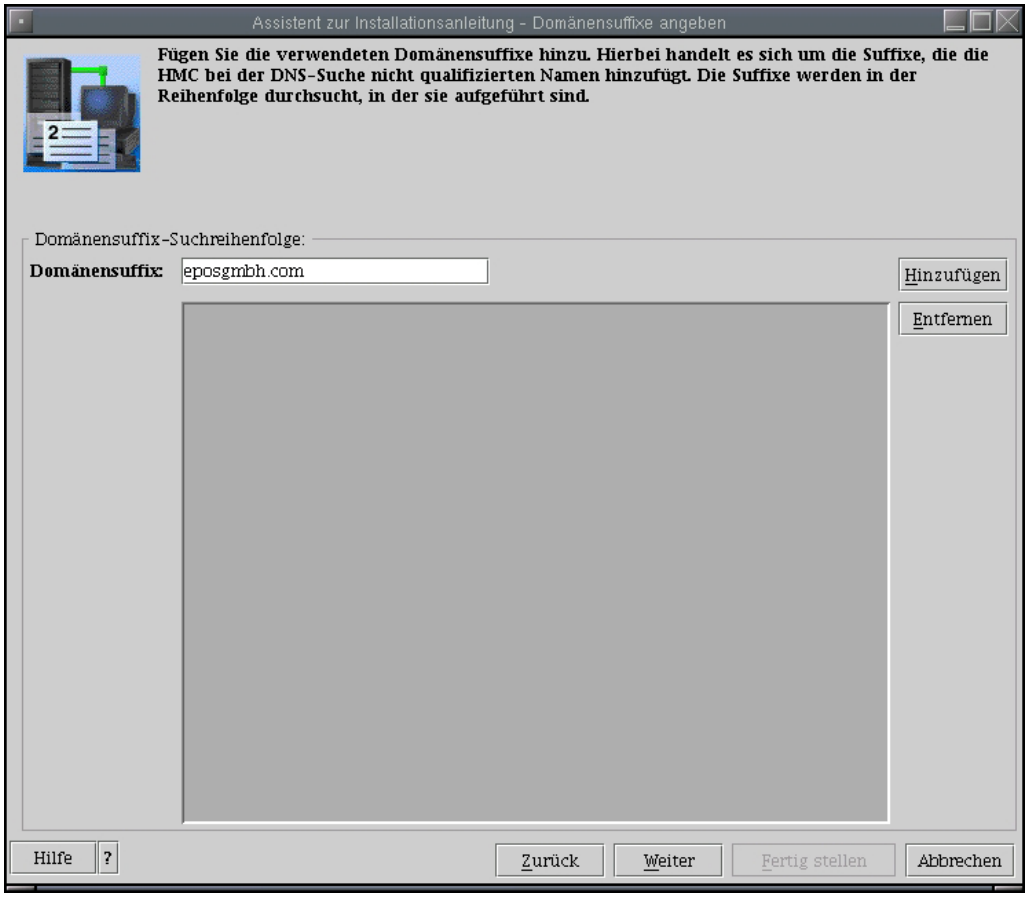
DNS-Server zuordnen

Sie sehen, dass ich hier die Adressen für zwei DNS-Server zugeordnet habe.

Hier: 172.16.110.1 und 172.16.110.2

4.2.4
Seite 20

Gegebenenfalls ordnen Sie einen oder mehrere Domänen-Suffixe zu, so dass ein unqualifizierter Zugriff von der HMC auf Hosts Ihres Netzwerks nicht mit einem Fehler quittiert wird.

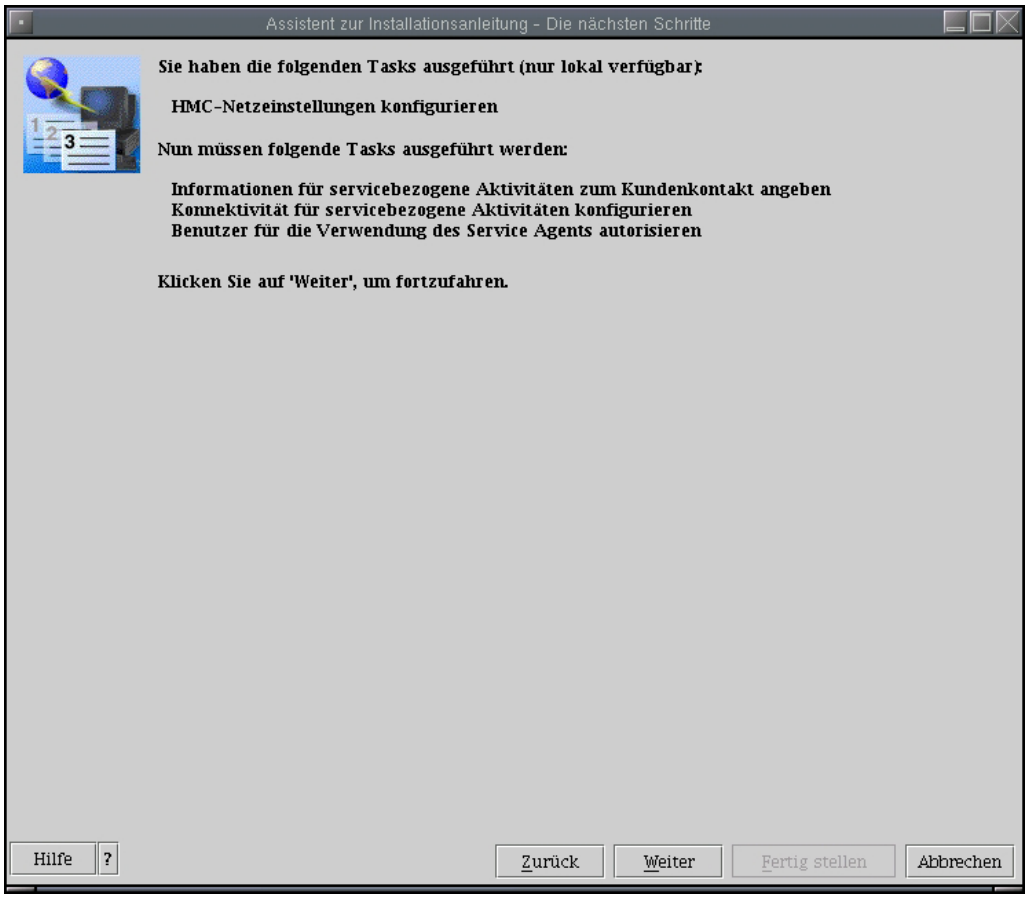


Domänen-Suffix zuordnen

Damit ist die Netzwerkkonfiguration Ihrer HMC beendet.

Klicken Sie auf „Weiter“.

Sie erhalten jetzt noch einmal eine Zusammenfassung der bisher geleisteten Konfigurationsschritte.



Zusammenfassung

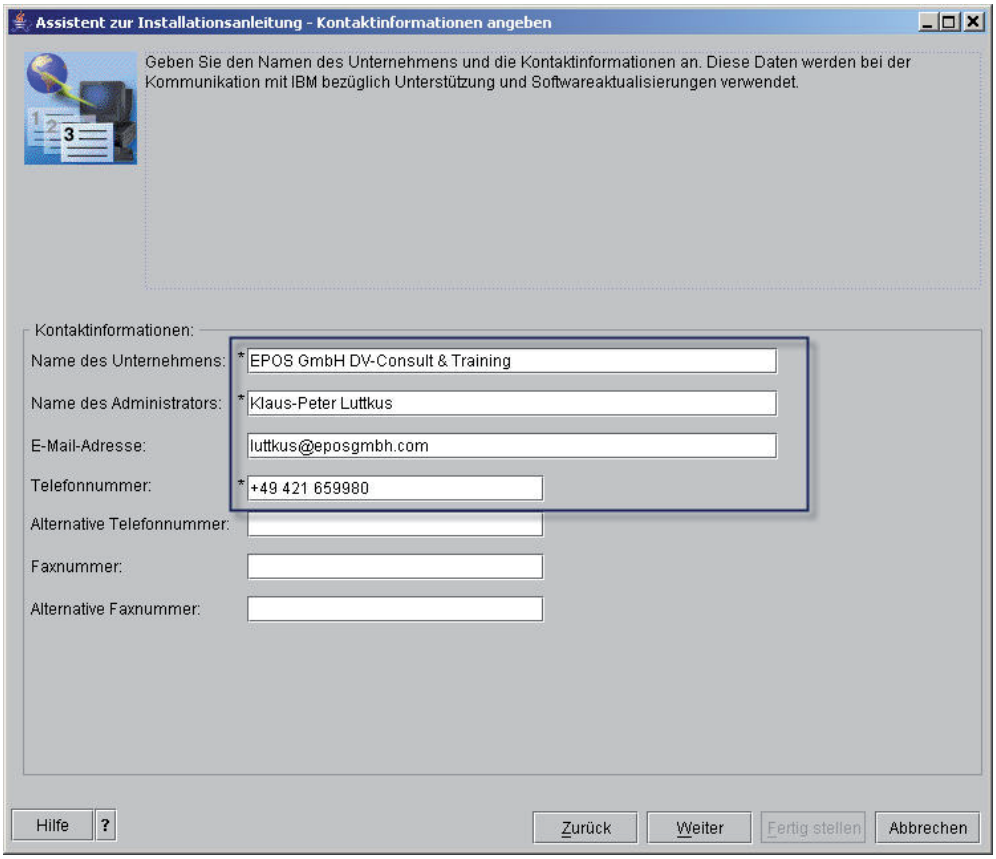
Klicken Sie im Zusammenfassungsfenster „Die nächsten Schritte“ auf „Weiter“.

4.2.4
Seite 22

Verbindung zu Service-Provider konfigurieren

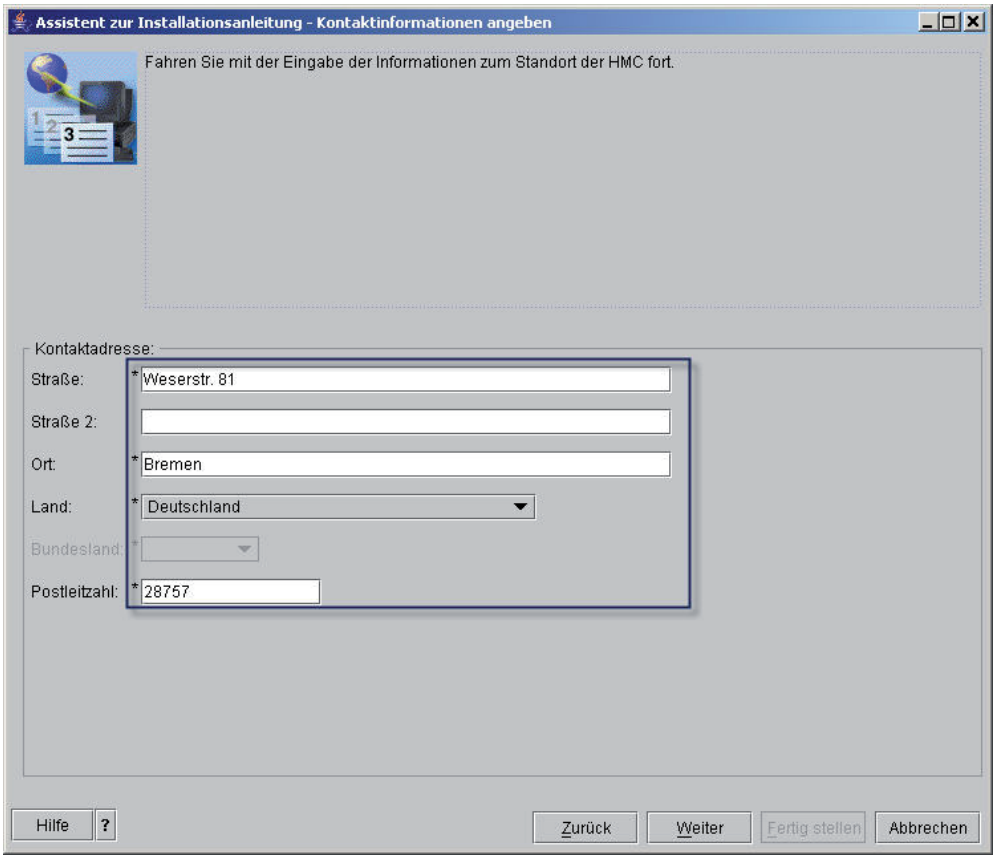
Alle weiteren Konfigurationen sind nicht zwingend notwendig für den Betrieb der HMC, bieten sich jedoch an, falls Sie Probleme beim Betrieb der HMC haben und einen elektronischen Zugriff von IBM oder einem Service Provider auf Ihre HMC wünschen.

1. Geben Sie im Fenster „Kontaktinformationen angeben“ die Kontaktinformationen für den HMC-Administrator ein. Klicken Sie auf „Weiter“.



Kontaktinformationen für HMC-Administrator

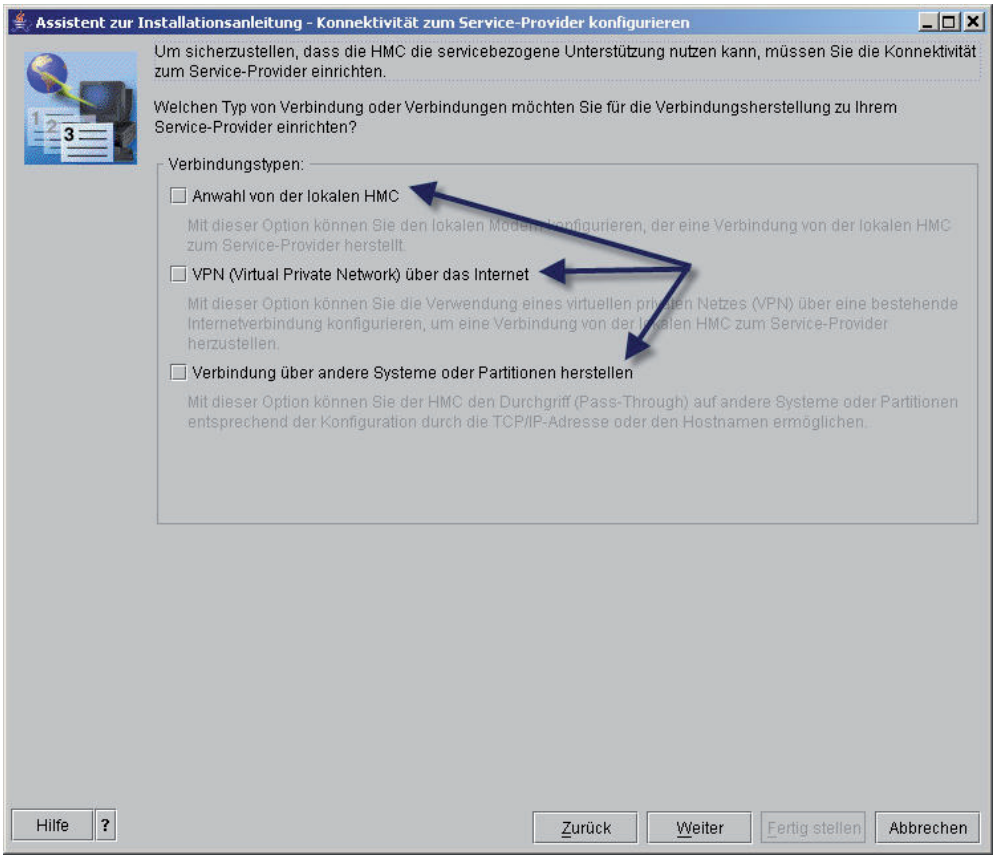
4.2.5
Seite 2



Adressinformationen für HMC-Administrator

2. Geben Sie im Fenster „Kontaktinformationen angeben“ die Adressinformationen der Kontaktperson für den HMC-Standort ein. Klicken Sie auf „Weiter“.

   Sie gelangen sodann in Definitionsdialoge, mit deren Hilfe Sie die Verbindung zu Service-Providern herstellen können.



Konnektivität zum Service-Provider

Folgende Möglichkeiten bestehen:

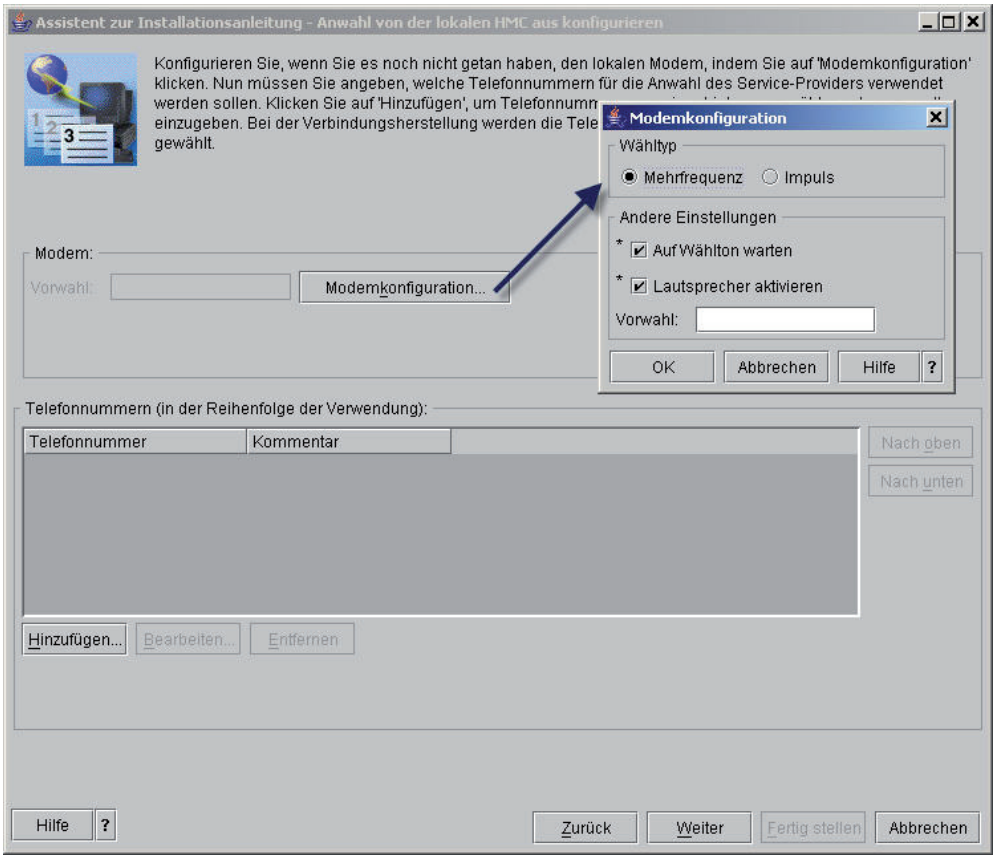
Anwahl der lokalen HMC:
Mittels eines an der HMC angeschlossenen Modems stellen Sie die Verbindung zum Service-Provider her.

VPN über das Internet:
Mittels eines VPN-Clients in der HMC stellen Sie die Verbindung zum Service-Provider her.

Verbindung über andere Systeme oder Partitionen herstellen:
Mittels eines System i als Router stellen Sie die Verbindung zum Service-Provider her.

Im folgenden Bild sehen Sie den Dialog, um mit Hilfe eines an der HMC befindlichen Modems die Verbindung zum Service-Provider herzustellen.

4.2.5
Seite 4



Modemkonfiguration 1

Tragen Sie hier für das von Ihnen verwendete Modem die entsprechenden Kennwerte ein.

Mittels „Hinzufügen“ fügen Sie die Anwahlnummer des Service-Providers der Konfiguration hinzu.

3. Hier noch einmal der Konfigurationsablauf:

   Gehen Sie im Fenster „Anwahl von der lokalen HMC aus konfigurieren“ wie folgt vor:

   a. Klicken Sie auf „Modemkonfiguration“. Um in Ihrem Unternehmen einen externen Telefonanruf zu tätigen, geben Sie die Nummer in das Feld „Vorwahl“ ein. Klicken Sie auf „OK“.

   b. Um eine Telefonnummer zu Ihrem Service-Provider hinzuzufügen, klicken Sie auf „Hinzufügen“.

      i. Wählen Sie das gewünschte Land oder die gewünschte Region aus.

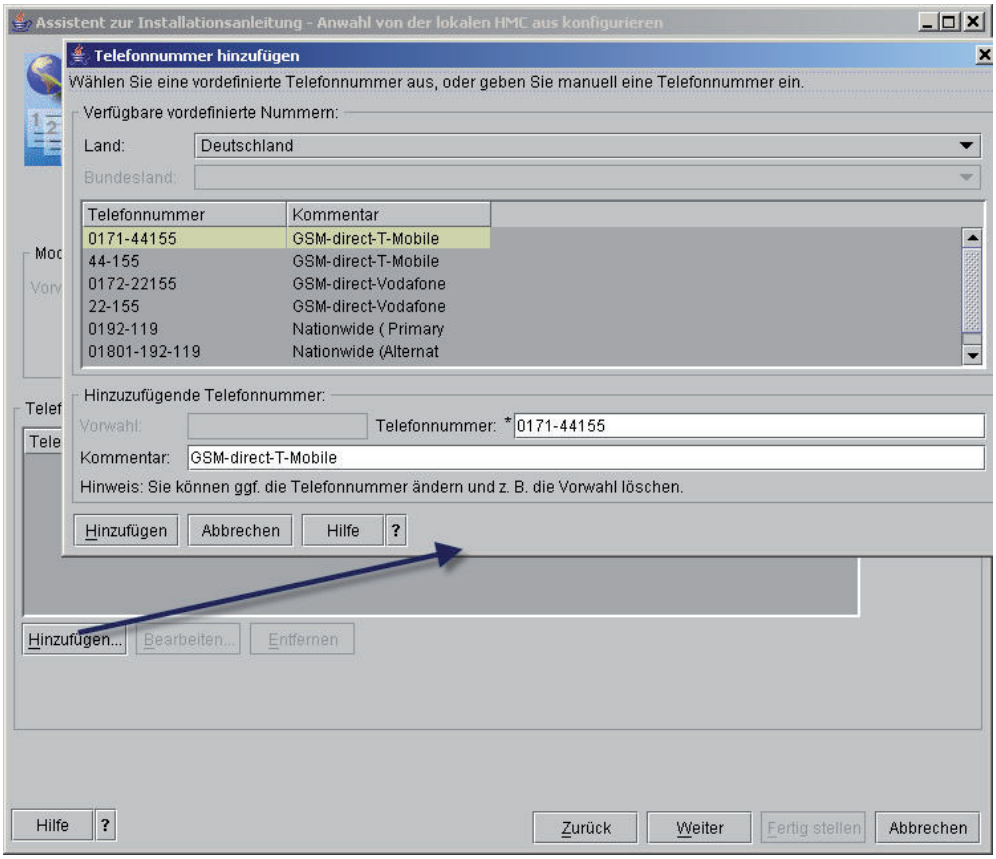
      ii. Wählen Sie den gewünschten Ort aus.

      iii. Wählen Sie die gewünschte Telefonnummer aus der Liste aus.

iv. Bearbeiten Sie im Feld „Telefonnummer“ ggf. die ausgewählte Telefonnummer. Wenn es sich bei der ausgewählten Telefonnummer beispielsweise um eine Telefonnummer im Ortsnetz handelt, müssen Sie die Vorwahl möglicherweise löschen.

v. Klicken Sie auf „OK“.

c. Klicken Sie auf „Weiter“.



Modemkonfiguration 2

4. Klicken Sie im Fenster „Konnektivität zum Service-Provider konfigurieren“ auf „Weiter“.

5. Gehen Sie im Fenster „Anwahl von der lokalen HMC aus konfigurieren“ wie folgt vor:

   a. Klicken Sie auf „Modemkonfiguration“. Um in Ihrem Unternehmen einen externen Telefonanruf zu tätigen, geben Sie die Nummer in das Feld „Vorwahl“ ein. Klicken Sie auf „OK“.

   b. Um eine Telefonnummer zu Ihrem Service-Provider hinzuzufügen, klicken Sie auf „Hinzufügen .

      i. Wählen Sie das gewünschte Land oder die gewünschte Region aus.

      ii. Wählen Sie den gewünschten Ort aus.

4.2.5
Seite 6

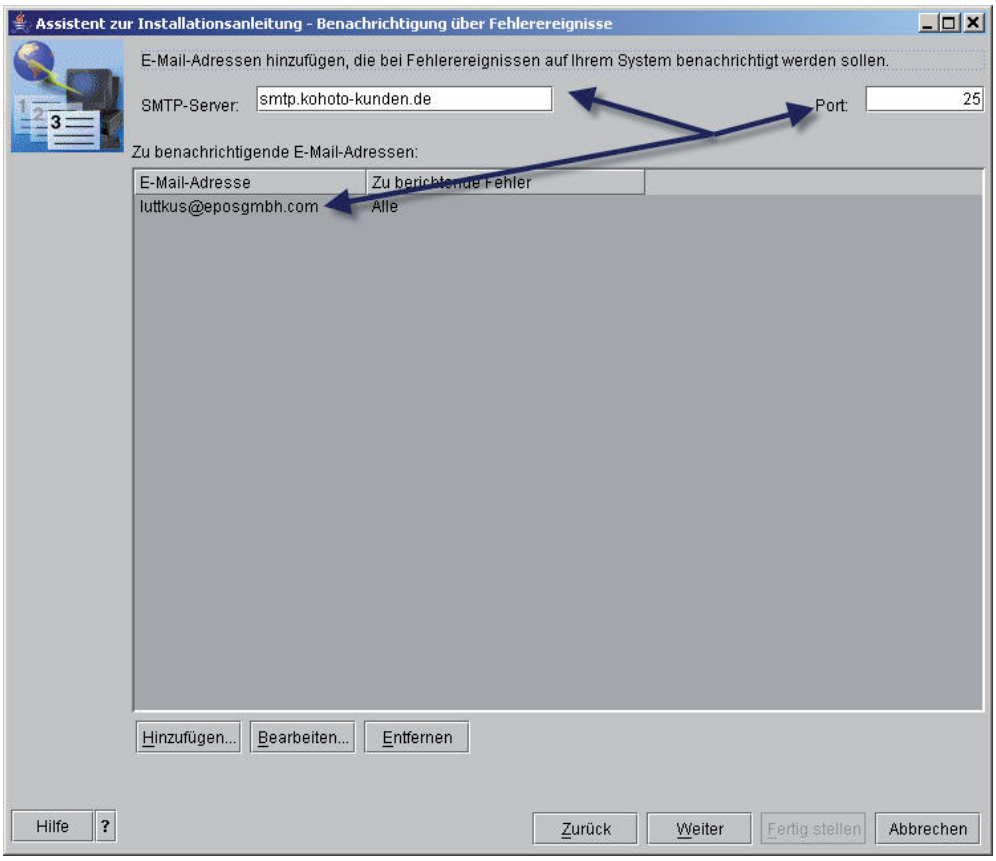
iii. Wählen Sie die gewünschte Telefonnummer aus der Liste aus.

iv. Bearbeiten Sie im Feld „Telefonnummer“ ggf. die ausgewählte Telefonnummer. Wenn es sich bei der ausgewählten Telefonnummer beispielsweise um eine Telefonnummer im Ortsnetz handelt, müssen Sie die Vorwahl möglicherweise löschen.

v. Klicken Sie auf „OK“.

c. Klicken Sie auf „Weiter“.

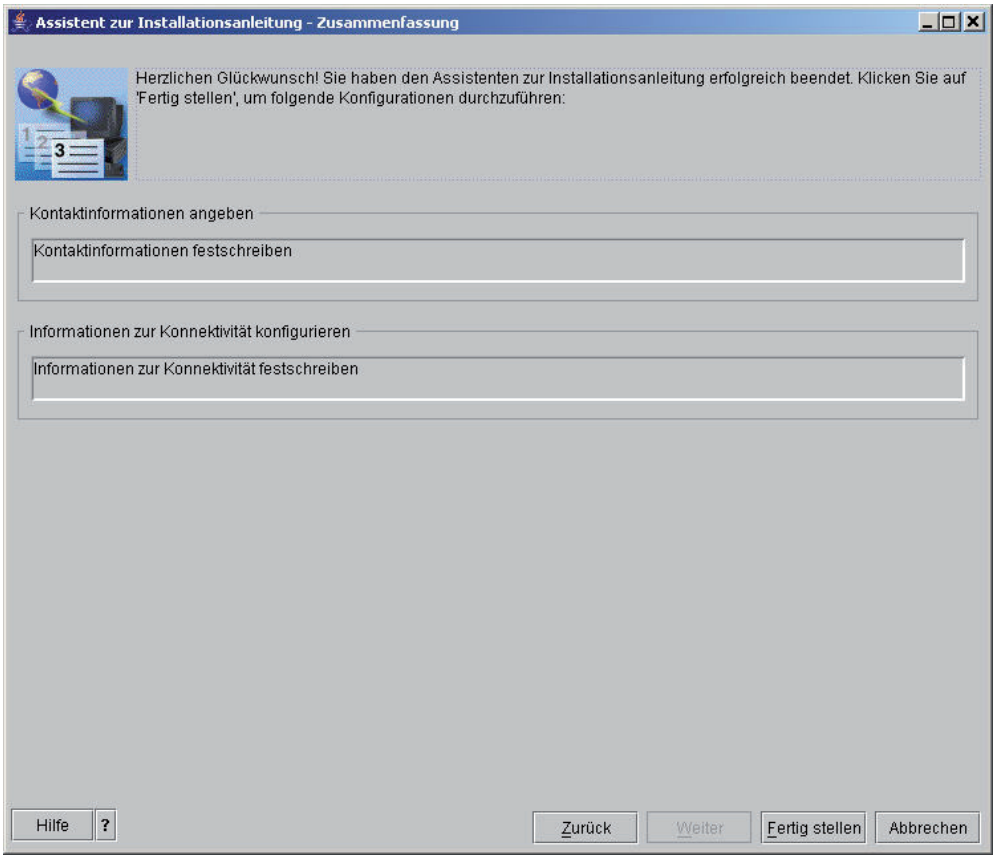
Weiterhin besteht die Möglichkeit Fehlermeldungen der HMC via e-Mail direkt an Sie weiterleiten zu lassen.



SMTP-Konfiguration

6. Geben Sie im Fenster „Benachrichtigung über Fehlerereignisse“ den von Ihnen genutzten SMTP-Server und -Anschluss an.

7. Klicken Sie auf „Hinzufügen“, um mindestens eine e-Mail-Adresse zur Benachrichtigung über Fehlerereignisse anzugeben. Das wird im Regelfall die e-Mailadresse des HMC-Administrators sein. Klicken Sie auf „Weiter“.



Abschluss der Konfiguration

8. Klicken Sie im Fenster „Zusammenfassung“ auf „Fertig stellen“.

Damit ist die eigentliche Anfangskonfiguration der HMC abgeschlossen.

Danach kehren Sie zur HMC-Oberfläche zurück.

Sollten alle Verbindungskabel korrekt gesteckt und die soeben durchgeführte Konfiguration ebenfalls korrekt sein, so kann die Verbindung zwischen HMC und System i hergestellt werden.

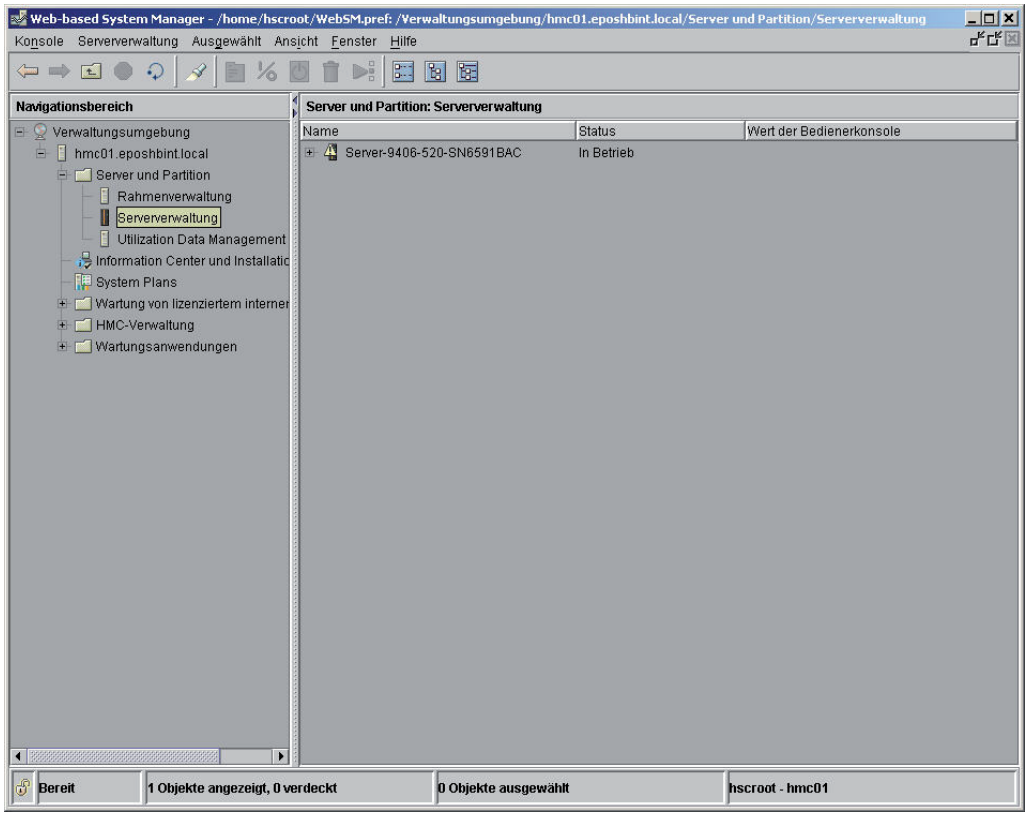
4.2.5
Seite 8

Logische Verbindungsherstellung zwischen HMC und System i

Ich hoffe, Sie haben Ihr System i noch nicht mit dem Stromnetz verbunden. Besitzen Sie ein Modell 520, so ist es nicht so schlimm. Ziehen Sie einfach den Stromstecker aus dem System heraus. Warten Sie einen kurzen Moment (Zählen Sie bis 10, aber schön langsam!). Dann stellen Sie die Stromverbindung wieder her.

Jetzt fordert der so genannte Service Processor als DHCP-Client von der HMC eine IP-Adresse an und eine logische Verbindung kann zwischen den beiden Systemen hergestellt werden.

Danach sollte der HMC-Hauptdialog wie folgt aussehen:



HMC-Oberfläche

Klicken Sie im Navigationsbereich auf den Punkt „Serververwaltung“, dann sollten Sie links die Kenndaten Ihrer iSeries sehen.

4.2.6
Seite 2

Damit wissen Sie, dass beide Systeme Kenndaten ausgetauscht haben. Sollten Sie hier keinen entsprechenden Eintrag sehen, so sind Fehler in folgenden Bereichen möglich:

1. Verkabelung fehlerhaft
2. Verbindungskonfiguration fehlerhaft
3. HMC-Software muss via Fixes aktualisiert werden.

HMC-Webclient installieren und konfigurieren (WebSM)

Eine weitere gute Funktion der HMC besteht darin, dass auf fast alle ihre Funktionen von ferngesteuerten (remoten) Clients aus zugegriffen werden kann. Das gilt sowohl für ein lokales Netzwerk als auch für ein WAN-Umfeld. Dabei ist möglicherweise in einem LAN die Konfiguration einfacher zu gestalten als im WAN, da bei einem lokalen Netzwerk gegebenenfalls auf die Verschlüsselung der zwischen HMC und WebSM ausgetauschten Daten verzichtet werden kann.

Um remote HMC-Clients betreiben zu können, muss auf den entsprechenden Clients eine Software installiert werden, die mit dem Namen WebSM bezeichnet wird.

Um WebSM einsetzen zu können muss eine lauffähige HMC vorliegen.

Auf jeder HMC steht die WebSM-Software zum Download bereit.

Das Herunterladen der Software erfolgt mit Hilfe Ihres Browsers.

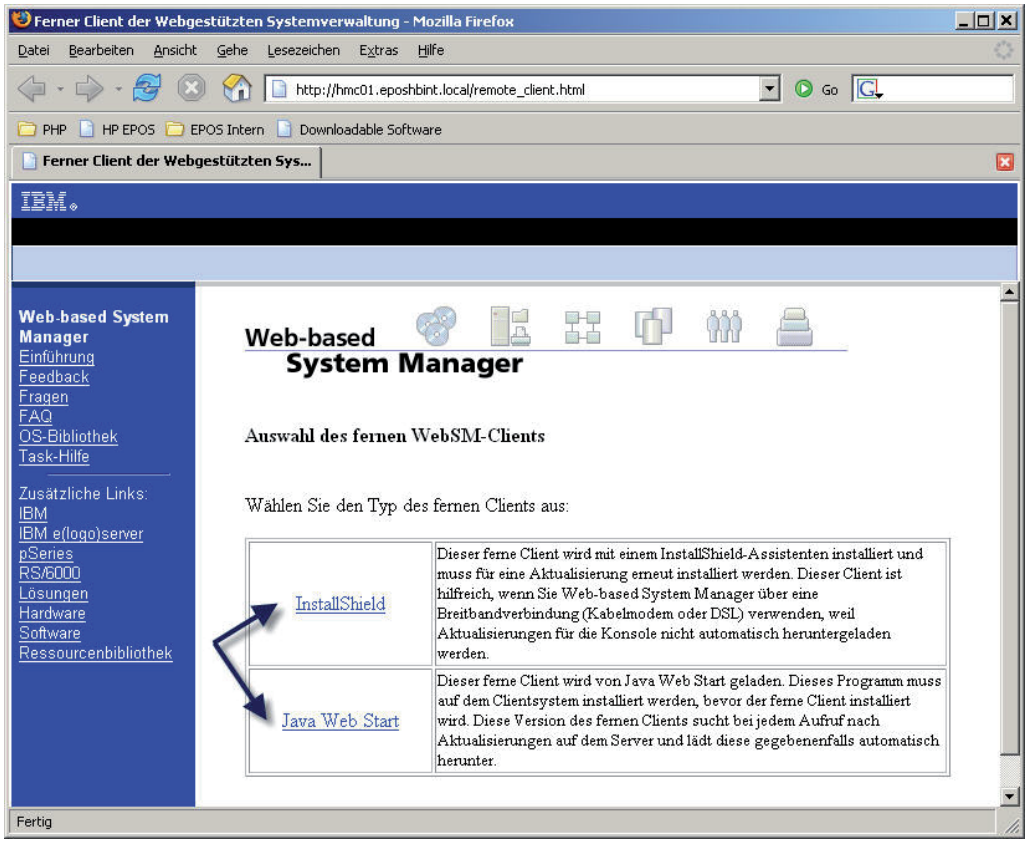
Ich rufe also meinen Browser auf und gebe folgende URL ein:

Formal: http://<IP-Adresse der HMC>/remote_client.html

In meinem Fall: http://hmc01.eposhbint.local/remote_client.html

Danach müssen Sie sich an der HMC mit einem HMC-Benutzer wie z. B. HSCROOT anmelden, dann öffnet sich folgender Dialog:

4.2.7
Seite 2



WebSM downloaden 1

Es gibt zwei Arten des WebSM-Download:

1. WebSM wird inklusive des InstallShield-Setups heruntergeladen.

   Dieser ferne Client wird mit einem InstallShield-Assistenten installiert und muss für eine Aktualisierung erneut installiert werden. Dieser Client ist hilfreich, wenn Sie ein Web-based System Manager über eine Breitbandverbindung (Kabelmodem oder DSL) verwenden, weil Aktualisierungen für die Konsole nicht automatisch heruntergeladen werden.

2. WebSM wird inklusive des Java Web Starts heruntergeladen.

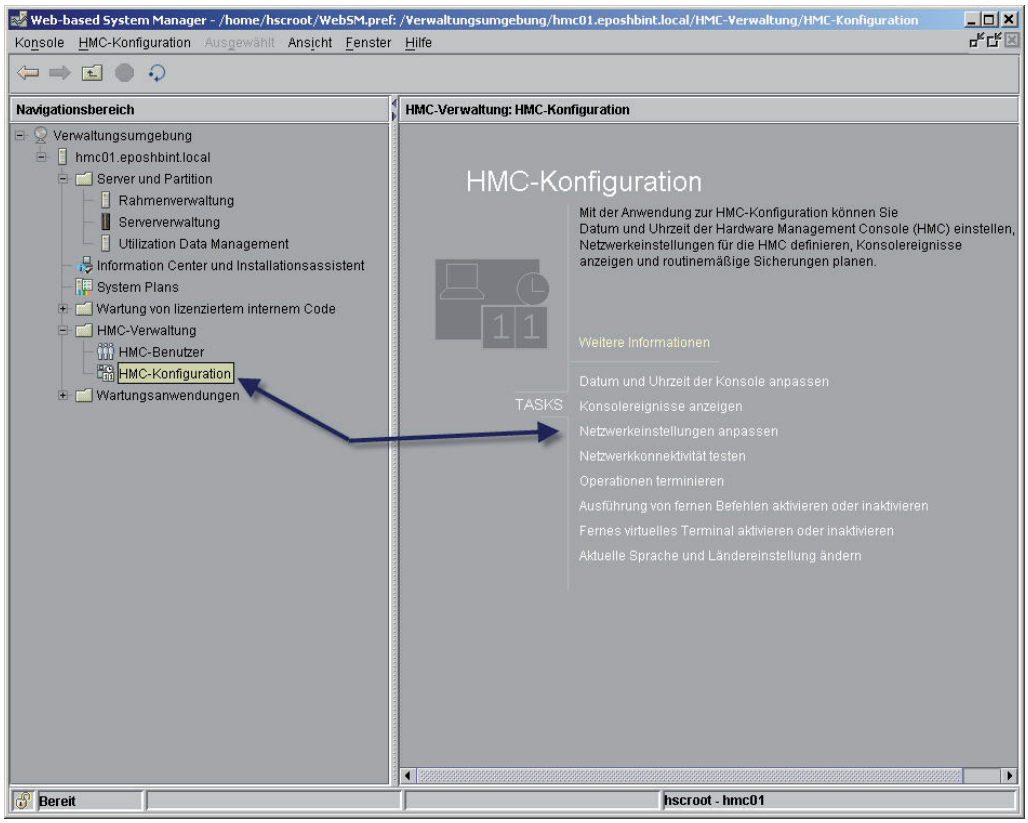
   Dieser ferne Client wird von Java Web Start geladen. Dieses Programm muss auf dem Client-System installiert werden, bevor der ferne Client installiert wird. Diese Version des fernen Clients sucht bei jedem Aufruf nach Aktualisierungen auf dem Server und lädt diese gegebenenfalls automatisch herunter.

Firewall-Einstellungen der HMC anpassen

Sollte der oben gezeigte Dialog bei Ihnen nicht angezeigt werden, stattdessen jedoch die Meldung, dass aufgrund fehlender Berechtigungen der Zugriff verweigert wurde, so müssen Sie in der HMC den so genannten Web Access freischalten.

Dazu melden Sie sich an Ihrer HMC an und öffnen das Verzeichnis:

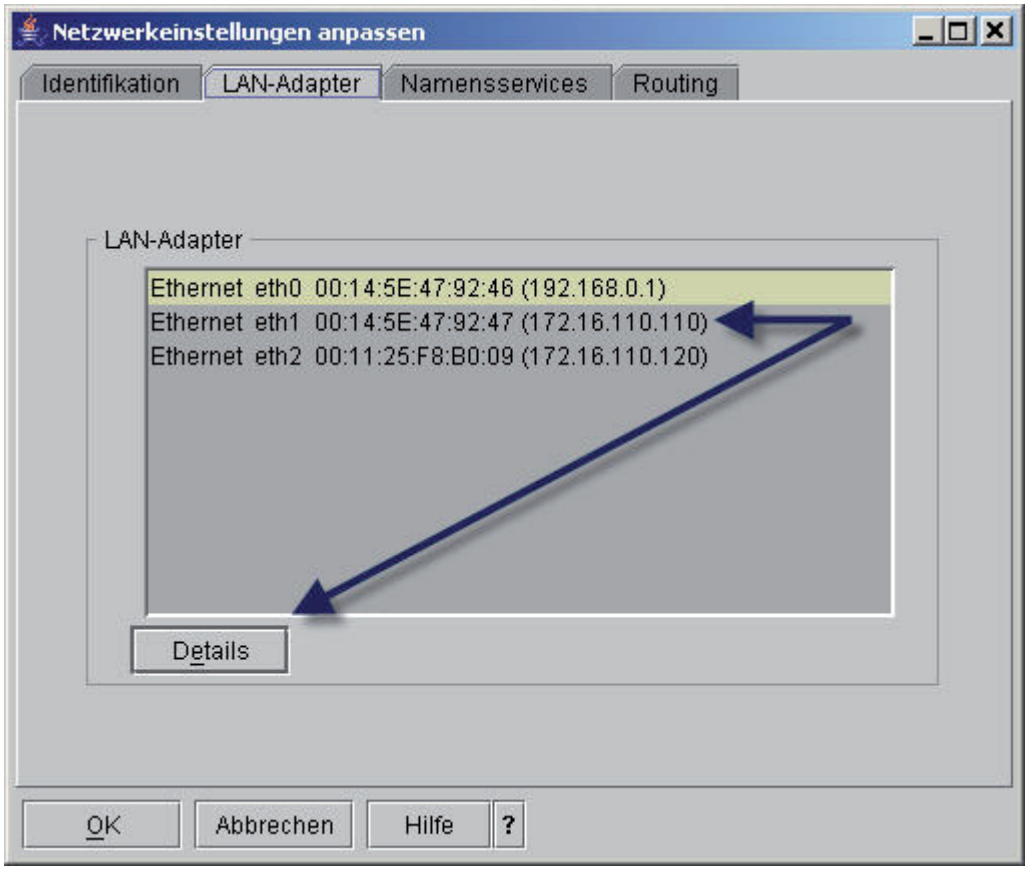
Verwaltungsumgebung -> <hmc-Name> -> HMC-Verwaltung -> HMC-Konfiguration.



HMC-Konfiguration

Dann klicken Sie auf „Netzwerkeinstellungen anpassen“.

4.2.7.1
Seite 2



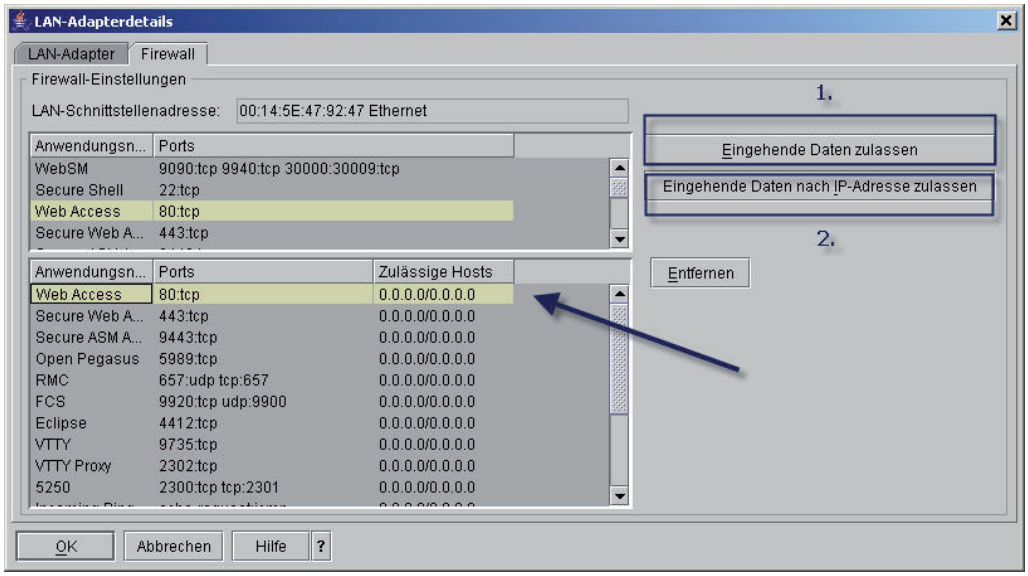
LAN Adapter-Konfiguration

Markieren Sie jetzt die Ethernet-Schnittstelle/IP-Adresse über die von Clients auf die HMC zugegriffen werden soll.

Klicken Sie auf „Details“.

Jetzt öffnet sich die Detailkonfiguration für diesen Land Adapter.

Wechseln Sie in das Register „Firewall“.



Firewall-Konfiguration

Markieren Sie im unteren Teildialog „Web Access“. Danach haben Sie zwei Möglichkeiten:

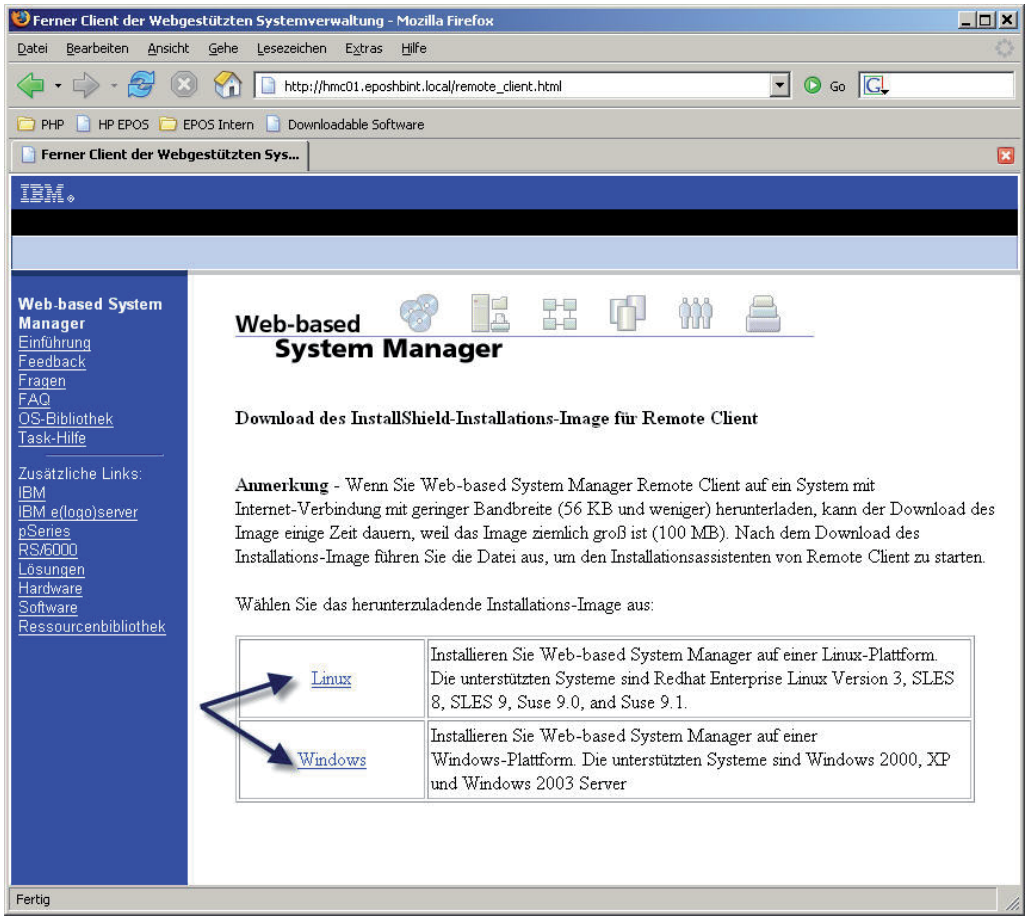
1. Eingehende Daten zulassen: Drücken Sie diesen Knopf, so kann jeder Client Ihres Netzwerks an den WebSM-Download herankommen.
2. Eingehende Daten nach IP-Adresse zulassen: Sie benennen IP-Adressen von Clients, denen die Erlaubnis gegeben wird, den WebSM-Download durchzuführen.

Um am Ende mit dem WebSM-Client auf die HMC zugreifen zu können, muss der WebSM-Dienst ebenfalls durch diese Firewall-Einstellungen konfiguriert werden.

Sie sollten dieses jetzt erledigen. Verfahren Sie analog zu den Einstellungen die für Web Access notwendig sind.

In meiner Konfiguration wähle ich für die weitere Installation des WebSM-Clients die Option „InstallShield“.

4.2.7.1
Seite 4



WebSM downloaden 2

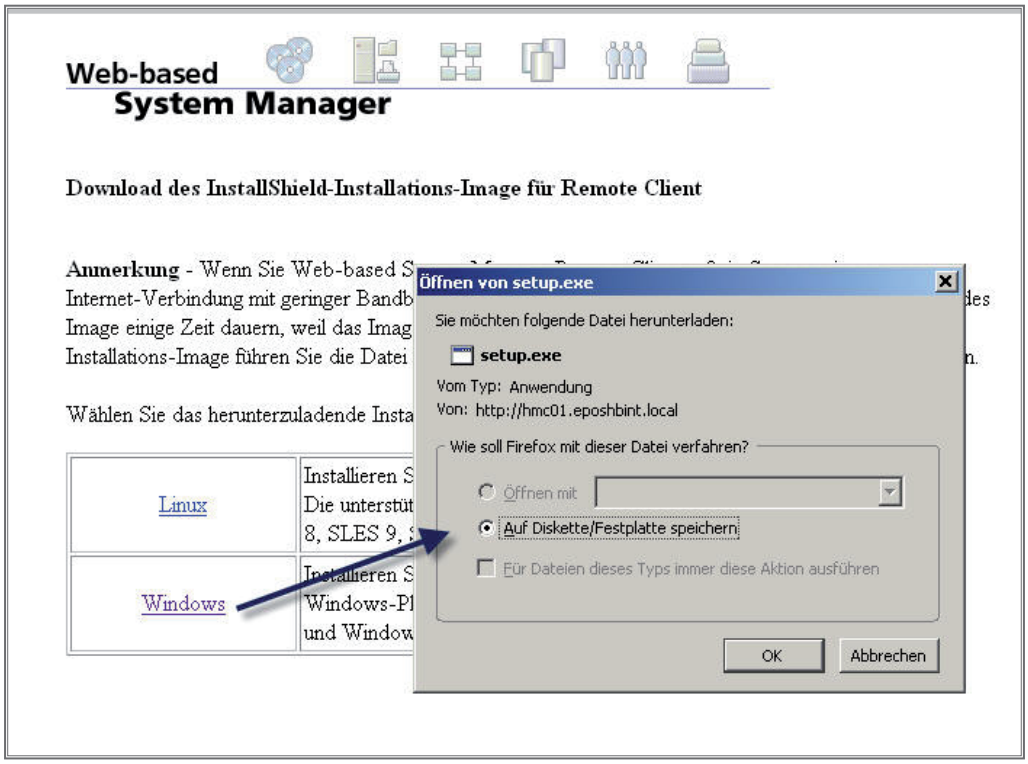
Weitere Fragen ergeben sich:

Soll WebSM unter Linux oder Windows installiert werden?

Linux:
Installieren Sie Web-based System Manager auf einer Linux-Plattform. Die unterstützten Systeme sind Redhat Enterprise Linux Version 3, SLES 8, SLES 9, Suse 9.0, and Suse 9.1.

Windows:
Installieren Sie Web-based System Manager auf einer Windows-Plattform. Die unterstützten Systeme sind Windows 2000, XP und Windows 2003 Server.

Ich wähle Windows.



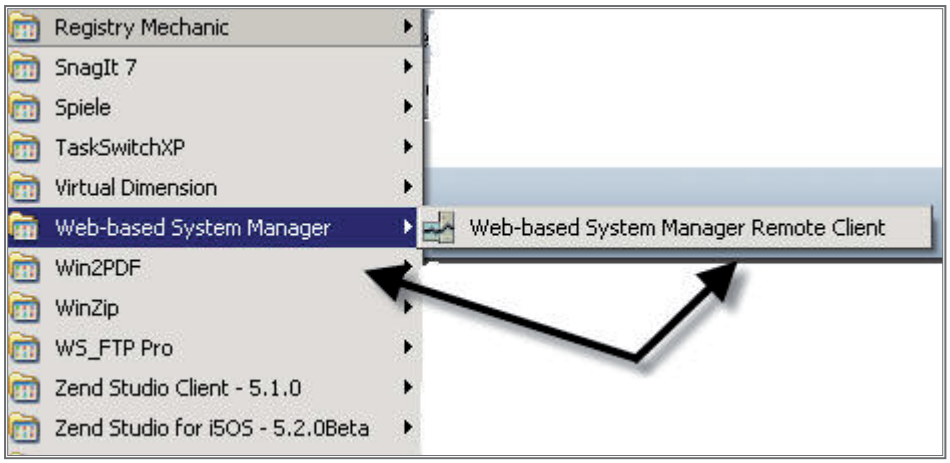
WebSM downloaden 3

Laden Sie jetzt WebSM auf Ihren PC herunter.

In dem Download finden Sie jetzt die Datei Setup.exe.

Führen Sie Setup exe aus.

Nachdem der Setup durchgeführt wurde, finden Sie für WebSM einen Eintrag auf Ihrem Desktop.



WebSM aufrufen

4.2.7.1
Seite 6

Aufruf von WebSM:

Start -> Programme -> Web-based System Manager -> Web-based System Manager Remote Client

Doppelklicken Sie auf diesen Eintrag.

Danach öffnet sich ein Dialog, der von Ihnen fordert, den Hostnamen der HMC bzw. die IP-Adresse der HMC anzugeben.

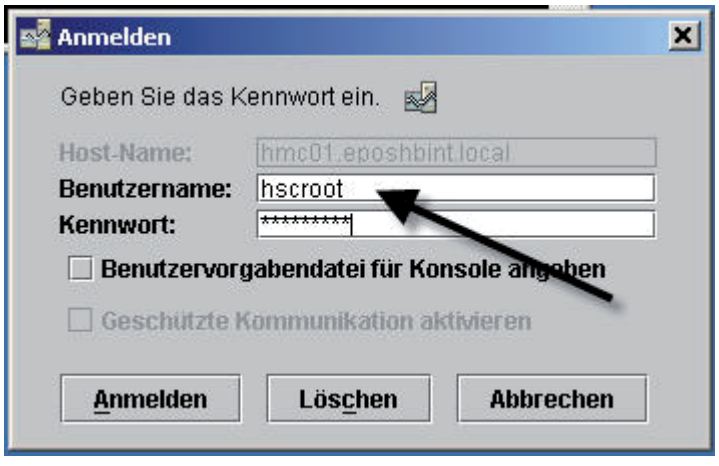
Hostnamen angeben

In meinem Falle gebe ich hier den qualifizierten Hostnamen der HMC an.

Dann einen Doppelklick auf „Anmelden“ ausführen.

Danach öffnet sich dieser Dialog.

Jetzt erfolgt die Anmeldung an der HMC.

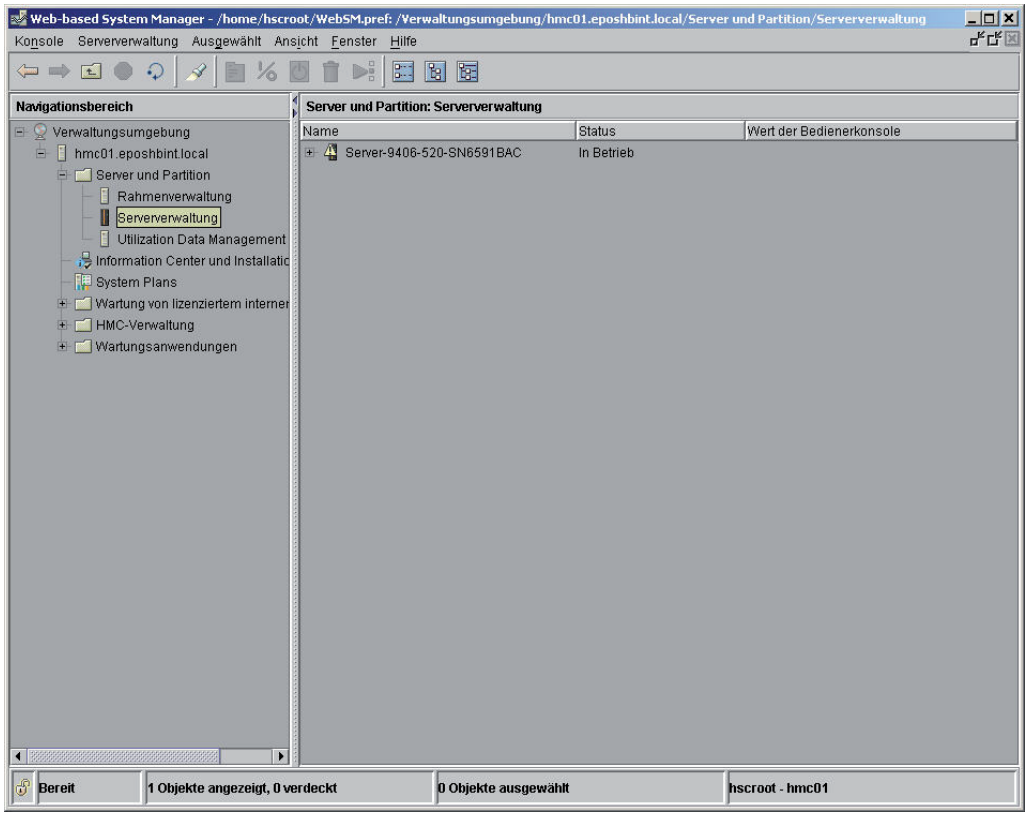


Anmeldung an WebSM-Client

Melden Sie sich mit einem registrierten HMC-Benutzer an.

Wie Sie in obigem Bild sehen, führe ich die Anmeldung als hscroot durch.

Danach öffnet sich WebSM.



Der WebSM-Client

Wie ich am Anfang dieses Abschnitts bereits sagte, ermöglicht WebSM den Zugriff auf **fast** alle Funktionen der HMC.

Eine der Funktionen, die die HMC von WebSM unterscheidet besteht darin, dass HMC für iSeries Systeme über eine eigene 5250-Emulation verfügt, so dass über eine HMC direkt eine Systemkonsole für ein System i zur Verfügung steht. Diese Funktion steht jedoch für WebSM nicht zur Verfügung.

Im nächsten Kapitel werde ich Ihnen zeigen, dass Sie auch von Ihrem Arbeitsplatz aus eine Systemkonsole erzeugen können.

4.2.7.1
Seite 8

Remote Systemkonsole für System i (DSP01)

Beabsichtigen Sie von Ihrem Arbeitsplatz aus eine Systemkonsole zu betreiben, so ist dies mit Hilfe einer HMC und einer 5250-Emulation möglich.

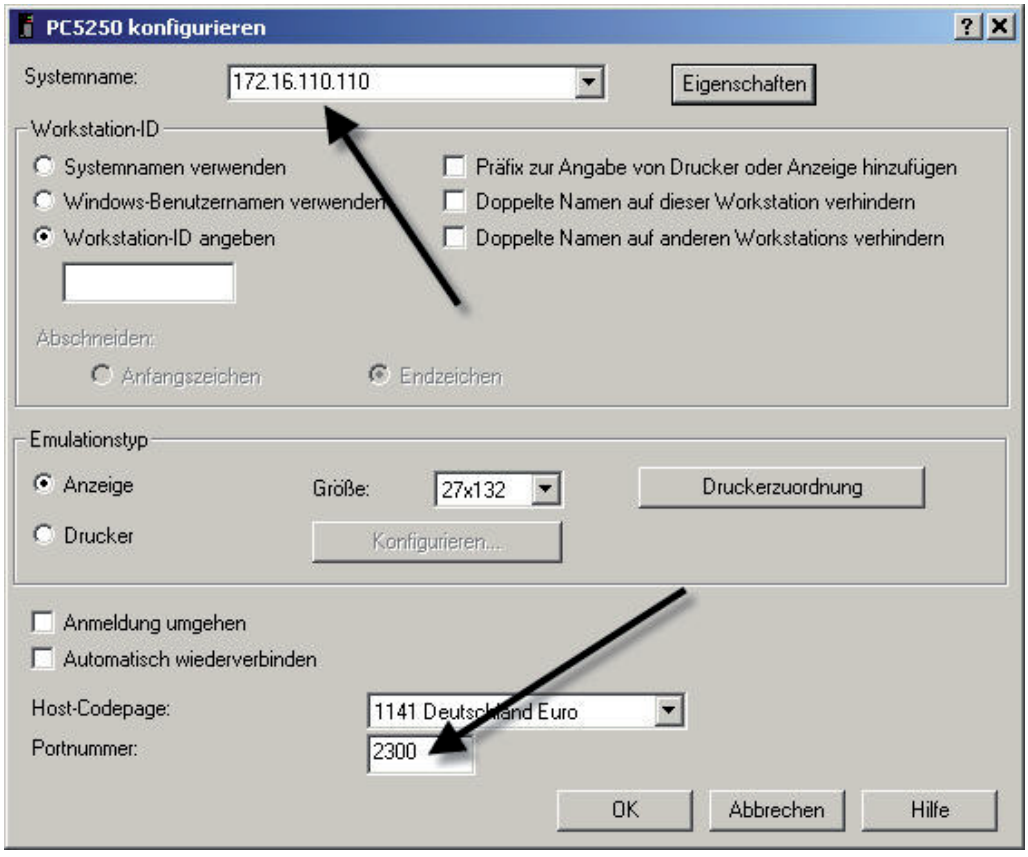
Voraussetzungen:

1. Funktionsfähige HMC
2. Eine Bildschirm-Emulation, bei der Sie die Angabe des zu verwendenden Ports frei vergeben können.
3. Eine Bildschirm-Emulation, die die Angabe eines Standardbenutzers zur Verbindungsanmeldung zulässt.

Ich verwende für dieses Beispiel iSeries Access für Windows in der Version V5R4. Die hier getätigten Konfigurationen gelten genauso für iSeries Access V5R3.

Öffnen Sie den Verbindungskonfigurationsdialog für iSeries Access aus der iSeries Access Programmgruppe -> Emulation -> Sitzung starten oder konfigurieren -> Neue Sitzung.

Folgender Dialog öffnet sich.



iSeries Access Verbindungskonfiguration 1

4.2.8
Seite 2

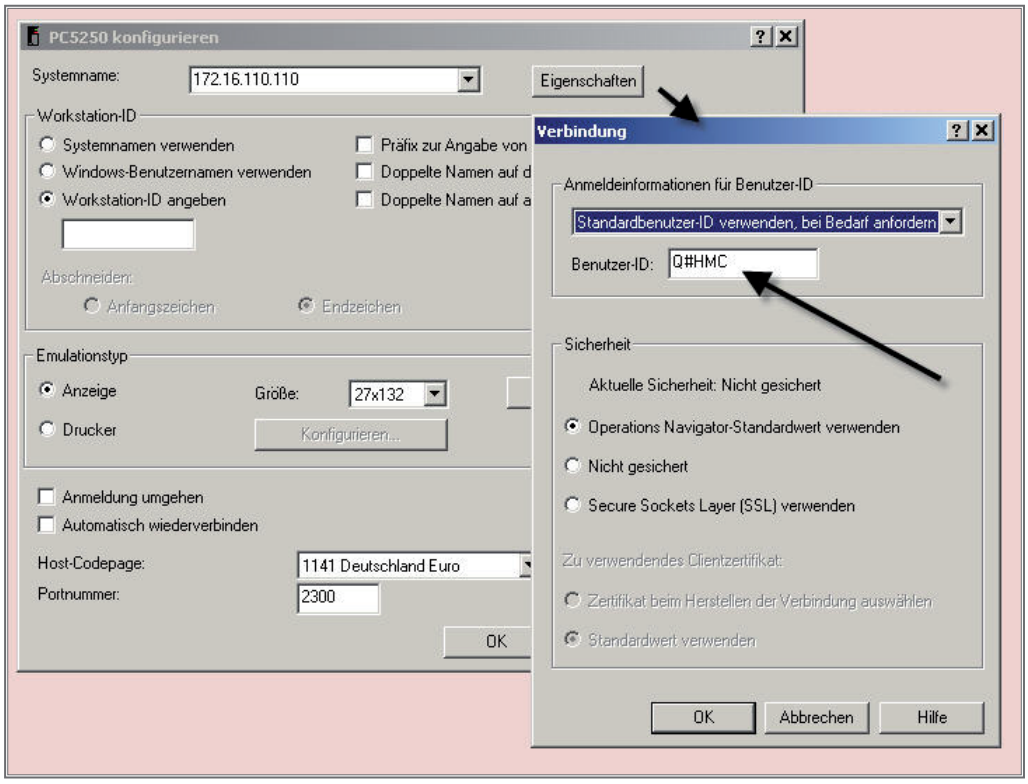
Versorgen Sie folgende Parameter:

Systemname: IP-Adresse oder Hostname der HMC

Hier: 172.16.110.110

Portnummer: 23000

Klicken Sie dann auf den Eigenschaften-Knopf.



iSeries Access Verbindungskonfiguration 2

Tragen Sie in den Eigenschaftendialog folgende Werte ein:

Anmeldeinformationen für Benutzer-ID:

Standardbenutzer verwenden

Benutzer-ID: Q#HMC

Bestätigen Sie alle offenen Dialoge mit „OK“ und speichern Sie die getätigte Konfiguration.

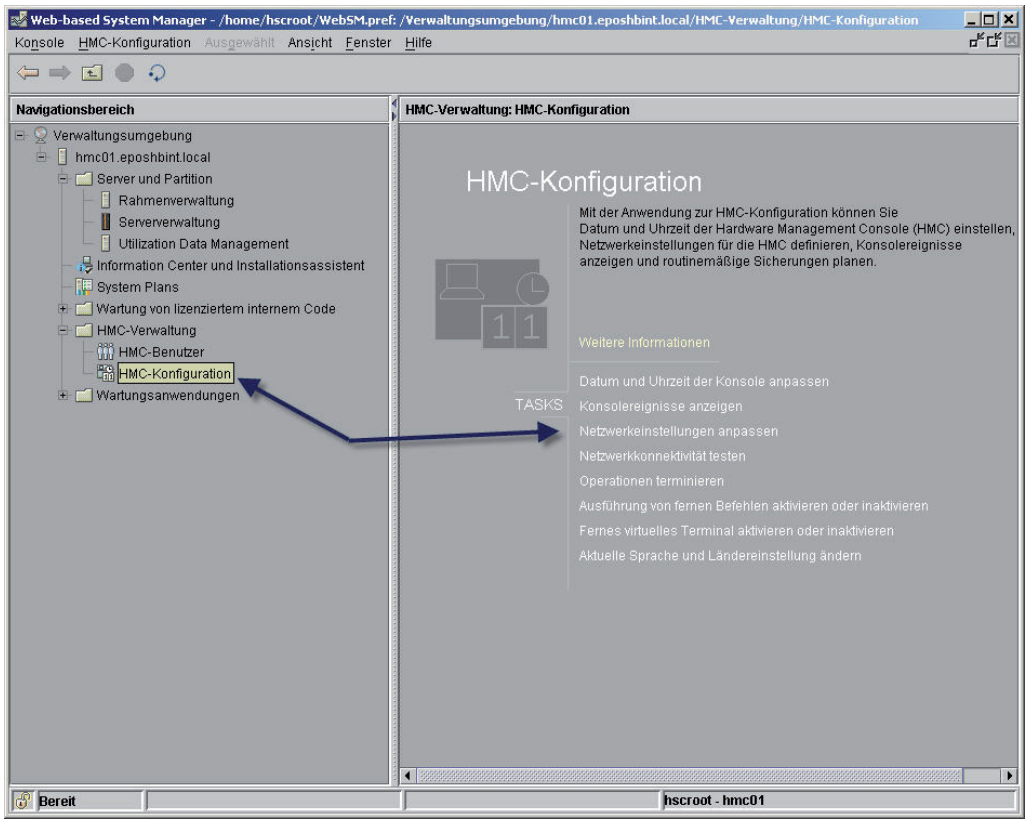
Sollte sich der Bildschirm nicht automatisch mit der HMC verbinden, so haben Sie ein Problem mit den Firewall-Einstellungen der HMC.

Firewall-Einstellungen der HMC anpassen

Sollte der oben gezeigte Dialog bei Ihnen nicht angezeigt werden, stattdessen jedoch die Meldung, dass aufgrund fehlender Berechtigungen der Zugriff verweigert wurde, so müssen Sie in der HMC den 5250-Zugang freischalten.

Dazu melden Sie sich an Ihrer HMC an und öffnen das Verzeichnis:

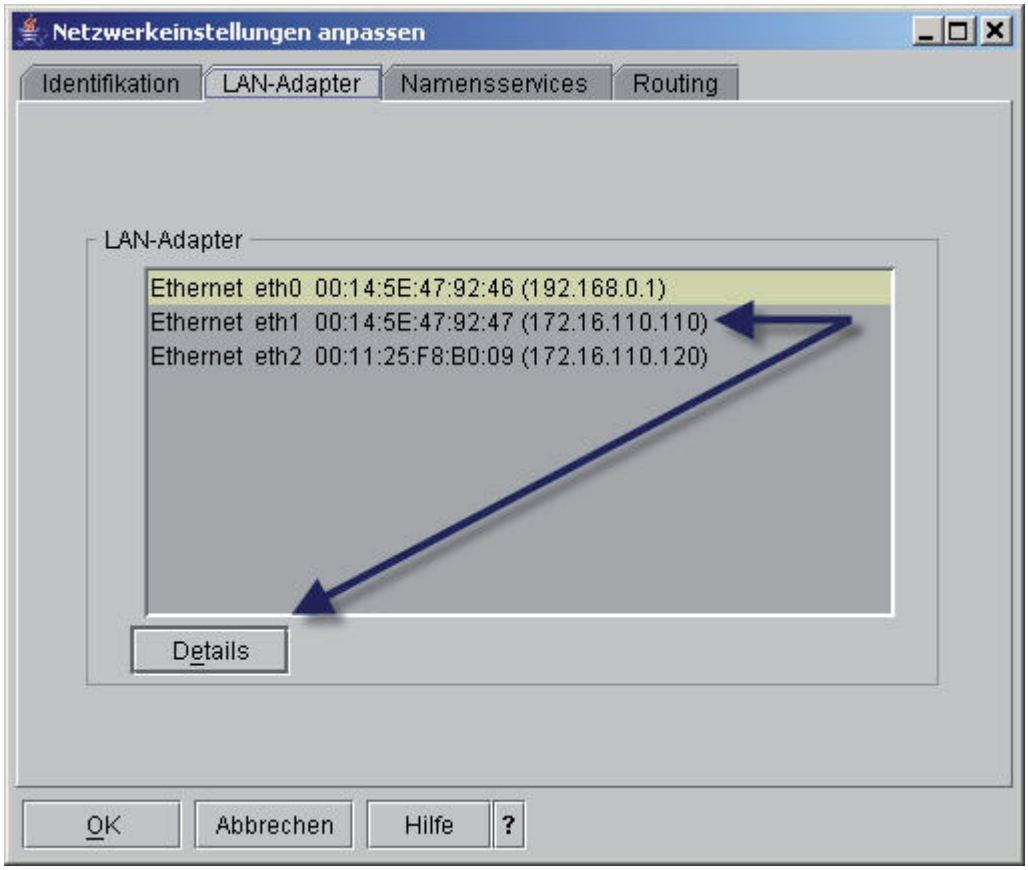
Verwaltungsumgebung -> <hmc-Name> -> HMC-Verwaltung -> HMC-Konfiguration.



HMC-Konfiguration

Dann klicken Sie auf „Netzwerkeinstellungen anpassen“.

4.2.8.1
Seite 2



LAN Adapter-Konfiguration

Markieren Sie jetzt die Ethernet-Schnittstelle/IP-Adresse, über die von Clients auf die HMC zugegriffen werden soll.

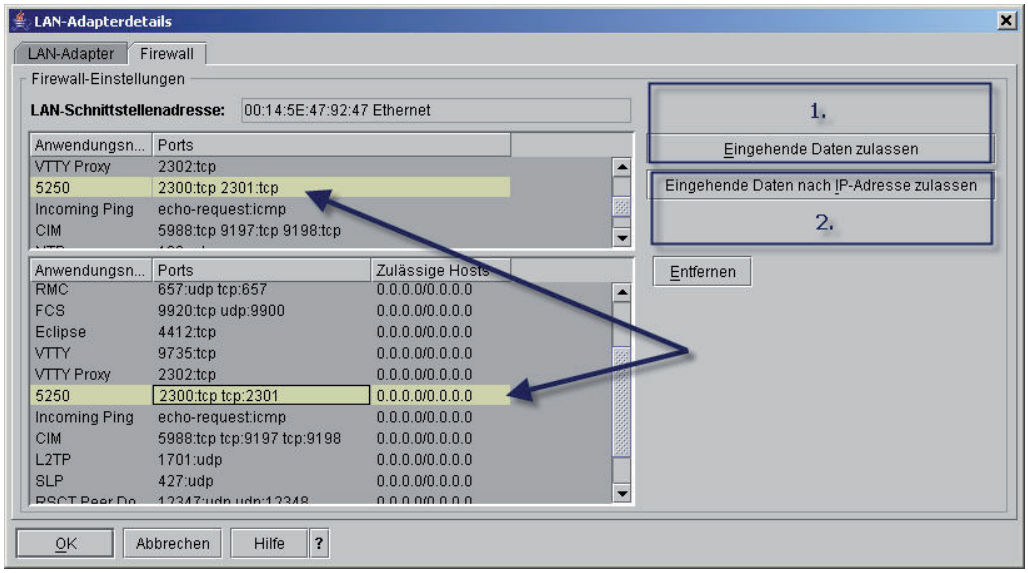
Klicken Sie auf „Details“.

Jetzt öffnet sich die Detailkonfiguration für diesen Land Adapter.

Wechseln Sie in das Register „Firewall“.

Markieren Sie im unteren Teildialog „5250“. Danach haben Sie zwei Möglichkeiten:

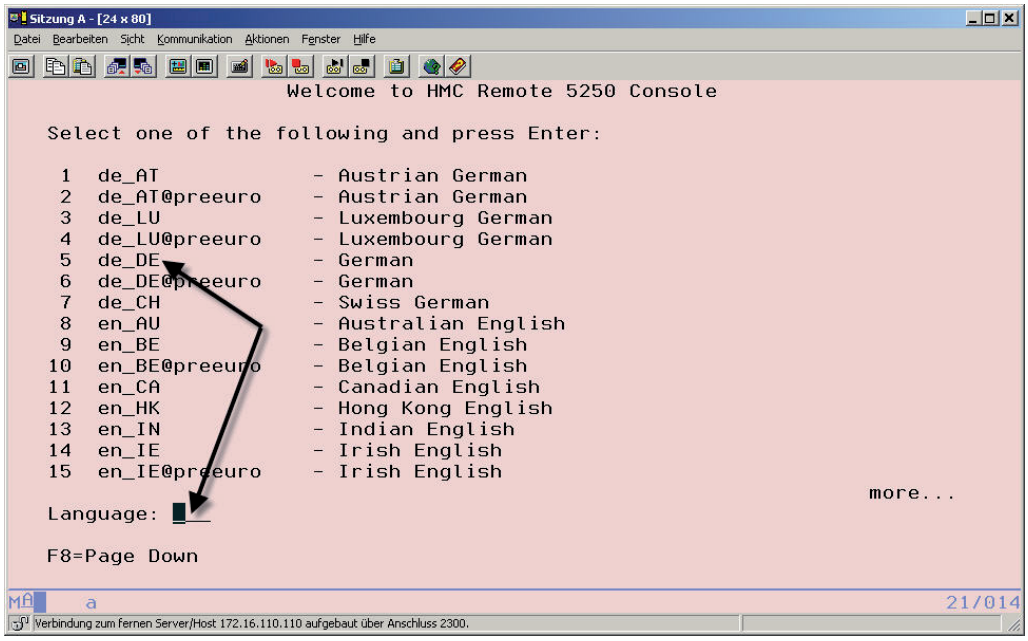
1. Eingehende Daten zulassen: Wenn Sie diesen Knopf drücken, kann jeder 5250-Client Ihres Netzwerks, der entsprechend konfiguriert wurde, die remote 5250-Konsole nutzen.
2. Eingehende Daten nach IP-Adresse zulassen: Sie benennen IP-Adressen von Clients, denen die Erlaubnis gegeben wird, die remote 5250-Konsole zu nutzen.



Firewall-Konfiguration

Sind die Firewall-Einstellungen korrekt, so sollte sich beim Verbindungsaufbau zwischen remote 5250-Konsole und HMC folgender Bildschirm öffnen.

In diesem Bildschirm wählen Sie aus, in welcher Sprache die 5250-Dialoge präsentiert werden sollen.

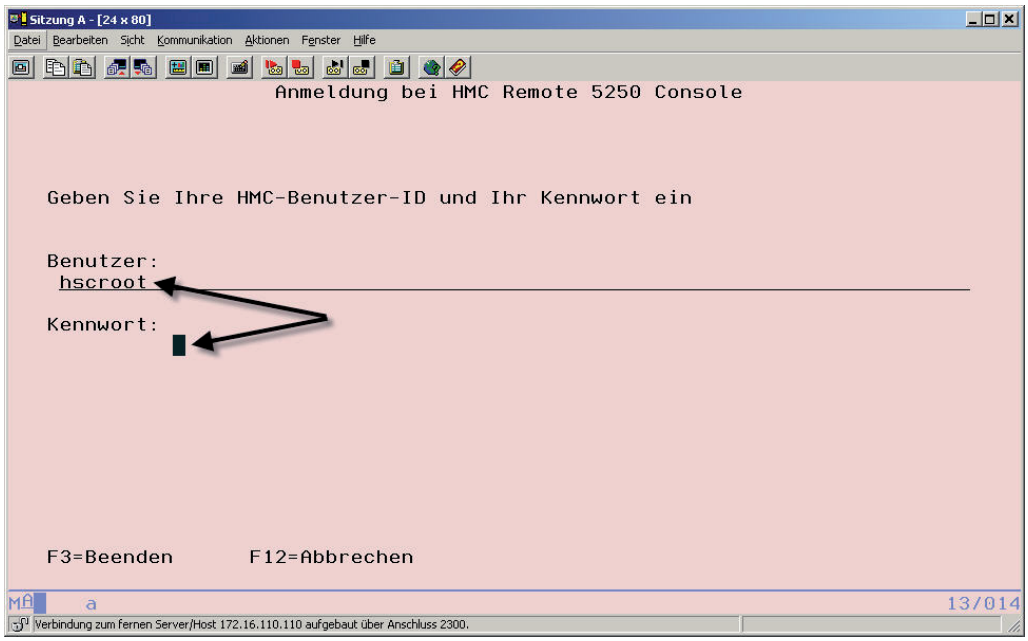


Sprache wählen

Ich wähle hier die Option 5: de_DE

4.2.8.1
Seite 4

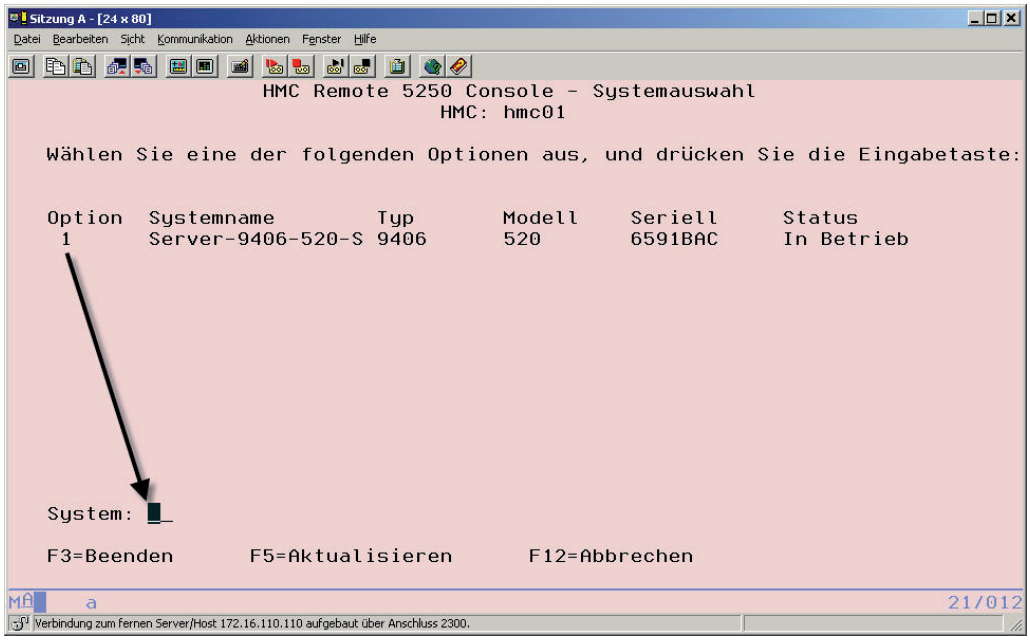
Jetzt öffnet sich der nachfolgende Bildschirm, mit dem Sie sich an der HMC anmelden.



Anmeldung an der HMC

Ich gebe hier den Benutzer „hscroot“ und sein Kennwort ein.

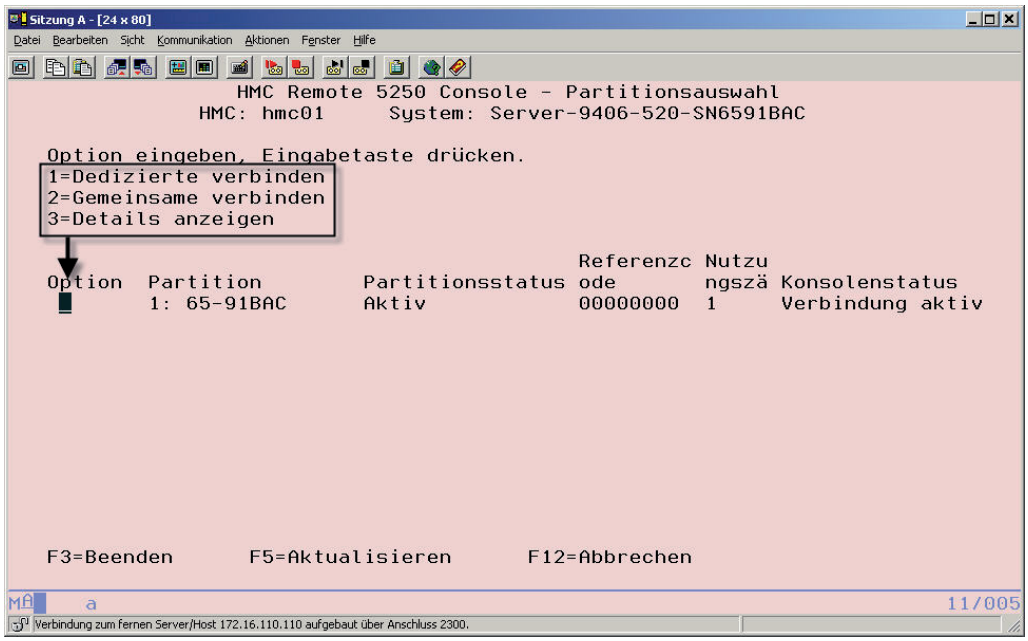
Der nächste Dialog zeigt Ihnen die durch die HMC verwalteten Systeme an. In diesem Fall wird nur ein Server, ein System i-Modell 520, Typ 9406 mit der Seriennummer 6591BAC durch die HMC verwaltet. Dieses System ist betriebsbereit.



Systemauswahl

Ich trage die Option 1 in das Feld „System“ ein und betätige Datenfreigabe.

Jetzt öffnet sich ein weiterer vorbereitender Dialog, in dem die Partitionen des Servers, den ich soeben angewählt habe, angezeigt werden.



Partitionsauswahl

Hier haben Sie die Möglichkeit, drei Werte einzutragen:

3: Detaildaten der gewählten Partition anzeigen. Dieser Punkt ist in diesem Zusammenhang nicht besonders interessant, deshalb zeige ich Ihnen hier keine Hardcopy an und verzichte auch auf eine Erläuterung des Dialogs.

1: Stellen Sie eine Dedizierte Konsolverbindung zum System i her. Es kann dann keine weitere Konsolverbindung aufgebaut werden.

2: Stellen Sie eine Gemeinsame Konsolverbindung zum System i her. Jetzt können mehrere Benutzer gemeinsam eine Verbindung zum System i nutzen.

Dazu ist es jedoch notwendig, einen Sitzungsschlüssel zu benennen.

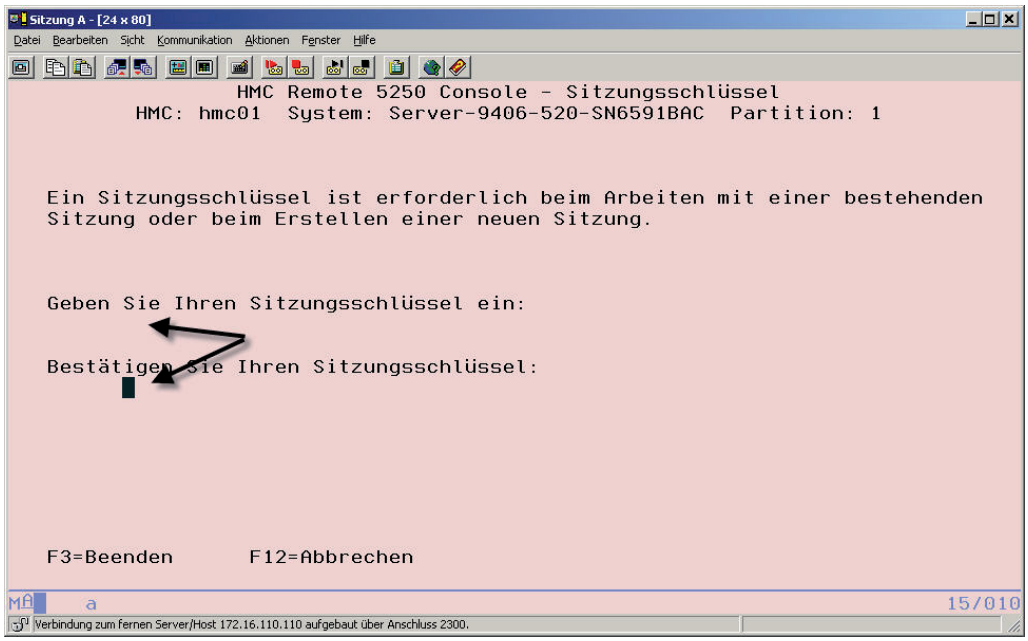
Dieser Sitzungsschlüssel ist eine frei wählbare Zeichenfolge, die aus mindestens 6 Zeichen bestehen muss.

Bei einer dedizierten Sitzung ist kein Sitzungsschlüssel zu benennen.

Ich wähle die Option 2 = Gemeinsame verbinden

4.2.8.1
Seite 6

Sie sehen sodann die Eingabe des Sitzungsschlüssels.

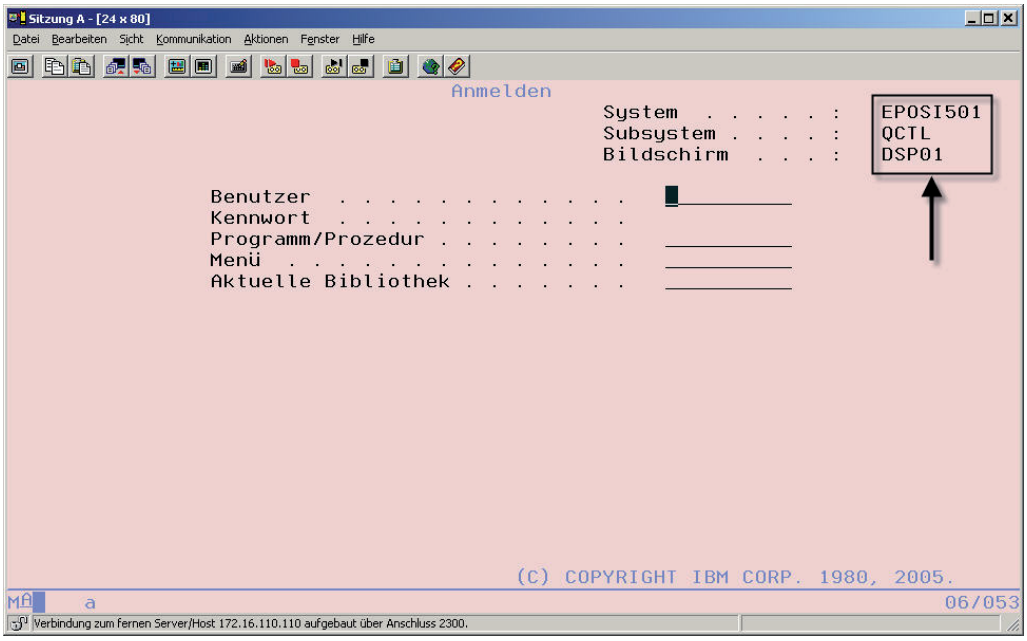


Sitzungsschlüssel benennen

Ich gebe zweimal den von mir gewählten Sitzungsschlüssel in den oben gezeigten Bildschirm ein.

Es werden keine Schutzzeichen angezeigt. Trotz Eingabe des Sitzungsschlüssels werden die Daten nicht visualisiert.

Ich drücke die Datenfreigabetaste und der Konsolbildschirm wird angezeigt.



Konsolbildschirm DSP01

Alles Weitere ist gewohnte Administratorentätigkeit, mit einem Unterschied: Ich brauche jetzt nicht mehr als Administrator zur Systemkonsole des Systems i hinzulaufen, sondern rufe sie von meinem Arbeitsplatz aus auf.

4.2.8.1
Seite 8

IBM Director – Grundlagen

Die Vielfältigkeit der IT ist in den meisten Unternehmen mittlerweile an Grenzen gelangt, welche eine Optimierung und Vereinfachung der Administration dringend erforderlich machen.

Zwar stehen uns gerade mit dem System i Möglichkeiten der Integration zur Verfügung, die gerade auch das Ziel verfolgen, den administrativen Aufwand zu minimieren – doch leider werden diese Möglichkeiten nur selten genutzt. Fakt ist, dass in den meisten Unternehmen ein mehr oder weniger heterogenes Umfeld von unterschiedlichsten IT Systemen, beginnend mit verschiedenen Servern auf Intel basierten Prozessoren oder anderen Systemen, eine individuelle Aufgabe zu bewältigen hat, über Rechner, die beispielsweise für Konstruktionssysteme im CAD Umfeld eingesetzt werden, bis hin zu den Hostsystemen, wie sie auch in Form der Power Systeme (System i, System p, System x und System z) zur Verfügung stehen.

Die Administration dieser IT Landschaften stellt heute einen großen Kostenfaktor dar. Dieser Kostenfaktor wird aber von Unternehmen aufgrund der Bedeutung der IT häufig „gerne“ in Kauf genommen – bildet doch die Verfügbarkeit der einzelnen Maschinen und der darauf laufenden Anwendungen das Rückgrat eines Unternehmens. Doch selbst die sinnvollsten Kosten bleiben, was sie sind: Kosten – und diese gilt es zu reduzieren.

Doch welche Möglichkeiten bestehen, die Kosten, die in der immer aufwendigeren Überwachung von Systemen entstehen, in einem gewissen Rahmen zu halten?

So heterogen wie die Server und die Anwendungen, so unterschiedlich sind auch die Werkzeuge und Techniken, mit denen der Betrieb und die Verfügbarkeit der einzelnen „Inseln“ sichergestellt werden müssen.

Wäre es nicht ideal, eine zentrale Überwachungsstation – eine Art „Notrufzentrale“ – für die gesamte IT Landschaft zu haben, mit der alle kritischen Prozesse und Systeme überwacht werden könnten? Wie wäre es, wenn beispielsweise beim Überschreiten von Plattenauslastungen eine Mail erzeugt werden würde, welche die Verantwortlichen darüber informieren würde, dass vielleicht eine Schleife in einem Programm eine riesige Datenmenge erzeugt, die droht, dass System zum Überlaufen zu bringen? Eine Meldung über eine CPF Meldung, die einen Job in der weiteren Verarbeitung hindert, wäre sicher auch als Mail oder SMS wünschenswert – ähnlich wie ein Abbruch auf einem Windows Server…

Sicher werden Sie sagen, kann man dafür Lösungen kaufen, die speziell in diesem Segment Hilfestellung leisten und in der Lage sind, die unterschiedlichen Systeme übergreifend und mit den feinsten Funktionen ausgestattet zu überwachen und zu managen. Doch es muss nicht immer das große teure Tool sein, das eine Lösung für die Anforderungen sein kann. Es kann auch eine vielleicht weniger bekannte Möglichkeit geben, die in der Lage ist, die Ziele zu erreichen.

IBM ist als Meister der „heimlichen Lösungen" bekannt – und eine dieser Heimlichkeiten bietet genau diese Möglichkeiten. Mit „Heimlichkeiten" sind solche Lösungen gemeint, die gerade auch die System i Gemeinde in Form von Software und Funktionen offeriert bekommen, welche nur leider allzu oft in den CD Hüllen ihr Dasein fristen. Einige dieser CD Hüllen beinhalten eine Software, die gerade die einfache Administration heterogener Umgebungen unterstützt und im Lieferumfang des System i enthalten ist. Die Rede ist von „IBM Director for Multiplatforms". Diese Software arbeitet eng mit den unterschiedlichen Virtualization Produkten zusammen, die im System i Umfeld zur Verfügung stehen, und ist eine Ergänzung in Sachen „Notrufzentrale" für Server, Clients und andere kritische IT Komponenten.

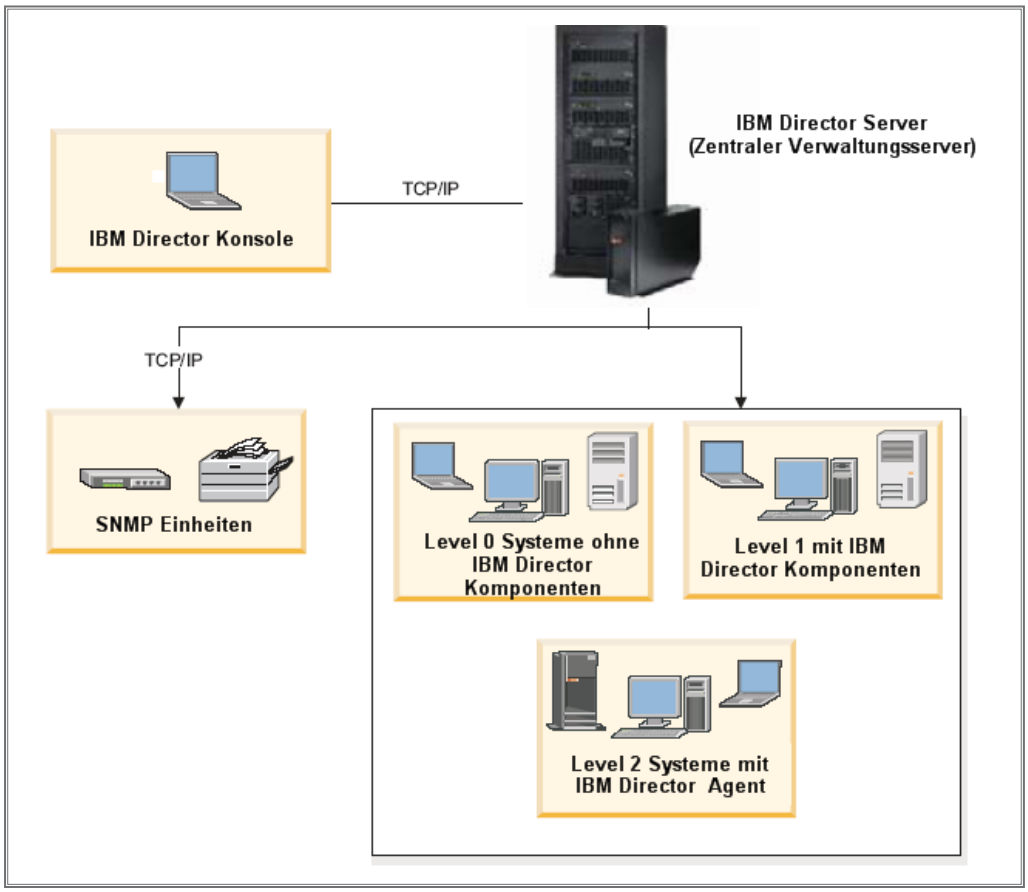
Wie der Name bereits erahnen lässt, handelt es sich dabei um eine Software, die für unterschiedliche Plattformen eingesetzt werden kann. Damit ist weniger die Installation der Software, sondern deren Einsatz in Bezug auf die Verwaltung und speziell auch die Überwachung von heterogenen Systemen gemeint.

Diese Software besteht sowohl aus Server- als auch Clientkomponenten.

Mit IBM Director for Multiplatform bietet IBM ein Produkt an, das unter anderem zwar auch auf bzw. mit dem System i eingesetzt werden kann, welches aber auch für Intel basierte Systeme, AIX, UNIX und noch einige andere Plattformen zur Verfügung steht. Mit IBM Director steht eine Werkzeugpalette zum Einsatz bereit, mit dem wir die unterschiedlichen Systeme zentral überwachen und in einem gewissen Rahmen auch zentral administrieren können. Der Vorteil zur jeweiligen Einzelüberwachung liegt auf der Hand: Es wird nur noch eine zentrale Stelle für die Überwachung der Systeme benötigt. Informationen über beispielsweise die Auslastungen der einzelnen Server, sich ankündigende Defekte oder auch Fehler- bzw. Ausnahmesituationen, werden auf diese Weise zentral gesammelt und können mit nur einer Oberfläche verwaltet werden. In einem weiteren Schritt lassen sich auch Aktionen definieren, die beispielsweise bei dem Eintreten eines Ereignisses wie zum Beispiel einer Fehlermeldung ausgeführt werden.

Dabei lassen sich nicht nur die Server in einem Unternehmen, sondern auch Clients oder Drucker in das IBM Director Netzwerk einbinden und zentral überwachen. Mit diesen Funktionen können wir beispielsweise auch den Ausfall eines Druckers feststellen, der für einen kritischen Prozess benötigt wird.

Die nachfolgende Abbildung zeigt schematisch die einzelnen Komponenten im IBM Director Umfeld:



IBM Director Topologie (Quelle: IBM)

Wie die vorhergehende Abbildung zeigt, kommunizieren die einzelnen Bestandteile einer IBM Director Installation über TCP/IP miteinander.

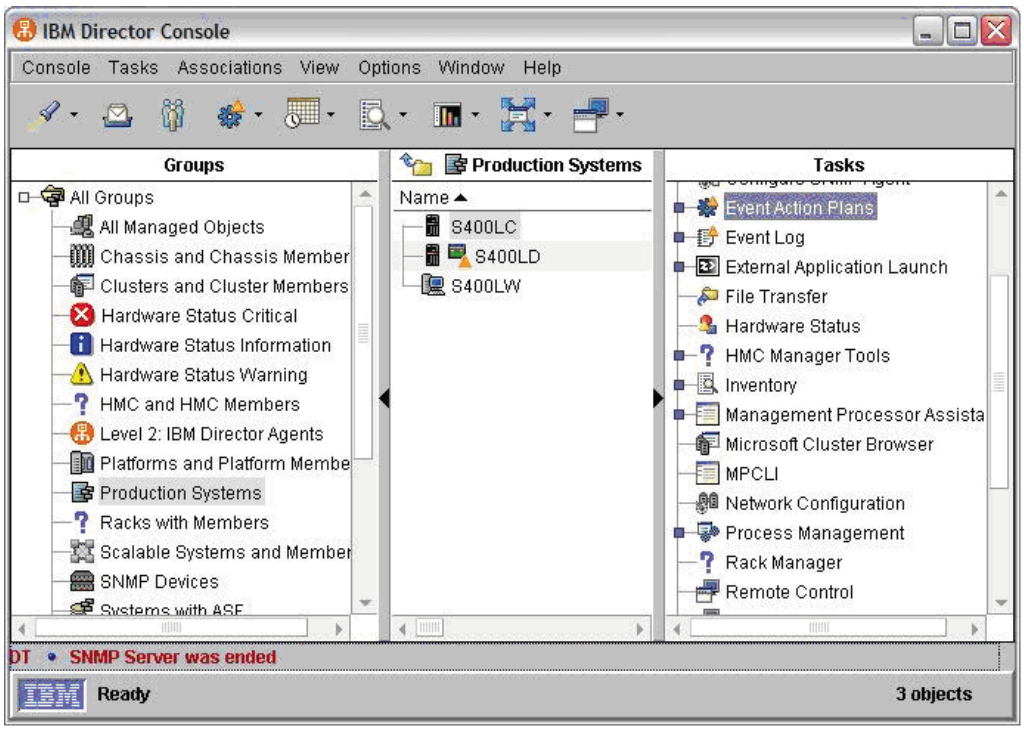
IBM Director kann als eine Art vollwertige Einstiegsversion der zentralen Systemüberwachung bezeichnet werden, lässt sich bei Bedarf aber auch in die „großen“ Lösungen wie zum Beispiel TSM einbinden. Wir wollen uns an dieser Stelle aber bewusst auf den IBM Director und seine Einsatzbereiche konzentrieren.

Wie bereits erwähnt, verfügt jeder System i Kunde über die Software, die für die zentrale Überwachung der gängigen IT Systeme eingesetzt werden kann. Der IBM Director for Multiplatforms besteht aus mehreren Installations-CDs, welche die verschiedenen Teilbereiche des IBM Directors abbilden. Natürlich kann die Software auch über das Internet herunter geladen werden. Diese Teilbereiche stellen sich wie folgt dar:

IBM Director Server
Bei dieser Komponente handelt es sich um das Herzstück des IBM Directors. Dieser Server wird zentral in einer Umgebung ausgeführt und sammelt die Informationen der übrigen Systeme innerhalb der Umgebung. Dieser Server ist auf allen eServer Plattformen – damit auch auf dem System i – verfügbar.

Der IBM Director Server verfügt über die Server Engine, die Datenbank und die Anwendungslogik, die für den IBM Director benötigt wird.

IBM Director Konsole
Bei dieser Komponente handelt es sich um das Verwaltungstool des IBM Directors, das als GUI Schnittstelle für die Administration dient. Diese Konsole kann beispielsweise auf einem Windows PC installiert werden, um die Einstellungen und Überwachungen der verschiedenen Server und Systeme innerhalb der IBM Director Umgebung durchführen zu können. Die Director Konsole wird derzeit neben Windows auch unter AIX und Linux unterstützt. Ein Beispiel für eine Anzeige der IBM Director Konsole:



IBM Director Konsole

Mit dieser Konsole lassen sich die Systeme definieren, die mit dem IBM Director überwacht werden sollen, und auch die Aktionen festlegen, die als kritische Situationen sowohl die Betriebssicherheit als auch die Verfügbarkeit überwachen. Diese Client Software ist ebenfalls mit im Lieferumfang des System i enthalten und muss auf dem verwaltenden PC installiert werden. Dabei lassen sich unterschiedliche PCs mit dieser Software ausstatten.

IBM Director Agents
Bei dieser Komponente handelt es sich um die Software, die auf den zu verwaltenden Clients installiert werden kann.

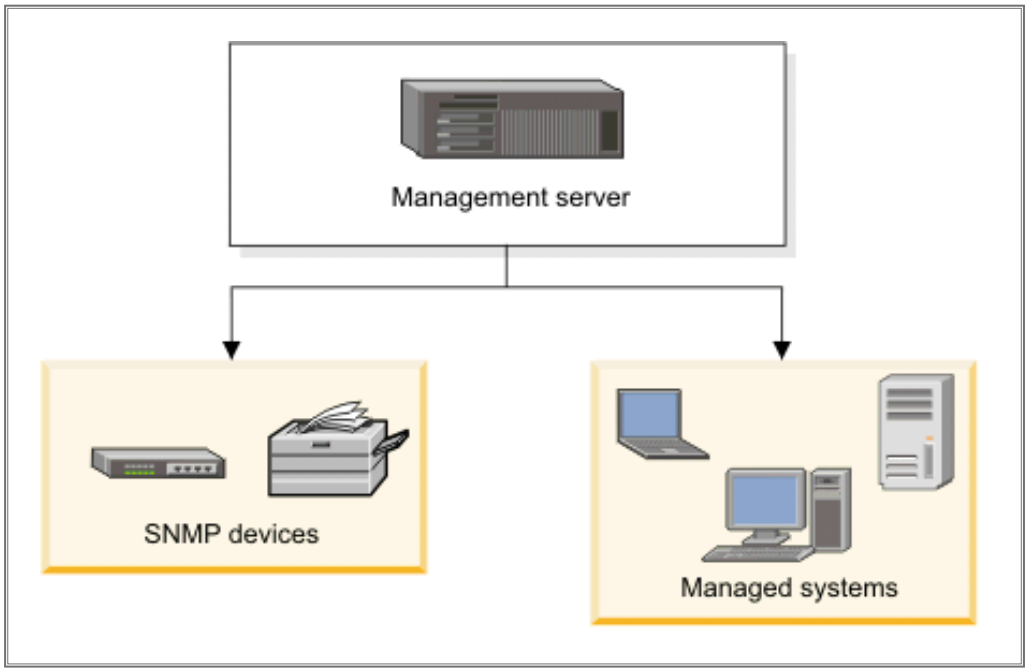
IBM Director Core Services
Dieser Bereich wird auf den Systemen installiert, um eine Kommunikation zu dem verwaltenden System herstellen zu können. Diese Core Services sind in der Lage, Informationen an das verwaltende System zu übertragen und auch bestimmte Aktionen auszuführen bzw. ausführen zu lassen:

- Inventarverzeichnis über die Hardware des jeweiligen Rechners
- Bereitstellen des Systemlogs
- Überwachung des Hardwarestatus
- Herunterfahren bzw. auch Neustarten des Systems
- Ausführen von Befehlen über eine Kommandozeile

In einem IT Umfeld können mit dem IBM Director die folgenden Einheiten bzw. Systeme eingebunden und damit überwacht werden:

1. Zentraler IBM Director Server (Management Server)
   Dieser zentrale Server wird als Verwaltungsserver bezeichnet und bildet das Kernstück der IBM Director Installation. Dieser Server kann auf einem beliebigen System installiert werden. Die übrigen zu verwaltenden Server (und bei Bedarf auch einzelne Clients) senden bestimmte Informationen an diesen zentralen Server, der diese anschließend für Administrations- und Verwaltungszwecke einheitlich aufbereitet für den Administrator zur Verfügung stellt.
2. Verwalteten Servern (Managed Server)
   Bei diesen Einheiten kann es sich um Server, aber auch um einzelne Clients handeln, deren Verfügbarkeit überwacht werden muss.
3. SNMP Einheiten
   Dies können beispielsweise Netzwerkkomponenten oder auch Drucker sein, die über SNMP verfügen und auf Basis dieses Protokolls Informationen mit dem IBM Director austauschen.

21. Ergänzung 12/2009



Management (Quelle: IBM)

In einem Umfeld, in dem der IBM Director zur Überwachung von unterschiedlichen Systemen eingesetzt werden soll, muss ein zentraler Server vorhanden sein, der die Informationen der übrigen Systeme zentral sammelt und die Schnittstelle zu dem Administrator bildet. Dieser zentrale Serverdienst sammelt die Informationen der übrigen Server und Clients und steuert auch die Aktionen, die als Unterstützung für den Administrator ausgeführt werden. So lassen sich beispielsweise auf einem solchen Server auch Schwellenwerte für die Auslastung einzelner Bereiche des System i definieren, welche bei einem Überschreiten als Auslöser für eine Aktion – wie das Erstellen einer Benachrichtigung an den Administrator – dienen.

Die Grundlage für diese Aktionen und auch das Sammeln der Informationen der Teilsysteme, die mit dem IBM Director zentral überwacht werden, basieren auf einer SQL Datenbank. Die Inhalte dieser Datenbank lassen sich mit dem Verwaltungswerkzeug in Form der IBM Director Konsole einsehen und verwalten.

Aber schauen wir uns zunächst die Voraussetzungen an, die auf dem System i erfüllt sein müssen, damit wir den IBM Director mit der Maschine nutzen können:

Grundsätzlich gilt: Der IBM Director Server ist eine überwiegend in Java geschriebene Anwendung. Damit verbunden sind alle Anforderungen an das System i, was die Ausführung von Java Anwendungen betrifft. So beispielsweise auch die erforderliche Java Virtual Machine und die Performanceeinstellungen, die für eine optimale Java Ausführung notwendig sind.

Installation des IBM Director Servers:
– Mindestens 150 CPW
– Freier Plattenplatz 500 MB
– Speicherpool mind. 350 MB

Installation des IBM Director Agents:
– Mindestens 75 CPW
– Freier Plattenplatz 350 MB
– Speicherpool mind. 300 MB

Die Installation kann auf Systemen mit der Betriebssystemversion V5R3 oder V5R4 des i5/OS erfolgen.

Die Installation selbst kann für den IBM Director Server über den Befehl RSTLICPGM über die mitgelieferte CD erfolgen. Damit wird auf dem System i ein Lizenzprogramm 5722-DR1 installiert. Bei der Installation des IBM Director 5.20 müssen die beiden Komponenten „IBM Director Server“ und „IBM Director Agent“ über dasselbe Release verfügen.

IBM Director ist ein Werkzeug, mit dem wir nicht nur System i, sondern mittels SNMP (Simple Network Management Protocol) andere in einem Netzwerk befindlichen Systeme überwachen und verwalten können. Dabei ist der Einsatz nicht nur auf PCs und Server beschränkt, sondern lässt sich auch bei entsprechender Eignung beispielsweise auf Drucker, Scanner etc. erweitern.

Im Jahre 2004 startete die erste Version des IBM Directors. Diese frühe Version unterstützte im Wesentlichen die Betriebssysteme OS/400 (das heutige i5/OS), Windows und AIX.

Heute steht der IBM Director in der Version 5.2x zur Verfügung, die über komplexe Funktionen und Werkzeuge verfügt, mit denen eine zentrale Überwachung der typischen Installationen von System i & Co. durchgeführt werden kann.

In diesem Kapitel erhalten Sie einen ersten Einblick über die Funktionsweise und die Einsatzgebiete des IBM Directors. Dabei setze ich Kenntnisse in der Administration des System i ebenso voraus, wie ein Grundverständnis für TCP/IP und i5/OS Themen.

21. Ergänzung 12/2009

4.3
Seite 8

Installation der Komponenten auf System i – der IBM Director Server und Agent

Die Installation kann auf Systemen mit der Betriebssystemversion ab V5R3 des Betriebssystems i5/OS bzw. IBM i erfolgen.

Prüfen Sie im Vorfeld der Installation, ob die erforderliche Betriebssystemsoftware installiert ist:

Produkte oder Optionen	Lizenzpgm.Nr.
(nur V5R3) IBM Cryptographic Access Provider (128 Bit) für iSeries. Ab der Version V5R4 sind die in diesem Lizenzprogramm enthaltenen Funktionen im Betriebssystem enthalten!	5722-AC3
IBM HTTP Server für iSeries	5722-DG1
Extended Base Directory Support, Option 3	5722-SS1
Java Developer Kit 1.4, Option 6	5722-JV1
OS/400 – Qshell, Option 30	5722-SS1
OS/400 – Digital Certificate Manager, Option 34	5722-SS1

Neben den Lizenzprogrammen muss das Benutzerprofil auf dem System i über ausreichende Berechtigungen für die Installation verfügen. Zwar kann die Installation mit jedem beliebigen Benutzerprofil durchgeführt werden, das mindestens über die Sonderberechtigungen *SECADM und *ALLOBJ verfügt, ich verwende allerdings immer gerne den QSECOFR, um etwaige Berechtigungsprobleme von vornherein auszuschließen.

Prüfen Sie außerdem UNBEDINGT, ob vor der Installation des IBM Director Server nicht etwa ein Agent installiert ist. Dieser kann zum Beispiel in Form des Lizenzprogramms 5733-VE1 auf dem System existieren. In einem solchen Fall muss der Agent unbedingt vor der Installation des Servers deinstalliert werden!

Mit der Version 5.2x des IBM Directors werden verschiedene CDs mit dem Betriebssystem i5/OS ausgeliefert, die im Rahmen der Virtualization Engine genutzt werden können und die Installationssoftware für den IBM Director Server und den IBM Director Agent auf dem System i enthalten. In Abhängigkeit der Konfiguration, die später mit dem IBM Director realisiert werden soll, können Sie wahlweise den IBM Director Server auf dem System i als zentrale Stelle für die Verwaltung der gesamten IBM Director Umgebung verwenden, oder lediglich den IBM Director Agent installieren, um die Kommunikation mit anderen IBM Director Server zu realisieren. Davon zunächst unabhängig liefert IBM die Software für die beiden Bereiche auf einer CD „IBM Director for i5/OS“ aus.

21. Ergänzung 12/2009

4.3.1
Seite 2

Auf dieser CD befinden sich zwei Lizenzprogramme:

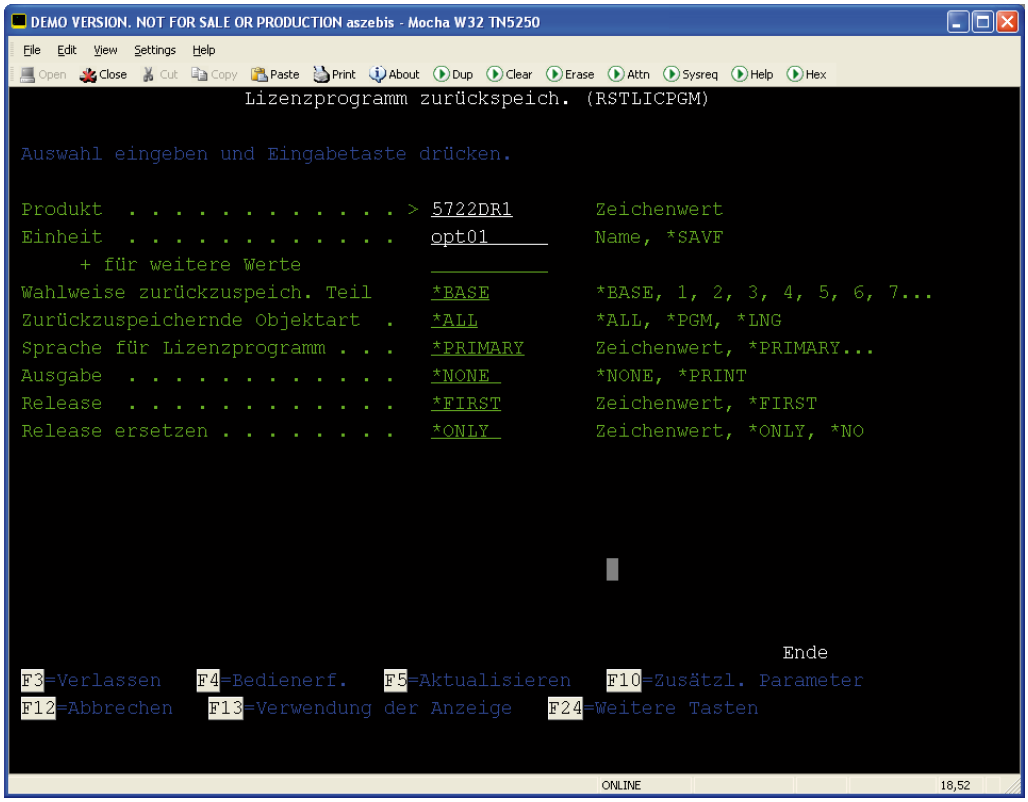
– 5722-DR1
  hierbei handelt es sich um die Kombination von IBM Director Server und IBM Director Agent.

– 5722-DA1
  Dies ist der IBM Director Agent – dieser liefert beispielsweise die Informationen über das System an die IBM Director Konsole.

Anmerkung

Prüfen Sie unbedingt vor der Installation der Software, ob auf dem System möglicherweise eine ältere Version des IBM Directors installiert war. Es ist zu empfehlen, ältere Versionen vor der Installation zu entfernen!

Die eigentliche Installation der Software ist für einen erfahrenen Administrator des System i sicher kein Problem. Nach dem Einlegen der CD „IBM Director for i5/OS“ in das CD/DVD Laufwerk der Maschine, der Anmeldung mit einem Benutzerprofil, das mindestens über die Sonderberechtigung*ALLOBJ, *SECADM und *IOSYSCFG verfügen sollte, wird die Installation mit dem Befehl RSTLICPGM gestartet.



RSTLICPGM

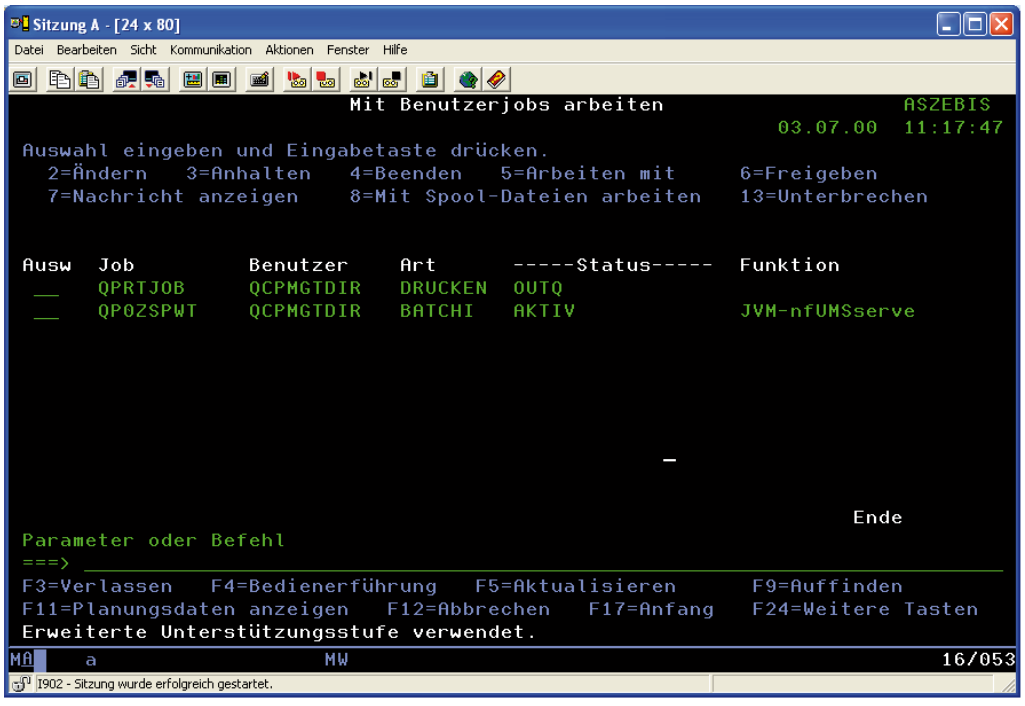
In Abhängigkeit dessen, welche Komponente Sie installieren wollen, geben Sie als Parameter „Produkt“ für den Befehl RSTLICPGM den Wert 5722-DR1 (Server und Agent) oder 5722-DA1 (nur Agent) ein.

Da wir in unserem Beispiel das System i zum zentralen Server auch für die Ausführung der Überwachungsdienste mit dem IBM Director definieren wollen, speichern wir mit dem Befehl RSTLICPGM das Lizenzprogramm 5722-DR1 zurück – darin enthalten: Der Server als auch der Agent.

Der IBM Director Server wird in Form eines speziellen TCP/IP Servers installiert. Dieser kann individuell verwaltet – das heißt auch gestartet bzw. gestoppt werden. Bevor wir den IBM Director Server bzw. den Agent nutzen können, starten wir diesen wahlweise mit dem Befehl STRTCPSVR *DIRECTOR.

Anmerkung

IBM liefert zusammen mit dem IBM Director ein separates Benutzerprofil QCPMGTDIR, das für die Ausführung der IBM Director Jobs auf dem System i genutzt wird. Ob der IBM Director läuft, kann beispielsweise mit dem Befehl WRKUSRJOB QCPMGTDIR geprüft werden:



IBM Director Server Jobs

4.3.1
Seite 4

Doch bevor wir nun den IBM Director nutzen können, müssen noch einige weitere vorbereitende Schritte ausgeführt werden. Zum einen benötigen wir noch eine IBM Konsole, die beispielsweise auf einem Windows Rechner als grafisches Interface für die Nutzung des IBM Directors genutzt wird, zum anderen sind zusätzliche Berechtigungseinstellungen erforderlich, ohne die der IBM Director nicht verwendet werden kann.

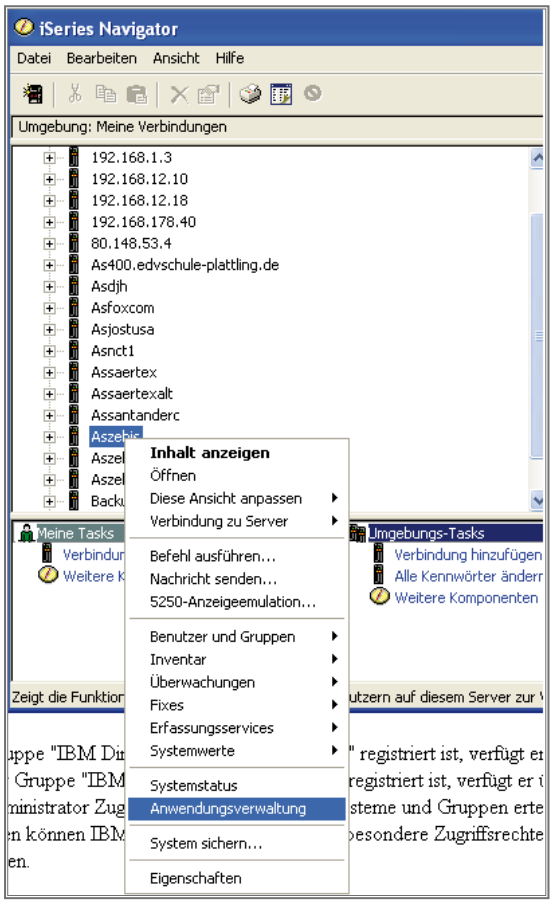
Benutzerverwaltung

IBM Director gestattet unterschiedliche Möglichkeiten der Authentifizierung. Neben dem klassischen Modell, welches i5/OS Benutzerprofile für den Zugriff auf die Funktionen des IBM Directors verwendet, können wir auch LDAP Authentifizierungen nutzen.

Für unser Beispielszenario wollen wir die einfachere Variante in Form der Nutzung von i5/OS Benutzerprofilen nutzen – denn in nur wenigen Fällen wird für die Authentifizierung LDAP genutzt.

Bevor ein Benutzer mit den Funktionen des IBM Directors arbeiten kann, muss er dafür berechtigt werden. IBM hat die Informationen für die Berechtigungsvergabe sehr gut verpackt, so dass man diese nicht auf Anhieb in der Dokumentation finden kann.

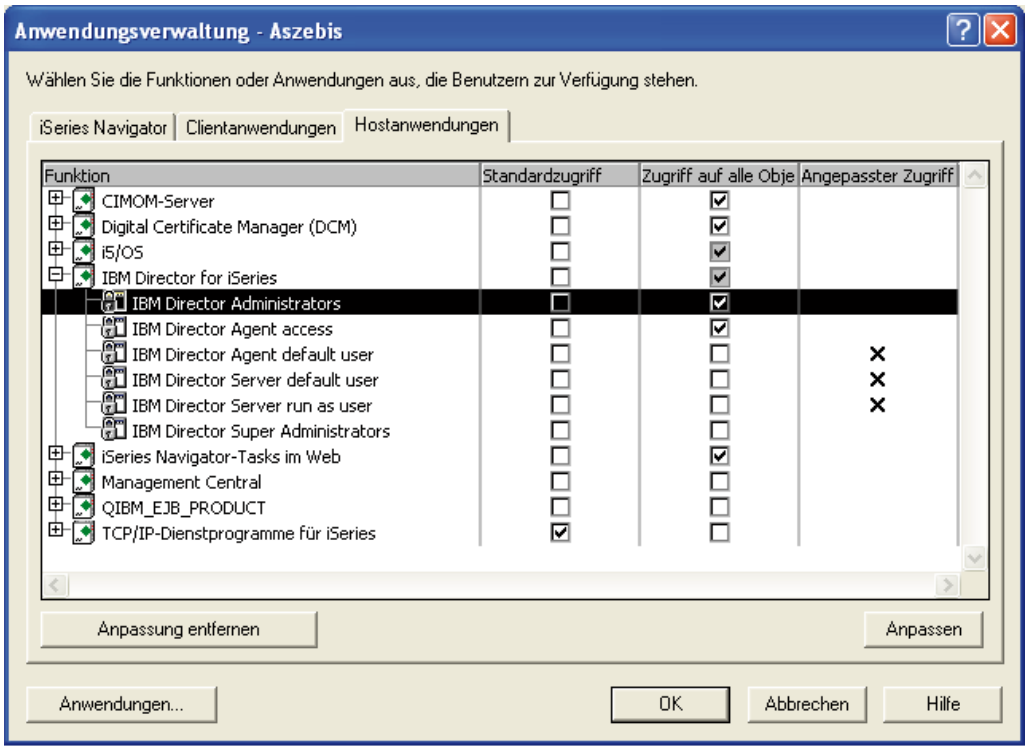
Für die grundsätzlichen Berechtigungseinstellungen verwenden wir den iSeries Navigator. Dort wählen wir das zu verwaltende System i – bzw. die Maschine aus, auf welcher der IBM Director Server installiert wurde. Mit der rechten Maustaste gelangen wir in ein Auswahlfenster, dass unter anderem den Eintrag „Anwendungsverwaltung“ beinhaltet. Diese Option selektieren wir nun:



Anwendungsverwaltung im iSeries Navigator

4.3.2
Seite 2

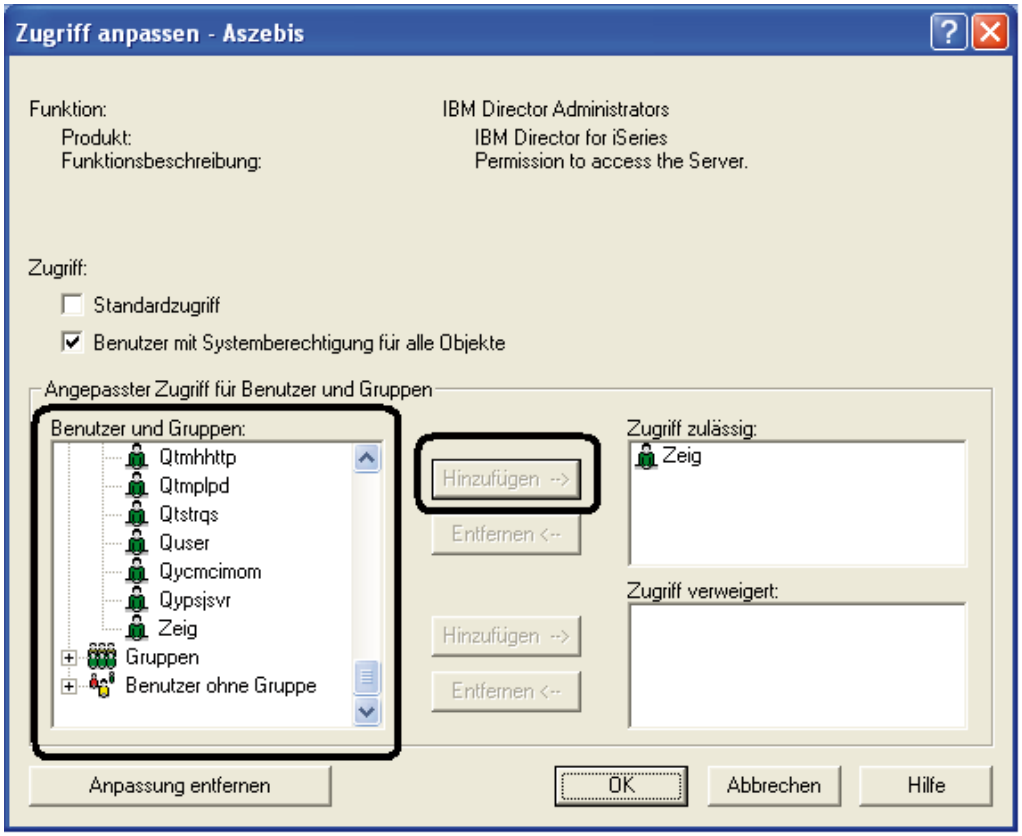
In dem sich nun öffnenden Fenster wählen wir den Bereich Hostanwendungen aus. Darin befindet sich unter anderem auch der Eintrag „IBM Director for iSeries“, den wir mit einem Klick auf das Erweiterungszeichen in Form des „+“ vollständig anzeigen lassen:



Berechtigungen anpassen

Die Untereinträge des IBM Directors for iSeries definieren die verschiedenen Standardbenutzergruppen, die mit dem IBM Director ausgeliefert wurden. Um die Einträge in diesen Gruppen zu verwalten, klicken Sie auf die gewünschte Gruppe. In unserem Beispiel habe ich die Gruppe „IBM Director Administrators“ markiert, weil ich dieser Gruppe einen Eintrag hinzufügen möchte.

Mit einem Klick auf die Schaltfläche „Anpassen“ gelangen wir dann in eine weitere Detailanzeige, mit der wir die Benutzer und deren Berechtigung festlegen können.



Benutzer selektieren

In dem linken Fensterbereich erhalten wir eine Auflistung aller i5/OS Benutzer. Um den gewünschten Benutzer auszuwählen, klicken wir auf einen der Einträge – in unserem Beispiel ist dies der Eintrag „Benutzer und Gruppen“, um den Inhalt in Form der angelegten i5/OS Benutzer anzuzeigen. In der Auflistung können wir nun das oder die Benutzerprofile auswählen, die Berechtigungen für das Arbeiten als Administratoren im Umfeld des IBM Directors erhalten sollen. Beachten Sie bitte, dass Sie Ihre Auswahl mit einem Klick auf „Hinzufügen“ bestätigen!

Anmerkung

Beachten Sie bitte, dass der IBM Director unterschiedliche Administrationsstufen kennt. Der einem QSECOFR auf dem System i am nähesten kommende Administrator im IBM Director Umfeld befindet sich nicht etwa in der Gruppe „IBM Director Administrators“, sondern in dem Bereich „IBM Director Super Administrators“.

4.3.2
Seite 4

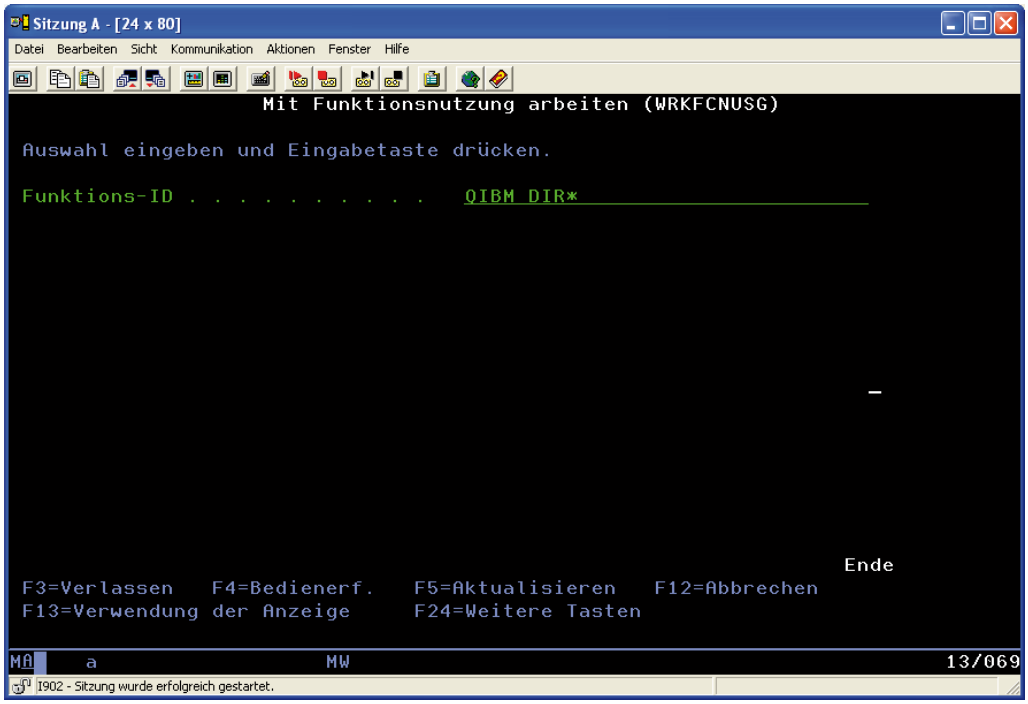
Die folgende Tabelle zeigt die unterschiedlichen Benutzergruppen innerhalb des IBM Directors:

Funktions-ID	Zweck
IBM Director Administratoren	Benutzer in dieser Gruppe können die Verwaltungsfunktionen mit Hilfe von Tasks, für die sie berechtigt sind, ausführen. Ein IBM Director-Superadministrator muss den Benutzern in dieser Gruppe besondere Zugriffsberechtigungen erteilen.
Zugriff auf den IBM Director-Agenten	Benutzer in dieser Gruppe können den Zugriff für ein verwaltetes System, auf dem der IBM Director-Agent ausgeführt wird, vom IBM Director-Server aus anfordern. **Anmerkung:** Standardmäßig verfügen alle Benutzer mit der Berechtigung *ALLOBJ über diese Funktion.
IBM Director-Agent-Standardbenutzer	QCPMGTDIR ist das Standardbenutzerprofil für diese Funktions-ID. Wenn ein anderes Benutzerprofil als das Standardprofil angegeben wird, können ferne Befehle über das angegebene Benutzerprofil auf einem verwalteten System ausgeführt werden. Beim Anfordern des Befehls sind keine Benutzer-ID und kein Kennwort erforderlich.
IBM Director-Agent als Benutzer ausführen	QCPMGTDIR ist das Standardbenutzerprofil für diese Funktions-ID. Wenn ein anderes Benutzerprofil als das Standardprofil angegeben wird, können Jobs auf dem verwalteten System unter diesem Profil ausgeführt werden. Um alle IBM Director-Tasks ausführen zu können, muss das Benutzerprofil über die Berechtigung *ALLOBJ verfügen.
IBM Director-Server-Standardbenutzer	QCPMGTDIR ist das Standardbenutzerprofil für diese Funktions-ID. Wenn ein anderes Benutzerprofil als das Standardprofil angegeben wird, werden Tasks, wie z. B. die Dateiübertragung, die Softwareverteilung und Ereignisaktionen, unter dem angegebenen Profil ausgeführt. Um alle IBM Director-Tasks ausführen zu können, muss das Benutzerprofil über die Berechtigung *ALLOBJ verfügen.
IBM Director-Server als Benutzer ausführen	QCPMGTDIR ist das Standardbenutzerprofil für diese Funktions-ID. Wenn ein anderes Benutzerprofil als das Standardprofil angegeben wird, können Jobs auf dem Verwaltungsserver unter diesem Profil ausgeführt werden. Um alle IBM Director-Tasks ausführen zu können, muss das Benutzerprofil über die Berechtigungen *ALLOBJ und *SECADM verfügen.
IBM Director-Superadministratoren	Benutzer in dieser Gruppe verfügen über die Berechtigung zum Konfigurieren eines Satzes von Zugriffsberechtigungen für die Gruppe „IBM Director-Administratoren", zum Bearbeiten von einzelnen Benutzerkonten und zum Verwenden der Funktionen des DIRCLI-Clients.

Anmerkung

IBM liefert zusammen mit dem IBM Director ein separates Benutzerprofil, das für die Ausführung der IBM Director Jobs auf dem System i genutzt wird. Dieses Benutzerprofil QCPMGTDIR sollten Sie nicht ändern – auch wenn IBM dies als Option in einigen Informationsschriften nicht ausdrücklich ausschließt!

Alternativ zur Berechtigungsvergabe mittels des iSeries Navigators (die zugegeben komfortabler ist als der ebenfalls zur Verfügung stehende Befehl) lassen sich die IBM Director Berechtigungen auch mit dem 5250-Befehl WRKFCNUSG (Mit Funktionsnutzung arbeiten) verwalten:

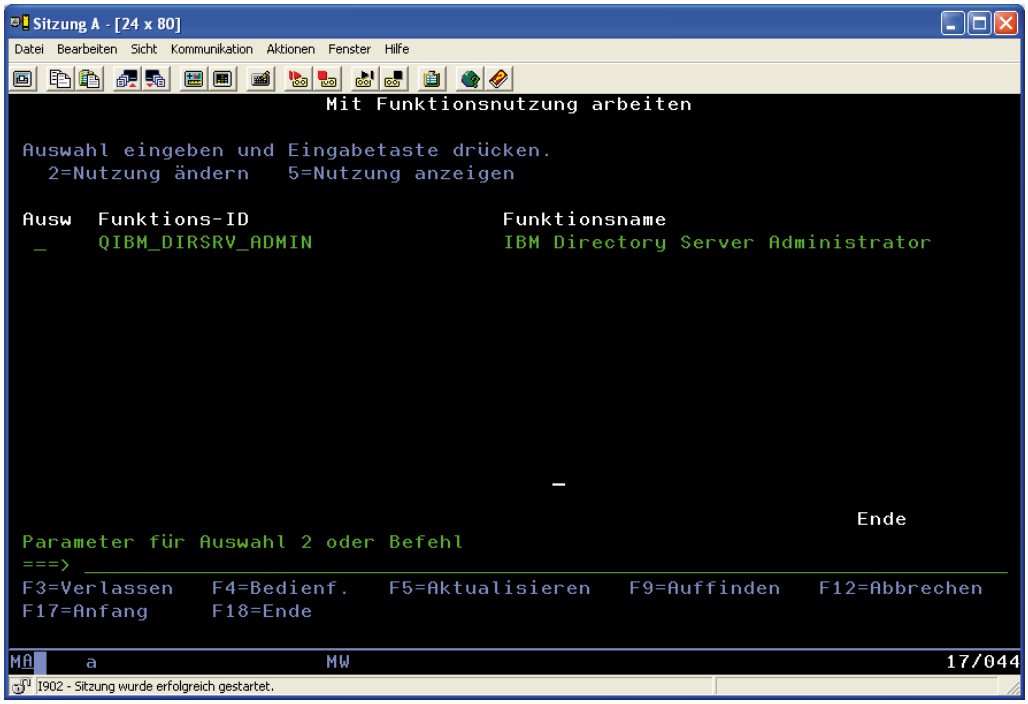


WRKFCNUSG

Die Berechtigungen des IBM Directors sind in dem Bereich „QIBM_DIR*“ zu finden, den wir als Parameter in dem Feld Funktions-ID angeben.

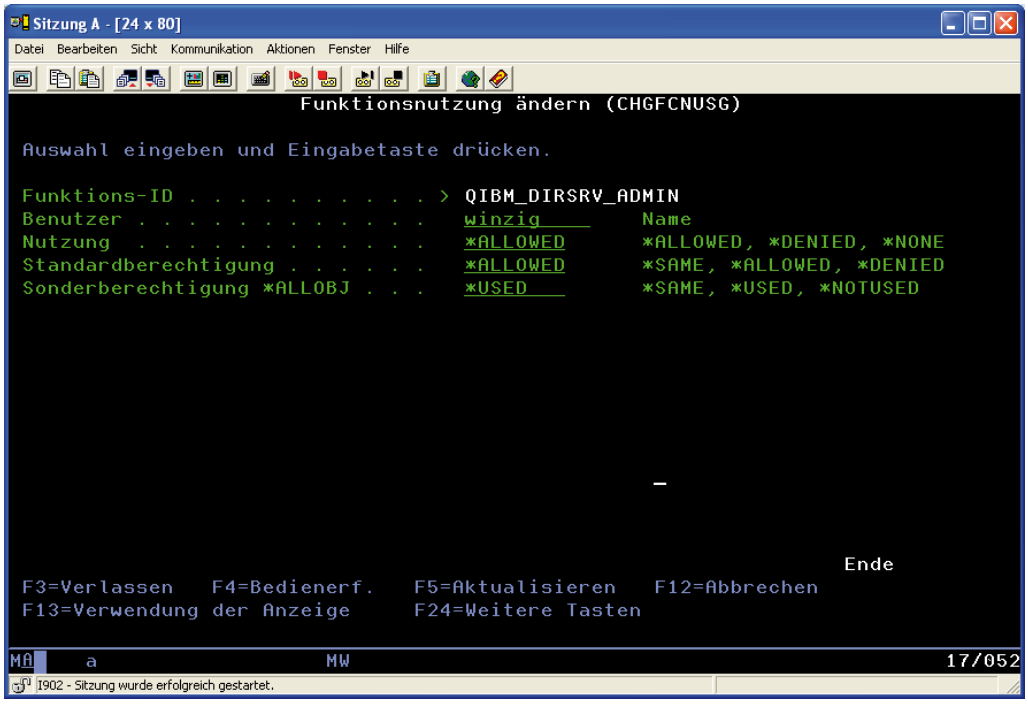
In der Folgeanzeige wählen wir nun die Funktions-ID QIBM_DIRSVR_ADMIN mit der Option 2 aus.

4.3.2
Seite 6



QIBM_DIRSVR_ADMIN

Mit der Auswahl „2“ gelangen wir in den Verwaltungsbereich, der sowohl zum Ändern bereits existierender Einträge, als auch zum Anlegen von neuen Benutzern für den Administrationsbereich des IBM Directors verwendet werden kann.



Benutzer berechtigen

Nach der Installation finden wir auf dem System i das Lizenzprogramm 5722-DR1. Für die Verwaltung einer IBM Director Umgebung benötigen Sie neben einem Server auch noch eine Konsole, mit der Sie die Umgebung steuern und überwachen können. Eine Besonderheit bei dem Einsatz des IBM Directors auf System i im Vergleich zu anderen Plattformen liegt darin, dass es keine native IBM Director Konsole für das System i gibt. Die Verwaltung des Servers lässt sich mit Hilfe von Befehlszeilenbefehlen direkt auf dem System i durchführen. Einfacher ist jedoch der Einsatz einer IBM Director Konsole in einer grafischen Form, wie sie beispielsweise für Windows Betriebssysteme angeboten wird. Diese Konsole ist im Lieferumfang der System i Software enthalten und kann auf einem beliebigen Windows Rechner mit aktuellem Betriebssystemstand (inkl. Servicepack!) installiert werden. Achten Sie unbedingt darauf, dass der Server und auch die Konsole des IBM Directors über dasselbe Release verfügen. IBM liefert auch die für die Konsole benötigte Software zusammen mit dem i5/OS aus. Die CD trägt die Bezeichnung „IBM Director Agents & Consoles 5.2x 2 of 2“.

Sollte der Installationsassistent für die Konsole nicht automatisch starten, dann können Sie diesen mit einem Klick auf das folgende Symbol starten.

Startbutton

Wie bei vielen anderen Softwarebereichen von IBM, so steht uns auch für die IBM Director Konsole ein Installationsassistent zur Verfügung, der uns durch die wenigen Schritte der Installation führt. Ich möchte an dieser Stelle nicht jeden einzelnen Schritt der Installation erläutern – denn die meisten Einzelschritte sind wirklich selbsterklärend.

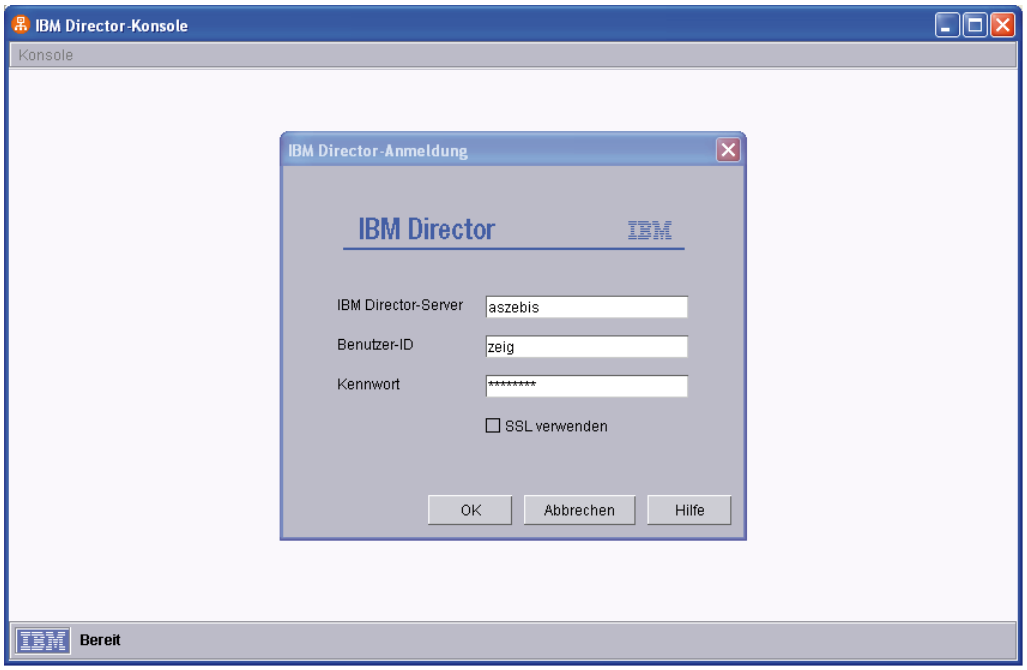
Anmerkung

Achten Sie unbedingt darauf, dass Sie die korrekte Version der IBM Director Konsole installieren. Hier muss unbedingt die Version mit der Version des IBM Director Servers übereinstimmen!

Wenn Sie nichts anderes während der Installation der IBM Konsole angegeben haben, dann lässt sich diese über den Windows Aufruf „Start / Alle Programme / IBM Director Console“ aufrufen.

4.3.2
Seite 8

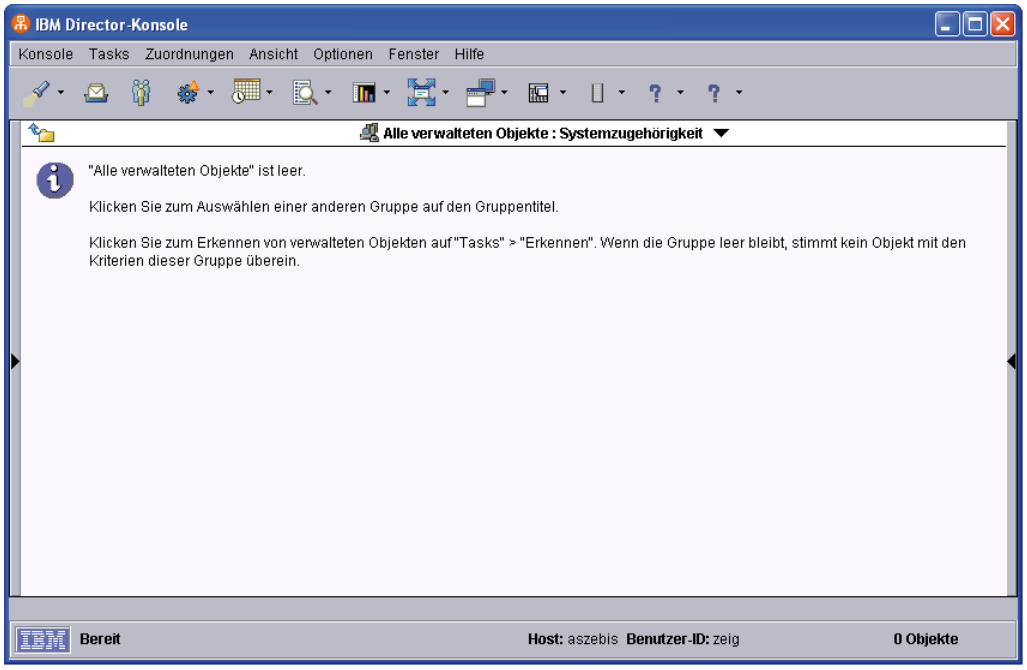
Zunächst müssen wir uns mit einem zuvor speziell für den Zugriff auf den IBM Director konfigurierten Benutzer anmelden. Als IBM Director Server muss der Hostname oder die IP Adresse des IBM Director Servers angegeben werden – dies ist in unserem Beispiel der Hostname des System i.



Anmeldung an der IBM Director Konsole

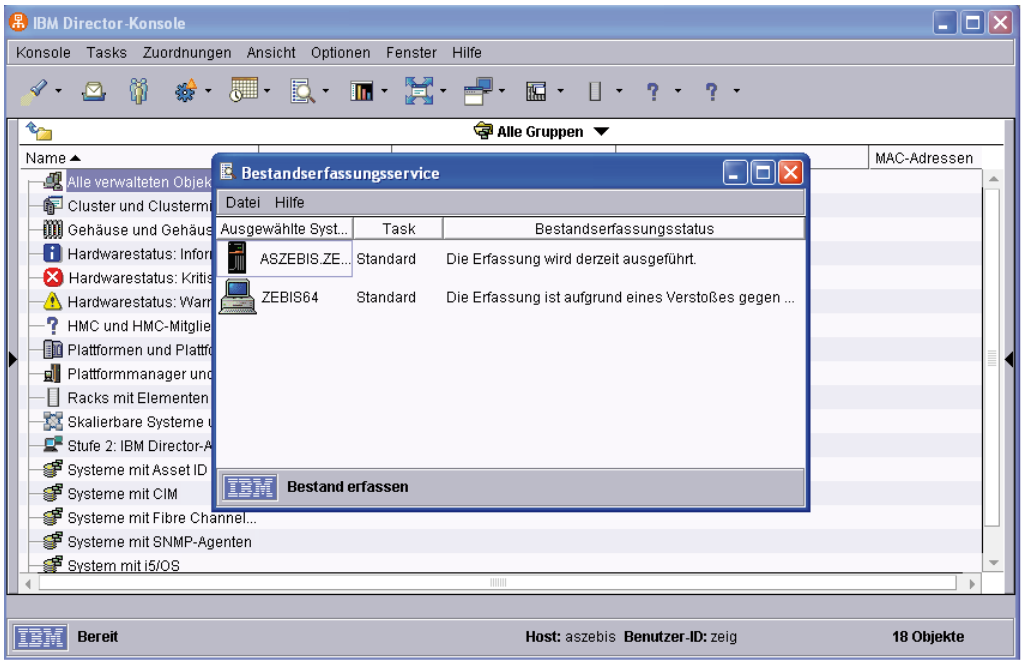
Wenn Sie eine sichere SSL Verbindung zwischen der Konsole und dem IBM Director Server aufbauen wollen, dann klicken Sie dazu auf das Optionskästchen „SSL verwenden“.

Je nach Voreinstellungen wird eine Auflistung der zu verwaltenden Systeme oder eine Hinweismeldung erscheinen, wie sie in der folgenden Abbildung zu sehen ist.



Leere Objekte

Sollte die IBM Director Konsole keine Einträge gefunden haben, so können wir diese einmalig erfassen lassen. Dazu wechseln wir in die Anzeige „Alle Gruppen“ und positionieren den Cursor auf den Eintrag „Alle verwalteten Objekte“. Hier selektieren wir mit einem Klick der rechten Maustaste die Option „Bestand erfassen“, wie es auch die folgende Abbildung zeigt.



Bestand erfassen

4.3.2
Seite 10

Nun kommt es auf die Umgebung des IBM Directors an. Mittels TCP/IP Verbindung wird nun auf den erreichbaren Systemen nach installierten IBM Director Komponenten gesucht, die als „Informationsgeber" genutzt werden können. Dies ist beispielsweise der IBM Director Agent, den wir zuvor bereits mit dem IBM Director Server auf dem System i installiert haben.

Diese Agent Software steht beispielsweise auch für Windows, UNIX, AIX und andere Betriebssysteme zur Verfügung. Bei der Erfassung der zu verwaltenden Objekte werden die Systeme aufgelistet, die in dem Netzwerk aktuell erreichbar sind und über eine Kommunikationsversion des IBM Directors verfügt.

Damit ist die Grundkonfiguration des IBM Directors für System i abgeschlossen.

IBM Director Konsole

In einer IBM Director Umgebung benötigen Sie neben einem Server auch noch eine Konsole, mit der Sie die IBM Dircetor Umgebung steuern und überwachen können. Eine Besonderheit bei dem Einsatz des IBM Directors auf System i im Vergleich zu anderen Plattformen liegt darin, dass es keine native IBM Director Konsole für das System i gibt.

Die Verwaltung des Servers lässt sich mit Hilfe von Befehlszeilenbefehlen direkt auf dem System i durchführen. Einfacher ist jedoch der Einsatz einer IBM Director Konsole in einer grafischen Form, wie sie beispielsweise für Windows Betriebssysteme angeboten wird. Diese Konsole ist im Lieferumfang der System i Software enthalten und kann auf einem beliebigen Windows Rechner mit aktuellem Betriebssystemstand (inkl. Servicepack!) installiert werden. Achten Sie unbedingt darauf, dass der Server und auch die Konsole des IBM Directors über dasselbe Release verfügen. IBM liefert auch die für die Konsole benötigte Software zusammen mit dem i5/OS aus. Die CD trägt die Bezeichnung „IBM Director Agents & Consoles 5.20 2 of 2“.

4.3.3 Seite 2

Die Verwendung der IBM Director Konsole

Für diesen Teilbereich benötigen Sie bestimmte Vorgaben, mit denen Sie sich im weiteren Verlauf an dem System anmelden und mit den entsprechenden Berechtigungen die Überwachungs- und Konfigurationsaufgaben durchführen können.

Bei der IBM Director Konsole handelt es sich um die grafische Benutzeroberfläche für die Verwaltung des IBM Directors. Diese steht in Form einer speziell zu installierenden Software für Windows und Intel Linux Betriebssysteme zur Installation zur Verfügung.

In der Regel erhalten Sie als System i Kunden die Software für die IBM Director Konsole in dem Lieferumfang Ihres i5/OS Betriebssystems.

Nach der Installation stellt die IBM Director Konsole eine Verbindung zu dem zentralen System her. Dies ist ein System, auf dem ein weiterer Teil der IBM Director Software installiert wurde – der IBM Director Server.

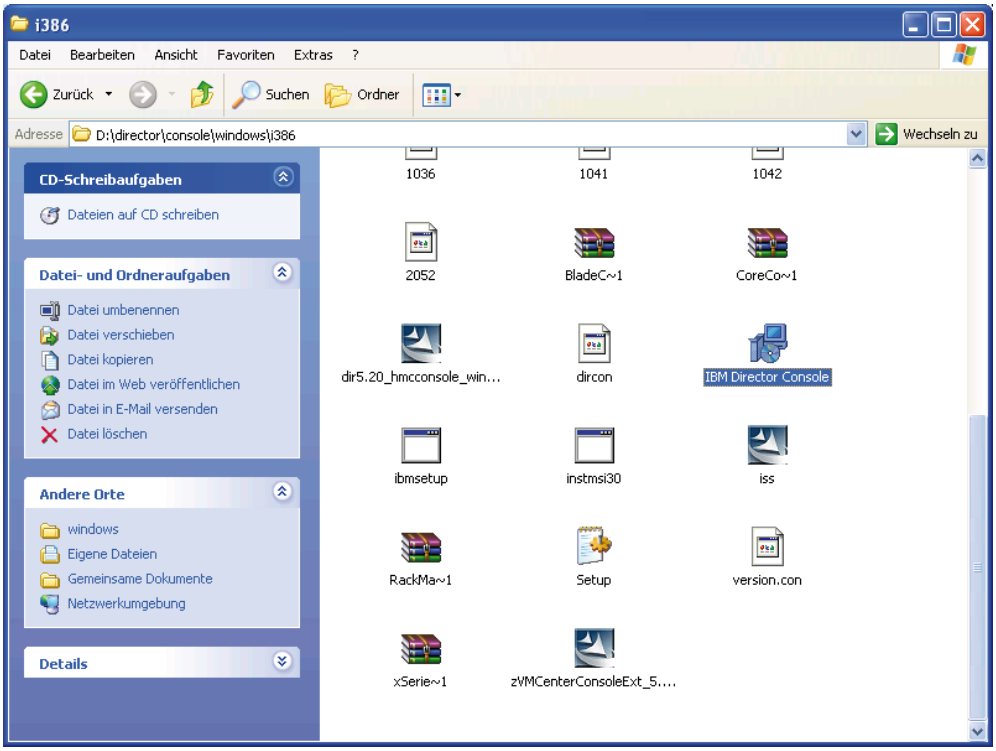
Inhalte dieses Teilbereichs
In diesem Abschnitt werden Sie lernen,
- wie man die IBM Director Konsole installiert,
- die erforderlichen Berechtigungen für den Zugriff definiert,
- die Anmeldung an der IBM Director Konsole durchgeführt werden kann.

4.3.3.1

Seite 2

Installation der IBM Director Konsole

1. Melden Sie sich auf dem Windows PC mit einem Benutzer an, der über Administrationsrechte verfügt.
2. Legen Sie die Installations-CD für die IBM Director Console ein. Die Software haben Sie i. d. R. mit der Lieferung des Betriebssystems i5/OS erhalten. Die CD trägt die Bezeichnung „IBM Director Agents & Consoles 5.20 2 of 2“. Meist startet damit der Installationsassistent automatisch. Sollte dies nicht der Fall sein, dann starten Sie den Installationsprozess mit einem Klick auf den Eintrag „IBM Director Console“ in dem CD Verzeichnis D:\director\console\windows\i386

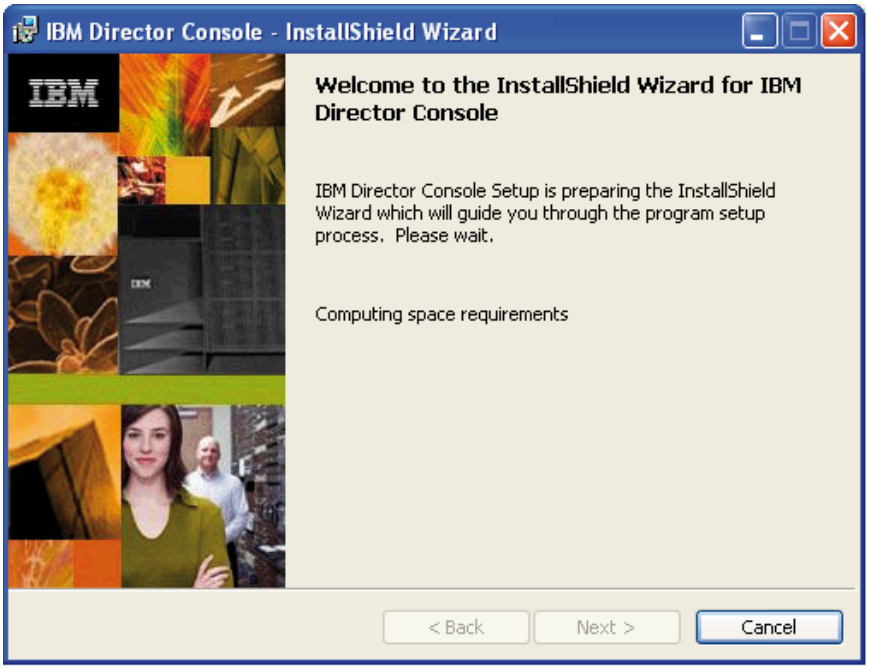


Installation IBM Director Konsole

3. Der Installations-Assistent führt Sie nun durch den Installationsprozess. Folgen Sie den Anweisungen.

4.3.3.2
Seite 2

4. Wenn noch keine vorherige Installation der IBM Director Konsole auf dem Rechner vorhanden ist, startet der Prozess der Installation nun mit der Willkommensseite, die Sie auch in der folgenden Abbildung sehen.



Start der Installation

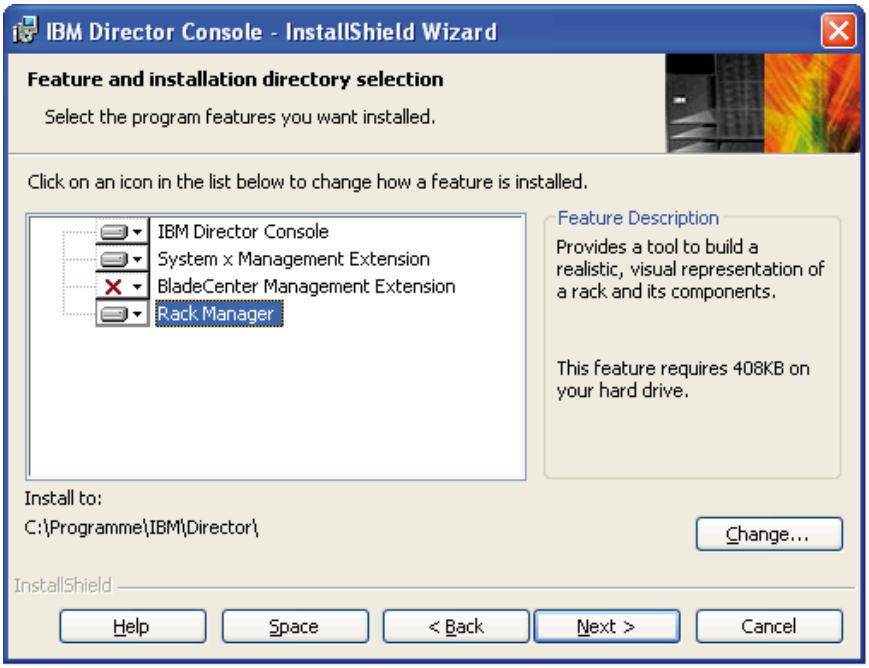
5. Mit einem Klick auf die Schaltfläche „Next“ setzen wir die Installation fort.

6. In dem folgenden Anzeigenbereich sind die Lizenzbestimmungen aufgelistet. Wenn Sie diese akzeptieren, dann wählen Sie die Option **„I accept the terms in the license agreement“** und bestätigen Sie Ihre Auswahl mit einem Klick auf die Schaltfläche „Next“.

Lizenzvereinbarungen

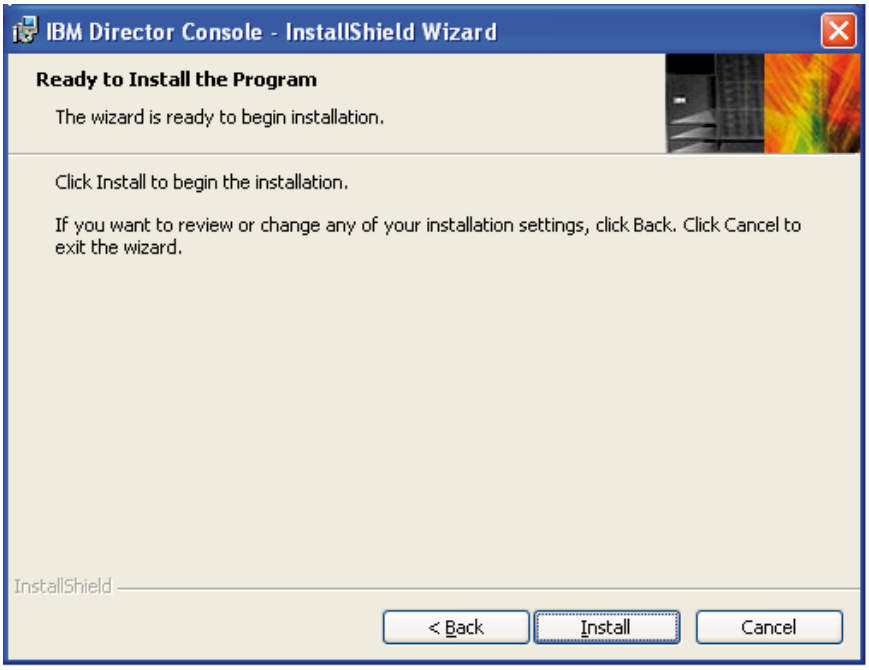
4.3.3.2
Seite 4

7. Mit der nächsten Anzeige haben Sie die Gelegenheit, die zu installierenden Komponenten der IBM Director Konsole festzulegen. Per Standard sind die beiden ersten Optionen vorselektiert. Sollten Sie beispielsweise ein Blade Center mit der IBM Director Installation überwachen wollen, dann wählen Sie dazu den entsprechenden Eintrag. Bestätigen Sie Ihre Auswahl mit einem Klick auf die Schaltfläche „Next“.



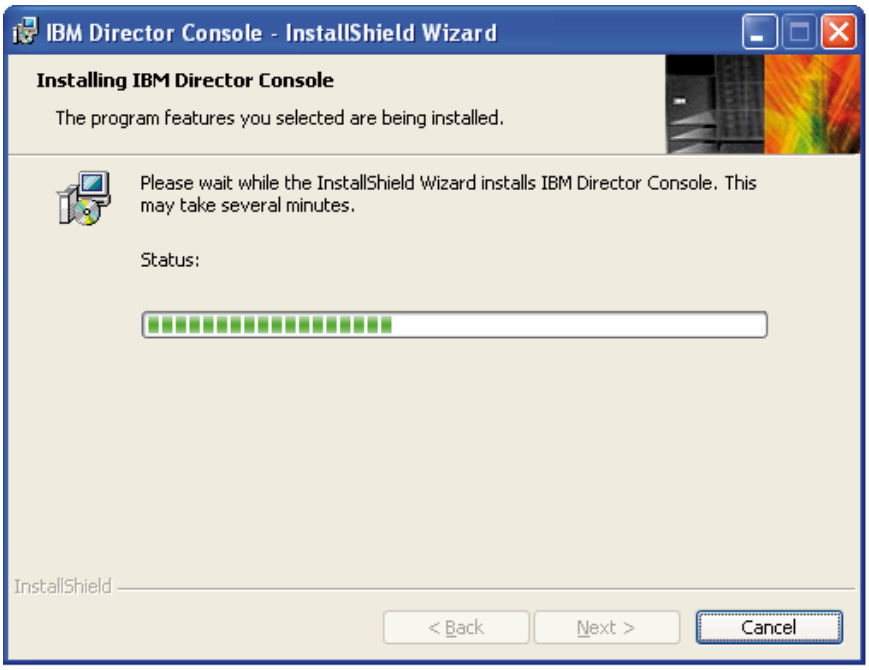
Zu installierende Optionen

8. Damit haben wir die erforderlichen Angaben getätigt. Mit der letzten Anzeige starten wir die eigentliche Installation mit einem Klick auf die Schaltfläche „Install“.



Starten der Installation

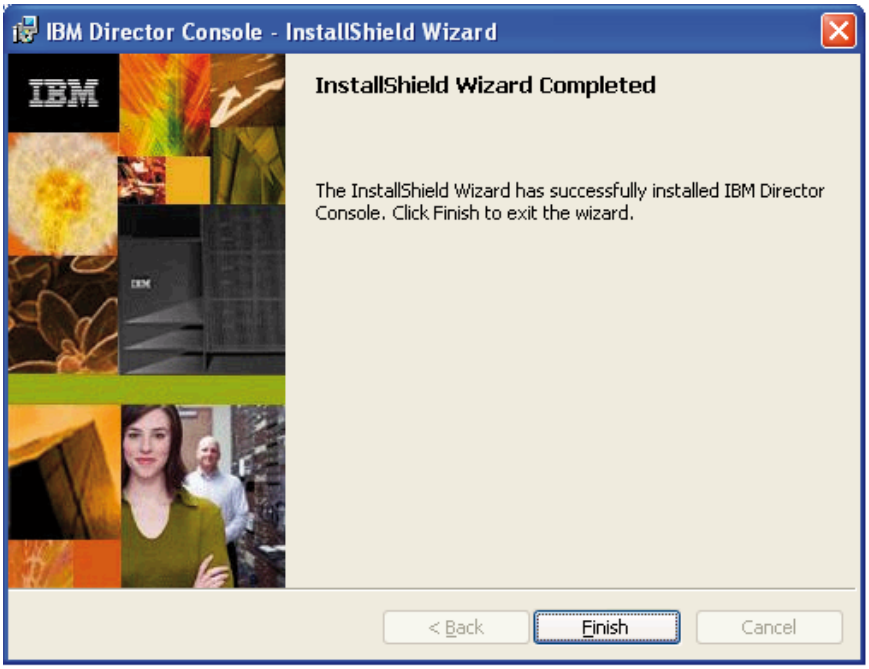
Eine Statusanzeige gibt uns Auskunft über den Status der Installation.



Installationsfortschritt

4.3.3.2
Seite 6

9. Nachdem die Installation abgeschlossen worden ist, wird eine entsprechende Meldung ausgegeben. Bestätigen Sie diese mit einem Klick auf die Schaltfläche „Finish“.



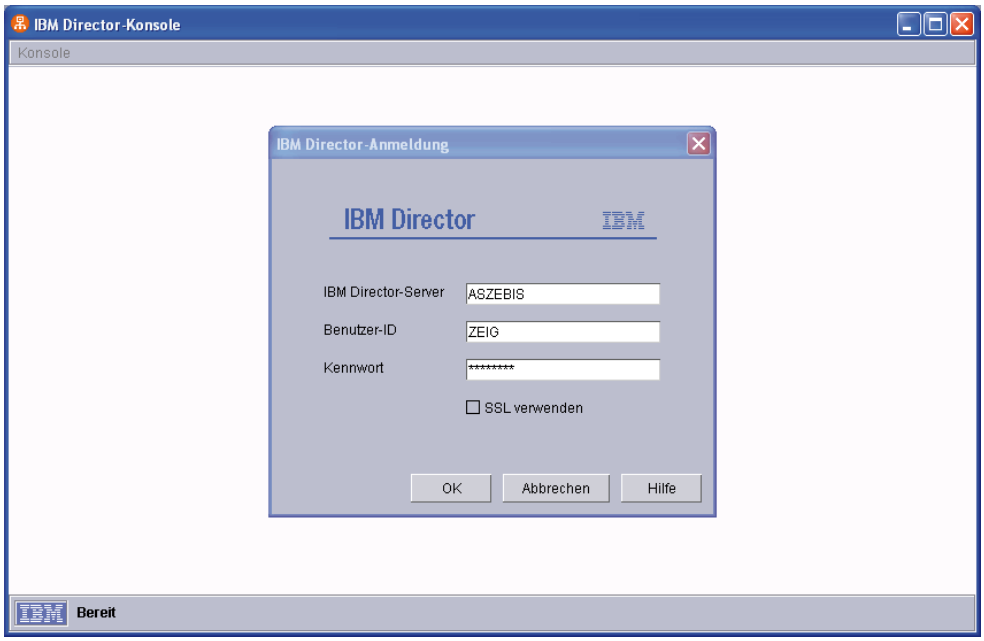
Abschluss der Installation

Damit ist die Installation abgeschlossen. Starten Sie in Abhängigkeit des PCs Windows nun neu.

Berechtigungssteuerung für die Verwendung mit IBM Director Server auf System i

Bevor wir mit den IBM Director Funktionen arbeiten können, müssen wir zunächst eine Verbindung zwischen der IBM Director Konsole und dem IBM Director Server herstellen. Die Kommunikation wird auch hier – wie in allen anderen IBM Director Teilbereichen auch – über TCP/IP durchgeführt. Für den Verbindungsaufbau zwischen IBM Director Server und Konsole benötigen wir neben den IP Vorgaben oder dem Hostnamen auch einen Benutzer mit entsprechenden Berechtigungen.

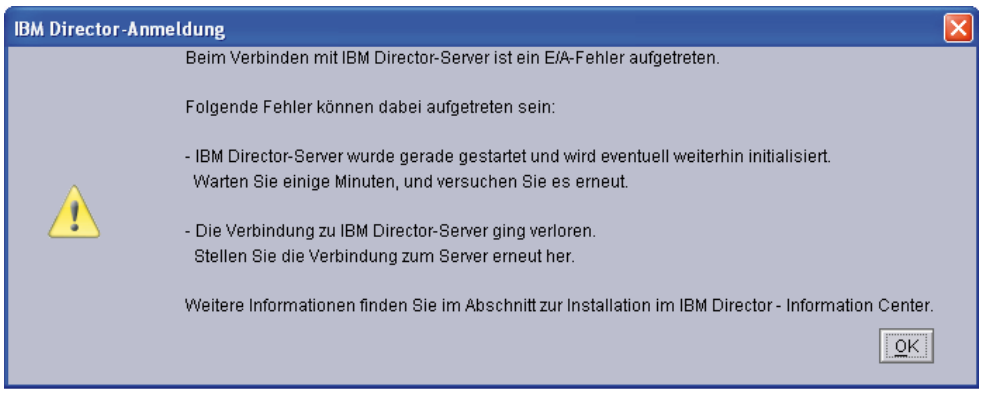
1. Starten Sie nun die IBM Director Konsole: „Start/Alle Programme/IBM Director Console" oder das zugehörige Symbol auf dem Desktop.
2. Als erstes werden Sie nun aufgefordert, eine Anmeldung mit einem gültigen Benutzerprofil durchzuführen. Geben Sie in der folgenden Anzeige Ihren System i Benutzer und das zugehörige Kennwort ein.



Anmeldung an der IBM Director Konsole

4.3.3.3
Seite 2

3. Bestätigen Sie Ihre Eingaben mit einem Klick auf die Schaltfläche „OK".



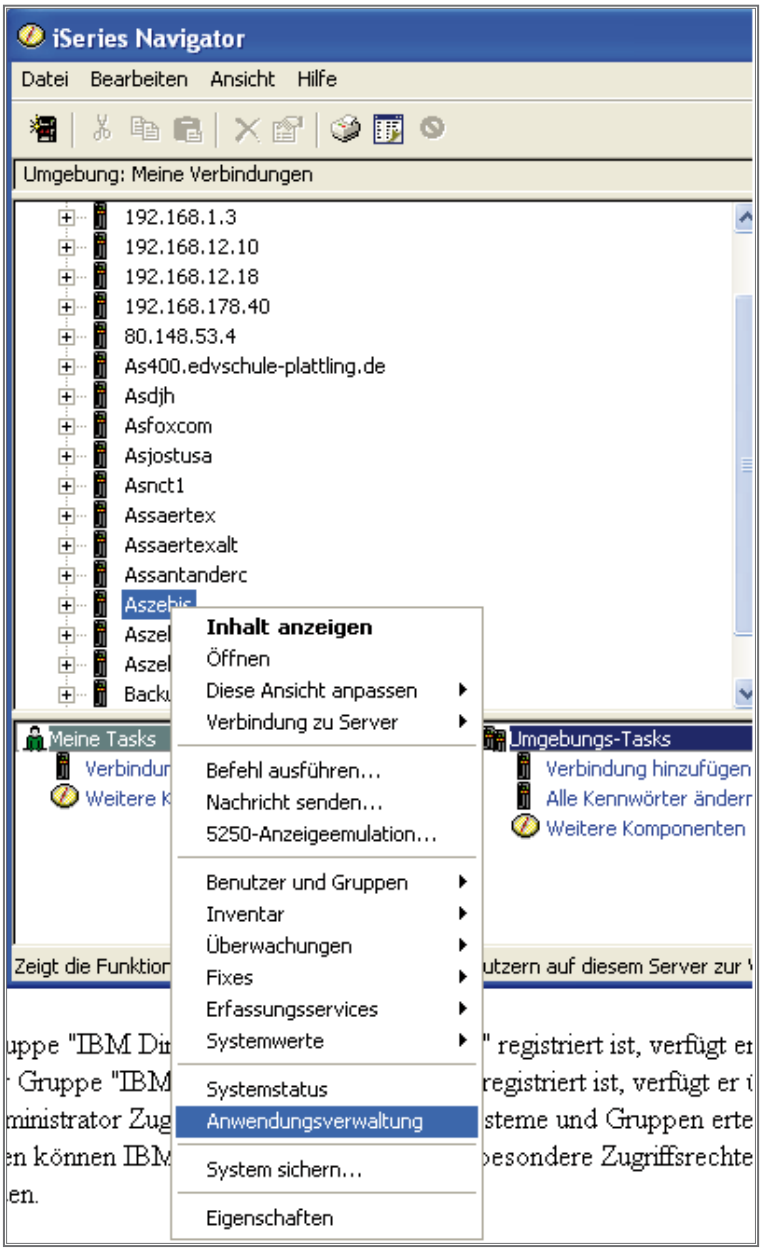
Fehlermeldung

Achtung!

Wenn Sie diese Fehlermeldung erhalten, dann kann dies unterschiedliche Ursachen haben. Zum einen kann es sein, dass die Verbindung zu dem zentralen IBM Director Server nicht hergestellt werden konnte (prüfen Sie in diesem Fall, ob der TCP/IP Serverdienst für den IBM Director Server auf dem System i gestartet ist).

Der Serverdienst lässt sich mittels des Befehls STRTCPSVR *DIRECTOR starten. Der Fehler kann aber auch darin begründet liegen, dass der Benutzer nicht für den Bereich IBM Director Server Administration registriert worden ist.

4. Klicken Sie unbeirrt auf die Schaltfläche „OK". Damit wird der Anmeldeprozess zunächst abgebrochen.
5. Wenn Sie die vorhergehende Meldung erhalten haben, ist eine Überprüfung der Einstellungen erforderlich. Wir gehen dabei davon aus, dass sowohl der IBM Director Server als auch die Verbindung zu diesem System eingerichtet sind. Beschäftigen wir uns deshalb mit den Berechtigungseinstellungen.
6. Öffnen Sie den iSeries Navigator.
7. Für die grundsätzlichen Berechtigungseinstellungen verwenden wir den iSeries Navigator. Dort wählen wir das zu verwaltende System i – bzw. die Maschine aus, auf welcher der IBM Director Server installiert wurde. Mit der rechten Maustaste gelangen wir in ein Auswahlfenster, das unter anderem den Eintrag „Anwendungsverwaltung" beinhaltet. Wählen Sie den Eintrag „Anwendungsverwaltung" nun aus.



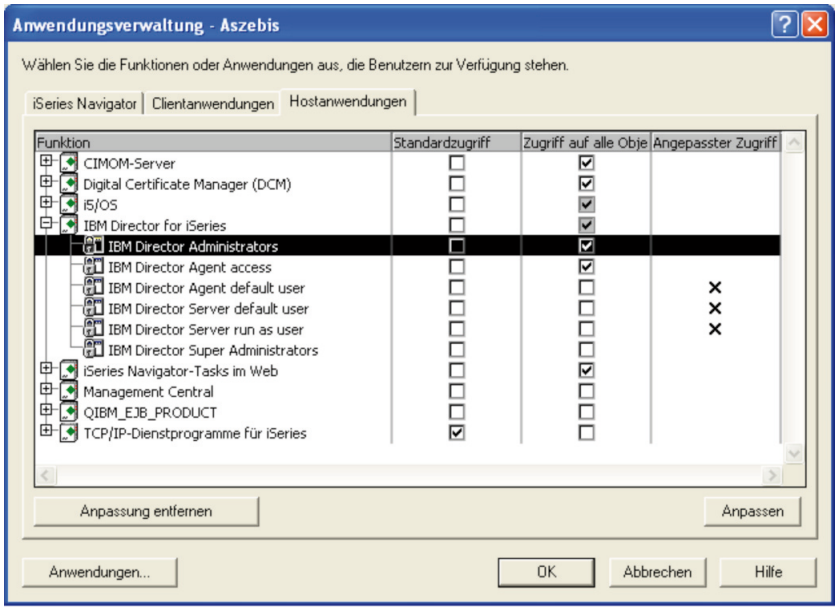
Anwendungsverwaltung

Anmerkung

Sollten Sie nicht über den iSeries Navigator verfügen, dann können Sie die Berechtigungsvergabe auch mittels des i5/OS Befehls “CHG-FCNUSG“ durchführen.

8. In dem sich nun öffnenden Fenster wählen wir den Bereich Hostanwendungen aus.

9. Darin befindet sich unter anderem auch der Eintrag „IBM Director for iSeries“, den wir mit einem Klick auf das Erweiterungszeichen in Form des „+“ vollständig anzeigen lassen.

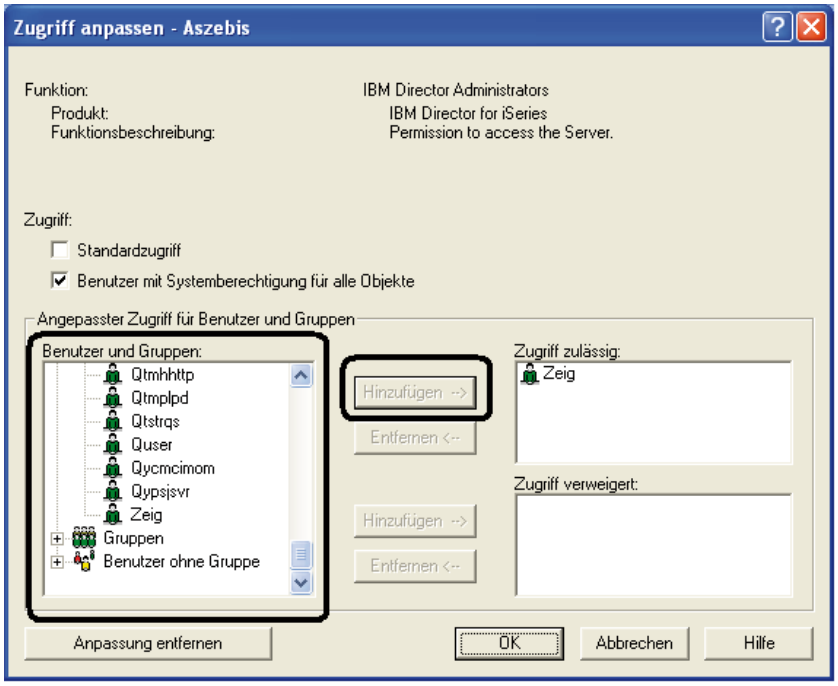


Anwendungen berechtigen

Die Untereinträge des IBM Directors for iSeries definieren die verschiedenen Standardbenutzergruppen, die mit dem IBM Director ausgeliefert wurden. Um die Einträge in diesen Gruppen zu verwalten, klicken Sie auf die gewünschte Gruppe.

Beachten Sie bitte, dass Sie für die Berechtigungsvergabe über ausreichende Berechtigungen verfügen müssen. Deshalb muss für die Anpassung der Berechtigungen des IBM Director ein Benutzerprofil verwendet werden, welches über *SECADM Berechtigungen verfügt.

10. Klicken Sie nun mit der rechten Maustaste auf den Eintrag „IBM Director Super Administrators“.
11. Wählen Sie die Option „Anpassen“
12. Erweitern Sie den Eintrag “Alle Benutzer”. Suchen Sie darin nach dem Benutzerprofil, welches für den Einsatz mit dem IBM Director genutzt werden soll.



Benutzerberechtigungen

13. Nachdem Sie den oder die Benutzer ausgewählt hatten, müssen Sie sicherstellen, dass diese mit einem Klick auf die Schaltfläche „Hinzufügen" in den rechten Übersichtsbereich übernommen wurden.
14. Bestätigen Sie Ihre Auswahl mit einem Klick auf die Schaltfläche „OK".
15. Schließen Sie den iSeries Navigator nun wieder.

Achtung!

Alternative Methode der Berechtigungsvergabe ohne iSeries Navigator:

CHGFCNUSG FCNID(QIBM_QDIR_SUPER_ADM_PRIVILEGES) USER(<Benutzername>) USAGE(*ALLOWED)

16. Nachdem wir nun die benötigten Berechtigungen vergeben hatten, wiederholen wir die Anmeldeprozedur nochmals und melden uns an der IBM Director Konsole an.

4.3.3.3
Seite 6

Einführung in die IBM Director Konsole

Nachdem wir die IBM Director Konsole installiert hatten und die Berechtigungen definiert wurden, möchte ich Ihnen an dieser Stelle einen ersten Einblick in die Oberfläche und die grundsätzliche Arbeitsweise dieses Tools geben.

Die IBM Director Konsole wird als Programmgruppe unter Windows installiert. Wenn Sie die Standardeinstellungen während des Installationsprozesses übernommen haben, dann finden Sie die IBM Director Konsole unter „Start/Alle Programme/IBM Director Konsole".

1. Starten Sie nun die IBM Director Console.
2. Melden Sie sich mit dem Benutzer und dem Kennwort an.

Anmeldung an der IBM Director Konsole

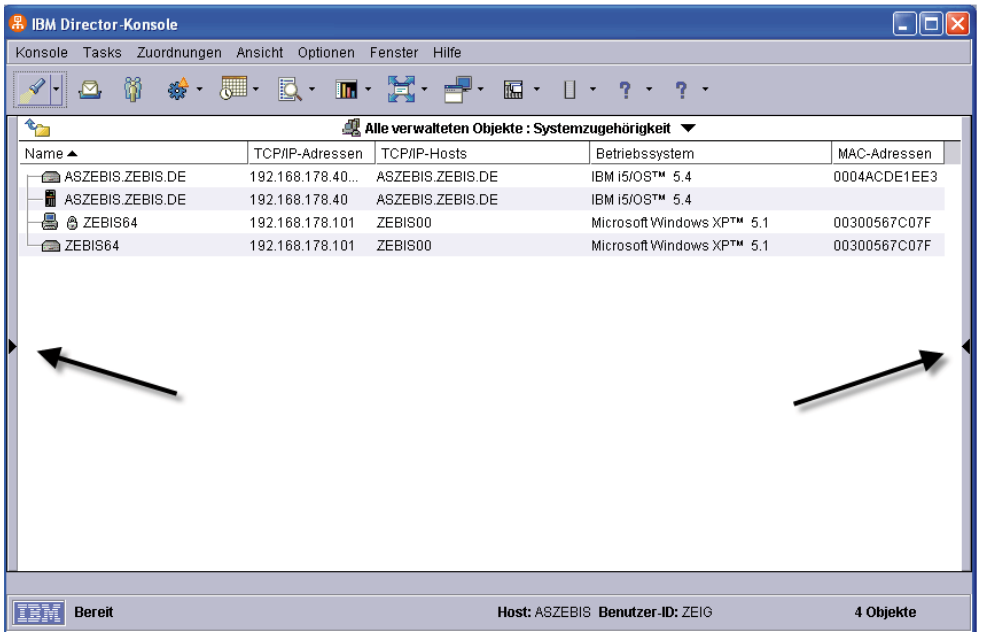
Nach dem Starten und Anmelden an der IBM Director Konsole kann, je nach Einstellungen, zunächst eine Willkommensseite dargestellt werden.

4.3.3.4
Seite 2

Willkommensseite

3. Sollte die Willkommensseite angezeigt werden, dann schließen Sie diese bitte.

   Nach dem Start der IBM Director Konsole ist die Darstellung auf einen Fensterbereich fokussiert. Ein Beispiel dazu finden Sie in der folgenden Abbildung:



Nicht angepasster Startbildschirm

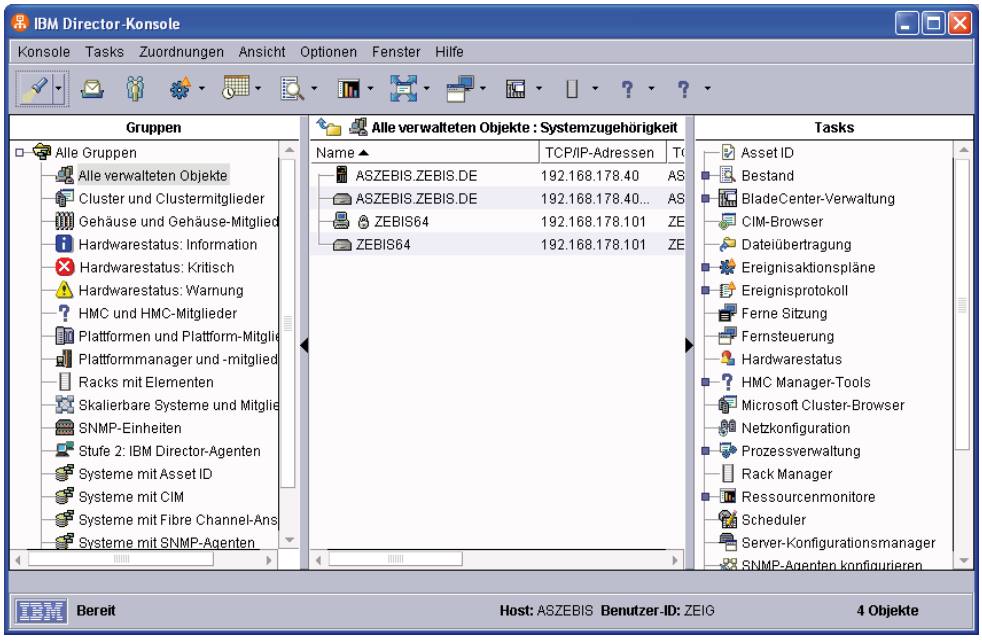
Tipp

Beachten Sie die beiden, an der rechten und linken Seite der Anzeige befindlichen, schwarzen Pfeile. Diese erlauben es uns – zugegeben ein wenig eigenwillig – die Anzeige in verschiedene Fensterbereiche aufzuteilen. Wenn Sie auf diese beiden Pfeile klicken, ändert sich die Anzeige der Konsole dahingehend, dass der Hauptanzeigenbereich zu Gunsten zweier weiterer Fenster zusammenschrumpft.

4. Erweitern Sie nun die Anzeige mit einem Klick auf den linken und rechten Pfeil in der Anzeige.

   Das Ergebnis ist die klassische Darstellung der IBM Director Konsole. Diese Darstellung ist in folgende drei Hauptbereiche gegliedert:

   - Gruppen – dabei handelt es sich um bereits von IBM vordefinierte Gruppen
   - Alle verwalteten Objekte (oder eine ausgewählte Detailanzeige)
   - Tasks



Vollanzeige

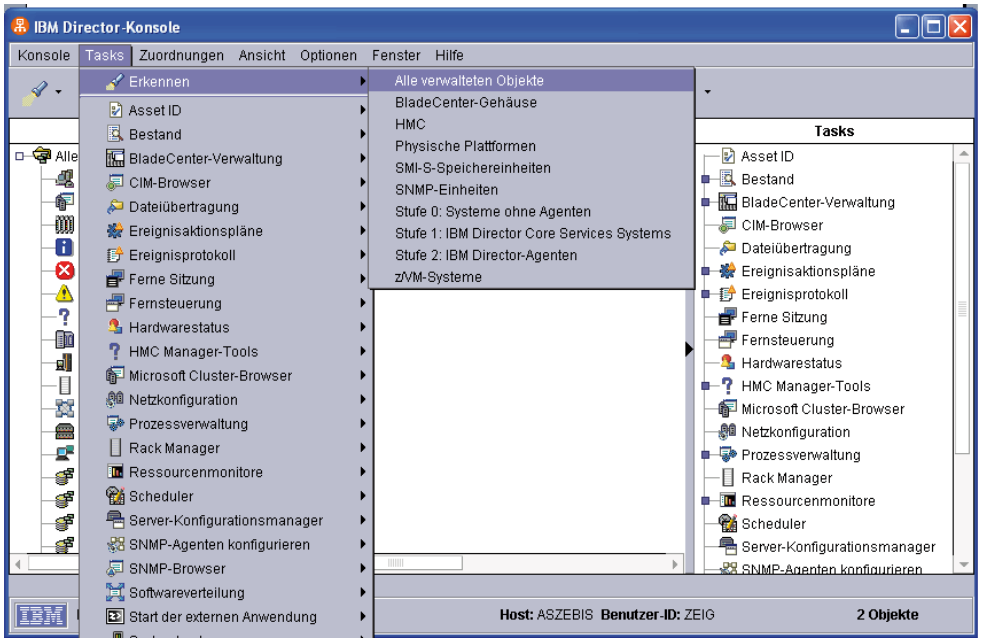
5. Klicken Sie nun in dem Bereich „Gruppen" auf den Eintrag „System mit i5/OS". Damit erhalten wir in dem mittleren Bereich eine Auflistung aller in der IBM Director Umgebung enthaltenen System i Maschinen.
6. Schauen wir uns noch einen weiteren Bereich an – klicken Sie nun auf den Eintrag „Hardwarestatus kritisch". Dieser befindet sich ebenfalls in dem linken Navigationsbereich.

7. Beachten Sie nun den Inhalt des mittleren Anzeigenbereichs. Dort wird das Ergebnis der kritischen Hardwaresituationen angezeigt. Sollten Probleme vorhanden sein, dann finden wir diese direkt auf einen Blick.

   Wenn die Anzeige der kritischen Hardwaresituationen leer ist, dann liegt das entweder daran, dass es keine solchen Zustände gibt, oder die Erfassung dieser Informationen nicht durchgeführt wurde. Der IBM Director basiert auf bestimmten erfassten Daten, welche automatisch durchgeführt werden. Diese Erfassung können wir aber auch bei Bedarf manuell initiieren.

8. Wählen Sie dazu den Menüeintrag „Tasks/Erkennen/Alle verwalteten Objekte“.

   Diese Aktion wird zur Verarbeitung an den IBM Director Server übergeben. Hier heißt es nun ein wenig warten.



Tasks

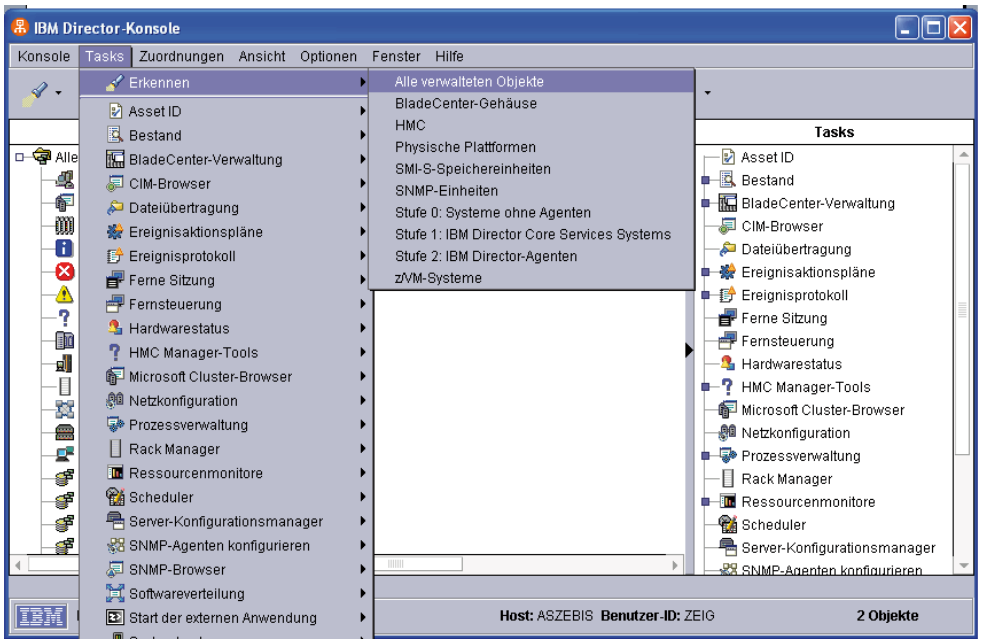
Schauen wir uns nun den rechten Bereich der Anzeige an – den Abschnitt „Tasks“.

Die darin enthaltenen Einträge können auf unterschiedliche Weisen aktiviert werden – entweder mittels eines Doppelklicks, mit einem Klick der rechten Maustaste oder durch das „Ziehen“ auf einen System- oder Gruppeneintrag im linken Fensterbereich.

Die Bereiche, die in der Darstellung erweiterbar sind, verfügen über ein blaues Symbol in Form eines Vierecks.

9. Erweitern Sie in dem rechten Anzeigenbereich „Tasks“ den Eintrag „Bestand“.
10. Klicken Sie mit der rechten Maustaste auf den Eintrag „Angepasste Erfassung“.
11. Wählen Sie die Option „Erstellen“.

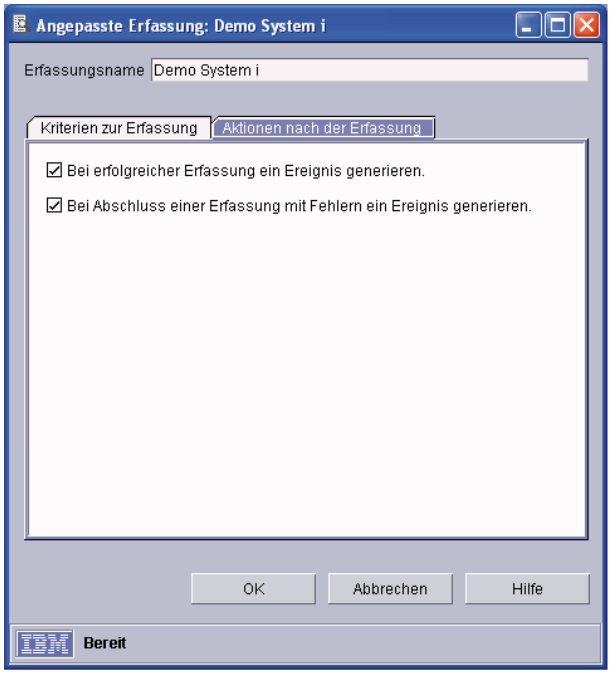
    Damit gelangen wir in den Definitionsbereich für das Erfassen von Beständen. Diese lassen sich differenziert nach Software und Hardware erfassen. Beide Hauptbereiche sind in unterschiedliche Komponenten aufgeteilt, welche wir für individuelle Erfassungen und spätere Überwachungen definieren können.



Angepasste Erfassung

12. Erweitern Sie in dem linken Navigationsbereich den Eintrag „Hardware/Hauptspeicher/Installierter Speicher“ und fügen Sie diesen dem rechten Auswahlbereich mit einem Klick auf die Schaltfläche „Hinzufügen“ zu.
13. Wiederholen Sie dieses Verfahren auch für den Eintrag „Speicher“. Achten Sie dabei darauf, dass Sie nicht den Untereintrag, sondern die gesamte Gruppe „Speicher“ selektieren! Das Ergebnis sollte dann so aussehen, wie es in der vorhergehenden Abbildung zu sehen ist.
14. Tragen Sie in dem im oberen Bereich der Anzeige befindlichen Feld „Erfassungsname“ als Namen für diese individuelle Erfassung „Team xx Erfassung 1“ ein. Ersetzen Sie den Platzhalter „xx“ mit der Teamnummer!

15. Wechseln Sie nun in den Tabellenbereich „Aktionen nach der Erfassung“.

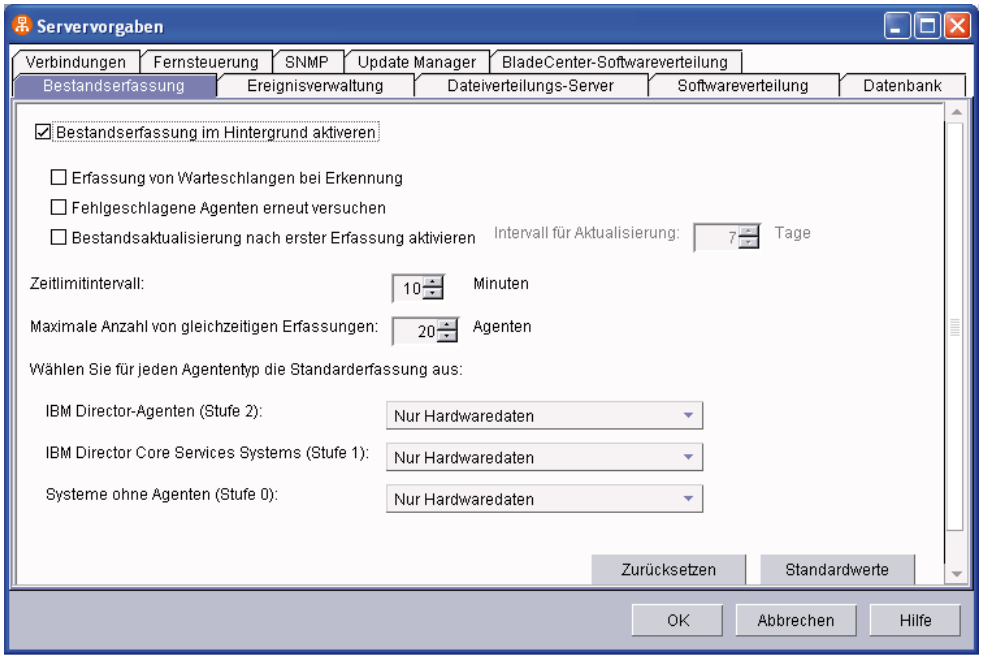


Aktionen der Erfassung

16. Wählen Sie die beiden zur Verfügung stehenden Optionen aus.
17. Bestätigen Sie Ihre Angaben mit einem Klick auf „OK“.
18. Schauen wir uns nun das Ergebnis an – lassen wir dazu dem System aber einen kurzen Augenblick Zeit.
19. Erweitern Sie in dem rechten Navigationsbereich den Eintrag „Bestand/ Angepasste Erfassungen“. Darunter befindet sich der von Ihnen definierte Erfassungsbereich.

    Wir werden uns später noch mit der definierten Erfassung beschäftigen.
20. Schließen Sie nun die Anzeige der Ergebnisprotokolleinträge wieder.

    Die globalen Einstellungen für die IBM Director Konsole können in einem gewissen Rahmen individuell angepasst werden. Die dazu möglichen Einstellungen lassen sich in dem Menübereich „Optionen“ vornehmen. Unter anderem sind in diesem Bereich auch die Vorgaben für die Bestandserfassung, Softwareverteilung, SNMP etc. möglich.



Servervorgaben

4.3.3.4
Seite 8

Ressource Überwachung

Die Vielfältigkeit der IT ist in den meisten Unternehmen mittlerweile an ihre Grenzen gelangt, wodurch die effektive Ausführung der Administration und Überwachung immer mehr erschwert wird.

Der IBM Director kann in Verbindung mit dem System i für eine Vielzahl administrativer Aufgaben eingesetzt werden. Dieser Einsatz ist nicht nur auf i5/OS beschränkt, sondern lässt sich auf alle unmittelbar und mittelbar auf dem System ausführbaren Betriebssysteme (wie z. B. Windows, AIX, Linux) erweitern.

Die Funktionen beinhalten unter anderem Überwachungswerkzeuge, Definitionen von Schwellenwerten, grafische Auswertungen und Benachrichtigungsfunktionen.

Inhalte dieses Teilbereichs

In diesem Abschnitt erhalten Sie folgende Informationen:

1. Erstellen von Ressource Überwachungen
2. Auswerten der überwachten Ressourcen
3. Einrichten von Schwellenwerten

4.3.3.4.1

Seite 2

Erstellen von Ressource-Überwachungen

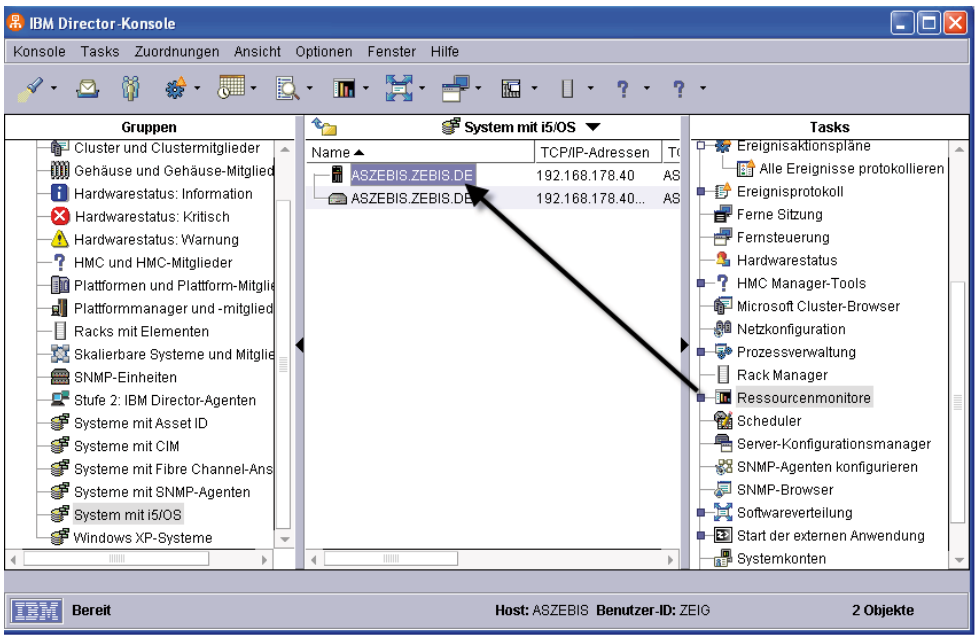
In diesem Abschnitt lernen Sie, wie ein Ressource-Monitor generiert werden kann. Solche Überwachungen lassen sich für unterschiedliche Bereiche definieren. Das in diesem Beispiel verwendete Szenario ist nur exemplarisch und kann beliebig angepasst werden!

Wir wollen mit dieser Übung ein Subsystem mitsamt den darin enthaltenen Jobs überwachen. Eine solche Überwachung kann beispielsweise das Ziel haben, den Start von geplanten Jobs zu überwachen oder beispielsweise auf eine Fehlersituation eines Jobs aufmerksam zu werden.

Mit den in diesem Kapitel beschriebenen Übungen überwachen wir unter anderem eine JOBQ. Dies ist lediglich ein Beispiel, wie mit diesen Komponenten des IBM Directors in Bezug auf den Einsatz mit dem IBM i gearbeitet werden kann. Ressourceüberwachungen stehen darüber hinaus natürlich für die unterschiedlichsten IBM i Teilbereiche zur Verfügung.

Die Jobwarteschlange wollen wir nun in Bezug auf deren Status und den darin enthaltenen Jobs überwachen.

1. Wählen Sie in dem rechten Anzeigenbereich den Eintrag „Ressourcemonitore" und ziehen Sie diesen in den mittleren Bereich auf den Eintrag des System i.

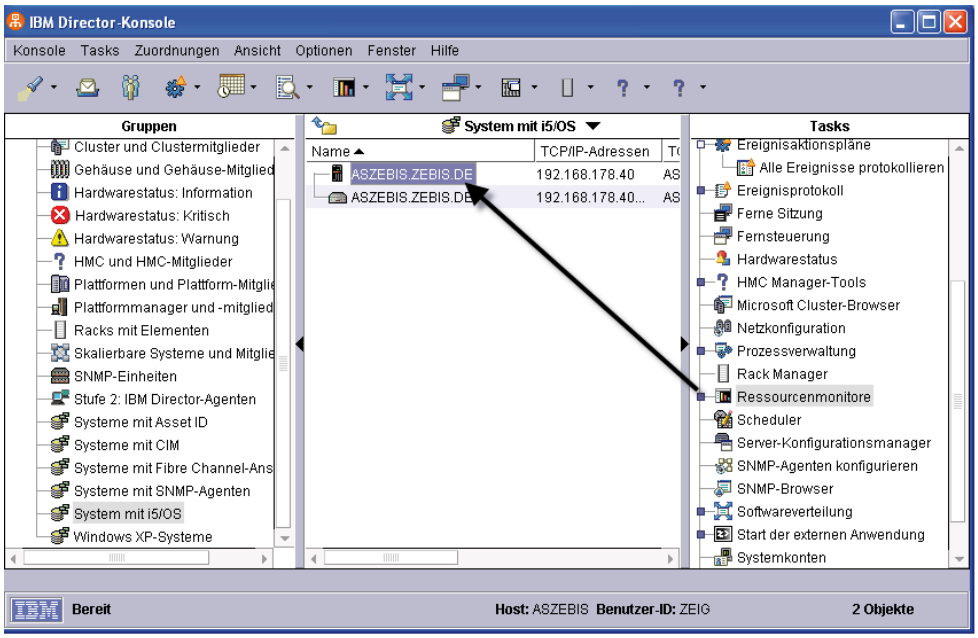


Ressourceüberwachung

Damit wird der Bereich des Ressourcemonitors für das ausgewählte System gestartet. Die Anzeige wird in einem separaten Fenster dargestellt.

4.3.3.4.2
Seite 2

2. Erweitern Sie in dem Fenster „Ressourcemonitor“ in dem rechten Abschnitt den Eintrag „Director-Agent“.
3. Erweitern Sie den Eintrag „i5OS Systemmonitore”
4. Erweitern Sie den Eintrag „Jobwarteschlangen”.
5. Erweitern Sie den Eintrag „TEAMxx/IDJOBQ“ (dabei handelt es sich um die Bibliothek und die JOBQ) und ziehen Sie die Einträge „Jobs in Warteschlange“ und „Jobwarteschlangenstatus“ in den rechten Bereich „Ausgewählte Ressourcen“.



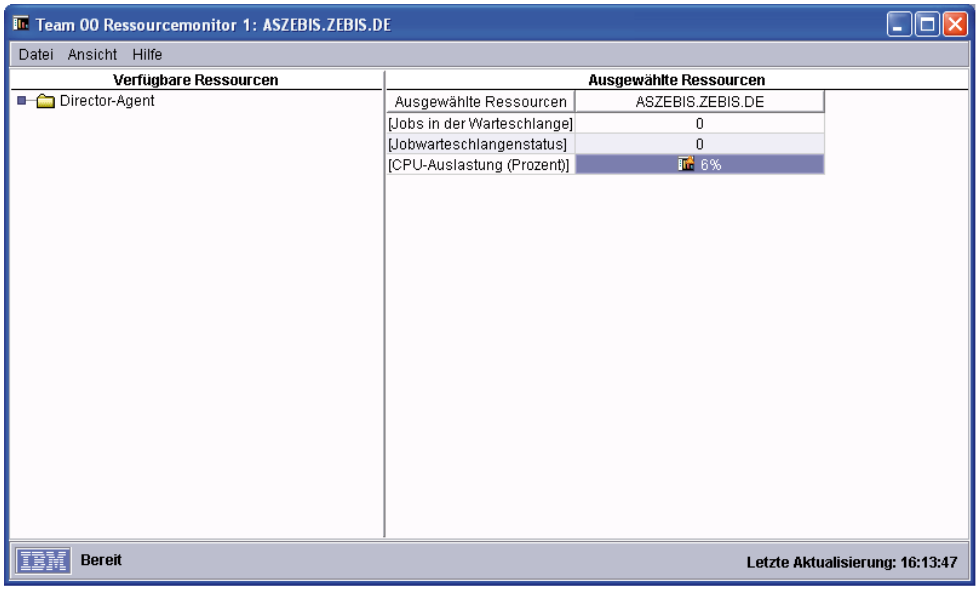
Jobwarteschlangenstatus auswählen

6. Zusätzlich wollen wir noch die Auslastung unserer CPU überwachen.

   Erweitern Sie den Eintrag „Systemstatistik“ in dem linken Anzeigenbereich und wählen Sie den dort enthaltenen Eintrag „CPU Auslastung“ mit einem Doppelklick aus, um ihn in den Bereich „Ausgewählte Ressourcen“ zu übertragen.

   Damit haben wir eine Anzahl unterschiedlicher Überwachungen definiert. Wir können die aktuellen Werte der ausgewählten Ressourcen auch direkt einsehen.

7. Klicken Sie auf den Menüeintrag „Ansicht/Aktualisieren“.



CPU Auslastung

Das Ergebnis ist die aktuelle Darstellung der ausgewählten Ressourcen – am Beispiel der CPU Auslastung sehen wir, dass diese auf dem System bei 6 % liegt.

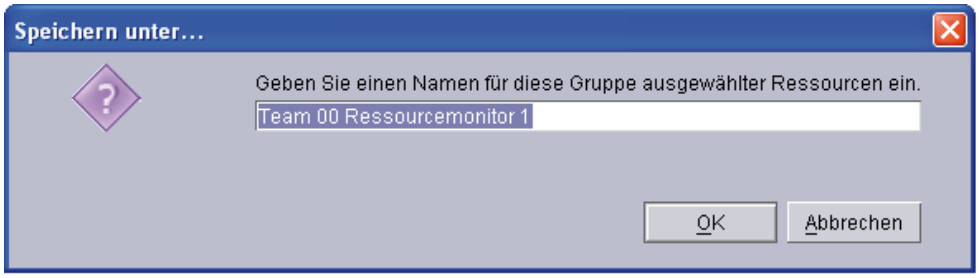
Der Inhalt der Anzeige wird per Defaulteinstellung alle 10 Sekunden aktualisiert. Im unteren Bereich der Anzeige finden Sie den jeweiligen Stand in Form der Angabe „Letzte Aktualisierung“.

Tipp

Wie auch viele andere Voreinstellungen verwaltet werden können, so lässt sich unter anderem auch die Periode der Aktualisierung der Messwerte über Benutzervorgaben anpassen. Details dazu finden Sie in einem weiteren Kapitel.

Damit wir die Einstellungen später weiter verwenden können, müssen wir diese speichern.

8. Wählen Sie den Menüeintrag „Datei / Speichern unter“.
9. Speichern Sie diese Definition unter „Team xx Ressourcemonitor 1“.

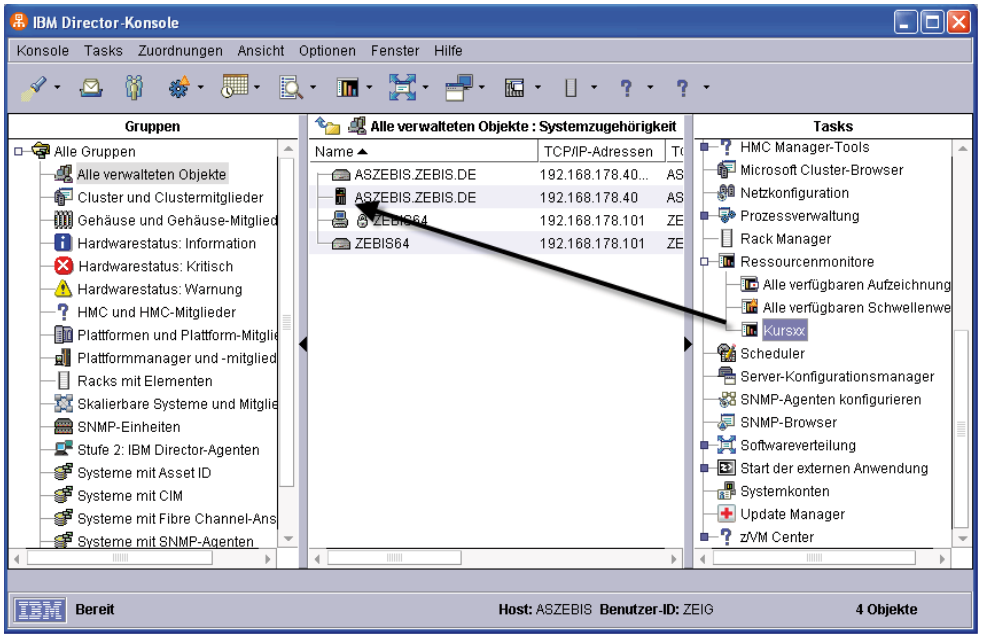


Speichern der Einstellungen

4.3.3.4.2
Seite 4

10. Schließen Sie nun das Fenster der Ressourceüberwachung. Die gespeicherte Definition finden wir unter anderem in dem rechten Anzeigenbereich „Tasks“ unterhalb des Abschnitts „Ressourcenmonitor“. Von dort aus können wir jederzeit auf die Definition und deren Ergebnisse zugreifen.

    Wenn Sie eine solche Definition starten und einsehen wollen, dann markieren Sie diese in der Übersicht und ziehen sie beispielsweise in dem mittleren Bereich auf das gewünschte System.



Überwachung starten

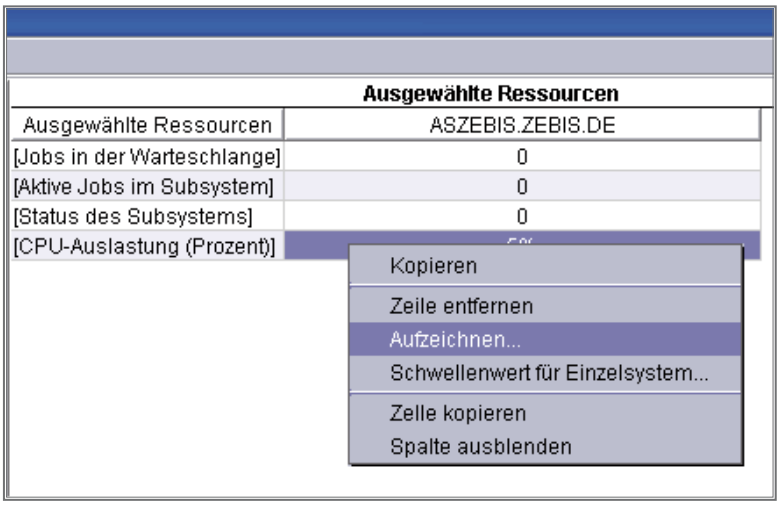
Damit haben wir die Basis für eine Überwachung geschaffen. Im weiteren Verlauf ist es für uns jetzt von Bedeutung, dass wir Schwellenwerte definieren und beispielsweise auch grafische Auswertungen der erfassten Daten erstellen können.

Einrichten von Schwellenwerten und Grafiken

Eine Eigenart des IBM Directors liegt darin, dass er keine Standardauswertungen in grafischer Form liefert. Auf Basis der gesammelten Daten lassen sich solche allerdings auch manuell erstellen.

Nachfolgend finden Sie eine kurze Beschreibung über das Einrichten von Schwellenwerten und der Verwendung von Grafiken.

1. Wählen Sie in der Anzeige der überwachten Ressourcen in dem rechten Bereich „Ausgewählte Ressourcen“ den Eintrag „CPU Auslastung“ mit der rechten Maustaste aus. Achten Sie darauf, dass der Cursor auf dem Feld mit der Prozentanzeige positioniert ist!

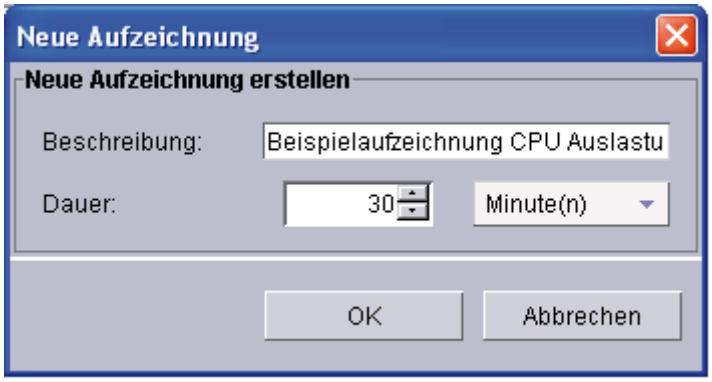


Funktion Aufzeichnen

2. Selektieren Sie den Eintrag „Aufzeichnen“.
   Damit gelangen wir in eine neue Anzeige „Aufzeichnung des Ressourcenmonitors…“.
3. Wählen Sie die Menüoption „Datei/Neu“.
   Damit gelangen wir in den Definitionsbereich für die Aufzeichnungen.
4. Geben Sie in dem Feld „Beschreibung” einen beschreibenden Text ein – die Aufzeichnung kann später in verschiedenen Ansichten ausgewählt werden.

4.3.3.4.3
Seite 2

5. Ändern Sie die Dauer der Aufzeichnung auf 30 Minuten und bestätigen Sie Ihre Auswahl mit einem Klick auf die Schaltfläche „OK”.



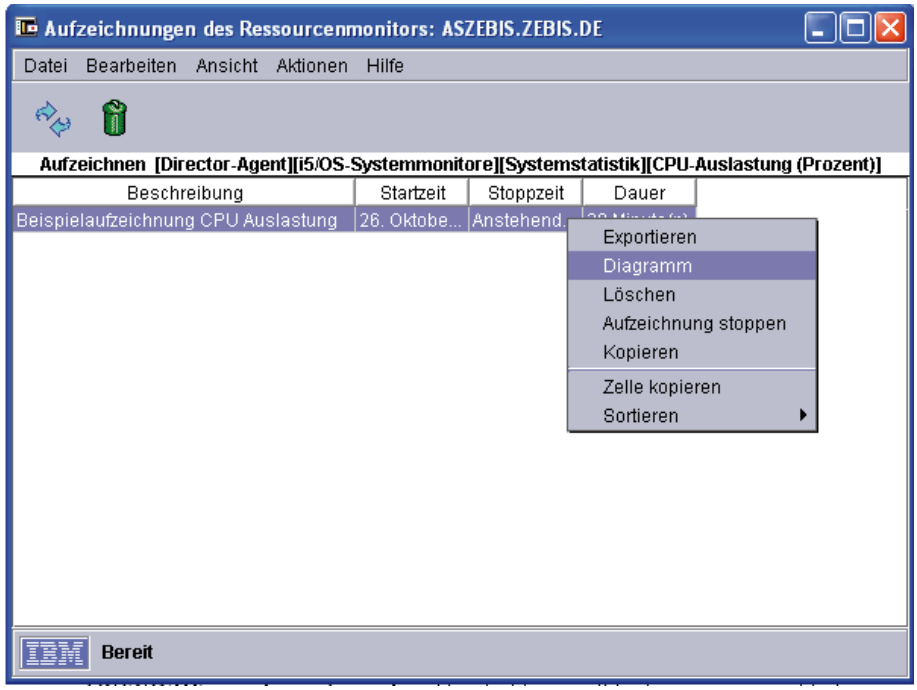
Neue Aufzeichnung erstellen

Eine weitere Eigenart des IBM Directors liegt darin, dass Sie lediglich eine Aufzeichnung starten können. Sollten Sie versuchen, mehrere solcher Aufzeichnungen parallel zu starten, erhalten Sie eine entsprechende Hinweismeldung.

In der Übersicht der ausgewählten Ressourcen finden Sie außerdem ein Symbol für die aktive Aufzeichnung.

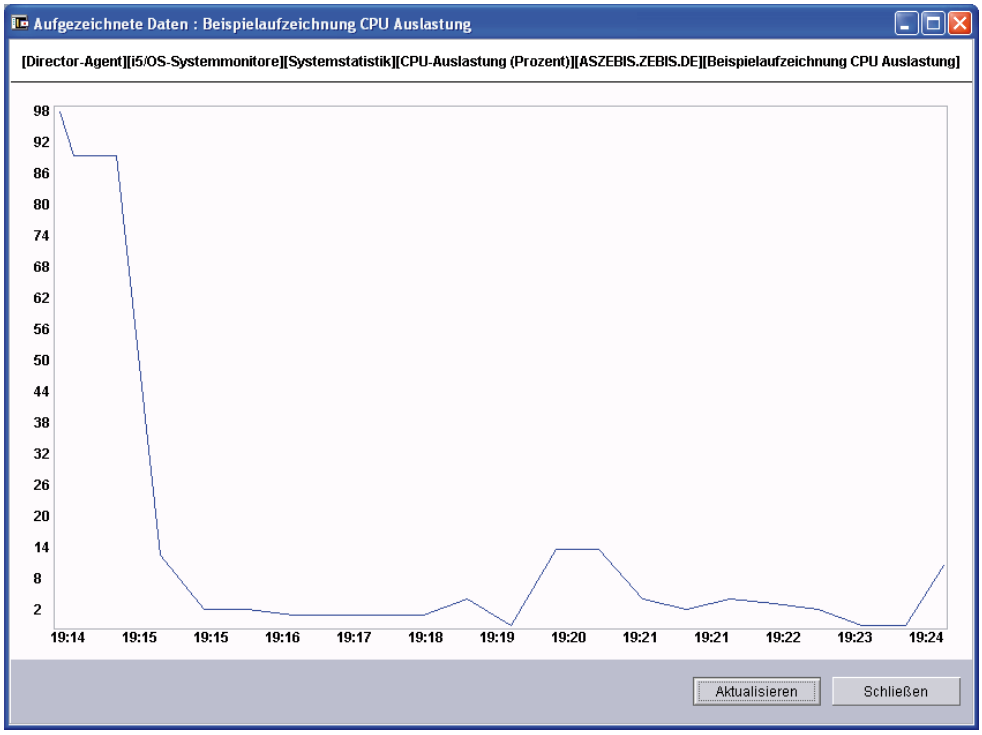
6. Klicken Sie nun in der Abbildung der Aufzeichnung mit der rechten Maustaste auf den Aufzeichnungseintrag. Damit erhalten Sie eine Übersicht der möglichen Optionen.

7. Selektieren Sie den Eintrag „Diagramm”.



Diagramm

8. Damit erhalten wir eine grafische Darstellung der Datensammlung:

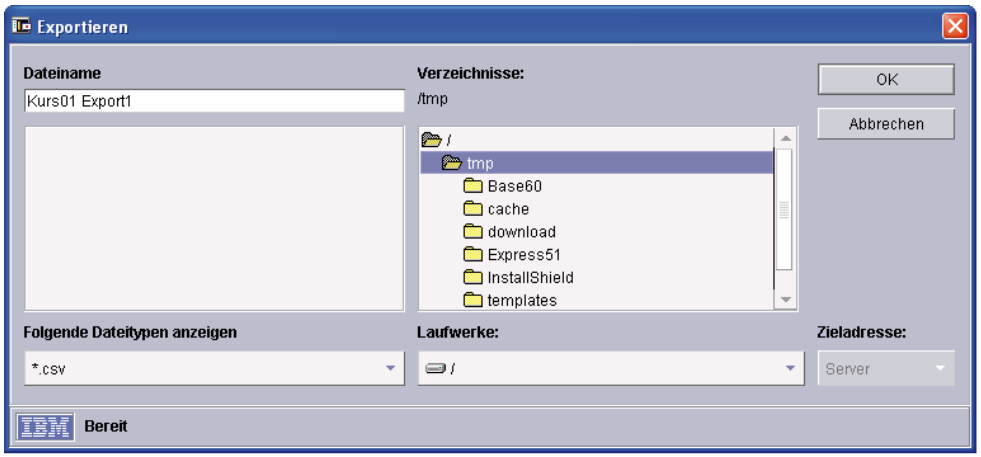


Ergebnis der Aufzeichnung als Diagramm

9. Diese kann bei Bedarf mit einem Klick auf die Schaltfläche „Aktualisieren" auf den neuesten Stand gebracht werden. Klicken Sie nun auf „Schließen".
10. Zurück in dem Fenster mit den Aufzeichnungen wählen wir mit der rechten Maustaste die Option „Exportieren".
11. Wir haben die Möglichkeit, die gesammelten Daten in das Verzeichnissystem auszugeben. Dabei stehen folgende Dateitypen für den Export zur Verfügung:
    - CSV
    - HTM
    - TXT
    - XML

4.3.3.4.3
Seite 4

12. Um die Daten beispielsweise in Excel weiter zu verarbeiten, wählen wir nun exemplarisch das Format CSV und geben der Datei einen entsprechenden Namen. Speichern Sie die Einstellungen nun an einem beliebigen Ort.



Exportieren der gesammelten Daten

13. Bestätigen Sie Ihre Auswahl mit einem Klick auf „OK".
14. Die auf diese Weise erstellten Daten lassen sich nun leicht in Excel & Co verarbeiten.

Damit haben wir die grafische Darstellung und auch den Export der gesammelten Daten behandelt. Diese stellen die Grundlage für weitere Auswertungen dar.

Einen wesentlichen Punkt, der insbesondere in komplexen Installationen von wesentlicher Bedeutung ist, sind die Schwellenwerte, die einen kritischen Zustand präventiv anzeigen sollen. Deren Definition ist Gegenstand des nächsten Abschnitts.

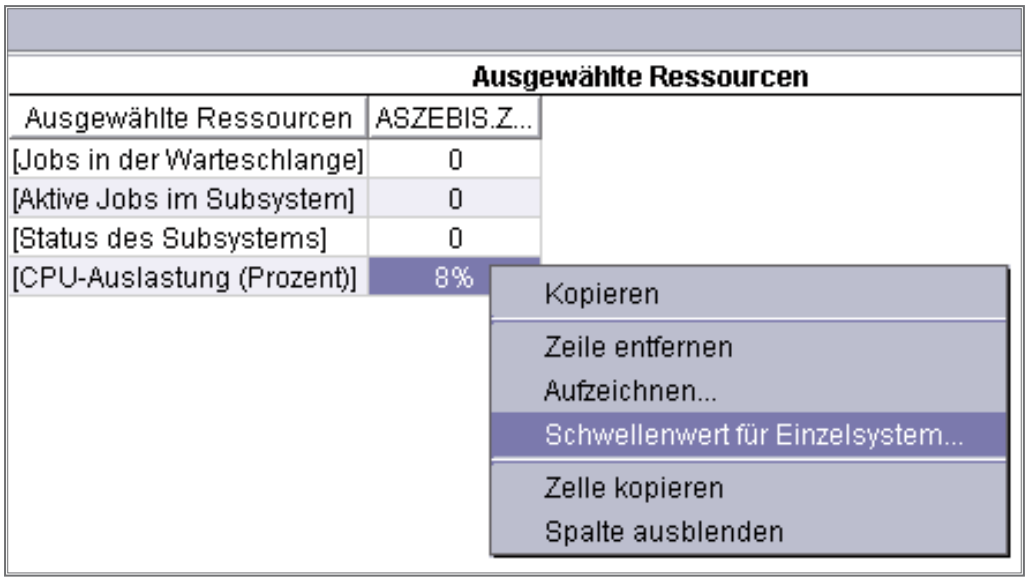
Definition von Schwellenwerten

Die bisherigen Tools des IBM Directors haben lediglich globale Werte ermittelt bzw. dargestellt. Dies ist zwar auch wichtig, jedoch im Rahmen von Überwachungsaufgaben zu allgemein. Für das – nach Möglichkeit frühzeitige – Erkennen von abnormalen Zuständen sind andere Werkzeuge erforderlich, welche abnormale Situationen oder Fehler aufzeichnen und ausgeben können.

Der IBM Director bietet für solche Überwachungen die Möglichkeit der Definition von Schwellenwerten. Diese lassen sich gezielt für separate Teilbereiche definieren und dienen der vorzeitigen Erkennung von Fehlern oder Problemen. Die Eigenart des IBM Directors bei dem Einsatz von Schwellenwerten liegt darin, dass der Wert dieses Schwellenwertes zwar festgelegt werden kann, eine Aktionsdefinition jedoch an dieser Stelle nicht möglich ist. Die Aktionen, welche bei dem Eintreten einer Schwellenwertkondition ausgeführt werden sollen, sind separat zu definieren.

– Klicken Sie in der Anzeige der Ressourcenüberwachung mit der rechten Maustaste auf den Eintrag „CPU Auslastung".

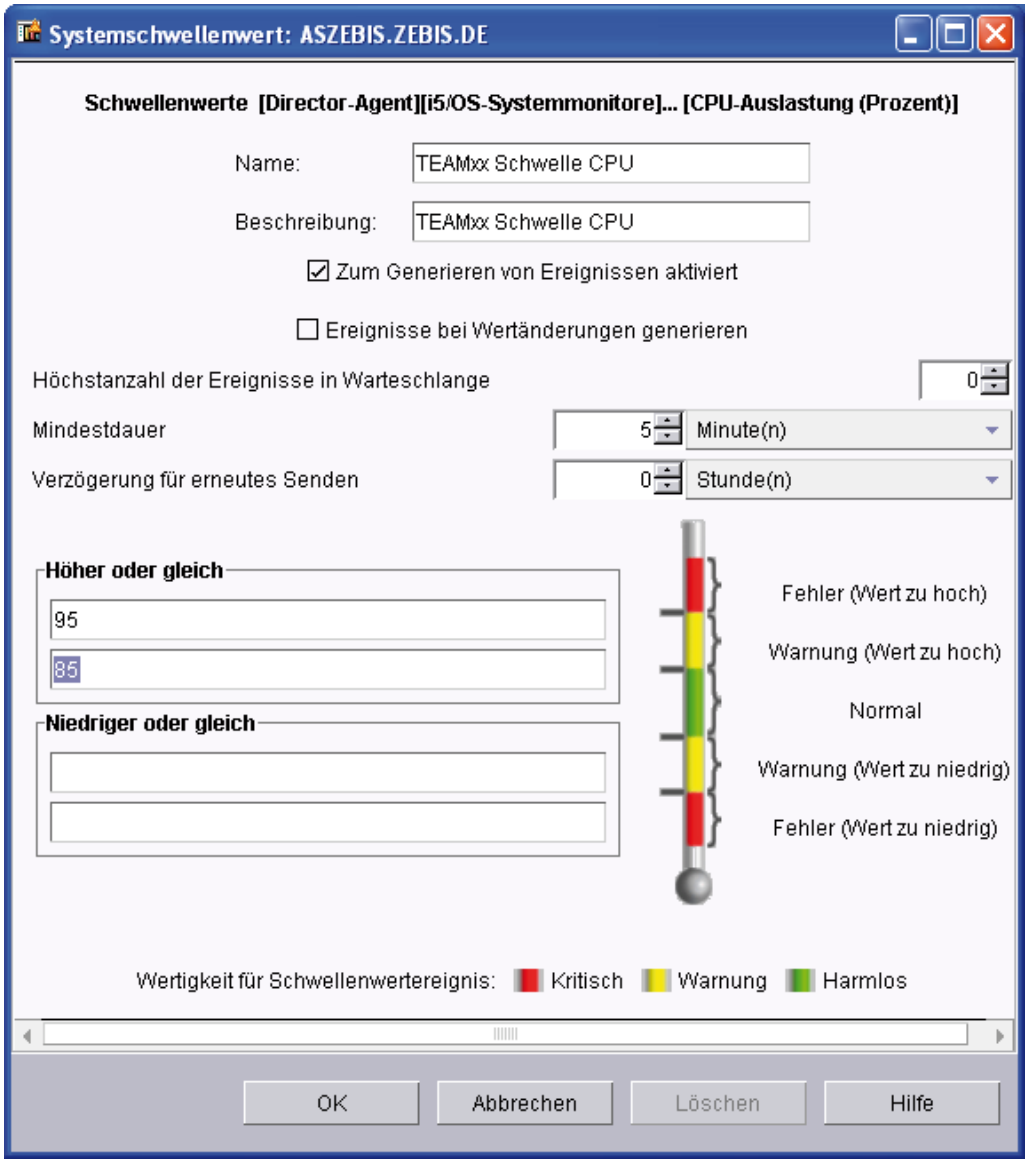
– Wählen Sie die Option „Schwellenwert für Einzelsystem".



Schwellenwerte

– Geben Sie in dem nächsten Fenster einen Namen ein, an dem man die Art des Schwellenwertes ersehen kann. Zum Beispiel: „Schwelle CPU Auslastung".

4.3.4
Seite 2



Details der Schwellenwerte

– Geben Sie als Namen und Beschreibung „Schwelle CPU“ an.

– Stellen Sie sicher, dass die Option „zum Generieren von Ereignissen aktivieren” gesetzt ist.

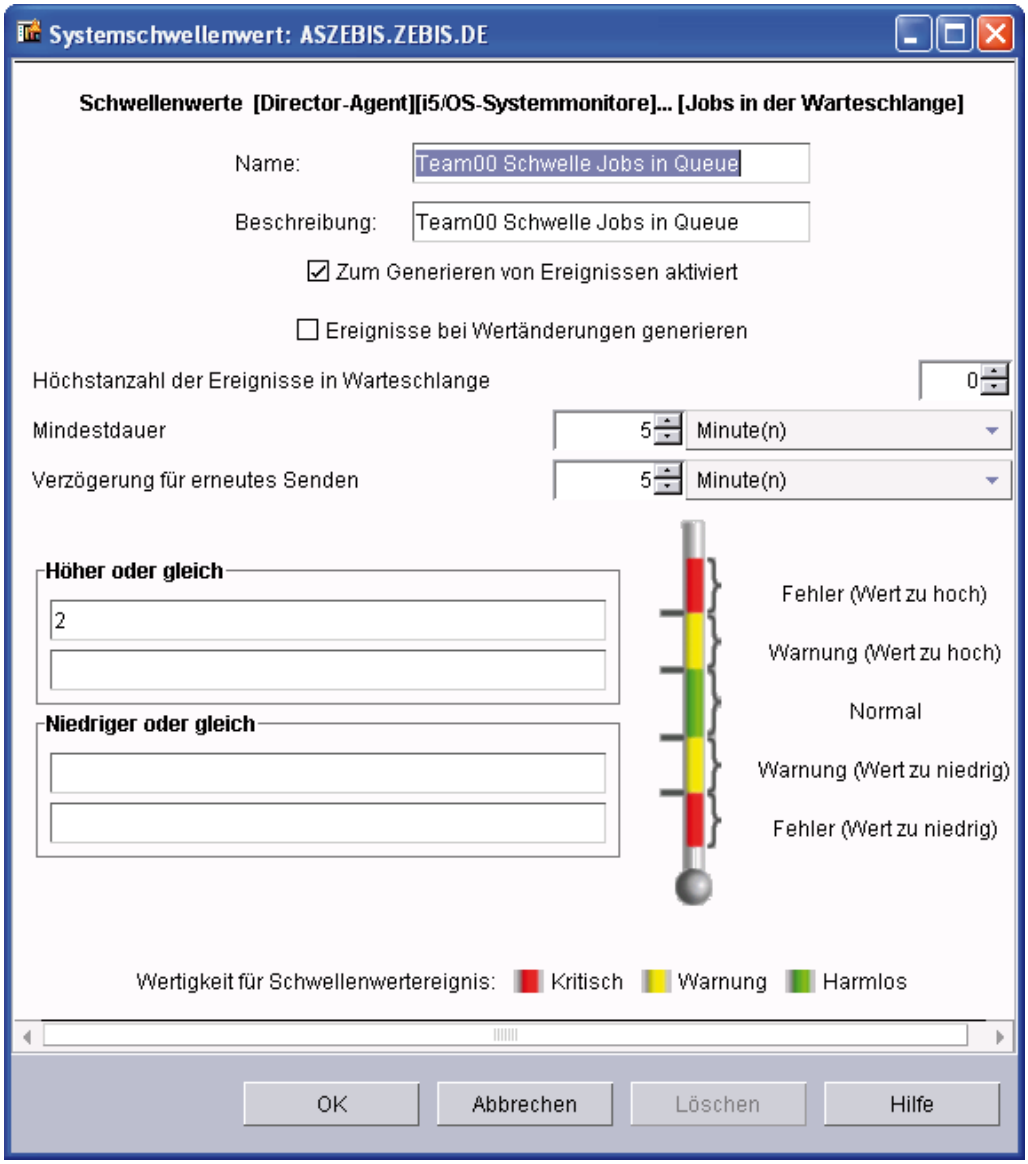
– Ändern Sie die Mindestdauer auf 5 Minuten.

– Geben Sie in dem Bereich „Höher oder gleich” in der ersten Spalte 95 und in der zweiten Spalte 85 ein.

– Übernehmen Sie alle weiteren Einstellungen und klicken Sie auf die Schaltfläche „OK”.

– Wiederholen Sie diese Angaben für den Eintrag „Jobs in Warteschlange“. Geben Sie als Namen und Bezeichnung auch hier einen sprechenden Namen an (z. B. „Schwelle Jobs in Queue“).

– Verwenden Sie als Schwellenwert Fehler „2“.
– Ändern Sie „Verzögerung für erneutes Senden“ auf 5 Minuten.



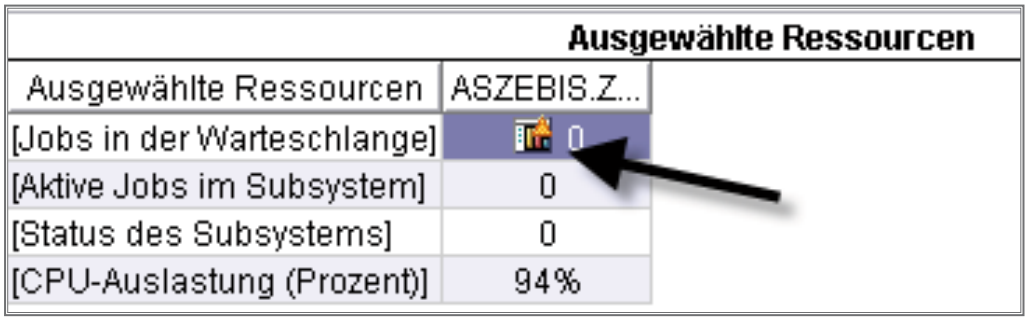
Schwellenwerte Höher-Angaben

– Übernehmen Sie alle weiteren Einstellungen und klicken Sie auf die Schaltfläche „OK”.

4.3.4
Seite 4

Verwalten von Schwellenwerten

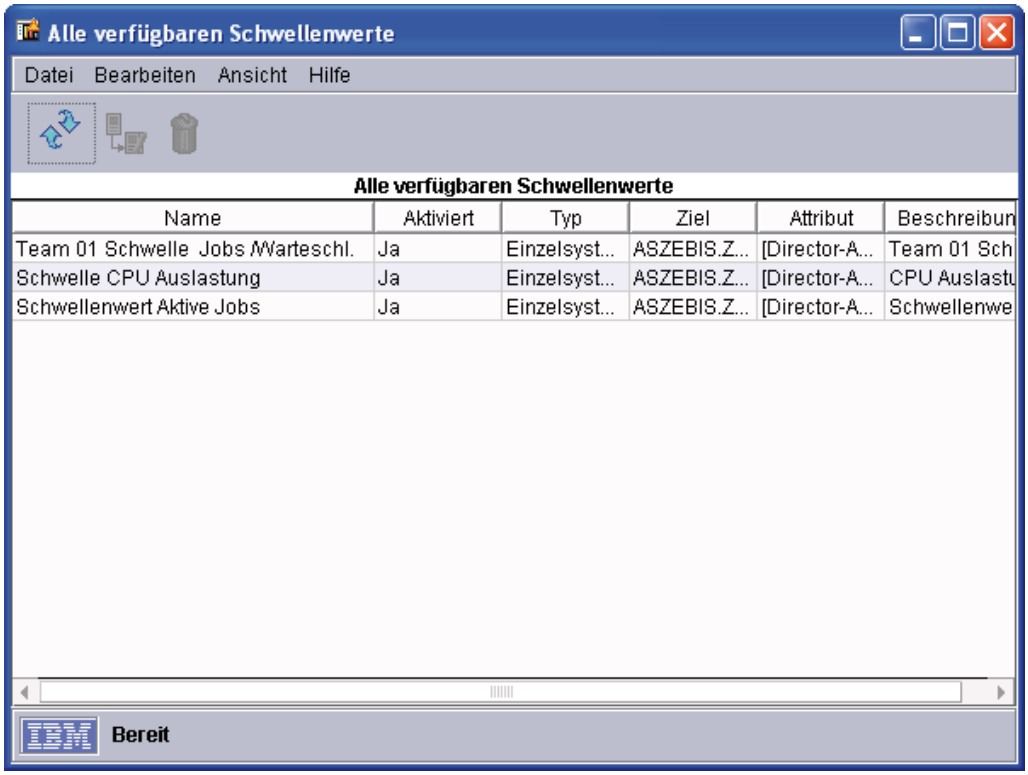
– Die definierten Schwellenwerte werden in der Übersicht der ausgewählten Ressourcen in Form eines Symbols dargestellt. Schauen Sie sich die Auflistung an – dort finden Sie nun bei dem Eintrag „Jobs in Warteschlange" ein Symbol, welches den definierten Schwellenwert darstellt.



Ausgewählte Ressourcen

– Schließen Sie nun das Fenster der Ressourcenverwaltung.

– Die gesetzten Schwellenwerte lassen sich in der Hauptansicht der IBM Director Konsole verwalten. Die Verwaltung ist über den Bereich „Ressourcenmonitor/Alle verfügbaren Schwellenwerte" möglich. Mit einem Doppelklick auf diesen Eintrag erhalten wir eine Übersicht aller definierten Schwellenwerte. Wählen Sie diesen Eintrag nun mit einem Doppelklick aus.

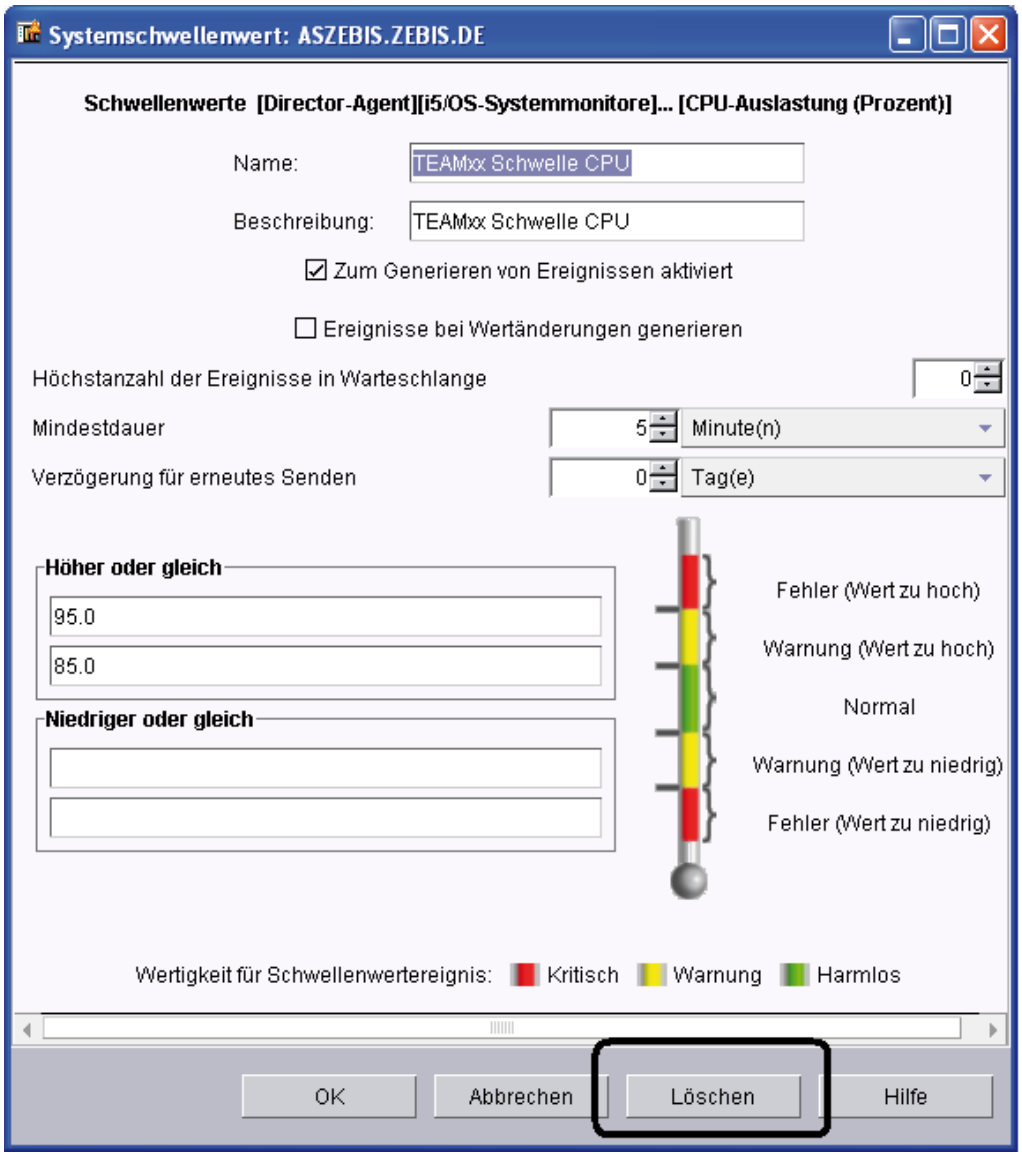


Alle verfügbaren Schwellenwerte

4.3.4.1
Seite 2

– Die Verwaltung der Schwellenwerte kann auch von dem zu überwachenden System ausgehend gestartet werden. Klicken Sie dazu in der Ansicht „Alle verwalteten Objekte“ auf das gewünschte System mit der rechten Maustaste und wählen Sie dort die Option „Alle verfügbaren Schwellenwerte“. Damit gelangen Sie ebenfalls wieder in die Übersicht der definierten Schwellenwerte.

Schwellenwerte lassen sich natürlich auch wieder entfernen. Wenn Sie dies ausführen wollen, wählen Sie einfach den Verwaltungsmodus für den Schwellenwert aus und klicken auf die Schaltfläche „Löschen“, wie es die folgende Abbildung zeigt.



Löschen

Ereignisaktionspläne

Zuvor haben wir die Schwellenwerte kennen gelernt. Diese dienen lediglich der Angabe von Schwellen. Die Aktionen, die bei dem Erreichen bzw. Über-/Unterschreiten der Werte ausgelöst werden sollen, müssen in Form der Ereignisaktionspläne definiert werden.

Inhalte dieses Teilbereichs
In diesem Abschnitt erhalten Sie Informationen über:

– Das Erstellen und Verwalten von Ereignisaktionsplänen

– Arbeiten mit den Ereignisprotokollen

22. Ergänzung 3/2010

4.3.4.2

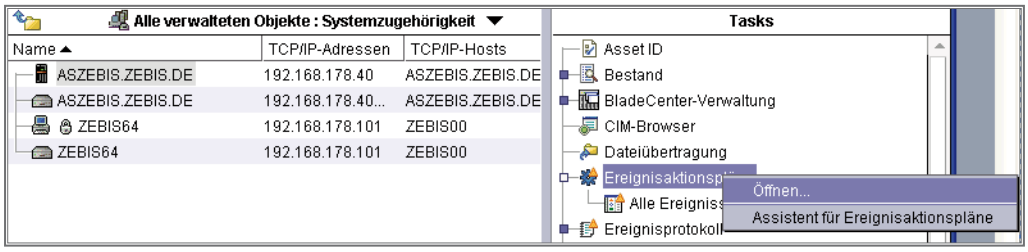
Seite 2

Erstellen eines Ereignisaktionsplans

In diesem ersten Teil der Betrachtung der Ereignisaktionspläne (kurz: EAP) werden wir die grundsätzliche Definition und Verwaltung der EAPs kennen lernen. Dabei erstellen wir mit Hilfe eines Assistenten einen EAP für die Überwachung eines Subsystems.

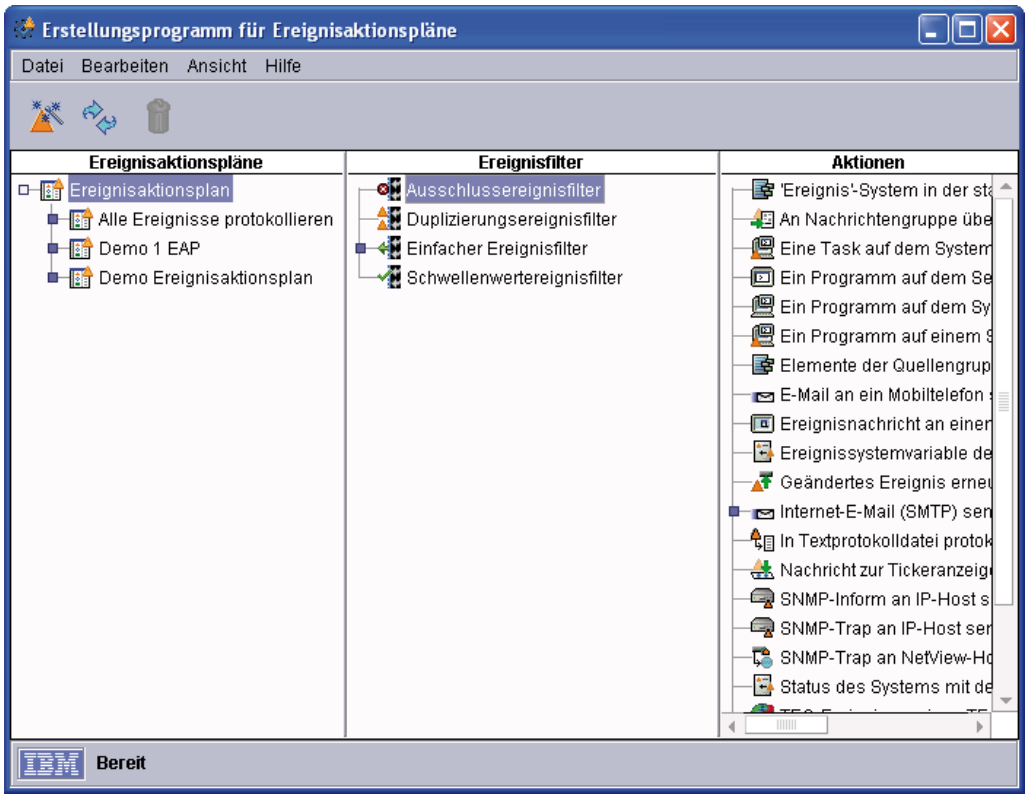
– Klicken Sie in der Ansicht der IBM Director Konsole in dem Bereich „Tasks" mit der rechten Maustaste auf den Eintrag „Ereignisaktionspläne".

– Wählen Sie die Option „Öffnen".



Ereignisaktionsplan

– In dem sich nun öffnenden Fenster „Erstellungsprogramm für Ereignisaktionspläne" erweitern wir den in dem linken Anzeigenbereich befindlichen Eintrag „Ereignisaktionsplan".



Erstellungsprogramm Ereignisaktionsplan

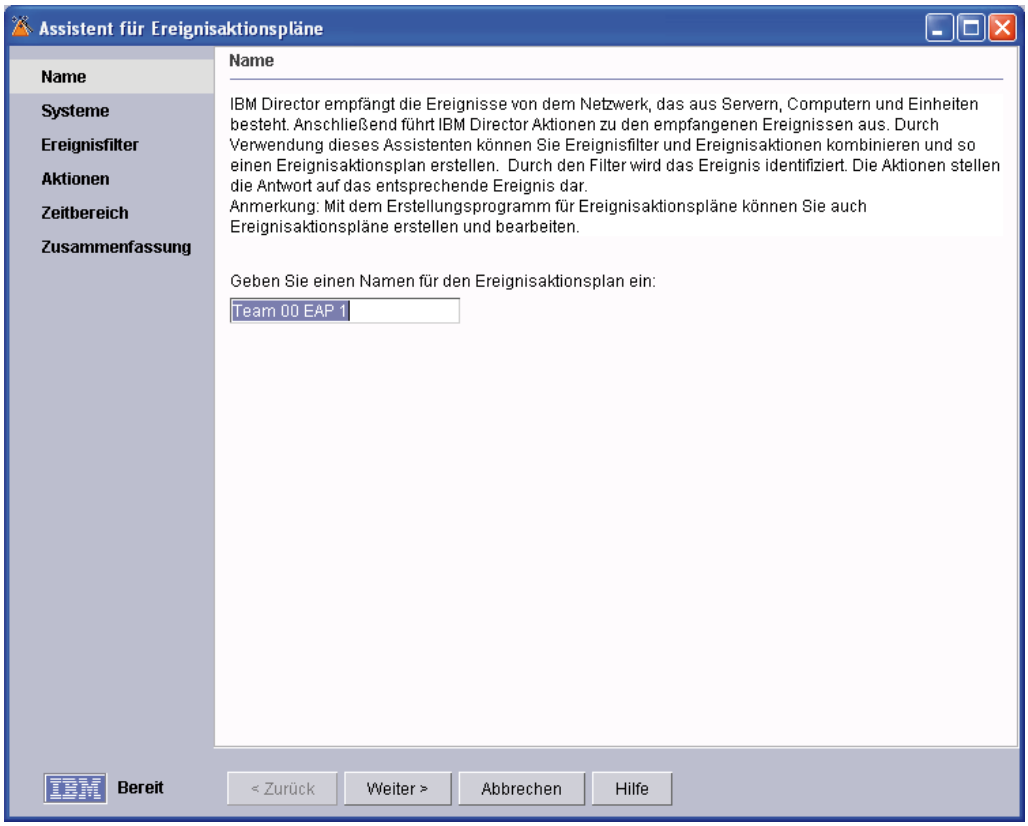
4.3.4.2.1
Seite 2

– Schauen Sie sich den Aufbau der Anzeige an – diese beinhaltet die drei wesentlichen Bereiche der EAP in Form des EAP, der Ergebnisfilter und der Aktionen.

– Lassen Sie uns nun einen eigenen EAP definieren. Dazu verwenden wir einen Assistenten, der über die Menüoption „Datei/Neu/Assistent für Ereignisaktionsplan“ aufgerufen wird.

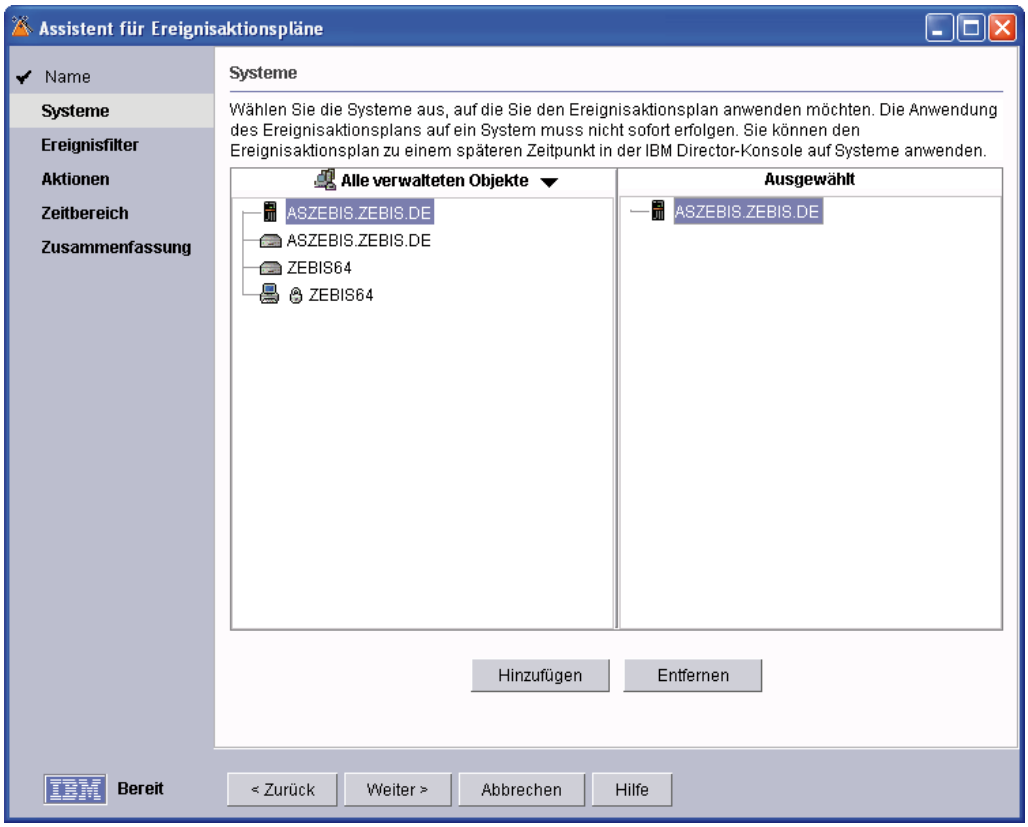
– Geben Sie in der nächsten Anzeige einen Namen für den Ereignisaktionsplan an, als Beispiel „EAP 1“.

– Bestätigen Sie die Angaben mit einem Klick auf „Weiter“.



Assistent für Ereignisaktionsplan

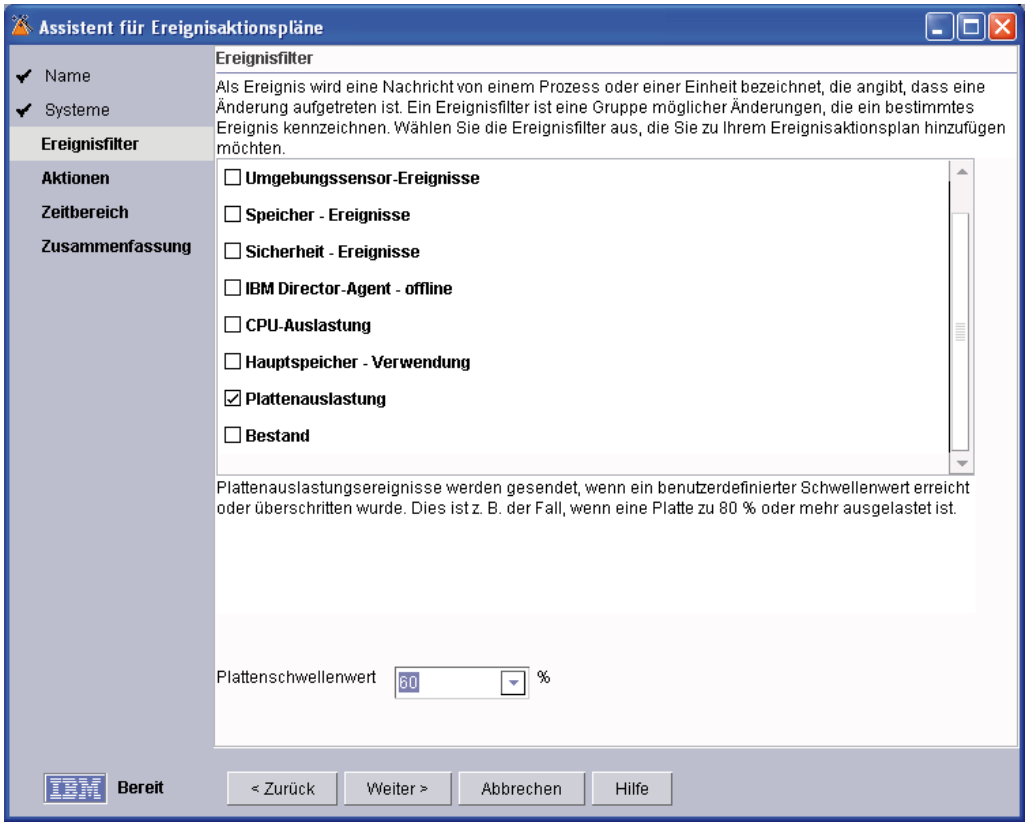
– Jetzt wählen wir das System aus, für das die Überwachung definiert werden soll und bestätigen die Auswahl mit einem Klick auf die Schaltfläche „Hinzufügen“. Klicken Sie erst dann auf „Weiter“, wenn das gewünschte System in dem rechten Anzeigenbereich steht!



Systemauswahl

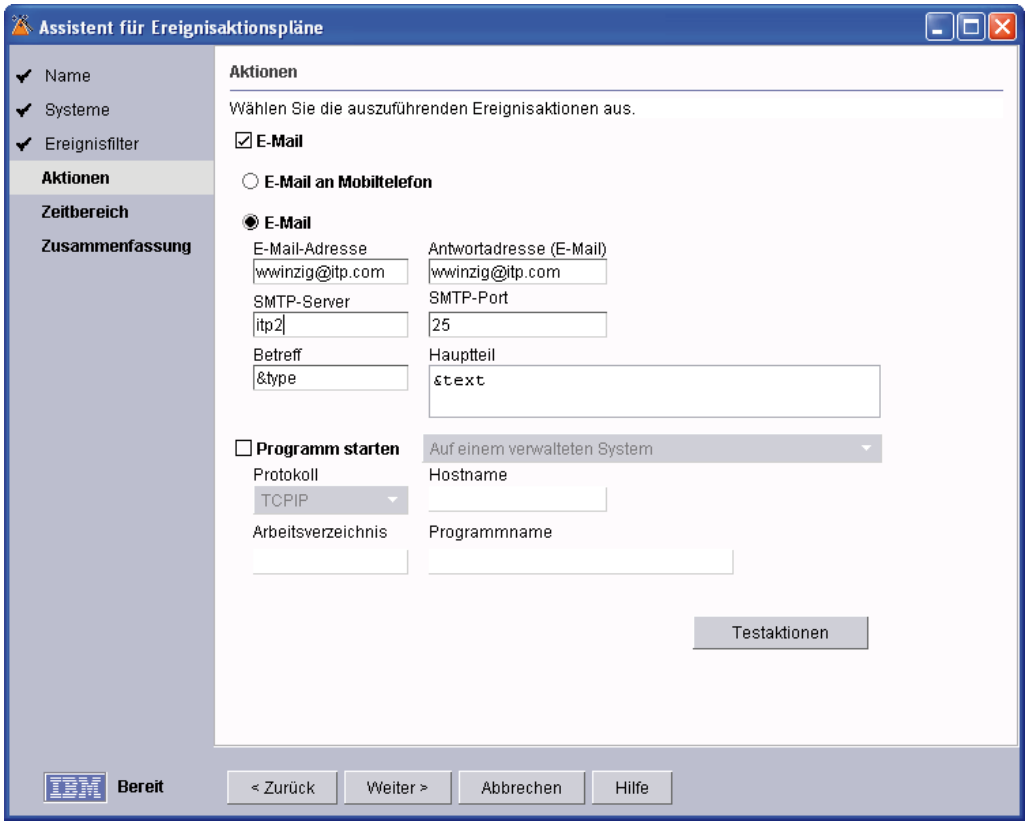
4.3.4.2.1
Seite 4

– In der nächsten Anzeige „Ereignisfilter“ klicken wir auf den Eintrag „Plattenauslastung“ und tragen als Schwellenwert 60 % ein. Bestätigen Sie mit einem Klick auf „Weiter“.



Ereignisfilter

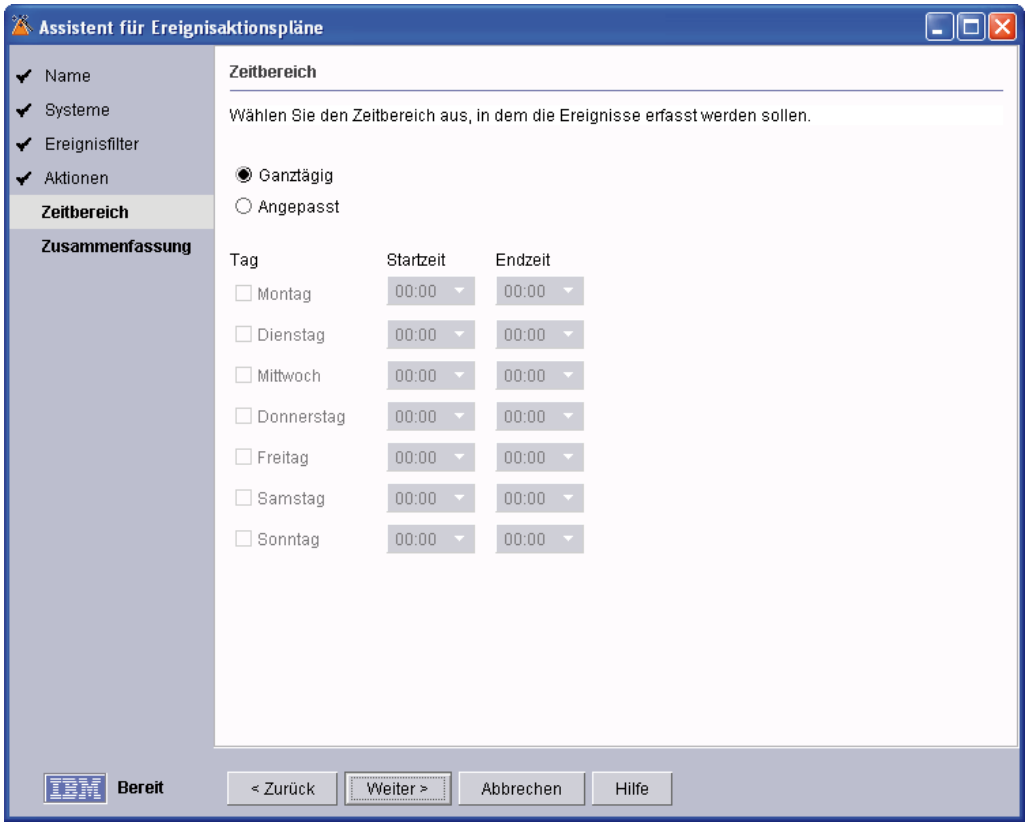
– Der IBM Director bietet als Ereignisreaktion die Möglichkeit der Benachrichtigung via Mail, SMS oder auch einem Programmstart. Wir wollen als Ergebnis eine Mail erhalten, welche uns darauf hinweist, dass die CPU-Auslastung den Schwellenwert erreicht bzw. überschritten hat. Füllen Sie dazu die Felder, wie es in der folgenden Abbildung zu sehen ist, und bestätigen Sie mit einem Klick auf „Weiter“.



Aktion

4.3.4.2.1
Seite 6

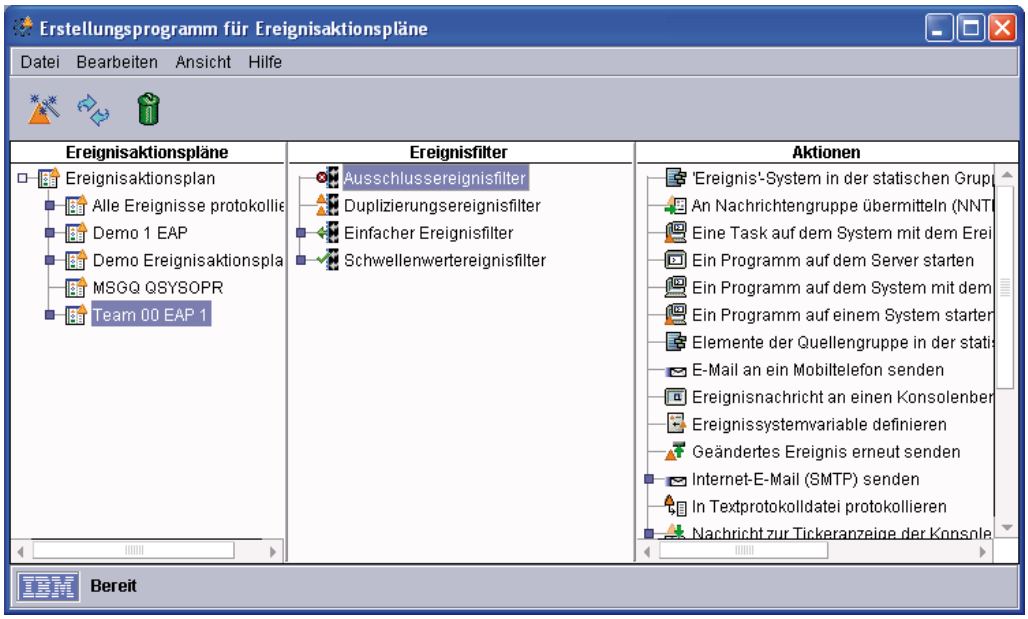
Der IBM Director bietet die Möglichkeit, die Erfassung bzw. die Reaktion auf Ereignisse einzuschränken. Eine solche Einschränkung kann in der nächsten Anzeige vorgenommen werden.



Zeitbereich

– Übernehmen Sie die Standardeinstellung „Ganztägig“ und klicken Sie auf „Weiter“.
– Die Zusammenfassung bestätigen Sie bitte mit einem Klick auf „Fertig stellen“.

– Aktualisieren Sie die Anzeige „Erstellungsprogramm für Ereignisaktionspläne“ – jetzt finden Sie den gerade erstellten Eintrag.



Aktuelle Übersicht EAP

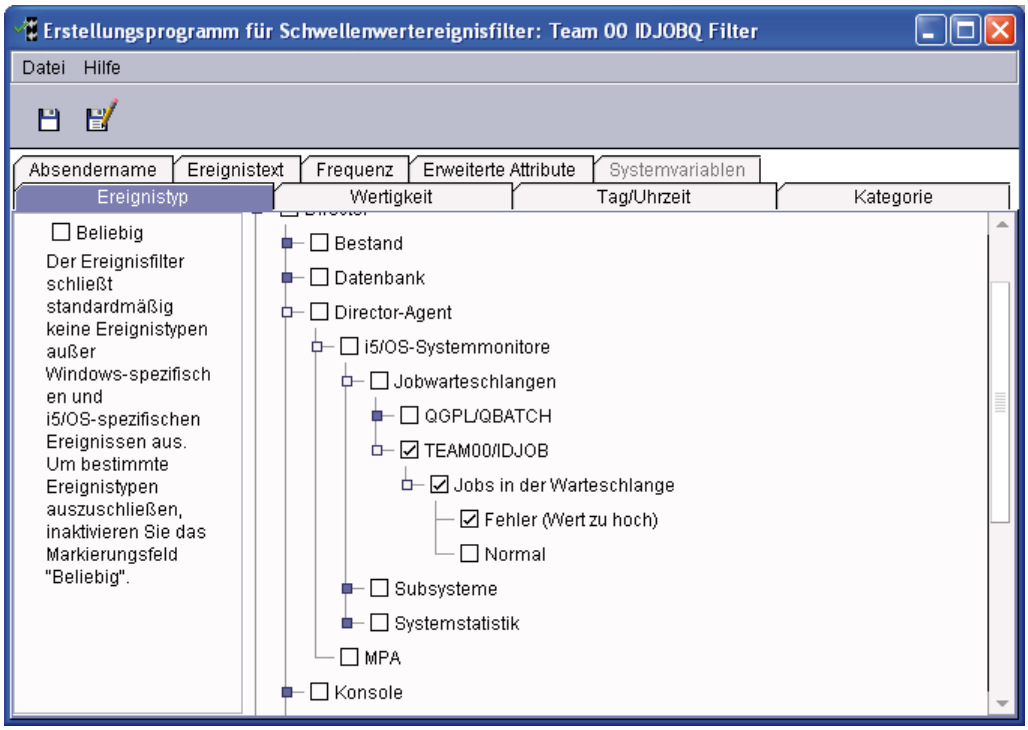
4.3.4.2.1

Seite 8

Erstellen eines EAP mit Schwellenwert

In der vorhergehenden Übung haben wir einen einfachen EAP erstellt. Nun wollen wir Ereignisse definieren, die auf Basis von Schwellenwerten ausgelöst werden.

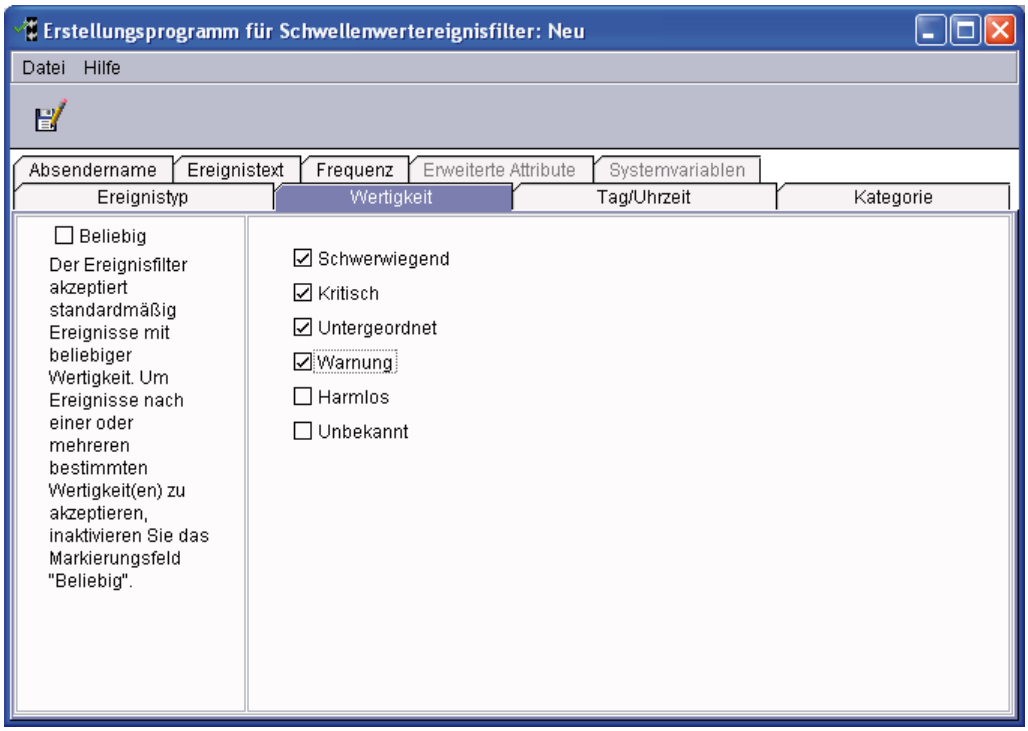
- Wählen Sie in der mittleren Sicht den Eintrag „Schwellenwertereignisfilter“ mit der rechten Maustaste und selektieren Sie anschließend die Option „Neu“.
- Selektieren Sie den Tabellenbereich „Ereignistyp“.
- Deaktivieren Sie die Auswahl „Beliebig“ in dem rechten Bereich.
- Erweitern Sie nun den Eintrag „Director/Director Agent/i5OS Systemmonitore/Jobwarteschlangen /xx“. (xx = Name der Jobwarteschlange)
- Erweitern Sie den Eintrag „Jobs in Warteschlange“ und dann den Eintrag für die Jobwarteschlange (z. B. IDJOBQ).
- Selektieren Sie dort den Eintrag „Fehler (Wert zu hoch)“.



Erstellungsprogramm für Schwellenwertereignisfilter

4.3.4.2.2
Seite 2

– Wir können an dieser Stelle auch die Schwellenwerte bestimmen. Wählen Sie den Tabellenbereich „Wertigkeit“ aus. Auch hier muss die Option „beliebig“ deaktiviert werden, um die Einstellungen anzupassen. Selektieren Sie anschließend einige Wertigkeiten.



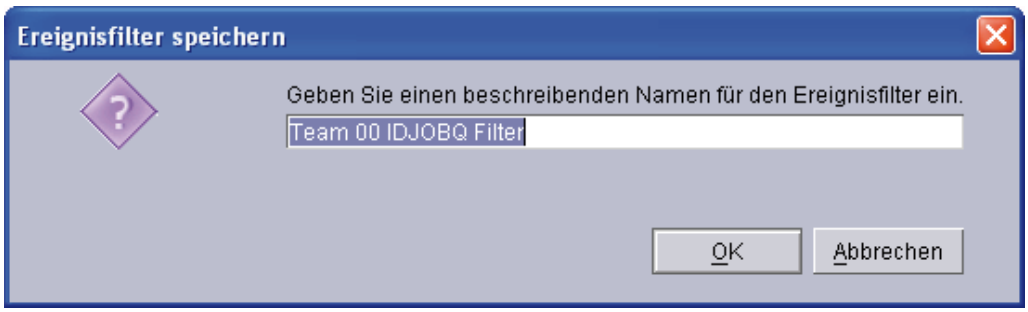
Wertigkeit

– In dem Bereich „Tag/Uhrzeit“ können wir den Zeitraum der Überwachung definieren. Übernehmen Sie die Werte.

– Schauen Sie sich die übrigen Definitionsbereiche an.

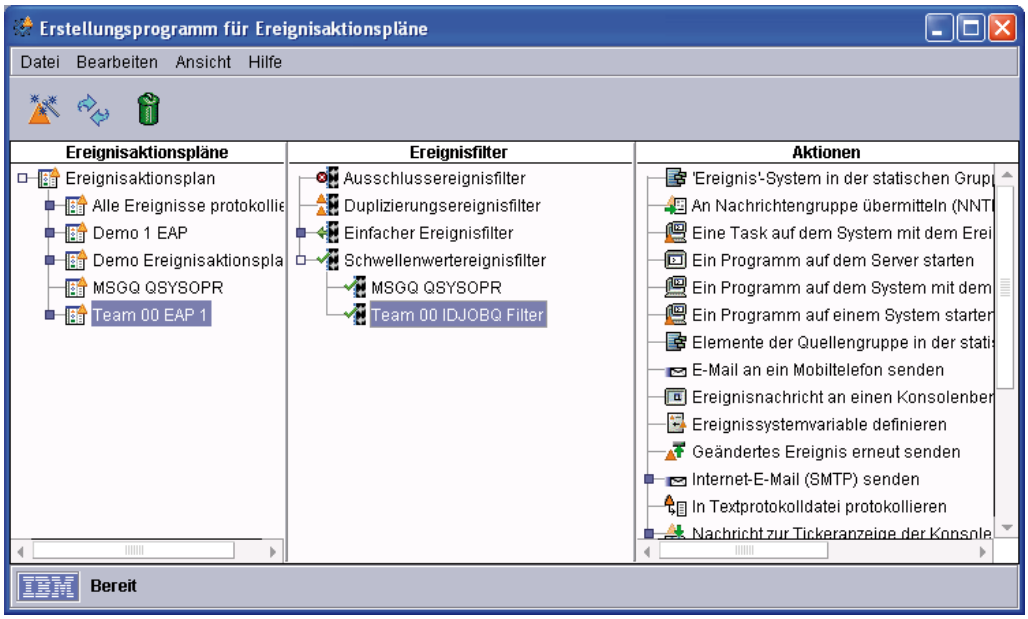
– Wählen Sie zum Speichern den Menüeintrag „Datei/Speichern unter“.

– Ordnen Sie der Definition den Namen „IDJOBQ Filter“ zu und bestätigen Sie die Eingabe mit einem Klick auf „OK“.



Name des Ereignisfilters

– Der neue Eintrag erscheint nun auch in der Übersicht der Schwellenwertereignisfilter.



EAP Übersicht

Auf dieselbe Weise können wir beispielsweise auch die Inhalte einer Nachrichtenwarteschlange überwachen. Leider hat IBM hier einige Verwirrung geschaffen, indem einige Einträge für das System i in dem Bereich „Director“ – und andere in dem Bereich „i5/OS“ vorzunehmen sind.

– Wiederholen Sie das Definieren für einen weiteren Schwellenwert, indem wir den Inhalt der Nachrichtenwarteschlange „QSYSOPR“ überwachen. Den Bereich der Nachrichtenwarteschlangen finden Sie in dem Bereich „i5/OS“.

Damit haben wir einen Schwellenwert definiert – jetzt fehlt noch die Aktion, die mit dem Erreichen des Schwellenwertes ausgeführt werden soll.

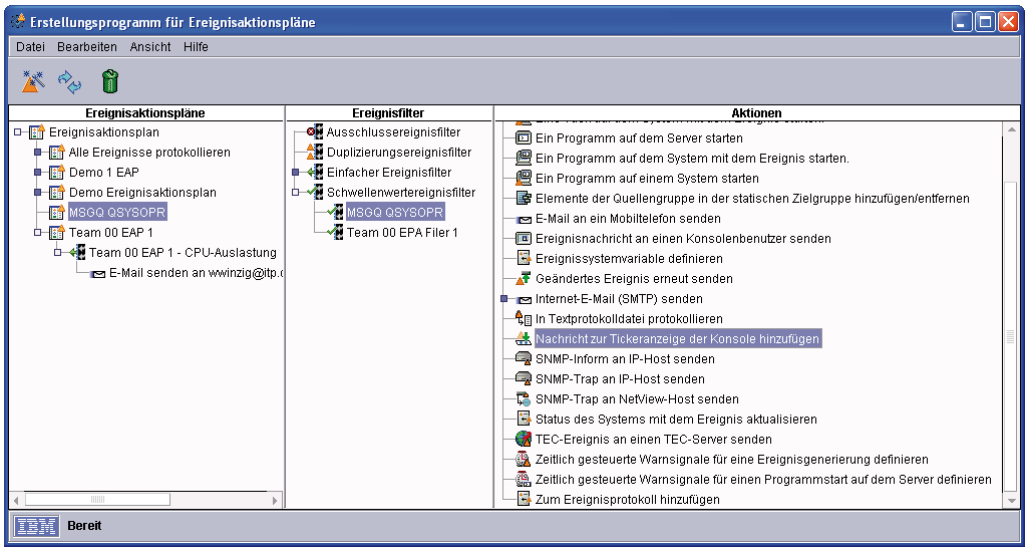
4.3.4.2.2

Seite 4

Erstellen einer Aktion

In den vorhergehenden Abschnitten haben wir die Erstellung von EAPs bzw. Filtern für Ereignisse kennen gelernt und auch Schwellenwerte definiert. In diesem Abschnitt lernen Sie, wie man Aktionen verwaltet.

– Schauen Sie sich in der Anzeige der Ergebnisaktionspläne die möglichen Aktionen an. Diese befinden sich in dem rechten Bereich der Darstellung.

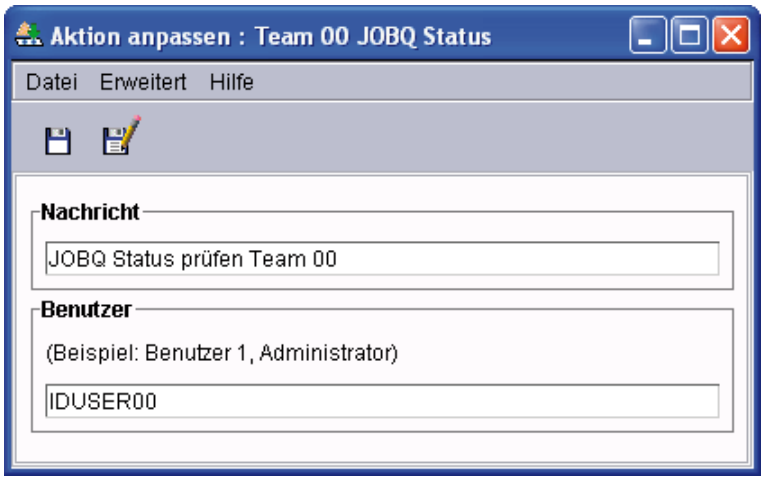


Aktionen

– Wählen Sie aus der Übersicht den Eintrag „Nachricht zur Tickeranzeige der Konsole hinzufügen“ mit einem Doppelklick aus.

– Geben Sie als Tickernachricht „JOBQ Status prüfen” ein.

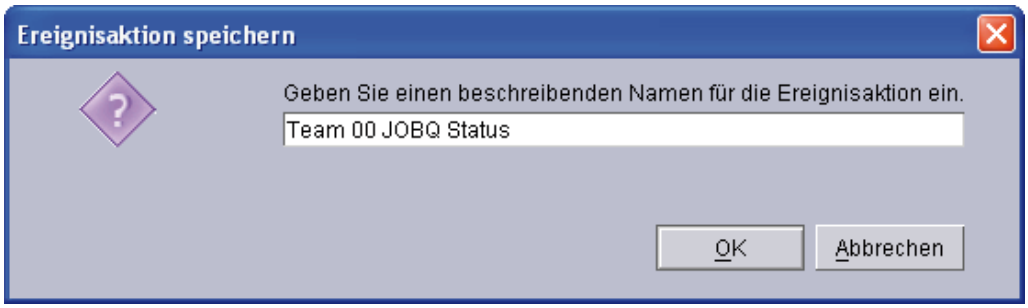
– Geben Sie in dem Feld „Benutzer“ den i5/OS Benutzer an (IDUSER).



Aktion anpassen

4.3.4.2.3
Seite 2

Schließen Sie die Anzeige mit der Menüoption „Datei/Speichern unter“ und vergeben Sie der Aktion den Namen „JOBQ Status“.



Ereignisaktion speichern

Mit einem Klick auf „OK” werden die Definitionen gespeichert.

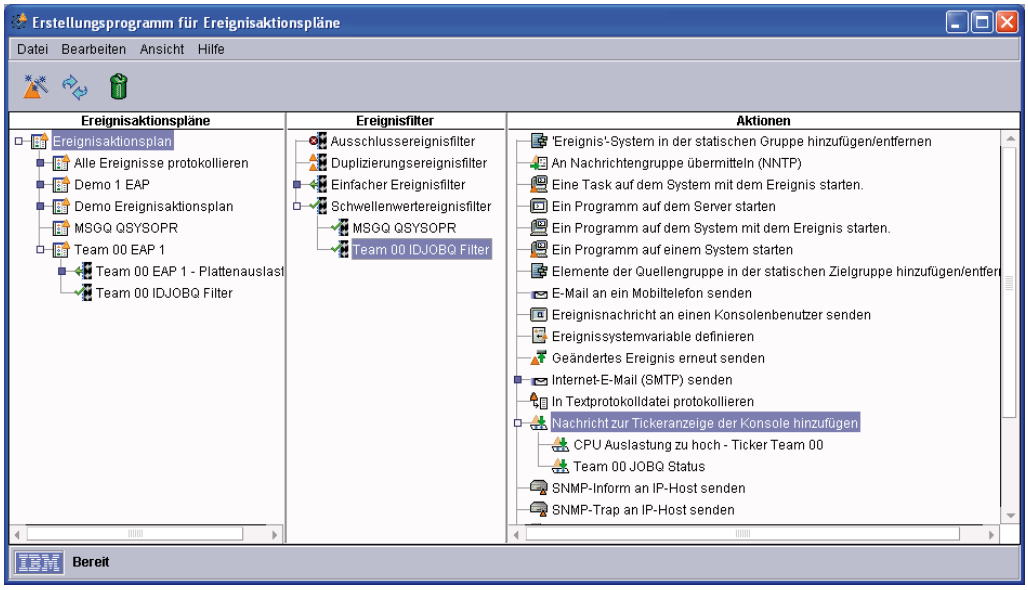
Hinzufügen von Filtern und Aktionen zu einem EAP

Nachdem wir nun mit Filtern und Aktionen gearbeitet haben, wollen wir diese nun unserem EAP hinzufügen.

Wir haben jetzt einige Komponenten erstellt, die in dem nächsten Abschnitt zusammengefügt werden.

– Wählen Sie nun in der mittleren Anzeige „Ereignisfilter" den zuvor erstellten Filter „IDJOBQ Filter" und ziehen diesen in den linken Bereich „EAP 1" unterhalb von „Ereignisaktionsplan".

– Ziehen Sie auf dieselbe Weise den Eintrag „MSGQ QSYSOPR" in den linken Bereich. Das Ergebnis sollte dann wie folgt aussehen:



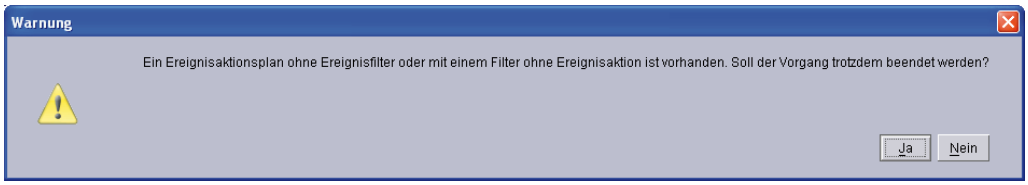
Aktionen

Damit haben wir unserem EAP mitgeteilt, welche Ereignisse überwacht werden sollen. Was fehlt, ist die Definition der Aktionen, welche bei dem Eintreten der Ereignisse ausgeführt werden sollen. Zwar haben wir zuvor für den Ereignisaktionsplan eine Aktion definiert – jedoch soll bei einer zu hohen Auslastung eine Tickeranzeige auf die zu hohe CPU-Auslastung hinweisen.

– Wählen Sie in dem rechten Anzeigenbereich den zuvor erstellten Eintrag „Ticker Auslastung zu hoch" und ziehen Sie diesen in den linken Bereich auf den Eintrag „EPA Filter 1".

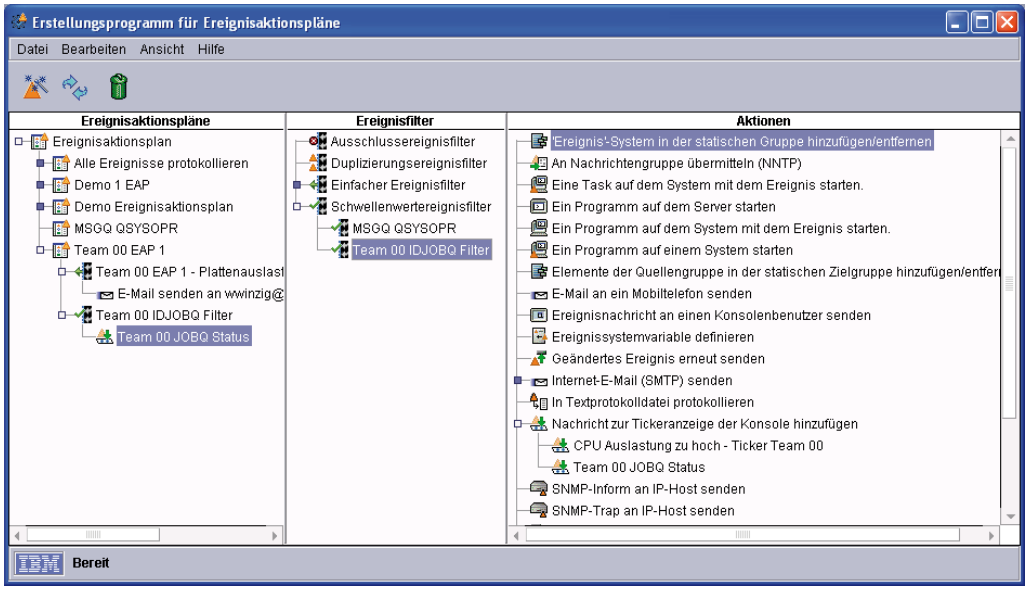
4.3.4.2.4
Seite 2

– Schließen Sie nun das Fenster und klicken Sie in der folgenden Anzeige auf „Ja“.



Warnhinweis

– Ziehen Sie nun in dem rechten Anzeigenbereich den Eintrag „JOBQ Status“ ebenfalls nach links in den Eintrag „IDJOBQ Filter“.



EAP Übersicht

Damit haben wir die Definition für den EAP und die Aktionen samt deren Zuordnung abgeschlossen.

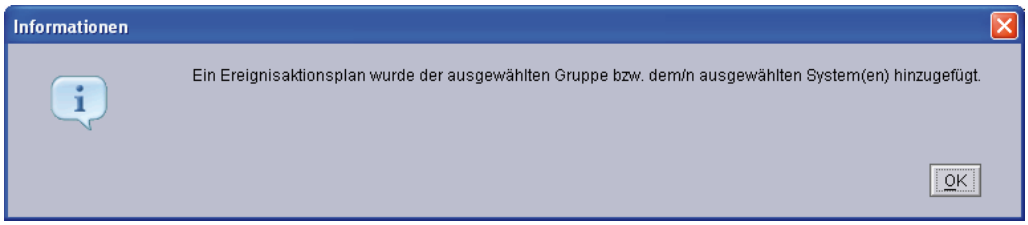
– Schließen Sie jetzt das Fenster „Erstellungsprogramm für Ereignisaktionspläne“. Sollte eine Meldung erscheinen, klicken Sie dort auf „Ja“.

Aktivieren des Ereignisaktionsplans

Ein EAP kann sehr einfach für ein System oder auch eine Gruppe aktiviert werden.

Die Basis dafür bildet nach der Definition des Ereignisaktionsplans die IBM Director Konsole – dort im speziellen der Bereich „Ereignisaktionspläne“ in der Ansicht „Tasks“.

– Wählen Sie nun Ihren Ereignisaktionsplan (EAP 1) aus der rechten Übersicht aus und ziehen Sie diesen auf das gewünschte System. Verwenden Sie dazu den Eintrag „System mit i5/OS“ im linken Bereich „Gruppen“.
– Bestätigen Sie die Meldung in der folgenden Anzeige mit einem Klick auf „OK”.



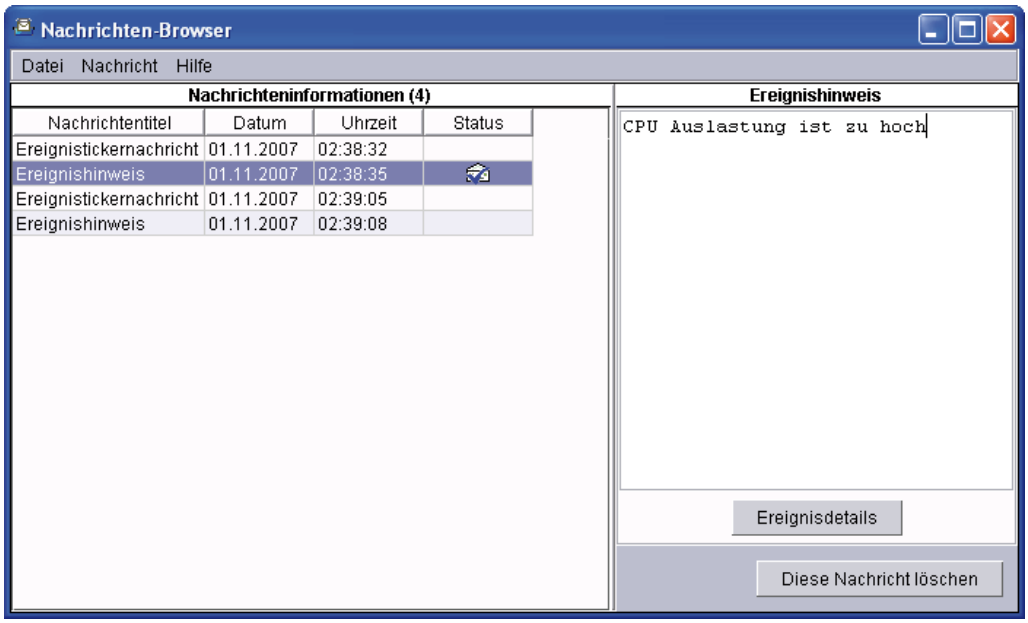
Hinzugefügt

Das war es auch schon. Damit haben Sie den EAP Ihrem System zugeordnet. Wenn nun der von Ihnen vorgegebene Schwellenwert erreicht wird, reagiert der IBM Director darauf und das definierte Ereignis wird ausgelöst.

Es sollte uns doch gelingen die CPU Auslastung der Maschine so zu steigern, dass wir jetzt das Ergebnis unserer Arbeit simulieren können.

4.3.4.2.5
Seite 2

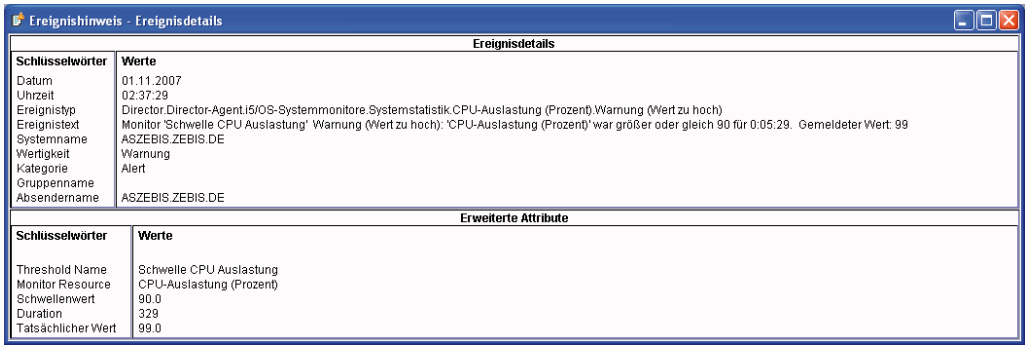
Wenn dies der Fall ist, erscheint ein zusätzliches Fenster mit einer Nachricht.



Nachrichten-Browser

In der vorhergehenden Abbildung sehen Sie den Nachrichten-Browser der IBM Director Konsole. Dieser beinhaltet eine Grobübersicht der eingetretenen Ereignisse. Details zu dem jeweiligen Ereignis bekommen wir mit einem Klick auf die Schaltfläche „Ereignisdetails“.

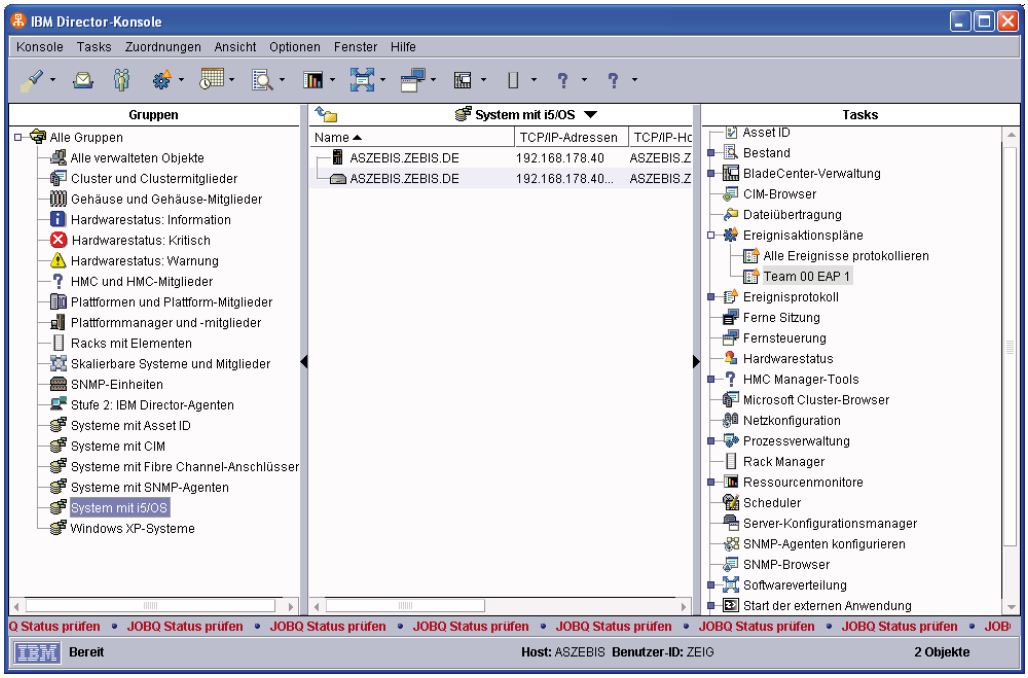
– Klicken Sie nun auf die Schaltfläche „Ereignisdetails“.



Ereignisdetails

Die Anzeige der Nachrichtenkonsole ist nur eine Möglichkeit, wie wir eingetretene Ereignisse mitgeteilt bekommen. Grundsätzlich lassen sich auf diese Weise alle Aktionen verwenden.

Sie erinnern sich noch – neben der Überwachung der CPU-Auslastung hatten wir die Überwachung der Jobwarteschlange definiert. Dabei sollte im Falle der Überschreitung der Anzahl der Jobs eine Tickermeldung generiert werden:



Tickeranzeige

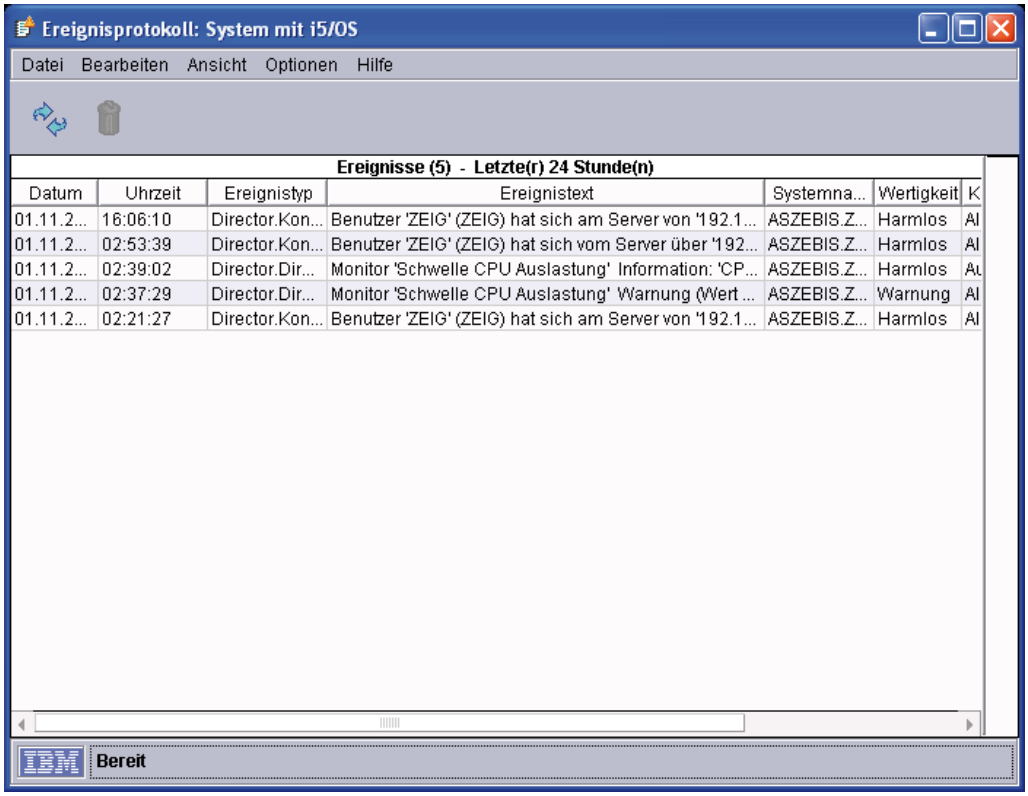
4.3.4.2.5

Seite 4

Anzeigen des Ereignisprotokoll

Die Ereignisse, die mit dem IBM Director aufgezeichnet wurden, das Ergebnisprotokoll, werden wir in der nächsten Übung kennen lernen..

– Wählen Sie in der linken Spalte „Gruppen“ den Eintrag „System mit i5/OS“ mit der rechten Maustaste aus.
– Selektieren Sie den Eintrag „Ergebnisprotokoll“.
– Damit öffnet sich ein weiteres Fenster mit der Anzeige des Ereignisprotokoll.



Ereignisprotokoll

Wie Sie erkennen können, sind hier unterschiedliche Einträge enthalten. Unter anderem finden wir hier Informationen über An-/Abmeldungen an dem IBM Director als auch beispielsweise das Überschreiten von definierten Schwellenwerten.

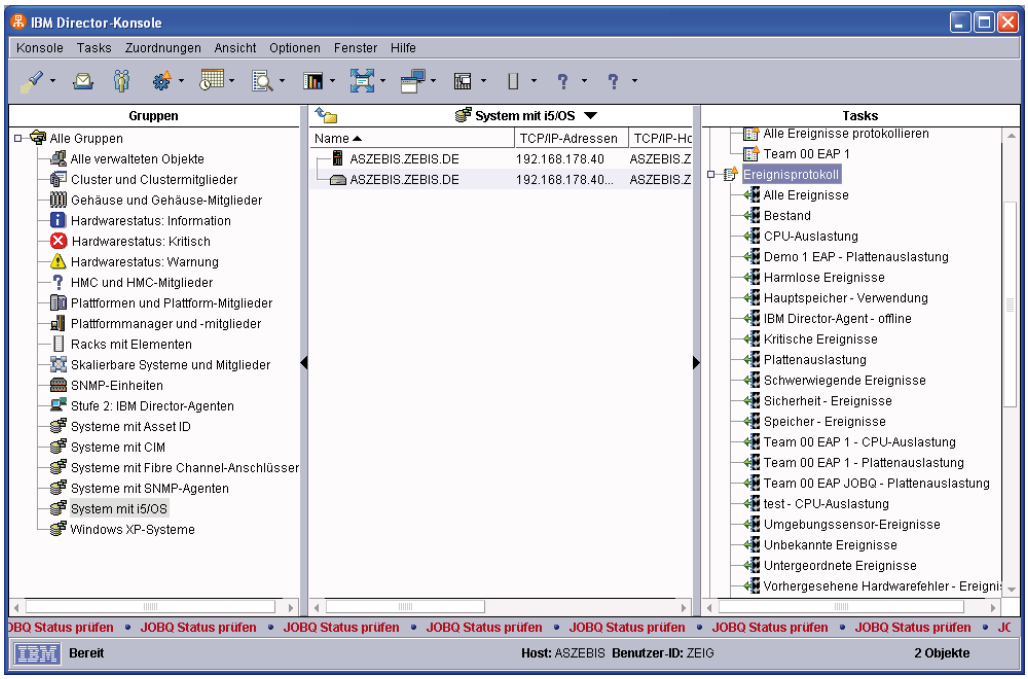
Mit der Zeit nimmt die Zahl der Einträge sicher zu – so dass eine Einschränkung bzw. Vorselektion sinnvoll ist. IBM Director bietet dazu vorgefertigte Möglichkeiten.

4.3.4.2.6

Seite 2

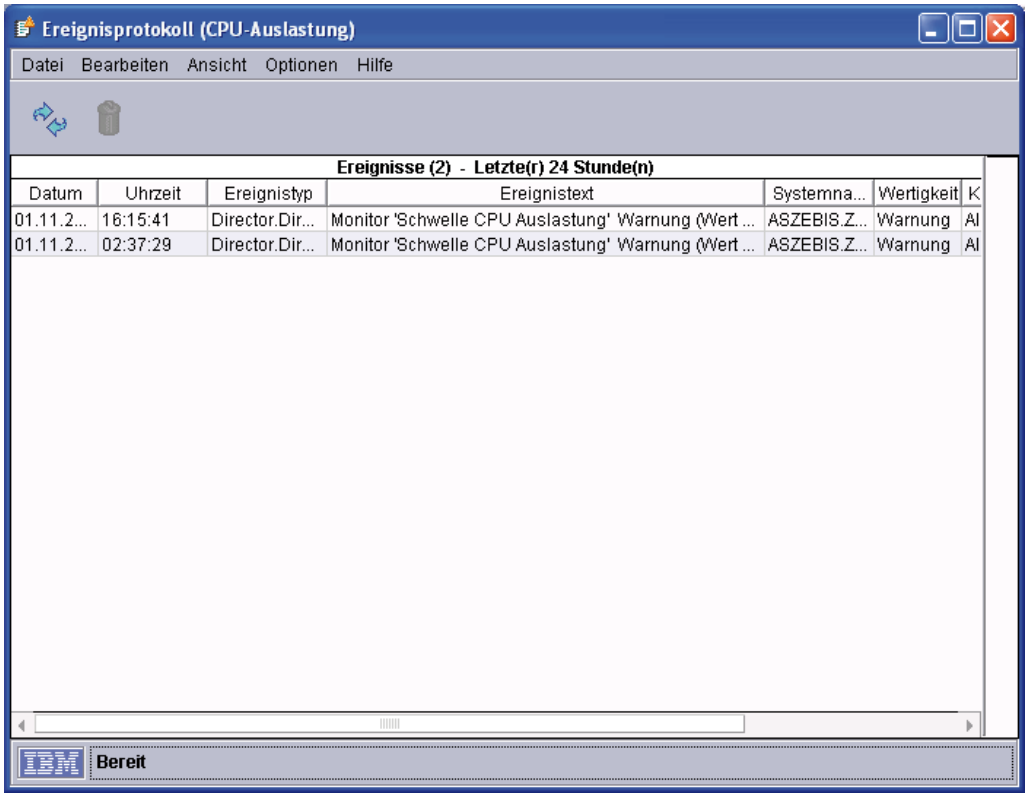
– Schließen Sie nun die Anzeige des Ereignisprotokolls.

– Erweitern Sie in der rechten Spalte der Anzeige der IBM Director Konsole den Eintrag „Ergebnisprotokoll“.



Tickeranzeige in der Konsole

– Wählen Sie den Eintrag „CPU-Auslastung” mit einem Doppelklick aus. Damit gelangen wir in die Anzeige des Ereignisprotokoll – diesmal allerdings eingeschränkt auf Einträge, welche sich auf die zu hohe CPU-Auslastung beziehen.



CPU-Auslastung

Auch hier sind weitere Einschränkungen in der Darstellung möglich, damit die Übersicht bei einer Vielzahl von Einträgen gewährleistet werden kann. Über die Menüoption „Optionen“ lässt sich beispielsweise der Zeitraum auswählen, für den die Ereignisprotokolleinträge aufgelistet werden sollen.

– Schließen Sie die Anzeige des Ereignisprotokolls nun wieder.

4.3.4.2.6

Seite 4

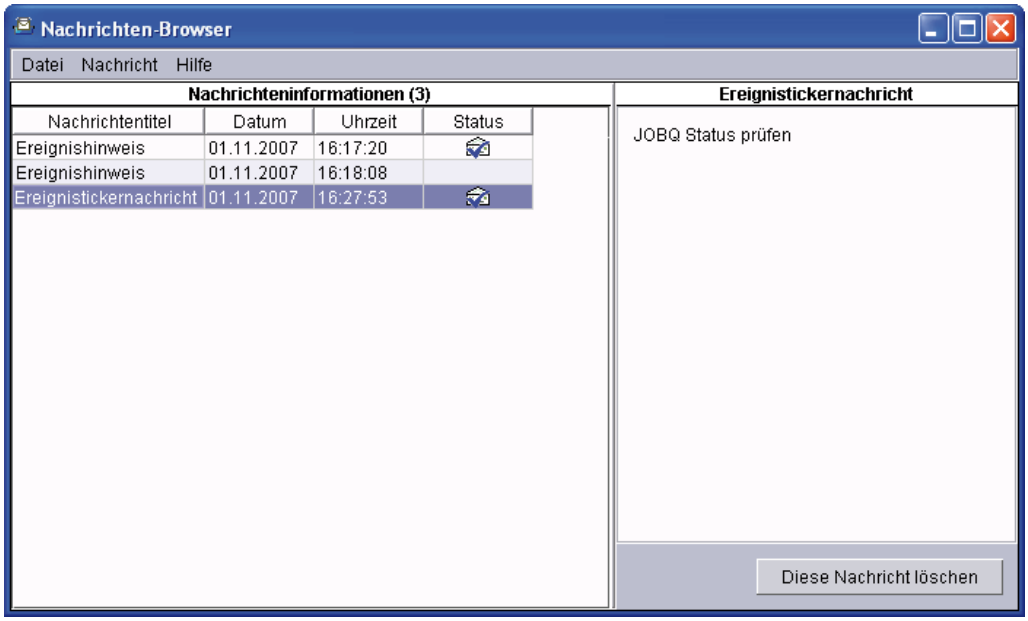
Entfernen von Nachrichten

Wie Sie sicher bemerkt haben, wird die Fehlernachricht noch immer in dem Ticker angezeigt. In diesem Abschnitt möchte ich Ihnen zeigen, wie man Nachrichten verwaltet bzw. entfernt.

– Klicken Sie mit der rechten Maustaste in die Tickerleiste der IBM Director Konsole.

– Wählen Sie die Option „Nachrichten-Browser".

Damit öffnet sich die Anzeige „Nachrichten-Browser".



Nachrichten-Browser-Inhalte

– Markieren Sie den Eintrag „Ereignistickernachricht" mit einem Doppelklick. Achten Sie bitte darauf, dass Sie die korrekte Nachricht auswählen.

– Mit der Auswahl wird in dem rechten Fensterbereich der Inhalt der definierten Nachricht dargestellt.

– Mit einem Klick auf die Schaltfläche „Diese Nachricht löschen" kann der Eintrag entfernt werden.

– Schließen Sie nun die Anzeige „Nachrichten-Browser".

– Wechseln Sie nun „schnell" in die Anzeige der IBM Director Konsole und achten Sie auf das Tickerlaufband. Die dort bisher ausgegebene Nachricht verschwindet nun.

4.3.4.2.7

Seite 2

Systeme und Gruppen

Der IBM Director Server ist in der Lage, zeitgleich unterschiedliche Betriebssysteme überwachen und managen zu können. Neben i5/OS können wir Systeme mit Windows, Linux und AIX einbinden. Zudem ist die Überwachung aller Bereiche, welche in Verbindung mit dem System i stehen – Beispiele sind „HMC und Blade Center" – mit zu implementieren.

In diesem Abschnitt erfahren Sie, wie man heterogene Betriebssysteme zentral mit dem IBM Director überwachen kann.

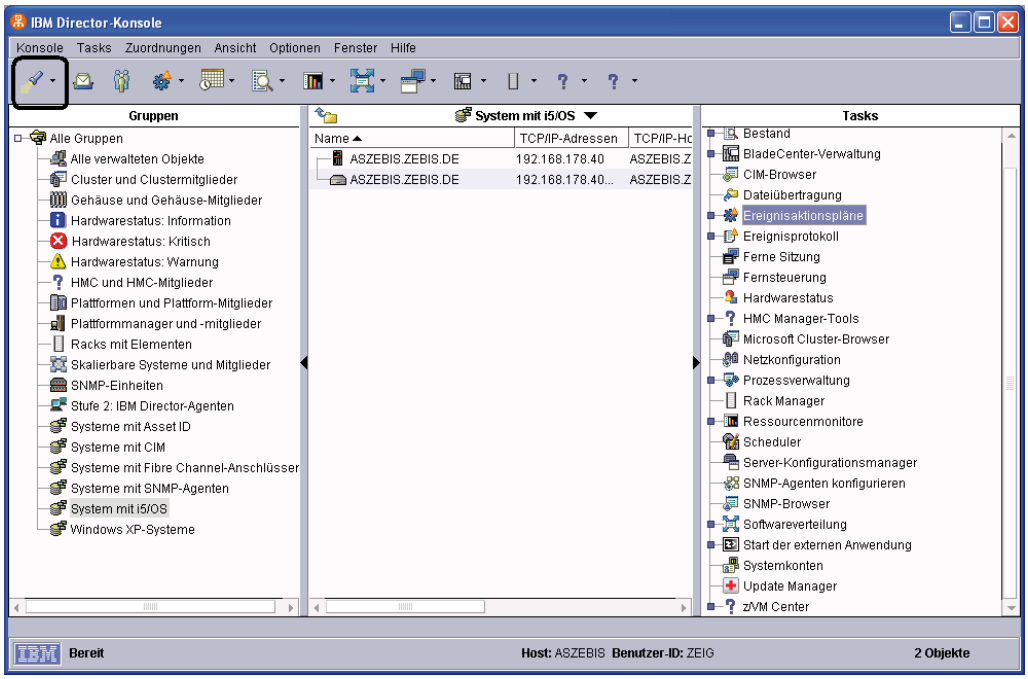
4.3.4.3

Seite 2

Hinzufügen von Systemen

Die zu überwachenden Systeme können manuell mittels der Angabe des Hostnamens bzw. der IP-Adresse dem IBM Director hinzugefügt werden. Alternativ dazu kann der IBM Director auch so konfiguriert werden, dass diese Systeme automatisch erkannt und für die Verwendung mit dem IBM Director vorbereitet werden.

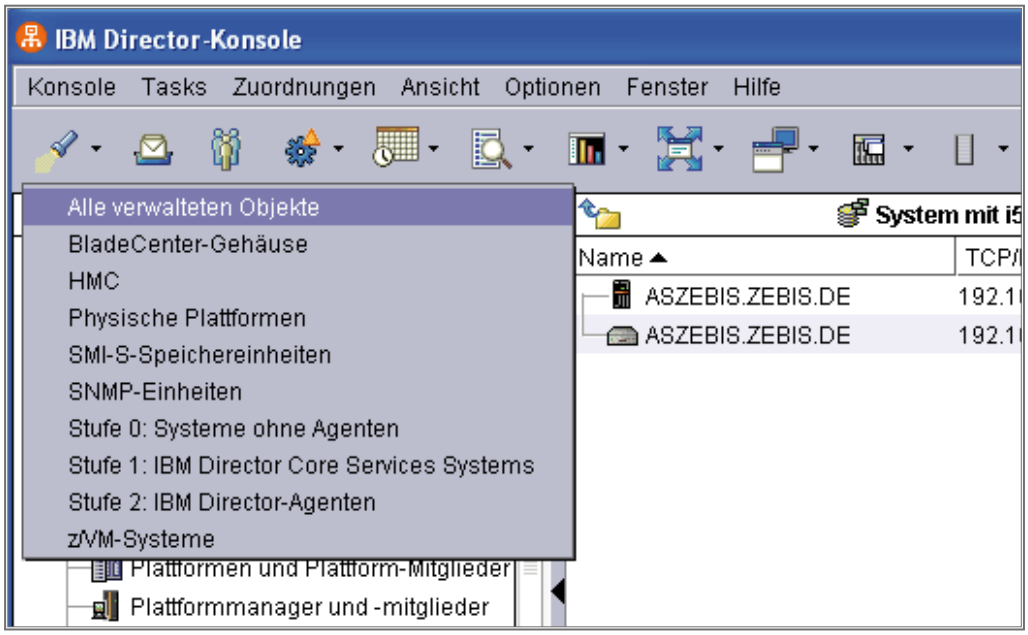
– Stellen Sie sicher, dass Sie sich in der Hauptanzeige der IBM Director-Konsole befinden.



IBM Director Hauptanzeige

– Wählen Sie in der Symbolleiste den linken Eintrag in Form der Taschenlampe mit einem Klick auf den Erweiterungspfeil aus.

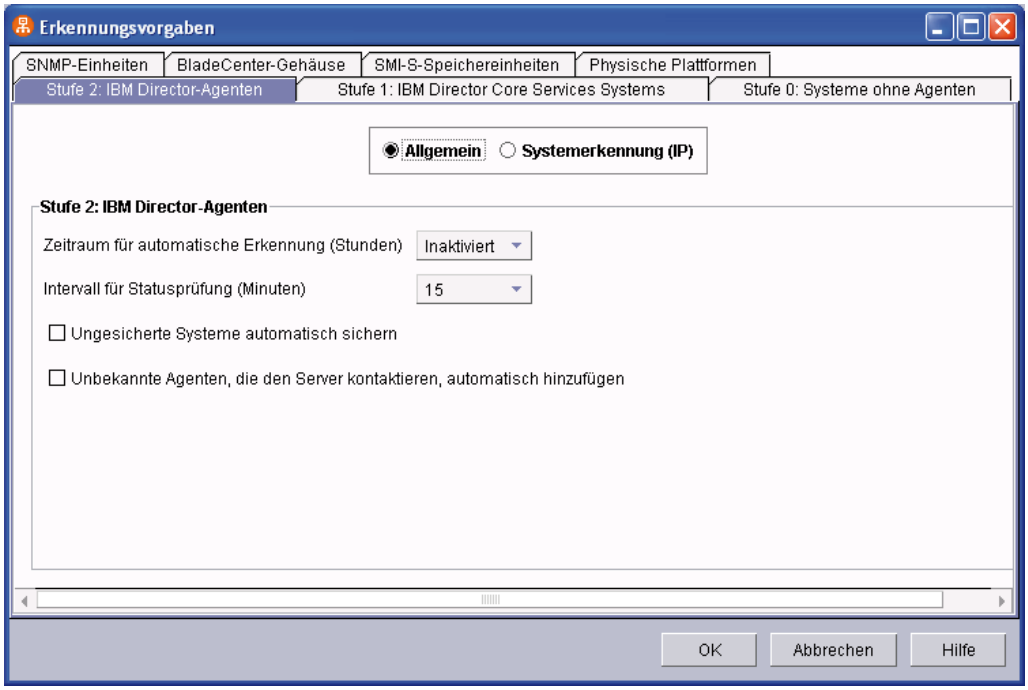
4.3.4.3.1
Seite 2



Alle verwalteten Objekte

– Schauen Sie sich die Optionen bitte nur an – die Erfassung wird zentral gestartet. Führen Sie bitte KEINE Erkennung durch und verlassen Sie die Optionsanzeige.

– Lassen Sie uns nun die Optionen für die Erkennung anschauen. Dazu selektieren wir die Menüoption „Optionen/Erkennungsvorgaben“.



Erkennungsvorgaben

Die automatische Erkennung lässt sich für unterschiedliche Bereiche definieren. Beispielsweise sind die Erkennungsangaben nach BladeCentern, SNMP Einheiten etc. möglich.

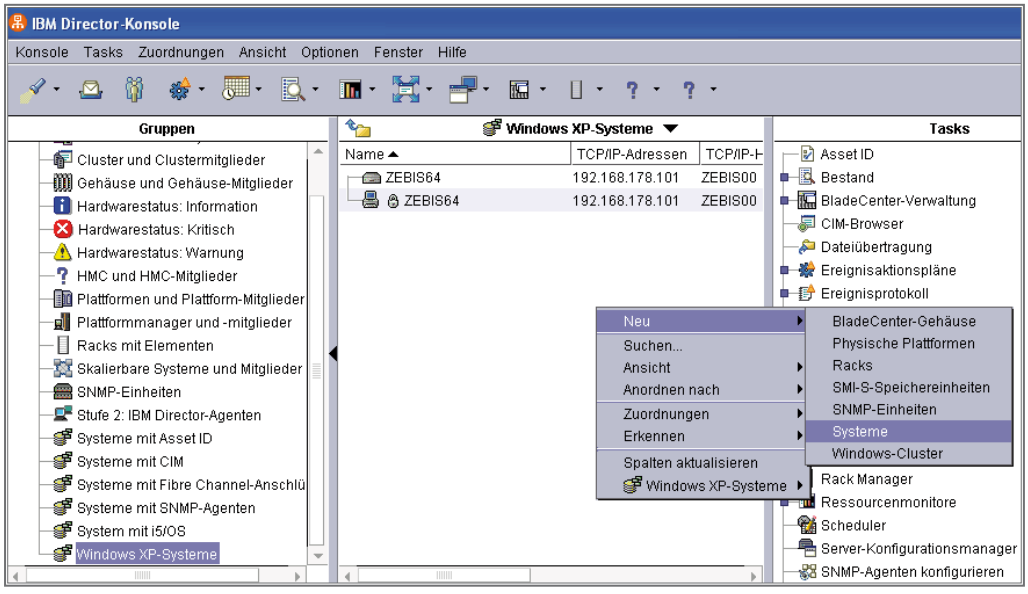
Die allgemeine Anzeige, die Sie in der vorhergehenden Abbildung finden, definiert den Prozess der Erkennung, der aktiviert werden kann. Mit der Vorgabe des Intervalls wird festgelegt, in welchen Zeitabständen die Umgebung nach neuen Systemen gescannt wird.

– Ändern Sie bitte KEINE Einstellungen und schließen Sie die Anzeige nun wieder.

Wenn Sie die automatische Erkennung aktivieren, werden die neu erkannten Systeme automatisch den Auflistungen in der IBM Director-Konsole hinzugefügt. Es kann Situationen geben, bei denen die automatische Erkennung nicht wünschenswert ist. Wahlweise kann ein weiteres System mittels eines manuellen Prozesses der IBM Director-Umgebung hinzugefügt werden.

– Kehren Sie in die Anzeige der IBM Director-Konsole zurück.

– Positionieren Sie den Cursor in der mittleren Spalte der Anzeige und wählen Sie nach einem Klick der rechten Maustaste den Eintrag „Neu/Systeme“.



Neue Systeme

– Geben Sie in dem folgenden Anzeigenbereich als Namen „PC xx“ ein.

– Übernehmen Sie den Eintrag „TCPIP”.

– Geben Sie als Adresse die IP-Adresse an, welche der Dozent Ihnen zugewiesen hat.

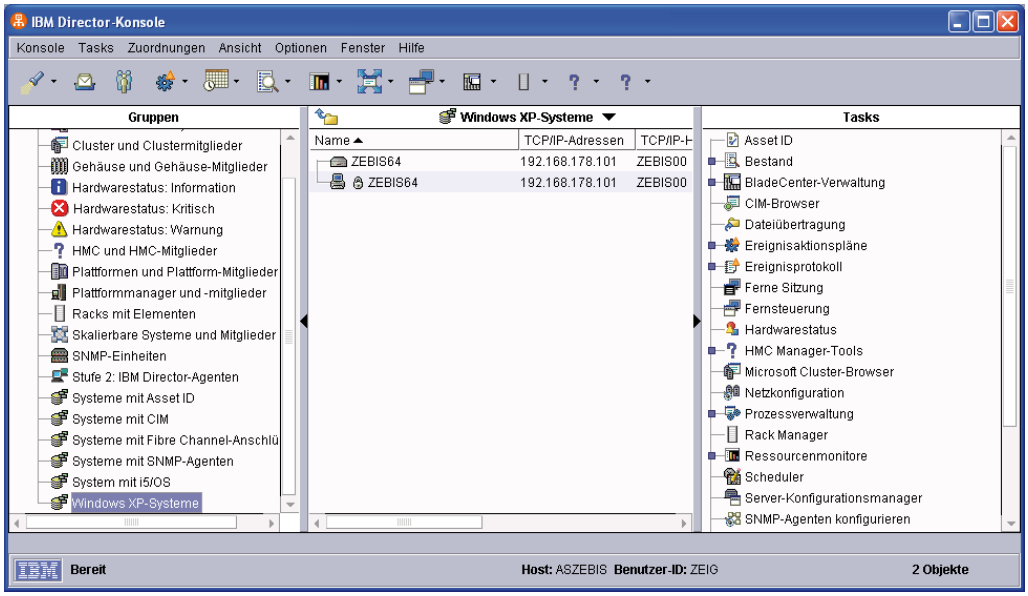
– Bestätigen Sie Ihre Angaben mit einem Klick auf „OK”.

4.3.4.3.1
Seite 4

Details des hinzuzufügenden Systems

Wenn Ihre Angaben korrekt waren, dann wird das neue System in der Auflistung der zu verwaltenden Systeme der IBM Director-Konsole erscheinen und steht damit zur Überwachung und Verwaltung bereit.

In der folgenden Abbildung sehen Sie ein System mit Windows XP zur Verwaltung in der IBM Director-Konsole:



Windows XP

Die Konfiguration der Überwachung von Nicht-i5-Systemen erfolgt ähnlich, wie wir es bereits bei der Konfiguration der i5-Umgebung kennen gelernt haben.

Erstellen von statischen und dynamischen Gruppen

Bei der Verwendung des IBM Directors lassen sich die Systeme in Gruppen zusammenfassen. Dabei unterscheiden wir statische und dynamische Gruppen.

– Eine statische Gruppe beinhaltet die zuvor definierten Mitglieder bzw. Systeme.

– Eine dynamische Gruppe dient der Definition von Systemen, welche bestimmten Vorgaben entsprechen. Prozesse innerhalb des IBM Directors verwalten dann die Zugehörigkeit der Systeme zu solch einer dynamischen Gruppe.

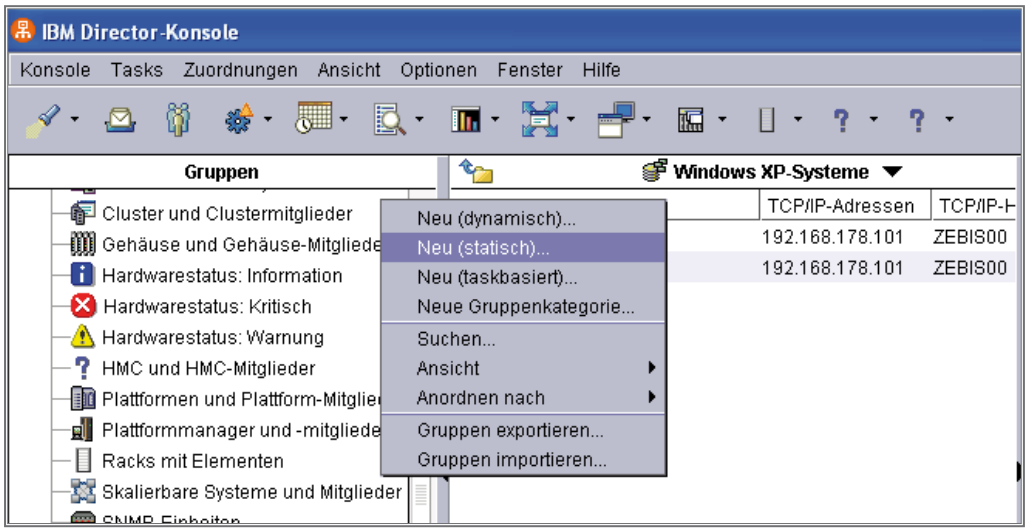
4.3.4.3.2

Seite 2

Erstellen einer statischen Gruppe

– Klicken Sie mit der rechten Maustaste in den linken Spaltenbereich „Gruppen" der IBM Director-Konsole.

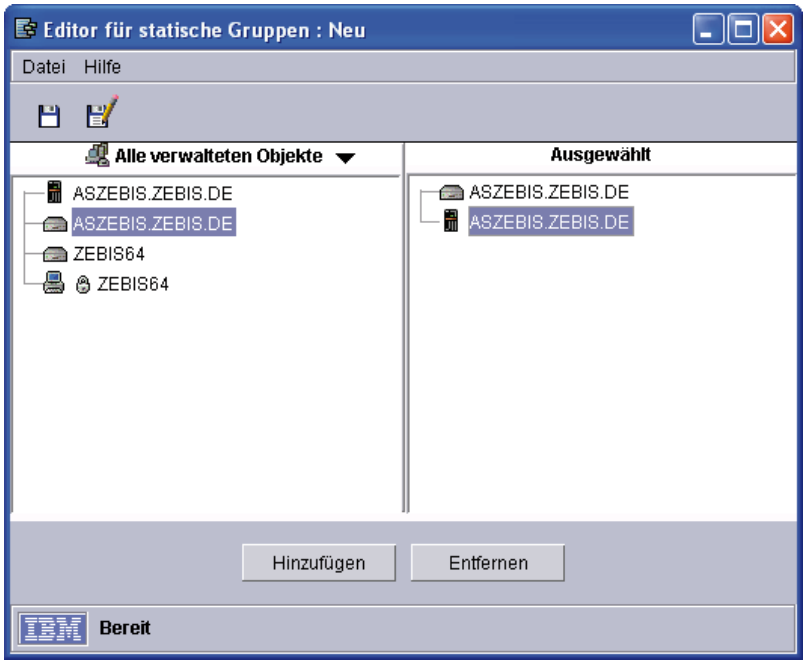
– Wählen Sie die Option „Neu (statisch)".



Neue statische Gruppe

– Wählen Sie das System i aus (den Eintrag mit dem IBM Director Agent) und bestätigen Sie diese Auswahl mit einem Klick auf „Hinzufügen".

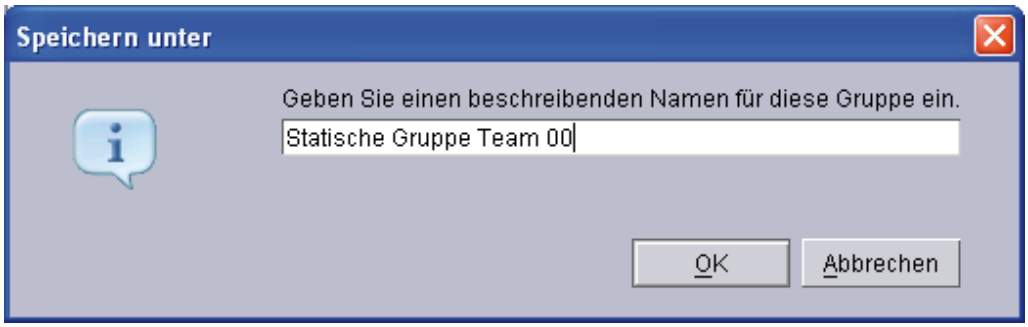
– Wiederholen Sie diesen Zuordnungsvorgang auch für den Eintrag des IBM Director Servers.



Editor für statische Gruppe

4.3.4.3.2.1
Seite 2

– Speichern Sie nun die Angaben und geben Sie als Namen für diese Gruppe „Statische Gruppe Team 00“ ein.



Speichern unter

– Schließen Sie das Fenster für die Definition der Gruppe.

Erstellen einer dynamischen Gruppe

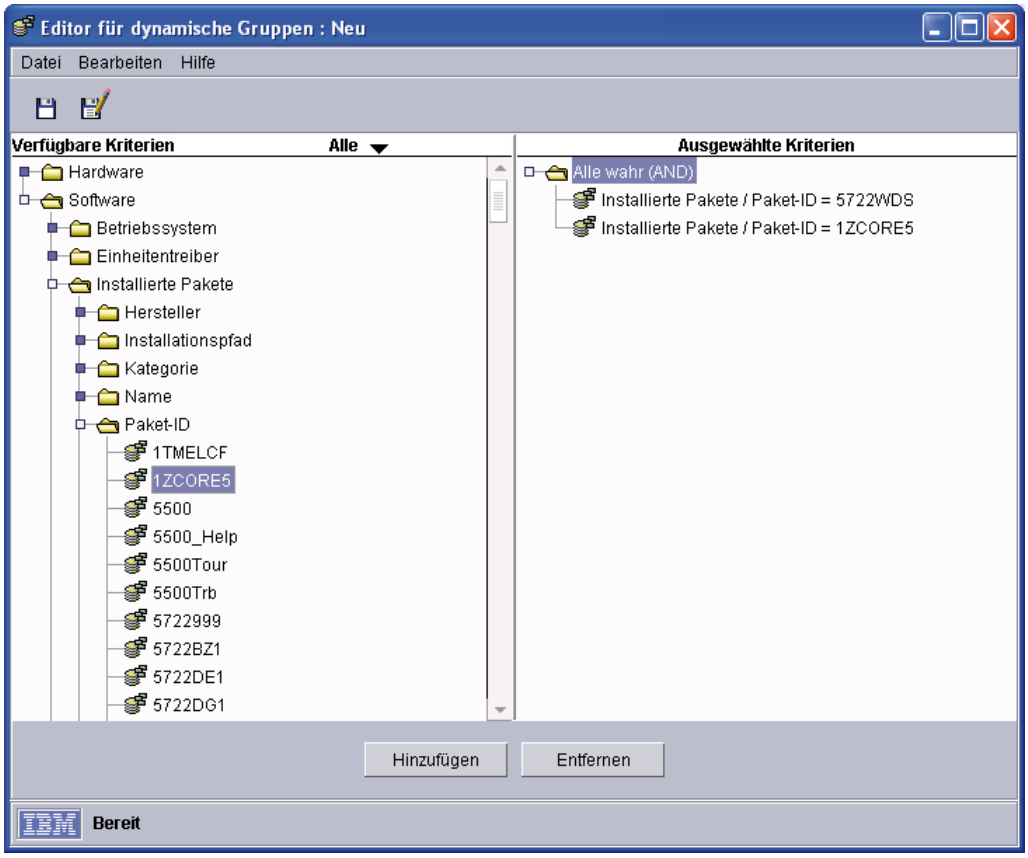
– Klicken Sie nun erneut mit der rechten Maustaste in den Spaltenbereich „Gruppen".

– Wählen Sie die Option „Neu (dynamisch)".

– Erweitern Sie den Eintrag „Software/Installierte Pakete/Paket-ID".

– IBM hat phantasievolle Namensvergaben – das Ergebnis dieses Eintrags liefert die installierten Lizenzprogramme auf dem System i.

– Selektieren Sie dort Eintrag „5722WDS" und klicken Sie auf die Schaltfläche „Hinzufügen".



Neue dynamische Gruppe

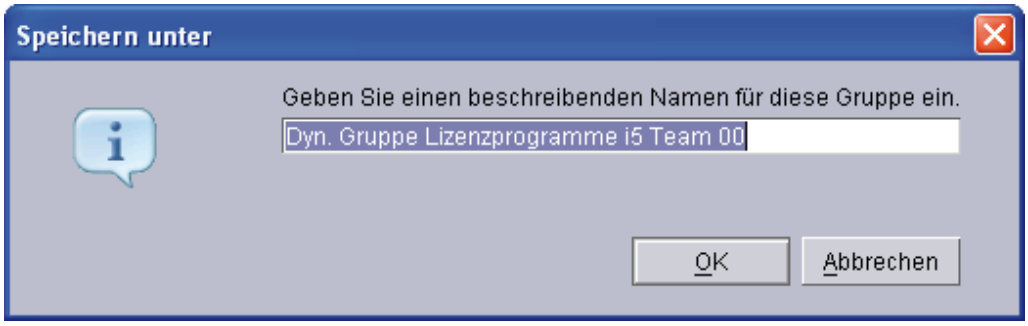
– Wählen Sie als zweiten Eintrag „5722XE1" aus und fügen Sie diesen dem Bereich „ausgewählte Kriterien" hinzu.

– Sie können entscheiden, ob die Bedingung eine „und" – bzw. „oder" – Bedingung darstellen soll. Selektieren Sie bitte „Und".

Damit legen wir fest, dass die Gruppe nur solche Systeme beinhalten soll, welche über die beiden installierten Lizenzprogramme verfügen.

4.3.4.3.2.2
Seite 2

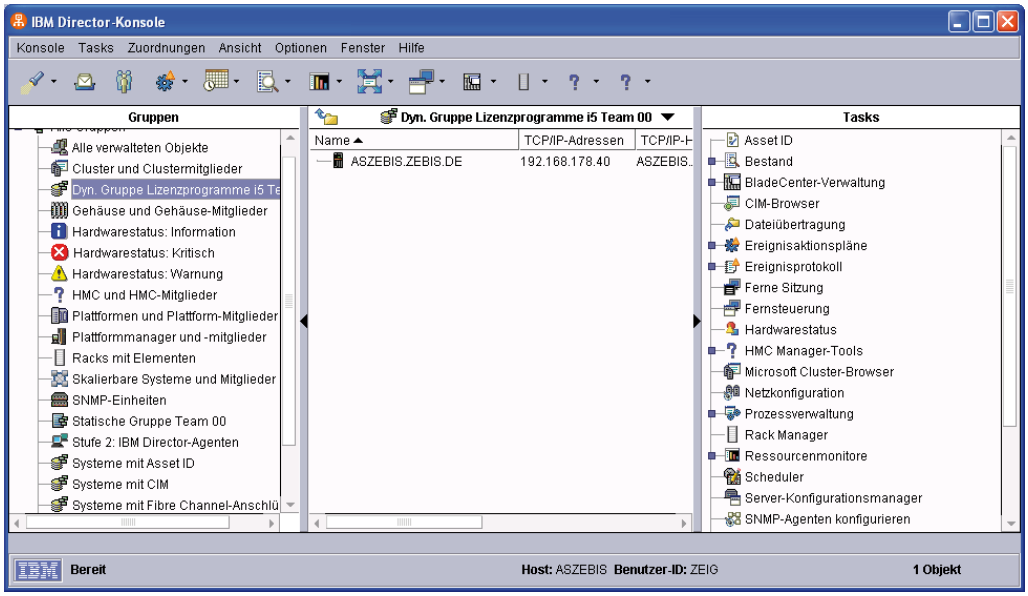
– Speichern Sie diese dynamische Gruppe unter dem Namen „Dyn. Gruppe Lizenzprogramme i5" ab.



Speichern unter

– Schließen Sie das Fenster „Editor für dynamische Gruppen".

– Schauen Sie sich nun in der Anzeige der IBM Director-Konsole den Bereich „Gruppen" an. Dort sind auch die beiden erstellten Gruppen enthalten.



Dynamische Gruppen

Damit haben wir eine Möglichkeit geschaffen, Systeme mit gleichen Kriterien zusammenfassend zu überwachen.

Zugriffssteuerung für IBM Director Server

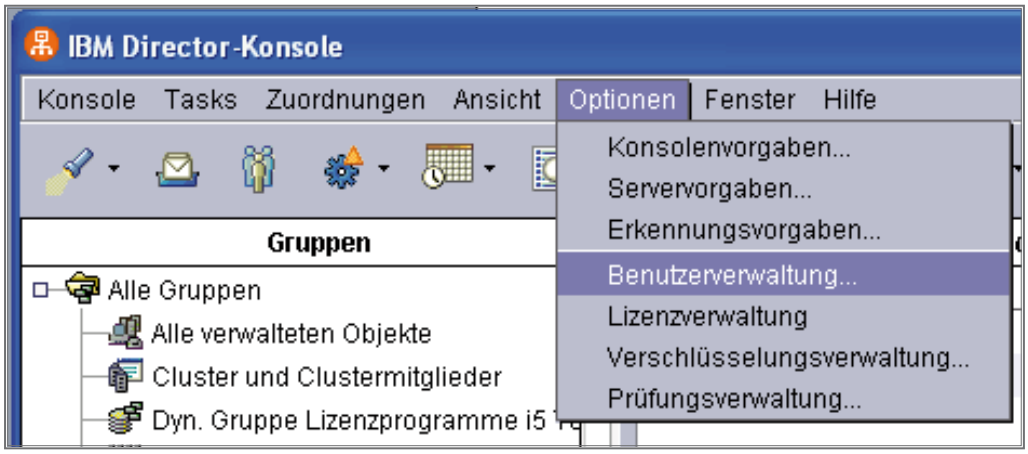
Der Einsatz des IBM Directors bedarf einiger Berechtigungen. Der Einsatz der IBM Director-Funktionen lässt sich darüber hinaus anpassen.

Die generellen Anpassungen für den Zugriff auf ein System i5 wird über die Berechtigungsvergaben im i5-Umfeld über 5250-Befehle bzw. den iSeries-Navigator durchgeführt. Bei dem Zugriff auf die Ressourcen des System i ist darauf zu achten, dass der Benutzer in der Gruppe der IBM Director Super Administratoren in der Lage ist, alle Bereiche des IBM Directors verwenden zu können. Einschränkungen lassen sich mittels Berechtigungsvergaben im IBM Director Umfeld definieren und steuern.

Beispielsweise kann in dem IBM Director unterbunden werden, dass bestimmte Benutzer einzelne Systeme ausschalten dürfen.

Nachfolgend lernen Sie die Berechtigungssteuerung im IBM Director grob kennen.

– Klicken Sie auf die Menüauswahl „Optionen/Benutzerverwaltung".



Benutzerverwaltung

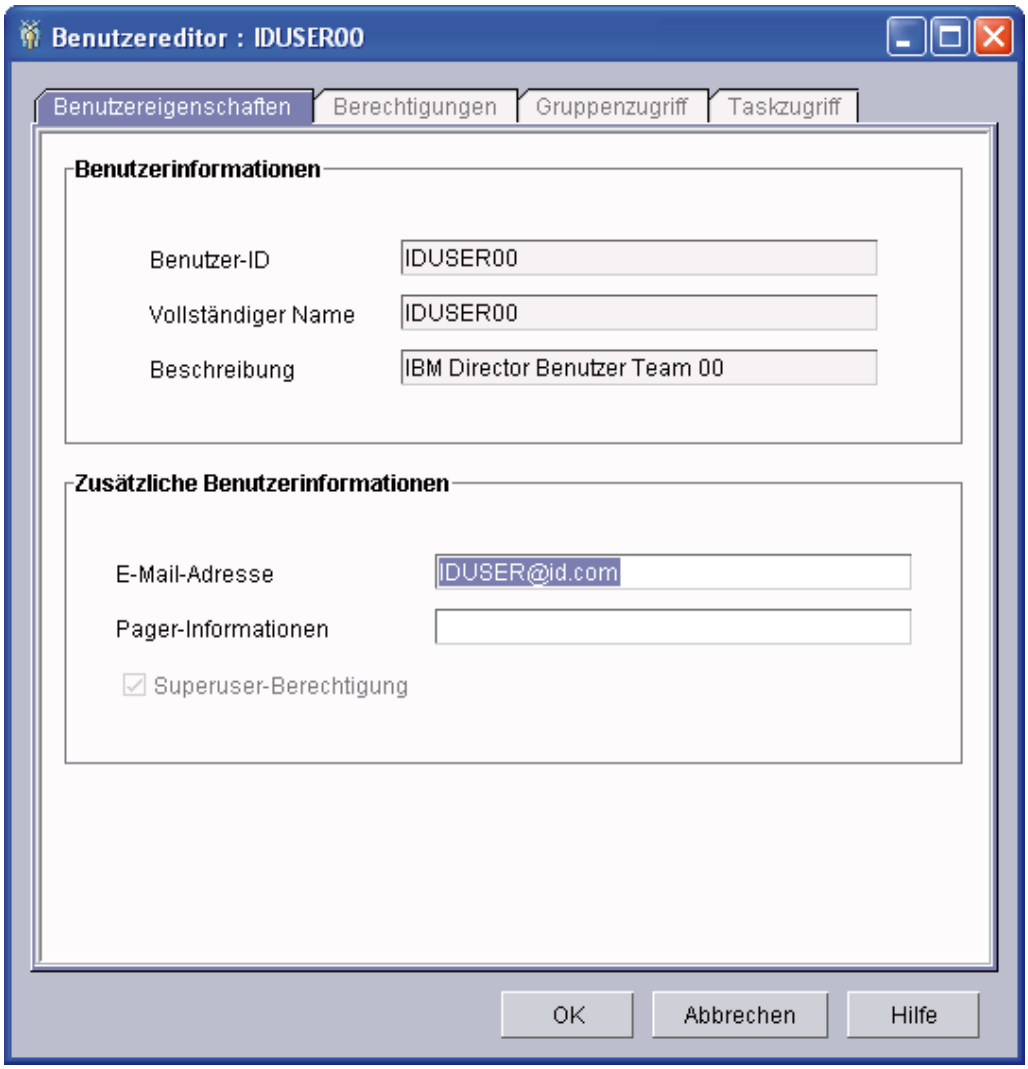
– Suchen Sie den Eintrag für Ihre Benutzer-ID und markieren Sie diesen mit der rechten Maustaste.

– Wählen Sie die Option „Bearbeiten".

Damit gelangen Sie in die Übersicht der Einstellungen für den Benutzer. Hier lassen sich die IBM Director-spezifischen Definitionen hinterlegen.

4.3.4.4
Seite 2

Da für spezielle Benachrichtigungen auch die Mailadresse von Bedeutung ist, kann diese in den allgemeinen Benutzerangaben verwaltet werden.



Benutzerdetails

– Selektieren Sie nun den Tabellenbereich „Berechtigungen".

In diesem Bereich finden Sie unterschiedliche Berechtigungsdefinitionen, welche Sie den einzelnen Benutzern oder Gruppen zuweisen bzw. entziehen können. Mit diesen Angaben lassen sich die Zugriffe der IBM Director Benutzer gezielt steuern.

5 Kapitel

Serverdienste / Webserver

5

Seite 2

Server

Denken wir einmal einige Jahre zurück: Es ist das Zeitalter der Twinax-Anbindungen. An die AS/400 sollen über „neue“ Verbindungstechnologien neben den guten alten Twinax-Bildschirmen auch PCs an die AS/400 (unsere heutige iSeries oder die neue i5) angebunden werden. Damals hielten erstmals Protokolle wie TCP/IP und auch SNA Einzug auf dem System. Eine der Voraussetzungen für die Anbindung dieser Clients sind besondere Server oder Server-Dienste auf der iSeries – denken wir hier an die Host-Server, TCP/IP Server etc. Zunächst von einigen AS/400-Betreibern als völlig unnütz und kaum brauchbar deklariert, ist diese Art der Anbindung externer Geräte heute längst Standard – und das trotz der zum Teil anfangs verhaltenen Akzeptanz dieser Technologien; dabei waren diese auf anderen Host-Systemen längst als quasi Standard etabliert.

Ein ähnliches Szenario scheint sich derzeit zu wiederholen:

Kennen Sie dies nicht auch: die Forderung nach „modernen“ Anwendungen, die uns helfen sollen, von den 5250-Darstellungen und den seit Jahren bewährten Systemen in eine neue, grafische und ergonomische Welt der Daten- und Informationsverarbeitung zu gelangen?

Diese Forderung nach einer ergonomischen Oberfläche hat einige Konzepte zur Folge gehabt, die die Lösung der Anforderungen auf unterschiedliche Arten und Weisen angehen. Dabei setzen viele dieser Lösungsansätze auf eine klare Trennung der Darstellungsschicht zur Logik- und Datenzugriffsschicht. Oft werden die 5250-Datenströme vor der Darstellung mit entsprechenden Werkzeugen abgegriffen, entsprechend umgesetzt und grafisch aufbereitet dem Endanwender als „neue“ Oberfläche zur Verfügung gestellt. Das ist sicher eine einfache Möglichkeit, ein neues Erscheinungsbild mit der unveränderten Logik der bisherigen Programmausführungen zu vereinen. Allerdings sind in den meisten Fällen allein für die Darstellungsebene bereits entsprechende Ausführungsumgebungen erforderlich, die die Basis für diese Tools darstellen.

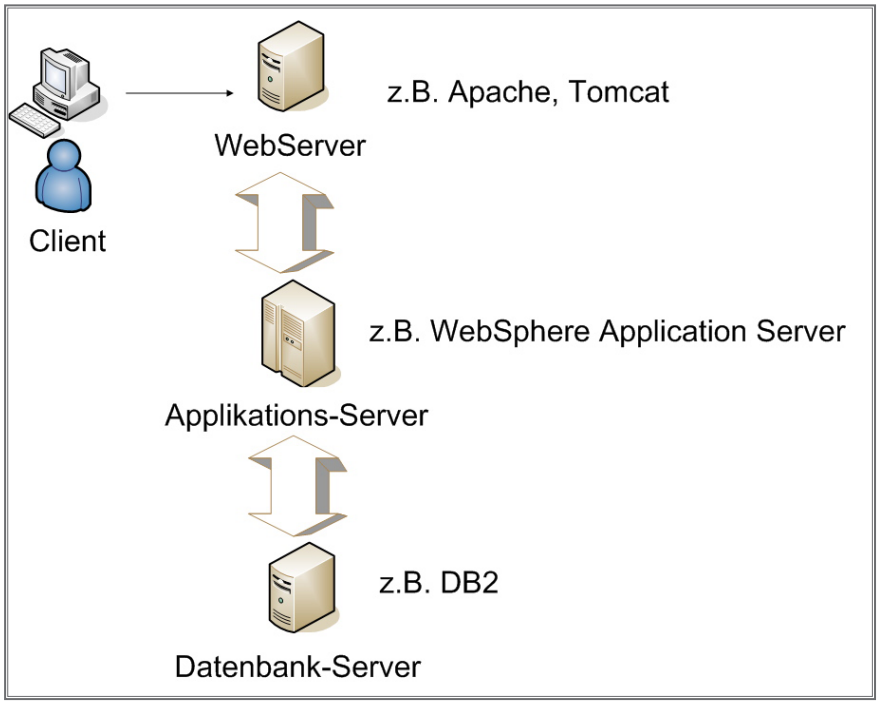
Es ist nicht allein der Ruf der iSeries-Anwender nach diesen neuen grafischen Anwendungen, sondern auch ein Trend der Entwicklung hin zur weitestgehend plattformunabhängigen Anwendung, die Informationen und Daten aus unterschiedlichen Systemen als Ziel hat und die sowohl entsprechende Umgebungen als auch Werkzeuge für die Darstellung und Realisierung bedeutet. Die Nutzung dieser Daten ohne Bindung an bestimmte Hardware – denken wir an die Darstellung von Informationen in Browsern – bestimmen mehr und mehr das tägliche Leben.

Der Grund dafür liegt in der Architektur der „modernen“ Anwendungen, die im Gegensatz zu den von der iSeries bekannten Ausführungsschichten eine strikte Trennung der einzelnen funktionalen Ausführungsebenen bedeuten.

Der Fachmann spricht hier von einem Schichtenmodell, bei dem die nachfolgenden Ebenen unterschieden werden:

- Darstellungsebene
- Logikebene
- Datenzugriffsebene

Die nachfolgende Abbildung soll dies verdeutlichen helfen:



Schema Server-Verwendung

Der Client stellt eine Anforderung in Form einer Datenanfrage. Dies kann zum Beispiel über eine spezielle Client-Software oder besser – weil unabhängig – über einen Web-Browser geschehen.

Diese Anforderung „Zeige mir Daten“ wird zunächst an einen so genannten Web-Server gesendet.

Dieser Web-Server übernimmt die Verarbeitung der Dialog-Anforderung, wertet diese aus und steuert unter anderem die weitere Verarbeitung.

Die Logik wird in der nächsten Schicht – dem Applikations-Server verarbeitet, der bei Bedarf auf die erforderlichen Informationen und Daten über den Datenbank-Server (das kann zum Beispiel die iSeries oder deren Datenbank DB2/400 sein) zugreift.

Ist doch eigentlich recht einfach – oder?

Aber ist diese Verarbeitungsart wirklich so gegensätzlich von der typischen iSeries Verarbeitung von klassischen RPG & Co.-Anwendungen? Kennen wir

dieses Schichtenmodell nicht auch in gewisser Weise bereits von der bisherigen Programmierung auf der iSeries?

Eine Darstellungsebene haben wir auch – die 5250-Bildschirmdarstellung – oder diverse SUI- und GUI-Derivate, die die Datenerfassung und Datenauswertung auf einem Bildschirm ermöglichen.

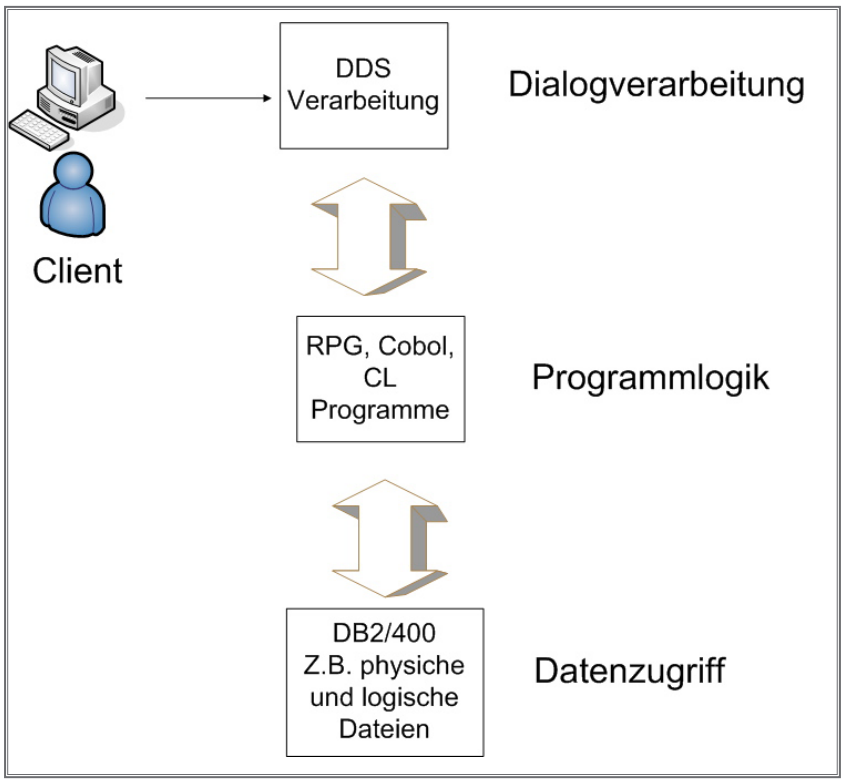
Die Logikebene befindet sich auf der iSeries in Form der RPG-, Cobol- und CL-Programme.

Die Datenzugriffsebene könnte durch SQL-Statements innerhalb der Programme beschrieben werden oder einfach über den Zugriff über die physischen und logischen Dateien.

Betrachtet man also die vielen neuen Begrifflichkeiten im Zusammenhang mit der Entwicklung neuer, vielleicht Browser- und Java-basierter Software, dann finden sich in der grundsätzlichen Struktur doch einige Parallelen zur bisherigen Verarbeitung wieder.

Wir wollen hier keine hinkenden Vergleiche schließen, denn in der Tat sind die Zugriffe und Werkzeuge nicht wirklich vergleichbar; oftmals helfen diese Eselsbrücken bei dem Verständnis der Strukturen der modernen Werkzeuge und Tools.

Schauen wir uns den bisherigen Ablauf eines einfachen Dialogprogramms einmal näher in Form einer Grafik an:

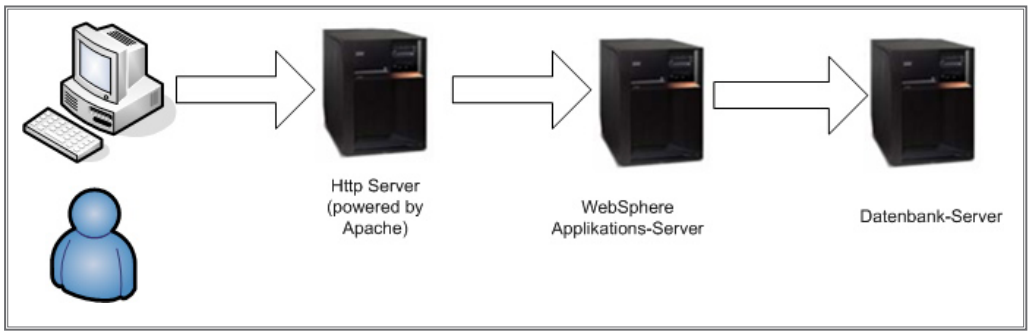


Ablaufschema „Klassisch“

Sie erkennen auch hier die drei Schichten:

- Darstellungsebene
- Logikebene
- Datenzugriffsebene

Und nun könnte ein solches Dialogprogramm folgendermaßen mittels der Verwendung eines Browsers und zum Beispiel einer Java-Logik grafisch dargestellt werden. Dabei wird die Ausführung über einen Applikations-Server gesteuert. In der nächsten Abbildung sehen sie – vereinfacht dargestellt – den Aufbau einer modernen Anwendung, ausgeführt auf einer iSeries:



Ablaufschema „Modern“

Eine Eigenschaft der „modernen“ Anwendungen liegt unter anderem darin, dass die Ausführung auf unterschiedlichen Applikations-Servern, Web-Servern und Datenbank-Servern aufbaut. Jede dieser Server-Ebenen für sich bietet die Möglichkeit, durch ihre Verfügbarkeit auf unterschiedlichen Hardware-Plattformen die Bindung der Anwendung auf eine einzige Hardware-Plattform zu eliminieren. Der Einsatz dieser Anwendungen ist damit weitestgehend plattformunabhängig möglich.

Voraussetzung für die Ausführung sowohl der „modernen“ Anwendungen auf der iSeries als auch dem Zugriff via Browser sind einige Server-Varianten, die wir in diesem Kapitel näher behandeln wollen.

Diese Server werden zunächst grob in folgende Klassen aufgeteilt:

- Web-Server
- Applikations-Server
- Datenbank-Server

Der wesentliche Unterschied zwischen diesen einzelnen Server-Klassen liegt in dem Funktionsumfang und dem Einsatzgebiet. Während Web-Server in der Regel für den Einsatz von statischen Web-Seiten ausgelegt sind (dies sind solche Web-Seiten, deren Inhalt sich nicht ändert und die einen gleich bleibenden Aufbau haben), sind die Applikations-Server in der Lage, dynamische Web-Seiten zu verwalten. Bei einer dynamischen Web-Seite können sich

die Inhalte der Web-Seite – also der Anzeige – verändern. Dabei sind zum Teil komplexe Datenzugriffe und Datenverwaltungen erforderlich, die über die Funktionen des Applikations-Servers realisiert werden können.

Betrachten wir heute diese Server, dann werden wir feststellen, dass es eine ganze Reihe unterschiedlicher Server oder auch Server-Dienste gibt, die für die unterschiedlichsten Anforderungen mehr oder weniger geeignet sind. Im Zusammenhang mit der iSeries können im Allgemeinen derzeit (Stand V5R3) die nachfolgenden Server genannt werden:

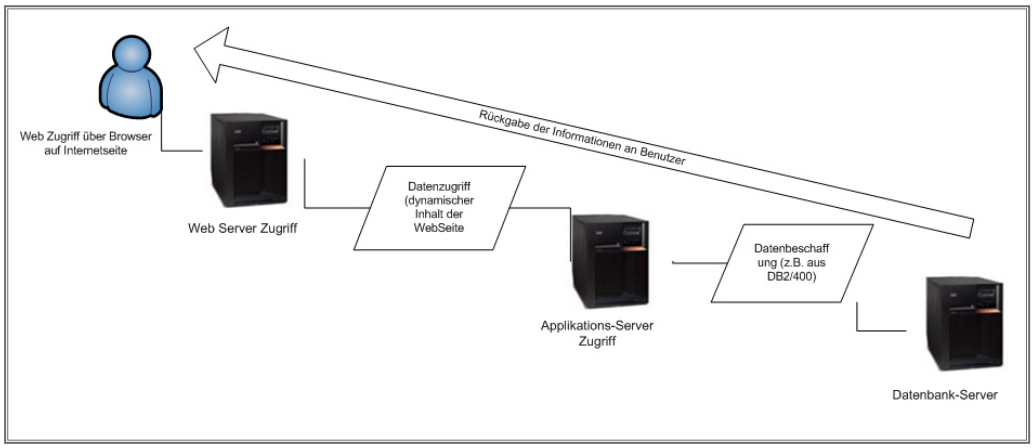
- WebSphere Applikation Server-Familie
- Domino-Server
- HTTP-Server

Um die Abläufe im Zusammenhang mit diesen Servern ein wenig zu verdeutlichen, schauen wir uns zunächst das nachfolgende Beispiel kurz an. In diesem Beispiel gehen wir davon aus, dass ein Benutzer via Internet auf eine Web-Seite zugreift.

Stellen wir uns vor, wir wollen in einem Web-Shop einen bestimmten Artikel kaufen. Also gehen wir zunächst auf die Einstiegsseite von unserem Web-Shop. Hier erscheint neben einigen Informationen unter anderem ein Eingabefeld, in dem ein Suchbegriff eingegeben werden kann. Die Darstellung der Web-Seite inklusive des Eingabefeldes kann über einen einfachen Web-Server realisiert werden. Ein solcher Web-Server auf der iSeries wäre der „HTTP-Server powered by Apache“.

Sobald der Suchbegriff nach dem gewünschten Artikel eingegeben worden ist, sind in der Regel die Funktionen des Web-Servers erschöpft. Wird dieser Suchbegriff eingegeben, dann muss aus der Fülle der vorhandenen Daten auf einem Datenbank-Server ein Extrakt der Informationen ermittelt werden, die der Benutzer angefordert hat. Diese Informationsbeschaffung wird von dem Web-Server zur weiteren Verarbeitung an einen Applikations-Server übergeben. Auf der iSeries haben wir zum Beispiel den WebSphere-Application-Server, der diese Aufgaben übernimmt. Dieser führt die entsprechenden Datenbankzugriffe aus und gibt die Ergebnisse der Zugriffe entweder direkt oder über den Web-Server wieder zurück an den Benutzer.

Grafisch könnte dies so aussehen:



Ablauf dynamischer Zugriff

Die zuvor genannten Server sind weitestgehend plattformunabhängige Ausführungsebenen für die unterschiedliche Ebene der Informationstechnologien und gerade auch auf den Nicht-iSeries-Plattformen seit langem feste Bestandteile der IT.

Bei dem Einsatz solcher Server hinkt die iSeries-Gemeinde den übrigen eServer-Plattformen derzeit noch deutlich hinterher. Denn neben zSeries und pSeries sind diese Server gerade auch auf den zSeries schon besser akzeptiert und auch eingesetzt.

Doch was versteht man im Detail unter diesen Servern? Wozu werden diese benötigt und wie wählt man unter der Vielzahl der unterschiedlichen auf dem Markt vorhandenen Derivate von Servern den oder die Server aus, mit denen die Anforderungen optimal gelöst werden können?

In dem nachfolgenden Kapitel wollen wir versuchen, diese Fragen zu beantworten und einen Einblick in den Aufbau, die Funktionen und die Administration zu gewähren. Nicht zuletzt wollen wir versuchen, die Vor- und Nachteile der Server aufzuzeigen und mit den Tipps und Tricks mögliche Stolperstellen im Vorfeld aufzeigen. Dabei wollen wir uns bewusst auf die derzeit von IBM als „strategische Produkte“ deklarierten Server-Versionen konzentrieren. Sicher können im Zusammenhang mit der Integration von iSeries-Daten – also DB2-Daten – auch weitere Server verwendet werden, die zwar nicht auf der iSeries ausgeführt werden, aber deren Daten benutzt werden können. Diese sollen aber nicht Gegenstand dieses Kapitels sein.

Bei den in diesem Kapitel besprochenen Servern wird davon ausgegangen, dass diese auch auf der iSeries installiert sind oder zumindest installiert werden können.

IBM hat in der Vergangenheit einige unterschiedliche Server auf den AS/400- und iSeries-Systemen integriert. In den letzten Jahren hat sich der Trend zur Vereinheitlichung und Standardisierung in diesen Bereichen stark fortgesetzt, so dass die strategische Ausrichtung der Server-Dienste klar in Richtung der in den nächstem Kapitel behandelten Server geht. Weil in absehbarer Zeit der Einsatz der Applikations-Server und der übrigen Server im Web- und HTTP-Bereich auf der iSeries wohl auf die strategischen Server-Produkte zusammengestrichen wird, werden wir uns auf diese Server beschränken.

Ich hoffe, dass Ihnen dieses nachfolgende Kapitel den Umgang mit den unterschiedlichen Server-Varianten erleichtern hilft oder auch die Entscheidung pro oder contra den Einsatz des einen oder anderen Servers erleichtert.

Grundwerk 10/2004

5

Seite 10

Web-Server

Mit dem Erscheinen des Internets mussten neben der Client-Software in Form von Browsern, einem gemeinsamen Protokoll (TCP/IP) auch entsprechende Voraussetzungen geschaffen werden, damit die Vielzahl der Benutzer auf die unterschiedlichsten Informationen im Netz zugreifen können. Eine dieser Voraussetzungen liefern die so genannten Web-Server.

Diese Web-Server sind Server, die im Internet oder auch innerhalb eines Unternehmens im Intranet eingebunden sind und die Web-Seiteninhalte, Daten und andere Dienste speichern und diese dann über HTTP den jeweiligen Clients zur Verfügung stellen.

Wer also Informationen im Internet (oder auch Intranet) zur Verfügung stellen möchte, benötigt in seinem Netz oder dem Netz eines Providers eine entsprechende Plattform oder genauer gesagt einen Server, der die Anfragen der Clients und deren Browser bearbeiten kann. Diese Funktionen und Aufgaben werden von den Web-Servern (oder auch HTTP-Servern) übernommen. Diese stellen somit eine Basis für den Datenzugriff mittels zum Beispiel eines Browsers dar.

Heute existiert eine Vielzahl unterschiedlicher Web-Server, die auf den unterschiedlichsten Plattformen eingesetzt werden können. Allerdings variieren die Eigenschaften dieser Server zum Teil erheblich. Bei den Eigenschaften der unterschiedlichen Server wird zum einen Wert auf die Verarbeitung von Indizes gelegt (für schnelle Zugriffe); bei einem anderen Server wird Wert auf große Zugriffssicherheit (z.B. Banken-Software) gelegt.

Die meisten Web-Server basieren auf den Konzepten und Standards, die ehemals von dem europäischen Labor für Teilchenphysik in Genf (CERN) festgelegt wurden. Diese wurden von dem W3-Konsortium und von der NCSA weiterentwickelt und stellen die heutigen Standards dar.

Schauen wir uns einige dieser „Standard-Web-Server“ an:

- CERN httpd 3.0
- NCSA httpd 1.4
- Apache
- Netscape Commerce Server
- WN
- …

Die wichtigste Voraussetzung für die Ausführung eines Web-Servers ist das Protokoll TCP/IP. Dieses Protokoll ist für den Transport von Informationen im Internet und Intranet zuständig.

5.1
Seite 2

Web-Server sind auf den unterschiedlichsten Plattformen verfügbar. Auf der iSeries haben diese erst in den letzten Jahren langsam Einzug gehalten. Betrachtet man die Ausrichtungen der neuen Anwendungen, die sich in den letzten Jahren auf dem Markt etablieren konnten, dann wird man schnell feststellen, dass diese in vielen Fällen eine Basis haben, die für die Ausführung dieser Anwendungen erforderlich ist: nämlich einen Web-Server.

In dem nachfolgenden Kapitel wollen wir uns nur mit den Web-Servern beschäftigen, die direkt auf der iSeries (im OS/400) installiert und ausgeführt werden können.

Die Server-Installationen, die in einem Netzwerk oder zum Beispiel in einer Linux-Partition auf der iSeries durchgeführt werden können, sollen nicht Gegenstand dieses Kapitels sein.

IBM hat in den letzten Jahren einige Web-Server auf der iSeries implementiert. Neben Produkten von Fremdanbietern hat IBM einen eigenen Web-Server für den Einsatz auf der AS/400 und iSeries entwickelt.

Auf der iSeries sind derzeit (noch) folgende HTTP-Server verfügbar:

- HTTP Server (powered by Apache) – das ist das strategische Produkt der IBM
- HTTP Server (Original)
- WebSphere Application Server
- Domino Server
- Tomcat 4.1 HTTP-Server
- i/net HTTP Server

Doch welchen dieser Web-Server sollte man einsetzen? Welche Vor- oder Nachteile haben die einzelnen Web-Server?

Diese Fragen müsste man sich sicher sofort stellen, wenn man die Vielzahl der unterschiedlichen Web-Server, die direkt auf der iSeries ausgeführt werden können, betrachtet. An dieser Stelle sei Folgendes anzumerken. Durch die Architektur der Web-Server und deren Einbindung in moderne Anwendungen muss der Web-Server nicht auf der iSeries installiert sein, um diesen in Verbindung mit iSeries-Anwendungen und Daten nutzen zu können. Aus diesem Grund wäre die zuvor genannte Auflistung der auf der iSeries verfügbaren Web-Server noch um die Web-Server zu erweitern, die auf allen übrigen Server-Plattformen verfügbar sind. Es existiert eine Vielzahl von unterschiedlichen Servern auf den verschiedenen Plattformen! Sicher können diese Web-Server, die nicht direkt auf der iSeries ausgeführt werden, die Daten und Informationen der iSeries nutzen, denn der Informationstransport erfolgt in der Regel über das Protokoll TCP/IP und macht die Installation plattformunabhängig. Diese Formen der Web-Server, die nicht direkt auf der iSeries

installiert und ausgeführt werden, sind aber nicht Gegenstand dieses Kapitels.

Die Entscheidung bezüglich des auf der iSeries einzusetzenden Web-Servers hat IBM den Anwendern bereits weitestgehend abgenommen:

Aufgrund verschiedener Aspekte wurde in den letzten Jahren die Entwicklung an dem IBM-eigenen Web-Server (HTTP-Server Original) jedoch weitestgehend eingestellt und die Unterstützung dieses Web-Servers auf der iSeries mittlerweile aufgekündigt.

Einer der Gründe liegt in der Konkurrenz aus dem eigenen Hause: Denn durch den Zukauf von Lotus hat man neben dem eigenen „Original HTTP-Server" einen weiteren HTTP-Server in Form des Domino-HTTP-Servers eingekauft. Dieser Server, der im Zuge einer Installation von Domino auf der iSeries ebenfalls genutzt werden kann, wäre zusätzlich zu der oben genannten Auflistung an HTTP-Servern zu nennen!

Verwunderlich – aber auch dieser Domino-HTTP-Server wurde nicht als neues strategisches Produkt der IBM im Bereich der Web-Server bestimmt. Vielmehr hat man sich dazu entschlossen, den „HTTP-Server powered by Apache" als Standard für die zukünftige Entwicklung in diesem Bereich zu definieren.

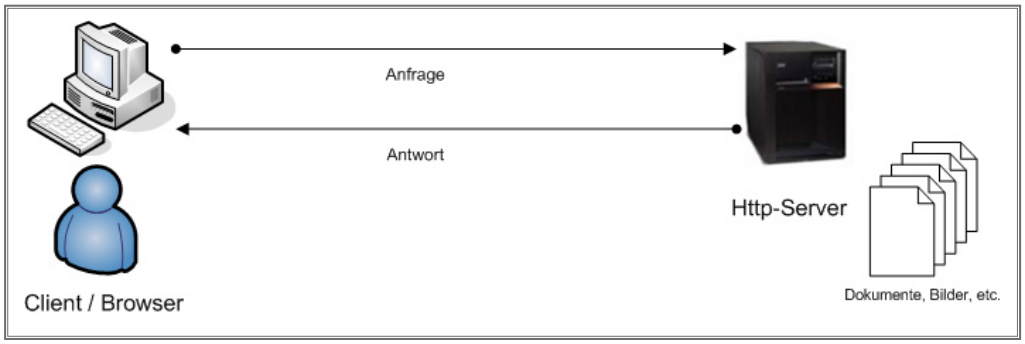
Aus diesem Grund werden wir uns in dem nachfolgenden Kapitel vornehmlich mit dem „HTTP-Server powered by Apache" beschäftigen.

Was macht nun ein solcher Web- oder HTTP-Server?

Wir kennen es eigentlich alle: das Internet.

Die größte Verbreitung oder auch Anwendung haben Web-Server durch den Einsatz des Internets bekommen. Zusätzlich finden Web-Server in den letzten Jahren Anwendung in den Bereichen der Intranets und hier speziell beim Einsatz von modernen Anwendungen. Diese basieren meist auf einem Schichtmodell, in dem eine Ebene die Darstellungsebene ist. Diese wird meist durch den Web-Server verarbeitet. Dabei werden von einem Client – oder genauer von einem Browser – Informationen zur Weiterverarbeitung an den Web-Server gesendet. Dieser wertet diese Informationen aus und gibt ein Ergebnis einer Verarbeitung zurück. Dabei kann je nach Definition das Ergebnis direkt von dem Web-Server ermittelt werden. Oder es kann bei der Ausführung von komplexen Programmen oder Datenzugriffen die Weiterverarbeitung der Informationen durch einen zusätzlich konfigurierten Applikations-Server ausgeführt werden.

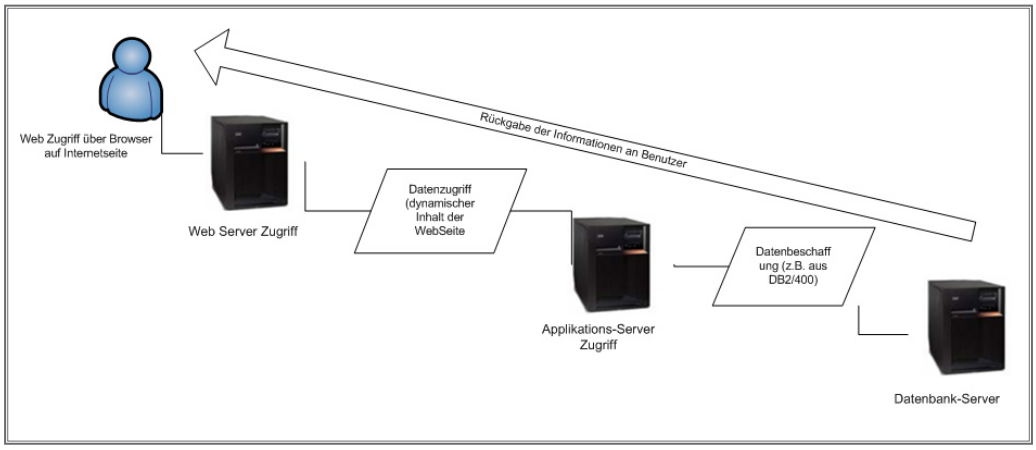
Bei einem einfachen Zugriff auf statische Informationen könnte ein solcher Ablauf grafisch wie folgt dargestellt werden:



HTTP-Server Zugriff

Ein Beispiel finden wir auf vielen einfachen Web-Seiten wieder, die keine veränderbaren Informationen enthalten.

Sobald sich Informationen innerhalb einer Web-Seite ändern, ist meist der Einsatz eines Web-Servers allein nicht mehr angebracht. In einem solchen Fall werden häufig zusätzliche Werkzeuge verwendet, die zum Beispiel in Form eines Applikations-Servers eingesetzt werden können:



Ablauf dynamischer Zugriff

Als grobe Funktionsbeschreibung eines Web-Servers könnte die folgende Auflistung dienen:

- Ausführen von Programmen
- Zugriff auf Daten
- Zugriffsschutz
- Zugriffskontrollen
- Speichern und Bereitstellen von Informationen
- HTML-Seiten
- Textdokumente
- Dynamische, datenbankbasierende Informationen

HTTP Server powered by Apache

Mit dem HTTP-Server „powered by Apache“ hat IBM die hauseigenen Produkte im Bereich der Web-Server (z.B. „IBM HTTP-Server Original“ und Domino-HTTP-Server) zugunsten dieses Servers quasi degradiert. Gründe für dieses Vorgehen sind sicher nur spekulativ zu suchen. Sicher ist aber, dass der Schritt der IBM, sich auf einen der am meisten verbreiteten Web-Server zu orientieren, durchaus sinnvoll ist.

Seit der Version V5R1 ist der „HTTP-Server powered by Apache“ der „strategische“ Web-Server der IBM und stellt somit die Basis für viele Neuentwicklungen und Informationssysteme dar. Einige der neuen Anwendungen, die auf der iSeries integriert sind (denken wir einmal an die Konfiguration der WebSphere Applikations-Server), benötigen für die Administration diesen Web-Server.

Was ist eigentlich dieser „HTTP-Server powered by Apache“?

Zunächst einmal ist die Entstehung des Namens ein wenig zum Schmunzeln.

Die Basis, von der sich neben vielen anderen Web-Servern auch die Entwicklung dieses HTTP-Servers ableiten lässt, war der NCSA-Server mit der Version 1.3. Dieser Server kann wohl als einer der Ur-Web-Server bezeichnet werden. Wie viele „Ur-Produkte“, so hatte auch dieser Web-Server in seinen Anfangszeiten einige Fehler, die entsprechend durch Programmkorrekturen behoben worden sind. Diese Korrekturen – englisch: „patches“ – geben dem HTTP-Server „powered by Apache“ seinen Namen: Denn „Apache“ wäre eigentlich korrekt als „a patch“ zu schreiben…

Der „HTTP-Server powered by Apache“ ist kostenlos und auf den meisten gängigen Hardware-Plattformen verfügbar. Die Entwicklung an diesem Server basiert auf einem „Open-Source-Projekt“, das seit 1995 ständig weiterentwickelt wird und unterstützt unter anderem auch den Standard HTTP/1.1.

Auf der iSeries ist der „HTTP-Server powered by Apache“ für die folgenden Betriebssystemversionen verfügbar:

- V4R5
- V5R1
- V5R2
- V5R3

Neben den unterschiedlichen Voraussetzungen auf der iSeries muss jeder PC, der mit dem HTTP-Server arbeiten soll, über einen Browser verfügen, der den jeweiligen HTTP-Standard des Servers unterstützt. Ein eingesetzter Browser sollte die nachfolgenden Anforderungen abdecken:

- HTTP 1.0 oder 1.1-Unterstützung
- Unterstützung von Frames
- Unterstützung von Java Script

Als gängigste Browser können in diesem Zusammenhang genannt werden:

- Microsoft Internet Explorer ab 5.5
- Netscape Navigator ab 4.75
- Domino HTTP-Server

Um bestimmte Protokolleinträge, die durch den HTTP-Server erstellt werden, auswerten zu können, benötigt der Browser außerdem eine Unterstützung der Java Virtual Machine 1.1.5 oder höher.

Die Installation des „HTTP-Server powered by Apache“ erfolgt nicht – wie bei anderen Produkten meist üblich – in Form eines Lizenzprogramms, sondern durch das Einspielen bestimmter Gruppen-PTFs, die in Verbindung mit weiteren Lizenzprogrammen, die auf der iSeries installiert sein müssen, eingesetzt werden.

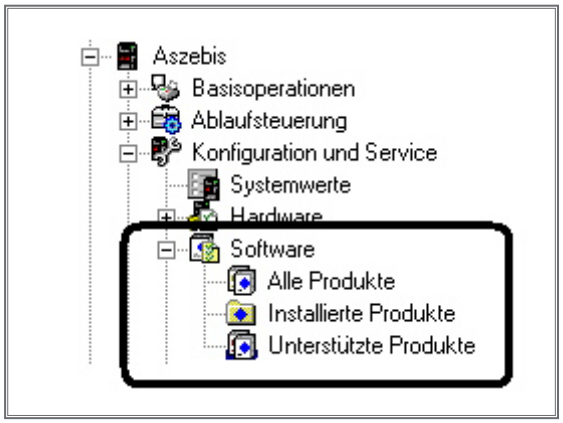
Bevor diese PTFs eingespielt werden, müssen neben einer der zuvor genannten Betriebssystem-Versionen die nachfolgenden Voraussetzungen erfüllt sein:

Erforderliche Lizenzprogramme

- 5722-DG1 HTTP Server for iSeries
- 5722-TC1 TCP/IP
- 5722-JV1 Java Dev. Kit 1.3
- Optionen 3 und 4
- 5722-AC3-Verschlüsselung
- Diese Option kann wahlweise installiert werden!

Die Prüfung, ob diese Lizenzprogramme installiert sind, kann wahlweise über einen 5250-Befehl (GO LICPGM – Auswahl 10) oder durch den iSeries Navigator durchgeführt werden.

In dem iSeries Navigator befinden sich diese Informationen unterhalb des Servers in dem Eintrag „Konfiguration und Services“ / „Software“ / „installierte Software“.



Lizenzprogramme unter iSeries Navigator

Aszebis: Installierte Produkte

Produkt	Option	Release	Beschreibung
5722de1	0002	V5R2M0	DB2 UDB XML Extender
5722de1	0003	V5R2M0	Steuerkomponente für Textsuche
5722dg1	0000	V5R2M0	IBM HTTP-Server für iSeries
5722dg1	0001	V5R2M0	Triggered Cache Manager
5722dp4	0000	V5R2M0	DataPropagator Relational für AS/400
5722ip1	0000	V5R2M0	IBM Infoprint Server für iSeries
5722iwe	0000	V5R1M0	WebSphere Application Server - Express
5722iwe	0002	V5R1M0	IBM WebSphere Application Server Express V5
5722jc1	0000	V5R2M0	IBM Toolbox für Java
5722js1	0000	V5R2M0	Advanced Job Scheduler für iSeries
5722jv1	0000	V5R2M0	IBM Developer Kit für Java
5722jv1	0003	V5R2M0	Java Developer Kit 1.2
5722jv1	0004	V5R2M0	Java Developer Kit 1.1.8
5722jv1	0005	V5R2M0	Java Developer Kit 1.3
5722jv1	0006	V5R2M0	Java Developer Kit 1.4

Installierte Lizenzprogramme

Fehlt ein Lizenzprogramm (oder mehrere) dann müssen diese vor der Installation des HTTP-Servers auf dem System installiert werden. Dazu sind die Installations-CDs des entsprechenden Betriebssystems erforderlich.

Die Installation fehlender Lizenzprogramme kann dann in dem Menü „GO LICPGM“ durchgeführt werden.

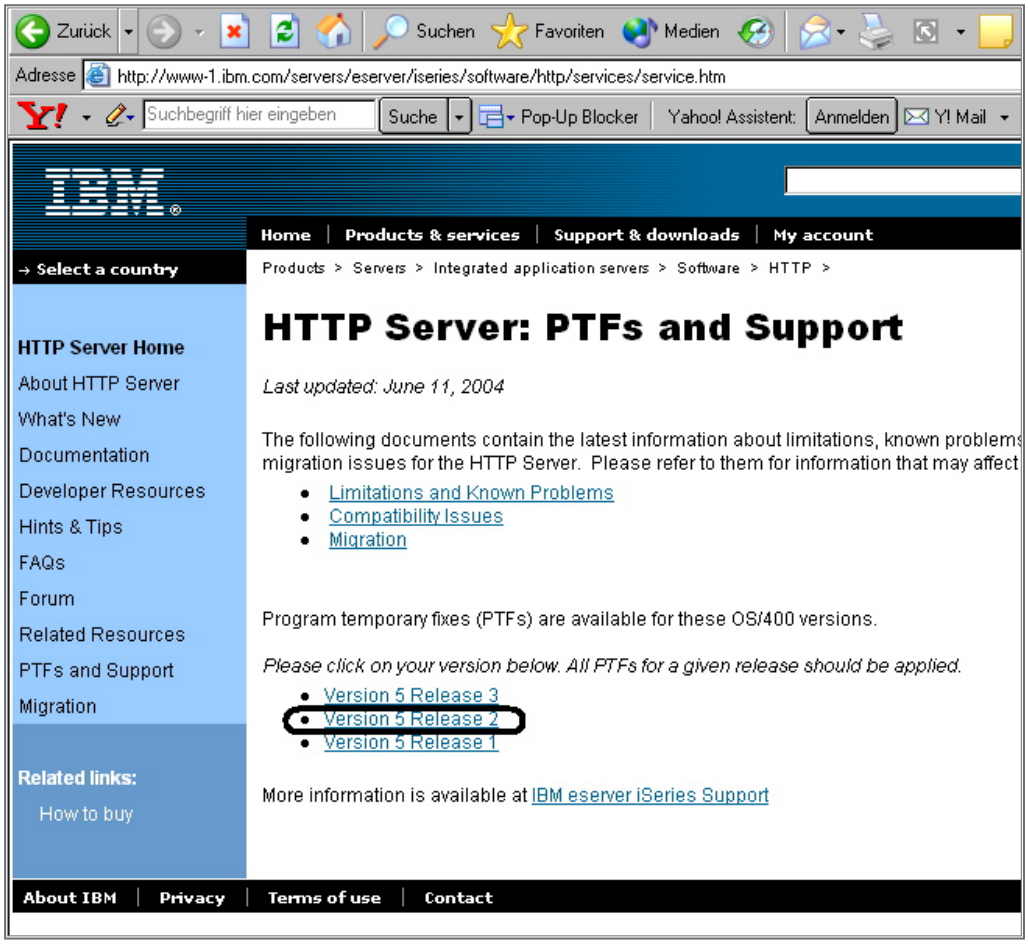
Dort wird mit der Auswahl „11“ die Liste aller installierbaren Lizenzprogramme angezeigt, aus der dann das oder die zu installierenden Lizenzprogramme für die Installation ausgewählt werden können. Meist wird für die Installation die von der IBM gelieferte Installations-CD verwendet. Eine Detailbeschreibung der Installation der erforderlichen Lizenzprogramme würde den Rahmen dieses Abschnittes überschreiten, deshalb wird hier auf die entsprechenden Grundkenntnisse bezüglich der Standard-Installationsverfahren für Lizenzprogramme verwiesen. Wir gehen in dem folgenden Bei-

5.1.1
Seite 4

spiel davon aus, dass alle erforderlichen Lizenzprogramme bereits auf dem System installiert sind.

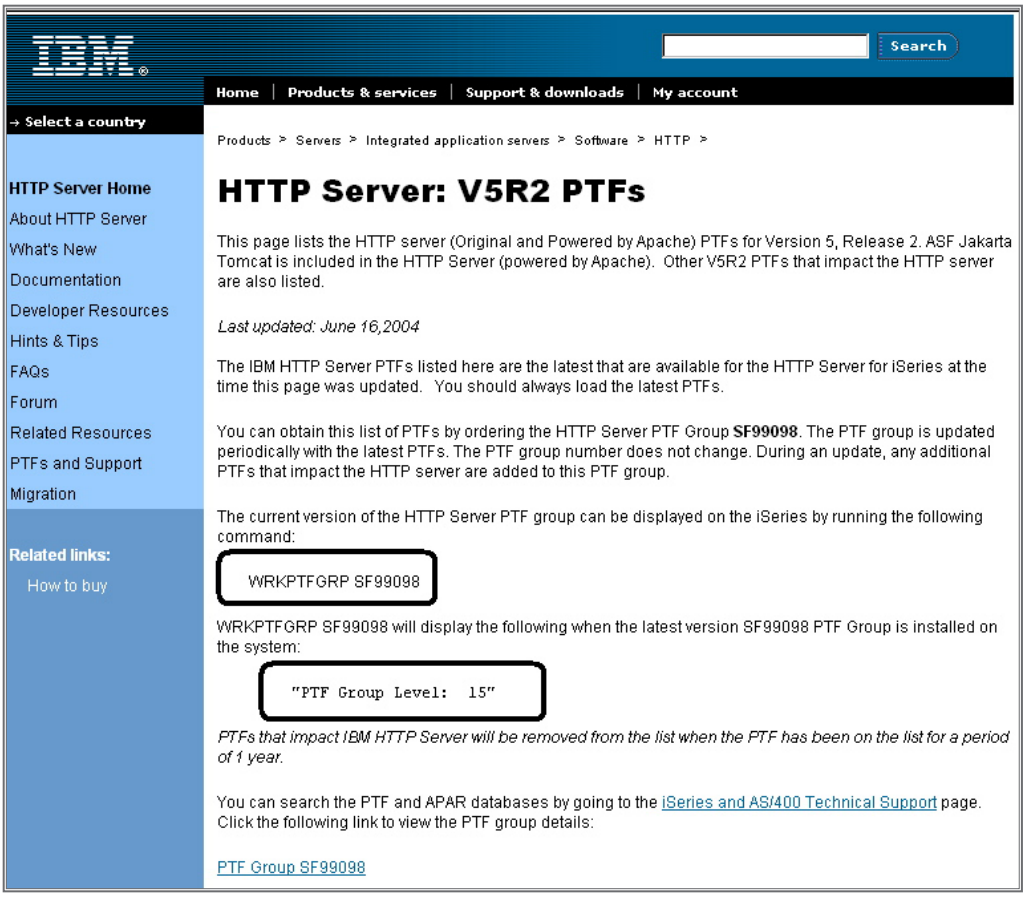
Bevor mit der Installation des Web-Servers begonnen werden kann, sollte in jedem Fall der aktuell auf dem System installierte PTF-Stand überprüft werden. Welcher PTF-Stand für eine erfolgreiche Installation und spätere sichere Ausführung des HTTP-Servers erforderlich ist, finden Sie auf der folgenden Web-Seite:

HTTP://www-1.ibm.com/servers/eserver/iseries/software/HTTP/services/service.htm



Erforderliche PTF

In dem folgenden Beispiel wird die Installation des Web-Servers auf einer iSeries 270 mit der Betriebssystemversion V5R2M0 durchgeführt. Deshalb wählen wir auf der Web-Seite den Eintrag „Version 5 Release 2“ aus, um zu sehen, welche PTFs für diese Betriebssystemversion erforderlich sind. Die erforderlichen Informationen befinden sich auf der Folgeseite der „HTTP-Server PTF und Support“-Seite:



Erforderliche PTF V5R2

Die Gruppen-PTFs für den HTTP-Server-Bereich sind für die OS/400 V5R2-Version unter „SF99098“ zu finden.

(Für V5R1M0 lautet die PTF-Nummer für den HTTP-Server: SF99156.)

Auf der Internet-Seite in der Abbildung oben ist als aktueller Gruppen-PTF-Stand der Level 15 angegeben. Dieser sollte als aktueller Stand auf der iSeries installiert werden, um alle Funktionen und auch Modifikationen des „HTTP-Servers powered by Apache“ nutzen zu können.

Anmerkung:

Der Grund für neue PTF-Auslieferungen liegt nicht immer allein in der Behebung von Fehlern in einem Produkt, sondern IBM ist in den letzten Jahren auch immer mehr dazu übergegangen, neue Funktionen in den einzelnen Produkten und Lizenzprogrammen durch PTFs verfügbar zu machen. Um also den vollen, aktuellen Funktionsumfang eines Produktes nutzen zu können, sollte der aktuelle PTF-Stand installiert sein.

Die Prüfung des auf der iSeries aktuell installierten Gruppen-PTF-Standes kann seit der Betriebssystemversion V5R2 mit dem neuen Befehl „WRKPTFGRP“ durchgeführt werden.

Geben wir diesen Befehl gefolgt von dem Gruppen-PTF (für den HTTP-Server ist das die Nummer: „SF99098“) ein:

```
WRKPTFGRP SF99098
```

Das Ergebnis wird wie folgt angezeigt:

```
                              Mit PTF-Gruppen arbeiten

Auswahl eingeben und Eingabetaste drücken.
  4=Löschen    5=Anzeigen    6=Drucken    9=Zugehörige

Ausw  PTF-Gruppe              Stand  Status
 =    SF99098                    13  Installiert
```

Gruppen-PTF anzeigen

Alternativ kann der aktuelle Stand der einzelnen Gruppen-PTFs mit Hilfe des iSeries Navigators ermittelt werden:

Aszebis: Fix-Gruppen

Fix-Gruppe	Ebene	Status	Beschreibung
Sf99039	2	Installiert	PERFORMANCE TOOLS FOR ISERIES PTF ...
Sf99085	5	Installiert	520 BACKUP RECOVERY SOLUTIONS
Sf99098	13	Installiert	IBM HTTP SERVER FOR ISERIES
Sf99149	9	Installiert	WAS - ADVANCED 4.0.7 5733WS4
Sf99166	1	Installiert	SF99166 - ISERIES CONNECT 2.0 PTF GRO...
Sf99169	15	Installiert	JAVA PTF GROUP
Sf99172	1	Installiert	IBM BUSINESS SOLUTIONS
Sf99232	3	Installiert	WEBSPHERE COMMERCE SUITE V5.4 FOR ...
Sf99245	5	Installiert	WEBSPHERE APP SERVER V5.0.2 5733WS5
Sf99246	4	Installiert	WEBSPHERE APP SERVER V5.0.2 ND 573...
Sf99271	12	Installiert	WEBSPHERE APP SERVER-EXPRESS V5.0....
Sf99502	10	Installiert	DB2 UDB FOR ISERIES
Sf99519	90	Installiert	GROUP HIPER

Gruppen-PTF im iSeries Navigator

Wie wir sehen, ist im Beispiel der zuvor gezeigten Abbildungen auf der iSeries momentan der Gruppen-PTF-Stand 13 installiert.

Um auf die aktuellste Version zu kommen, sollte also das aktuelle PTF SF99098 bestellt und auf der iSeries installiert werden.

Anmerkung:

Verwenden Sie die OS/400-Version V5R1, dann kann der Gruppen-PTF-Stand über den Befehl „DSPDTAARA QHTTPSVR/SF99156“ angezeigt werden. Für diese Betriebssystemversion gibt es keinen Befehl „WRKPTFGRP“.

Die Installation des HTTP-Servers wird durch das Einspielen und Anlegen des erforderlichen PTF SF99098 durchgeführt.

Die Installation von PTFs ist denkbar einfach:

Ausführungsschritte

① ▷ ② ▷ ③ ▷ ▷
Ausführungsschritte

- Melden Sie sich dazu mit einem Benutzerprofil an, das über ausreichende Berechtigung für die Installation von PTFs verfügt (Sonderberechtigungen *IOSYSCFG und *CHANGE-Berechtigung für die Bibliothek QUSRSYS) – im Idealfall verwenden Sie dazu das Benutzerprofil „QSECOFR“.
- Geben Sie in der Befehlszeile den Befehl „GO PTF“ ein.
- In dem folgenden Menü „Vorläufige Programmkorrekturen (PTF)“ wählen Sie den Eintrag „8 PTF-Pakete installieren“ aus.
- In dem nun folgenden Bild geben Sie als Installationseinheit die Einheit an, in der sich die PTFs befinden. Dies ist in der Regel das CD/DVD-Laufwerk der iSeries – in unserem Beispiel: „OPT01“.
- Soll der PTF-Code zunächst nur von der CD auf die iSeries kopiert und später angelegt werden, ändern Sie den Eintrag „Automatisches IPL“ auf „N“ ab. Anderenfalls werden Sie in einer weiteren Anzeige darauf hingewiesen, dass im Anschluss an das Einspielen des PTF-Codes ein IPL durchgeführt wird. Wird dieses bestätigt, dann werden alle Jobs auf dem System nach dem Kopieren des Codes von der CD beendet und das System fährt einen IPL! Wenn Sie direkt nach dem Kopieren des PTF-Codes diesen anlegen wollen, können Sie den Parameter „Automatisches IPL“ auf „J“ stehen lassen. Das System führt dann nach dem Kopieren einen IPL aus. Dabei muss an der Konsole der iSeries der Status auf jeden Fall auf „Normal“ gestellt sein, denn nur in diesem Status werden im Zuge eines IPL PTFs angelegt.

5.1.1
Seite 8

```
                        Installationsauswahl für PTFs
                                                          System:   ASZEBIS
Auswahl eingeben und Eingabetaste drücken.

  Einheit . . . . . .   OPT01      Name, *SERVICE

  Automatisches IPL . . . .  N      J=Ja
                                    N=Nein

  Art des Wiederanlaufs . .  *SYS   *SYS, *FULL

  PTF-Art . . . . . .   1          1=Alle PTFs
                                    2=Nur HIPER-PTFs und
                                      HIPER-LIC-Korrekturen
                                    3=Nur HIPER-LIC-Korrekturen
                                    4=LIC aktualisieren

  Weitere
    Auswahlmöglichkeiten    N      J=Ja
                                    N=Nein
```

PTF installieren

Nach der Installation der PTFs kann geprüft werden, ob diese erfolgreich angelegt worden sind. Dazu kann entweder der Befehl „WRKPTFGRP“ verwendet werden oder in dem Menü „GO LICPGM“ mit der Auswahl „50 Nachrichtenprotokoll anzeigen“ das Protokoll der Installationen auf der iSeries angezeigt werden.

Wenn die letztere Funktion verwendet wird, kann nach der Auswahl „50“ in der Folgemaske ein Datum und eine Uhrzeit eingegeben werden. Im Normalfall können die Vorgabewerte übernommen werden.

```
                              Installationschronik anzeigen

Auswahl eingeben und Eingabetaste drücken.

  Startdatum . . . . . .   12.08.04   TT.MM.JJ

  Startzeit  . . . . . .   10:52:25   HH:MM:SS

  Ausgabe  . . . . . .     *          *, *PRINT
```

Installationschronik

Das Protokoll sollte keinen Eintrag enthalten, der einen Hinweis auf einen Fehler oder eine nicht erfolgreiche Installation enthält:

```
                          Protokollinhalt anzeigen

Laden der PTFs gestartet.
Laden der PTFs erfolgreich beendet.
Markierung der PTFs für verzögertes Anlegen gestartet.
Markieren der PTFs für das verzögerte Anlegen erfolgreich beendet.
PTFs fehlerfrei installiert.
Anlegen der PTF gestartet.
Anlegen der PTFs für Produkt 5722IWE erfolgreich beendet.
Anlegen der PTFs beendet.
PTFs beim IPL angelegt oder entfernt.
```

Protokollinhalt

Nachdem nun alle Lizenzprogramme sowie der HTTP-Server durch die Installation der erforderlichen PTFs installiert worden sind, testen wir die Installation des HTTP-Servers „powered by Apache“:

Bevor mit dem Web-Server gearbeitet werden kann, muss dieser zunächst konfiguriert werden. Sowohl für die Konfiguration als auch für die weiterführende Administration wird eine für die iSeries spezifisch angepasste Oberfläche verwendet, die sich in der Administration von den Versionen anderer Nicht-iSeries-Plattformen unterscheidet. Diese Installations-Oberfläche hat sich zwischen dem Betriebssystem-Release V5R1 und V5R2 verändert. Wundern Sie sich also nicht, wenn die Anzeigen bei der Installation auf einem V5R1-System von den folgenden Beispielabbildungen abweichen. Die folgenden Abbildungen basieren auf einem OS/400 Release-Stand V5R2M0.

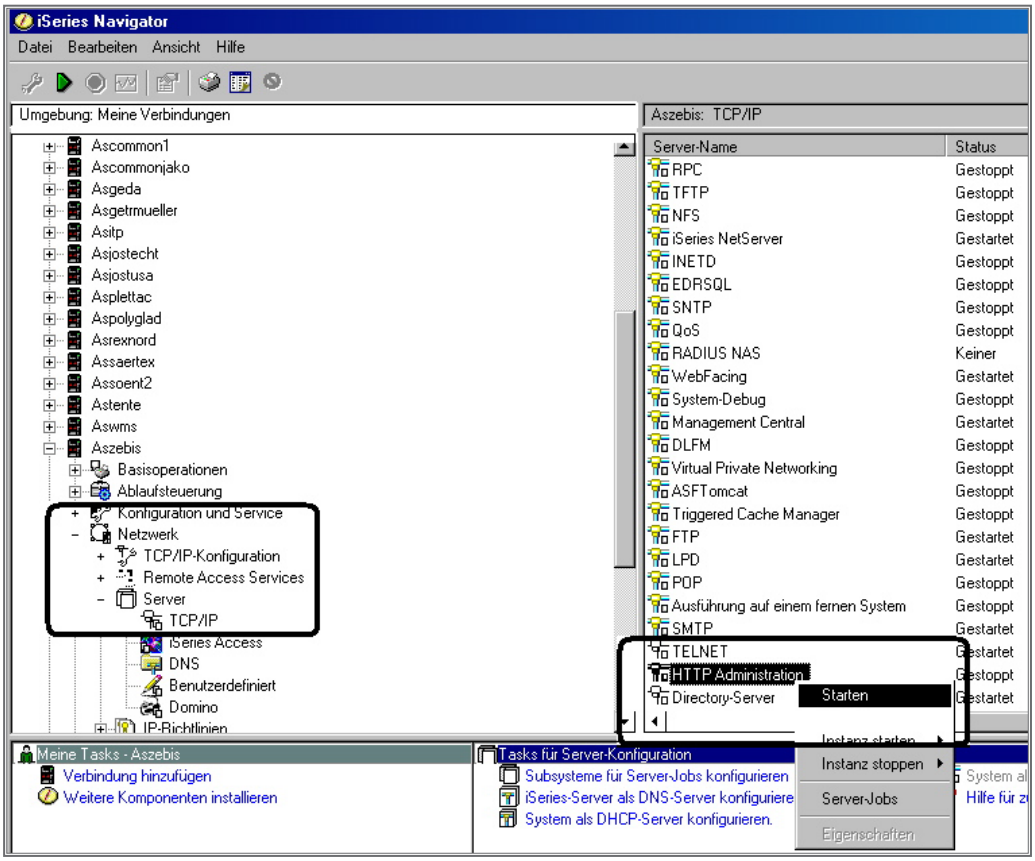
Damit die Administrations- und Konfigurationsoberfläche gestartet werden kann, muss zuerst die Kommunikation zwischen dem HTTP-Server und der iSeries gestartet werden. Grundlage für diese Kommunikation bildet der HTTP-Administrations-Server.

Ausführungsschritte

① ▷ ② ▷ ③ ▷ ▷
Ausführungsschritte

- Starten des HTTP-Administrations-Servers mit Verwendung des iSeries Navigators
- „Netzwerk / Server / TCP/IP“ auswählen
- Den Eintrag „HTTP-Administration“ mit einem Klick der rechten Maustaste markieren
- „Starten“ auswählen

5.1.1
Seite 10



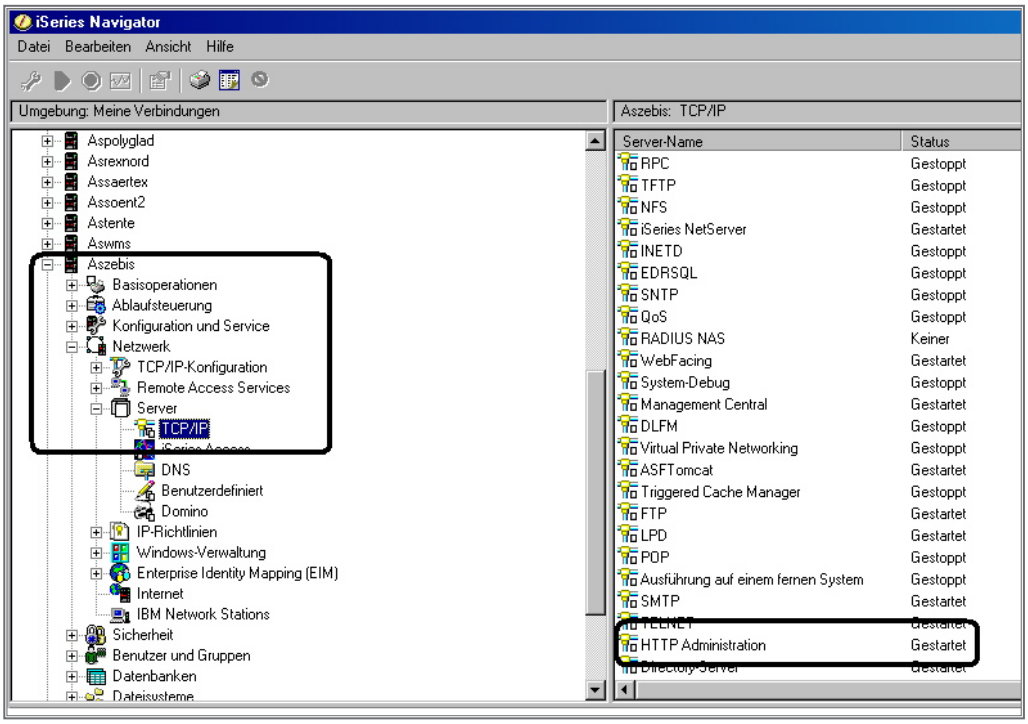
HTTP Administration

Alternativ zu dem Start über den iSeries Navigator kann auch folgender Green-Screen-Befehl verwendet werden:

```
STRTCPSVR SERVER(*HTTP)  HTTPSVR(*ADMIN)
```

Wurde der Administrations-Server korrekt gestartet?

Dies kann ebenfalls mit dem iSeries Navigator geprüft werden. Dazu wird dieser wieder geöffnet, unter demselben Verfahren wie zuvor beim Starten beschrieben (Auswahl „Netzwerk / Server / TCP/IP“) und es wird auf der rechten Seite der Anzeige des iSeries Navigators der Eintrag „HTTP Administration“ gesucht. Neben diesem Eintrag befindet sich der aktuelle Status dieses Servers. Dieser sollte nun auf „Gestartet“ stehen.



HTTP Administration – Gestartet

Für die Green-Screen-Fans: Auch hier gibt es eine Möglichkeit, den Status des HTTP-Administrations-Servers abzufragen.

Verwenden Sie den Befehl „WRKACTJOB SBS(QHTTPSVR). Das Ergebnis der Anzeige sollte in etwa so aussehen:

```
                    Mit aktiven Jobs arbeiten
                                                          12.08.04
   0,0      Abgelaufene Zeit:    00:00:00      Aktive Jobs:    203
 Subsystem/Job   Benutzer     Art   CPU %   Funktion          Status
QHTTPSVR         QSYS         SBS     0,0                     DEQW
  ADMIN          QTMHHTTP     BCH     0,0   PGM-QZHBHTTP      SIGW
  ADMIN          QTMHHTTP     BCI     0,0   PGM-QZSRLOG       SIGW
  ADMIN          QTMHHTTP     BCI     0,0   PGM-QZSRHTTP      SIGW
  APACHEDFT      QTMHHTTP     BCH     0,0   PGM-QZHBHTTP      SIGW
  APACHEDFT      QTMHHTTP     BCI     0,0   PGM-QZSRLOG       SIGW
  APACHEDFT      QTMHHTTP     BCI     0,0   PGM-QZSRHTTP      SIGW
```

HTTP Admin gestartet 5250

Damit ist die Kommunikation zwischen der iSeries und dem HTTP-Server für Administrationszwecke hergestellt.

Bevor wir nun die Konfiguration des HTTP-Servers durchführen, testen wir zunächst, ob wir diesen mit dem Administrator-Zugang erreichen können.

Für diese Prüfung und auch für die spätere Konfiguration muss ein OS/400-Benutzerprofil verwendet werden, das über die nachfolgenden Berechtigun-

5.1.1
Seite 12

gen verfügt. Auch an dieser Stelle sei darauf hingewiesen, dass für diese Aufgaben natürlich auch das Benutzerprofil „QSECOFR“ verwendet werden kann. Allerdings ist es in der Praxis meist üblich, gerade in diesem Bereich einen eigenen Benutzer zu definieren, der die administrativen Aufgaben im Bereich des Web-Servers übernimmt. Das macht insbesondere dann Sinn, wenn der Web-Server durch einen Nicht-iSeries-Bediener verwaltet werden soll.

Als Berechtigungen muss der Administrations-Benutzer über die folgenden Sonderberechtigungen verfügen:

- *IOSYSCFG
- *CHANGE auf die Bibliothek QUSRSYS
- *ALL Berechtigung auf die Objekte
  - QUSRSYS/QATMHINSTA
  - QUSRSYS/QATNHINSTC
- *USE Berechtigung auf die Befehle
  - CRTVLDL
  - STRTCPSVR
  - ENDTCPSVR

Sind nun alle Voraussetzungen geschaffen, dann versuchen wir eine erste Verbindung zwischen einem PC-Client und dem HTTP-Server herzustellen:

①▷②▷③▷▷
Ausführungsschritte

Ausführungsschritte

- Öffnen Sie den Browser – in unserem Beispiel verwendet wir den Microsoft Internet Explorer Version 6.0.
- Geben Sie als „Adresse“ den Eintrag

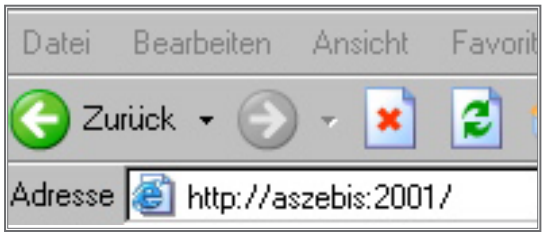
  HTTP://servername:2001

ein, wobei anstelle des Server-Namens (dieser muss in der Host-Tabelle des PC gepflegt sein) auch die TCP/IP-Adresse des HTTP-Servers angegeben werden kann.

Anmerkung:

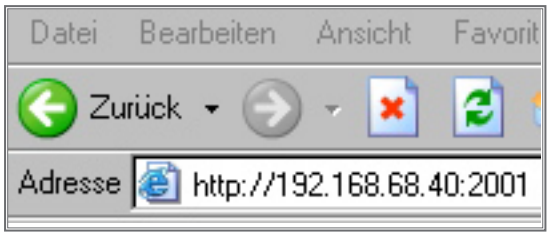
Bei dem Eintrag „:2001“ handelt es sich um den TCP/IP Port 2001, der als Administrations-Port für den Web-Server verwendet wird.

In unserem Beispiel geben wir den Host-Namen des Servers (der iSeries) „aszebis“ ein:



Server-Name

Alternativ hätte auch die TCP/IP-Adresse des Servers (oder der iSeries) angegeben werden können. Eine solche Eingabe sieht dann für die IP-Adresse 192.68.68.40 wie folgt aus:



IP-Adresse

Welche der beiden Eingabevarianten Sie auch wählen, in beiden Fällen sollte das Ergebnis dasselbe sein.

Als Ergebnis der Eingabe sollte nun ein Eingabebildschirm erscheinen, in dem Sie sich für die Administration des HTTP-Servers anmelden müssen.

Anmeldung

Für die Anmeldung muss ein Benutzerprofil verwendet werden, das über die erforderlichen Berechtigungen – wie zuvor beschrieben – verfügt.

Ist die Anmeldung erfolgreich, so erscheint die Einstiegsseite für die Administration der unterschiedlichen Server auf der iSeries:

Admin-Einstiegsseite

Wenn diese Einstiegsseite erscheint, dann wurden die Installation der Software sowie die Basiskonfiguration erfolgreich beendet.

Um nun einen (oder auch mehrere) Web-Server einzurichten, müssen bestimmte Konfigurationsschritte ausgeführt werden, die unter anderem davon abhängen, welche Aufgaben der Web-Server übernehmen soll.

Wie Sie in der Einstiegsanzeige für die Administration der „iSeries Tasks“ unschwer erkennen können, dient dieser Konfigurations- und Administrations-Client nicht nur für die Arbeit mit einem HTTP-Server, sondern stellt ein globales Werkzeug für das Arbeiten mit verschiedenen Servern (HTTP-Server, WebSphere Application Server Express, WebSphere Commerce Suite u. a.) dar.

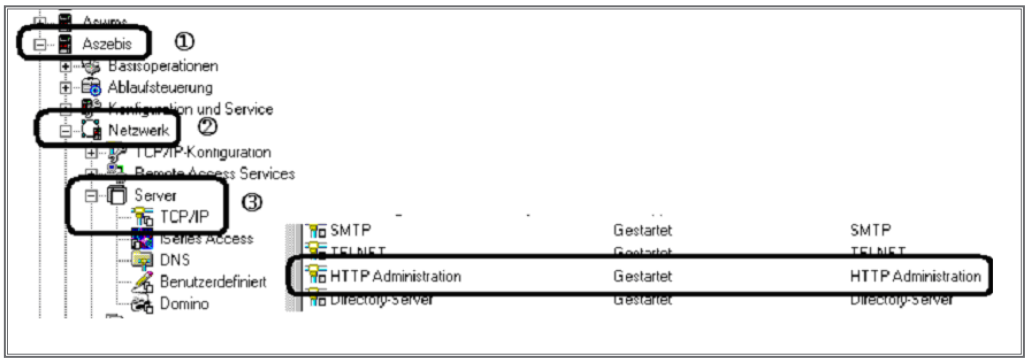
In dem nachfolgenden Abschnitt werde ich Ihnen die Konfiguration des HTTP-Servers näher beschreiben.

Browser-Administration

Bevor wir uns mit der Installation und Konfiguration eines neuen HTTP-Servers beschäftigen, wollen wir uns in diesem Abschnitt zunächst das Administrations-Werkzeug in Form der Browser-Administration anschauen.

Für die Arbeit mit dem HTTP-Administrations-Browser muss in jedem Fall sichergestellt sein, dass die in dem vorhergehenden Kapitel beschriebene Prozedur des Startens der Server-Instanz „Admin“ für den Bereich *HTTP durchgeführt wurde. Ist es nicht sicher, ob diese Admin-Instanz aktiv ist, wird die Prüfung mit dem iSeries Navigator wie folgt durchgeführt:

- Starten des iSeries Navigators
- Auswahl des i5-Systems, auf dem der HTTP-Server konfiguriert werden soll
- Erweitern des Eintrags „Server“ durch einen einfachen Klick
- Auswahl des Eintrages TCP/IP
  Im rechten Fensterbereich öffnet sich eine Übersicht, in der unter anderem auch der Eintrag „HTTP Administration“ zu finden ist. Dieser muss den Status „Gestartet“ haben.



HTTP Admin gestartet

Sollte der Status des Eintrages „HTTP Administration“ nicht „gestartet“ sein, so muss der Admin-Task zunächst gestartet werden, wie dies im vorhergehenden Kapitel beschrieben wurde.

Bevor wir in die Details der Arbeit mit dem Admin-Client einsteigen, noch einige wichtige Hinweise:

Die grundsätzliche Aussage „Für die Konfiguration ist ein Browser erforderlich“, sollte an dieser Stelle noch ein wenig konkretisiert werden. Denn der Browser, der für das Einrichten und spätere Administrieren des oder der HTTP-Server verwendet wird, muss einige Voraussetzungen erfüllen:

- HTTP Protokoll 1.0 oder besser 1.1
- Unterstützung von Frames

5.1.2
Seite 2

- Unterstützung von Java Script
- Unterstützung der Java Virtual Machine ab 1.1.5

Grundsätzlich ist jeder Microsoft Internet-Explorer ab 5.5 für die HTTP-Administration geeignet.

Das Starten des Admin-Clients erfolgt immer auf demselben Weg:

1. Starten des HTTP-Admin Bereichs unter TCP/IP
   - STRTCPSVR SERVER(*HTTP) HTTPSVR(*ADMIN)
2. Starten des Browsers (in unserem Beispiel des Internet Explorers)
3. Angabe der Start-URL für die HTTP-Server-Verwaltung (Host-Name und der Admin-Port 2001)
   - Zum Beispiel: http://aszebis:2001

!

Achtung

Für die Anmeldung am Administrations-Bereich stehen zwei TCP/IP-Ports zur Verfügung:

- Port 2001
  Aufbau einer nicht-sicheren Verbindung
- Port 2010
  Aufbau einer sicheren Verbindung

Sie können zwischen diesen beiden Ports wählen.

Nach Angabe des iSeries-Benutzernamens und Kennwortes gelangen wir auf die Startseite der Administration.

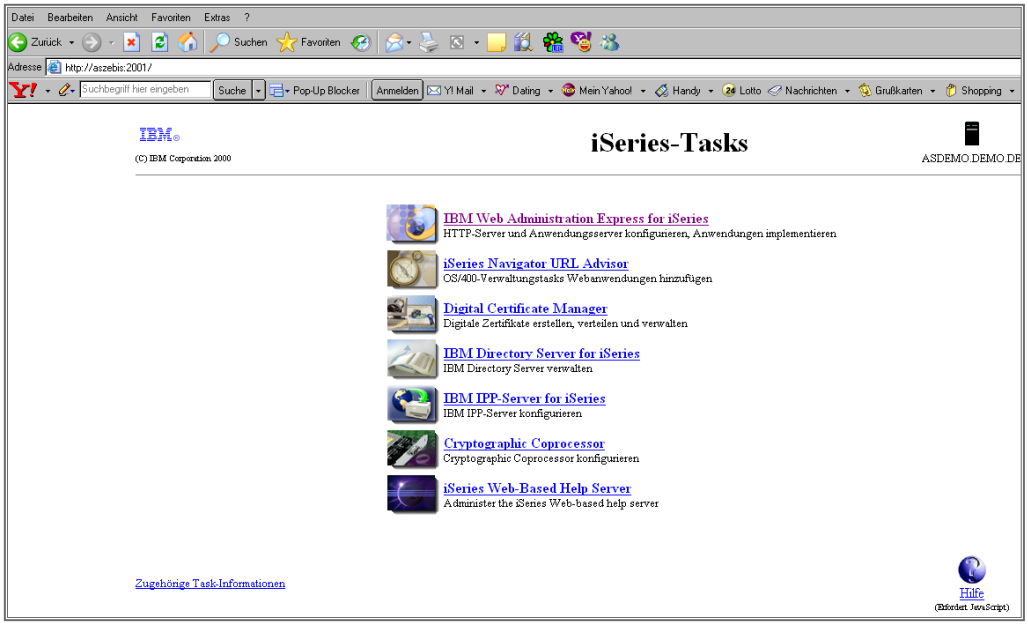
Diese kann je nach installierter OS/400- (oder i5/OS-) Version unterschiedlich aussehen.

Während unter V5R2 die Verwaltung der HTTP-Server noch als eigenständiger Eintrag in dem Einstiegsfenster aufgelistet ist, wurde dieser Punkt in der V5R3-Version umbenannt und beinhaltet dort die beiden Bereiche HTTP-Server und WebSphere Application-Server Express.

Die nachfolgende Abbildung zeigt die Einstiegsseite unter V5R2M0.

T5-K1-2-2 Admin-Einstiegsseite V5R2

Sollten sie V5R3M0 auf dem eServer verwenden, so erscheint die folgende Anzeige:



T5-K1-2-3 Admin-Einstiegsseite V5R3

Wir werden in diesem Kapitel auf das Installations- und Konfigurationsverfahren unter der Version V5R3M0 aufbauen. Sollten sie auf ihrem Server noch V5R2M0 einsetzen, so sind die Konfigurationsschritte ähnlich anzuwenden.

5.1.2
Seite 4

Des Weiteren dienen diese in dem folgenden Abschnitt beschriebenen Konfigurationsverfahren als Beispiel für den Aufbau einer kleinen Demo-Umgebung, mittels der sie am Ende dieses Kapitels in der Lage sind, eine Anwendung basierend auf einem HTTP-Server zu testen und zu konfigurieren.

HTTP Server – Der Administrationsbereich

In dieser allgemeinen Einstiegsseite befindet sich unter anderem der Eintrag, über den alle HTTP-Server verwaltet bzw. eingerichtet werden. Diesen Bereich „IBM HTTP-Server für iSeries“ unter V5R2 bzw. „IBM Web Administration Express for iSeries“ unter der Version V5R3M0 wählen wir mit einem Mausklick aus.

HTTP-Bereich

Oder für V5R3M0:

HTTP-Bereich V5R3M0

In der nächsten Anzeige sind die ersten wesentlichen Veränderungen zwischen den Versionen V5R2M0 und V5R3M0 zu erkennen.

Denn waren unter V5R2M0 noch die folgenden unterschiedlichen HTTP-Server-Einträge auswählbar, so sind diese unter V5R3M0 bereits konsolidiert.

Mit V5R2M0 kann die Verwaltung der nachfolgenden Server durchgeführt werden:

- HTTP Server (Powered by Apache)
- Original IBM HTTP Server
- Tomcat
- WebSphere Application Server

Unter V5R3M0 bekommen wir die folgende Einstiegsseite angezeigt:

5.1.2.1
Seite 2

Neuen HTTP-Server erstellen

Betrachten wir nun zunächst die Möglichkeiten, die sich mit diesem Verwaltungs-Client bieten, bevor wir uns mit der Konfiguration des ersten HTTP-Servers beschäftigen.

Im oberen Bereich innerhalb der Browser-Anzeige befindet sich eine allgemeine Auswahlleiste für:

Konfiguration

Dieser allgemeine Bereich dient der Konfiguration von den unterschiedlichen Servern.

Als Auswahlmöglichkeiten werden angeboten:

- HTTP-Server erstellen
  Diese Auswahl ist zu treffen, wenn ein neuer HTTP-Server erstellt werden soll.
- Original zu Apache migrieren
  Mit dieser Option lässt sich ein älterer „originaler“ IBM HTTP-Server in einen HTTP-Server „Powered by Apache“ migrieren.

Verwaltung

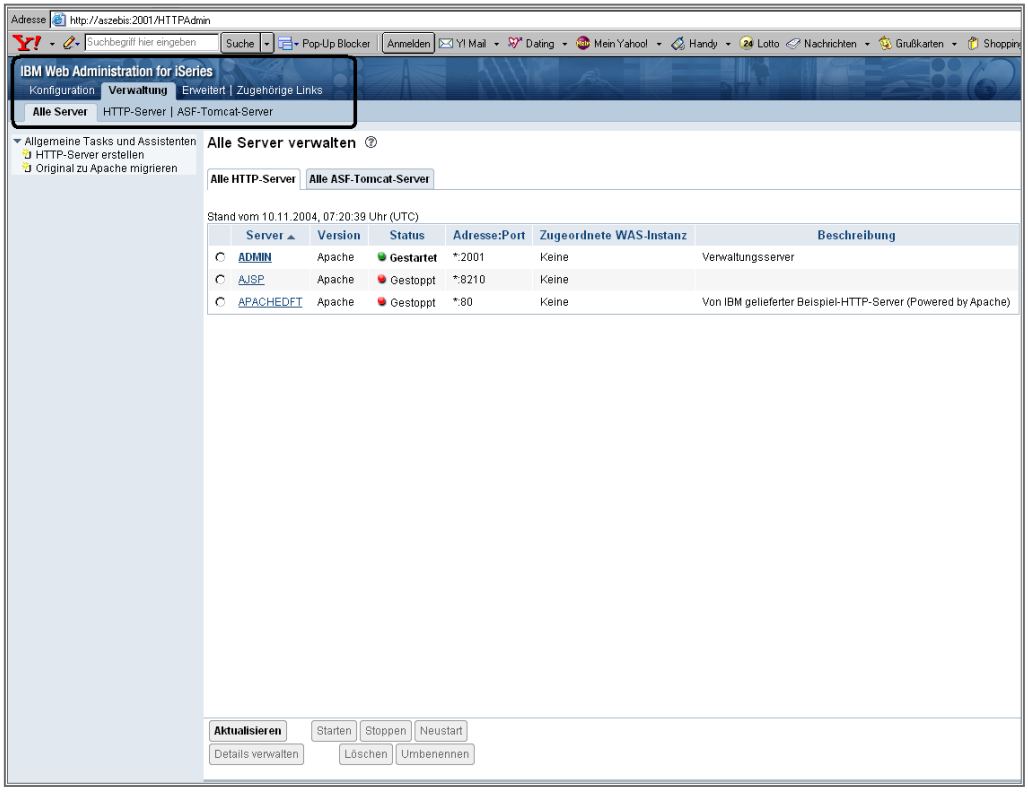
Der Verwaltungsbereich dient der Administration bereits bestehender Server. Wundern sie sich nicht, denn im linken Navigationsbereich befinden sich unter der Auswahl „Alle Server“ dieselben Einträge, wie in dem Oberbereich „Konfiguration“, mit denen sich auch neue Server erstellen lassen.

Der Verwaltungsbereich ist in drei Unterbereiche eingeteilt, die sich durch die verschiedenen Server-Typen klassifizieren lassen:

- Alle Server
- HTTP-Server
- Tomcat Server

Je nachdem, welchen Server sie gerade verwalten wollen, ist der entsprechende Bereich auszuwählen.

Die Auswahl „Alle Server“



Alle Server verwalten

Die Anzeige „Alle Server“ beinhaltet eine komplette Auflistung aller auf der i5 konfigurierten Server. Hier sind bereits einige Einträge in der Auflistung vorhanden, ohne dass sie diese erstellt haben. IBM liefert einen Default-Server-Eintrag mit (APACHEDFT). Zusätzlich finden wir in der Übersicht auch den Verwaltungs-Server – nämlich den mit STRTCPSVR gestarteten Admin-Server für die Verwaltung der HTTP-Einstellungen (also quasi den Bereich, mit dem wir uns aktuell auf dem System angemeldet haben).

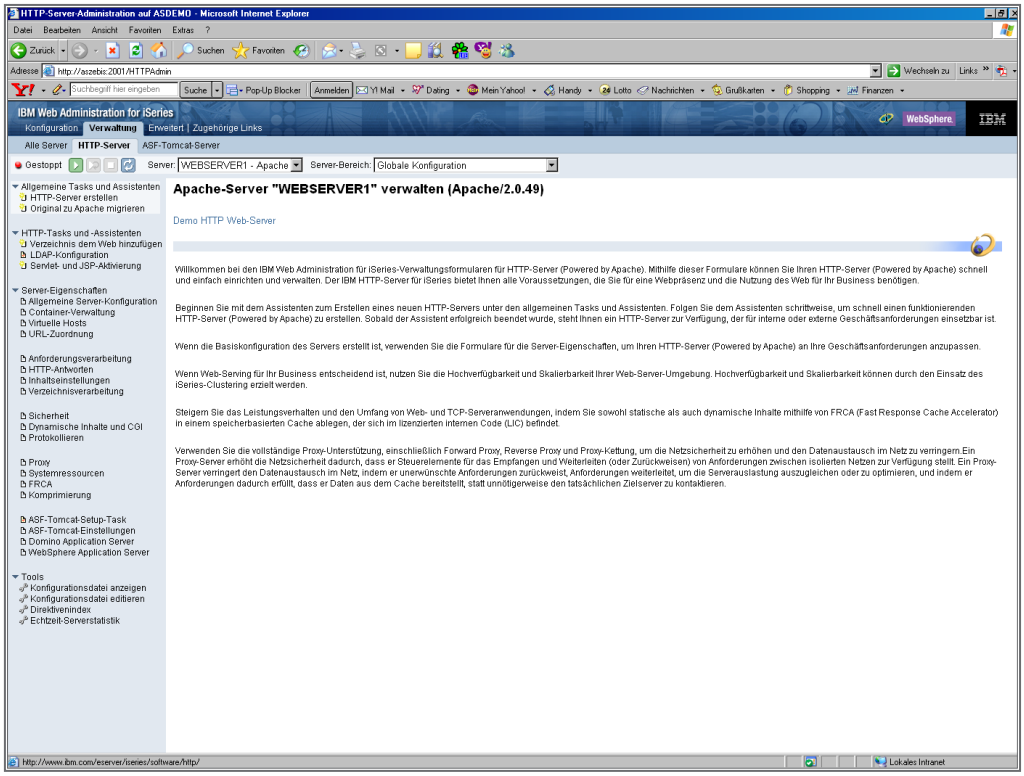
ADMIN Server-Eintrag

5.1.2.1
Seite 4

Die Auswahl „HTTP-Server“

Dieser Bereich stellt für unsere ersten Schritte und auch für die weitere Arbeit eine zentrale Administrationseinheit dar.

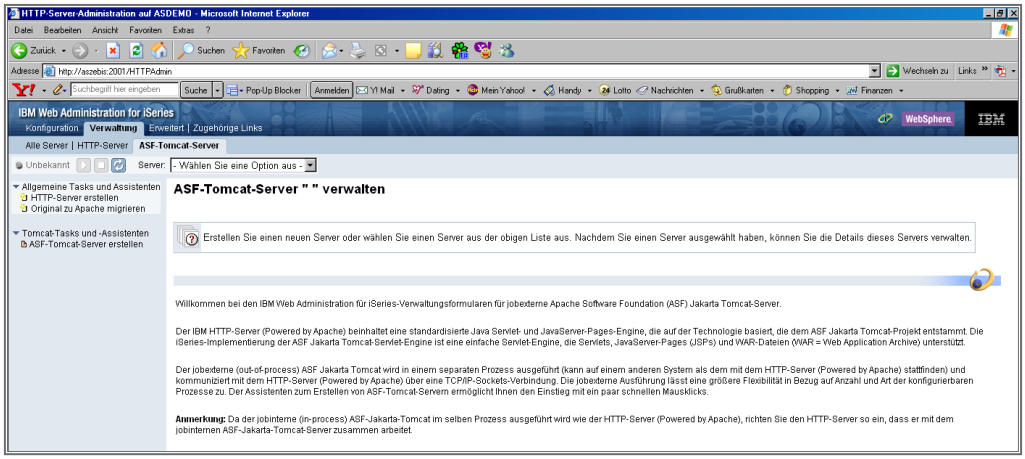
Von hier aus lassen sich die meisten Tätigkeiten, Einstellungen und Definitionen zu einem ausgewählten HTTP-Server vornehmen.



HTTP-Server verwalten

Wir wollen an dieser Stelle noch nicht in die Details der Verwaltung eines HTTP-Servers eingehen – diese behandeln wir später in diesem Kapitel.

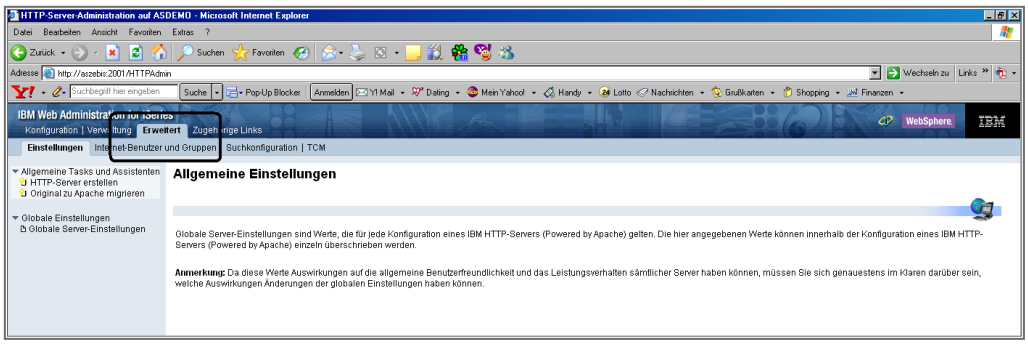
Die Auswahl „ASF-Tomcat-Server“



Tomcat

IBM unterstützt derzeit neben dem HTTP-Server (Powered by Apache) den ASF-Tomcat-Server, der ebenfalls mit diesem Werkzeug verwaltet und konfiguriert werden kann.

Der Eintrag „Erweitert“



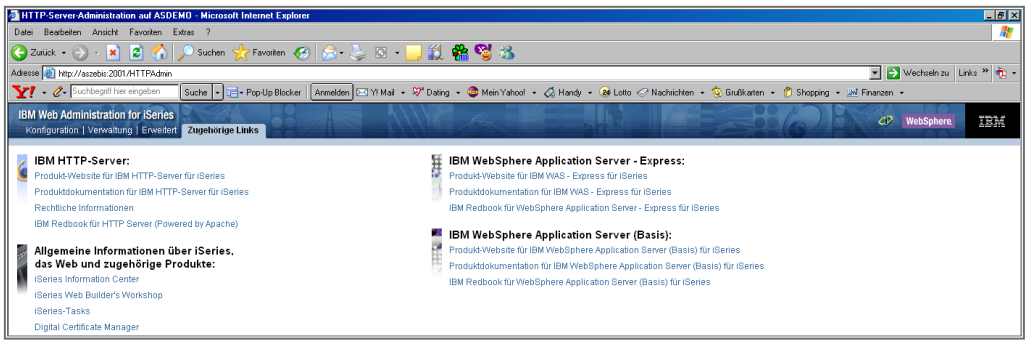
Erweitert

In diesem Verwaltungsbereich lassen sich globale Einstellungen vornehmen. Dies sind zum Beispiel die Zeichensatz-ID, mit denen die HTTP-Server arbeiten sollen (gerade auch im internationalen Einsatz ein wichtiger Punkt!), sowie die Einstellungsmöglichkeiten für externe Benutzer – um nur einen kurzen Überblick zu geben.

Ein sinnvolles Feature ist sicher die Web-Suchfunktion, mit der die Seiten des HTTP-Servers für eine Volltextsuche vorbereitet werden.

Der Eintrag „Zugehörige Links“

Hinter dieser Auswahl verbirgt sich eine Liste der wichtigsten Web-Seiten, die Informationen rund um den Bereich HTTP-Server beinhalten. Hier finden sie wichtige Informationen über HTTP-Server, Redbooks, Internet-Links usw.



Zugehörige Links

Nachdem wir uns eine kleine Übersicht über die verschiedenen Funktionsbereiche des HTTP-Admin Browser-Werkzeuges gemacht haben, wollen wir nun einen ersten eigenen HTTP-Server auf der iSeries erstellen.

5.1.2.1
Seite 6

Ohne zu viel belastende Theorie zu verwenden, werden wir einen einfachen, mit weitestgehenden Standardeinstellungen versehenen HTTP-Web-Server erstellen, der volle Funktionalitäten aufweißt und für unsere weiteren Beispiele in diesem Kapitel als Basis dient.

Keine Angst, es werden an dieser Stelle keine HTTP-Kenntnisse vorausgesetzt, sondern wir versuchen, diese durch „learning by doing“ zu erarbeiten und leicht verständlich zu vermitteln – sowohl für den klassischen iSeries-Bediener und Administrator als auch für den HTTP-Kenner, der bereits Erfahrungen im Einsatz von HTTP-Servern auf Nicht-iSeries-Plattformen besitzt.

HTTP-Server-Konfiguration

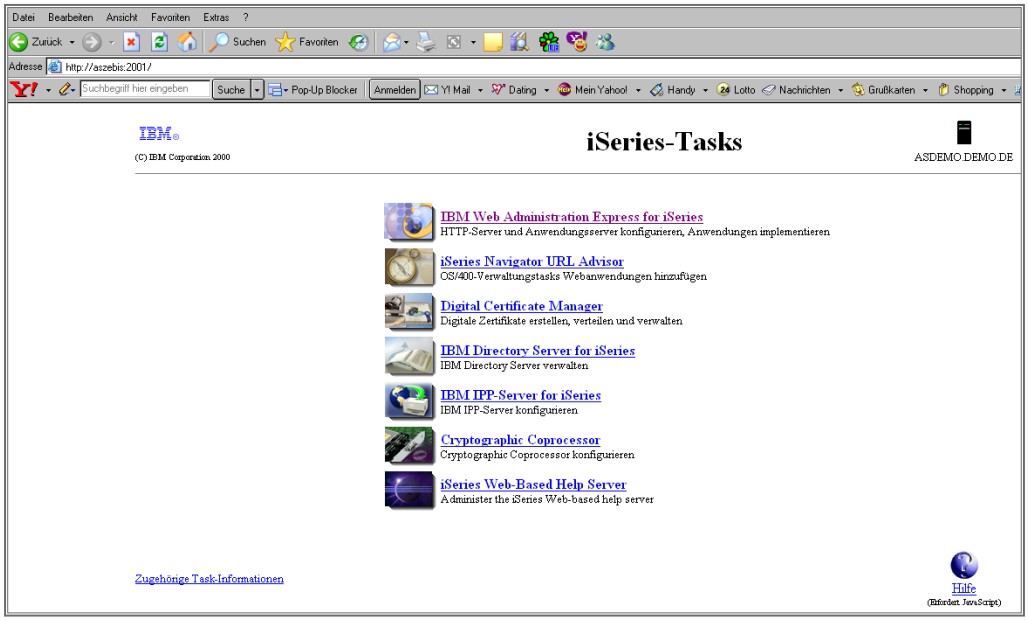
Nachdem die Installationsschritte des Kapitels 5.1 ausgeführt wurden, sind wir jetzt bereit, den ersten HTTP-Server auf der i5 zu konfigurieren.

Für die Konfiguration verwenden wir denselben Einstieg, wie wir ihn bereits während der Installation der Software kennen gelernt haben. Da dieses Verfahren immer wieder verwendet wird, kann die nachfolgende Aufgabenliste als Checkliste für die Verwaltung des HTTP-Servers genutzt werden.

1. Starten des HTTP-Admin-Bereichs unter TCP/IP
   STRTCPSVR SERVER(*HTTP) HTTPSVR(*ADMIN)
2. Starten des Browsers (in unserem Beispiel des Internet Explorers)
3. Angabe der Start-URL für die HTTP-Server-Verwaltung (Host-Name und der Admin-Port 2001)
   Zum Beispiel http://aszebis:2001 (für eine ungesicherte Verbindung) bzw. http://aszebis:2010 (für eine gesicherte Verbindung)

Nach Angabe des iSeries-Benutzernamens und Kennwortes gelangen wir auf die Startseite der Administration.

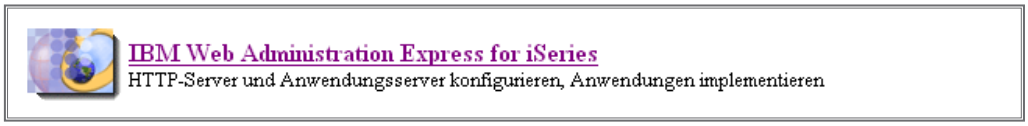
Wir beschränken uns in diesem Kapitel auf die Beschreibung des Installationsverfahrens unter der i5/OS, Version V5R3M0.



Admin-Einstiegsseite V5R3

In der Einstiegsseite wählen wir den ersten Eintrag „IBM Web Administration Express für iSeries“ aus.

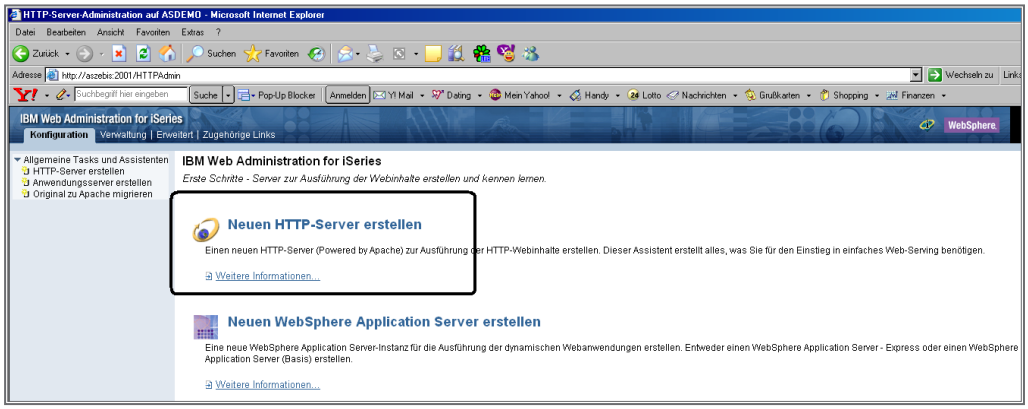
5.1.3
Seite 2



HTTP-Server-Bereich auswählen

Sollte ein Login-Fenster erscheinen, so muss hier die Anmeldung mit einem i5-Benutzer erfolgen.

Nun gelangen wir in den Bereich der Konfiguration eines neuen Servers. Dieser Bereich kann sowohl für die Konfiguration eines WebSphere Express Servers als auch für die Arbeit mit HTTP-Servern verwendet werden.



Neuen HTTP-Server erstellen

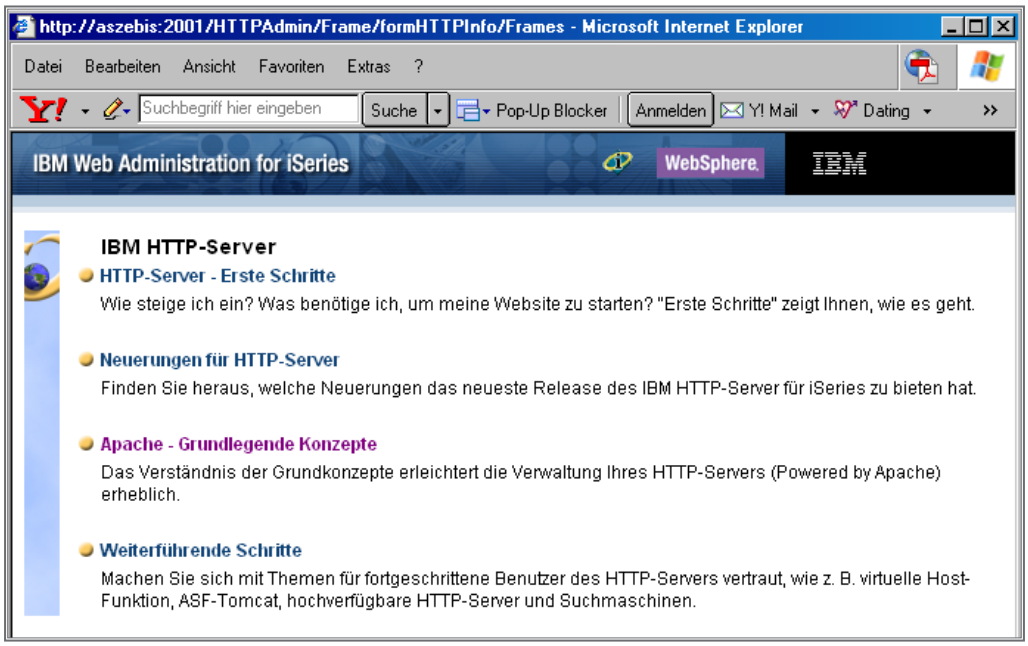
Alternativ zu der Auswahl in dem Hauptanzeige-Bereich lässt sich durch den Eintrag „HTTP-Server erstellen“ im linken Navigationsbereich der Browser-Anzeige derselbe Arbeitsschritt initiieren. Es macht keinen Unterschied, welche der beiden Auswahlmöglichkeiten verwendet wird.

Neuen HTTP-Server erstellen II

Sollten sie noch weitere Informationen für die Erstellung des HTTP-Servers benötigen, so lassen sich diese durch einen Klick auf „Weitere Informationen“ im Hauptfenster (direkt unterhalb des Eintrags „Neuen HTTP-Server erstellen“) anzeigen.

Weitere Informationen

Sollten sie diese Option auswählen, dann können Sie in dem sich öffnenden Fenster weitere Informationen zu verschiedenen Themenschwerpunkten abrufen.



Informationsseite

5.1.3
Seite 4

Einen neuen HTTP-Server erstellen

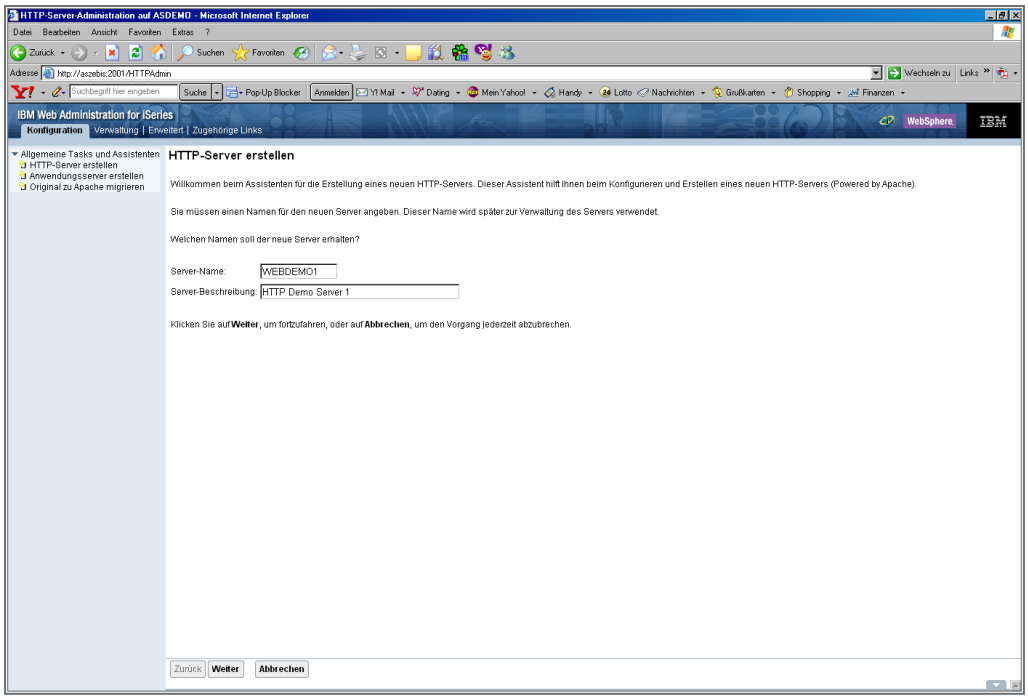
Allgemein gilt für das Arbeiten mit dem nachfolgend gezeigten Administrations-Werkzeug, dass jederzeit während der Konfiguration diese abgebrochen oder durch einen Klick auf die Schaltfläche „zurück“ auf einen der vorhergehenden Arbeitsschritte gesprungen werden kann. Jede Fortsetzung der Konfiguration wird über die Schaltfläche „Weiter“ im unteren Bereich des Browser-Fensters gesteuert.

Die folgenden Arbeitsschritte beschreiben die Definition eines neuen HTTP-Servers „Powered by Apache“. Dieser neue HTTP-Server bekommt den Namen „WEBDEMO1“.

Als Erstes wählen wir den Punkt „Neuen HTTP-Server erstellen“ aus.

Neuen HTTP-Server erstellen

Wir gelangen nun auf eine Seite, mit der wir dem neuen HTTP-Server einen Namen zuweisen. Wir verwenden den Namen „WEBDEMO1“ und geben zusätzlich eine Beschreibung des Servers an.



Name des HTTP Servers

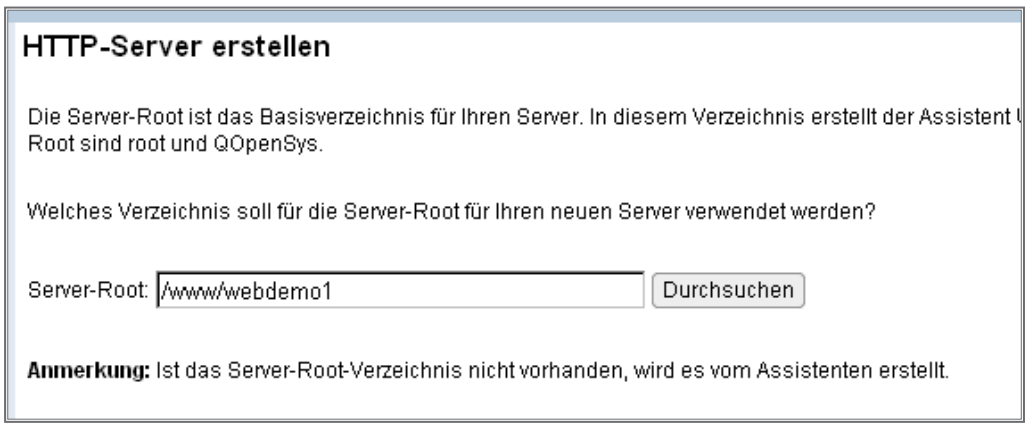
Nach erfolgter Eingabe wählen wir im unteren Fensterbereich die Schaltfläche „Weiter“ aus und gelangen so in die nächste Anzeige.

5.1.3.1
Seite 2

Jeder HTTP-Server wird auf der i5 in einem eigenen Verzeichnis angelegt und somit für die Verwendung verfügbar gemacht. Dieses Verzeichnis befindet sich in dem integrierten Dateisystem der i5 (IFS). Neben den Konfigurationsdaten werden in Unterverzeichnissen verschiedene Protokolle sowie auch die Web-Seiten gespeichert, die später für die Anwender bzw. die Anwendungen eingesetzt werden.

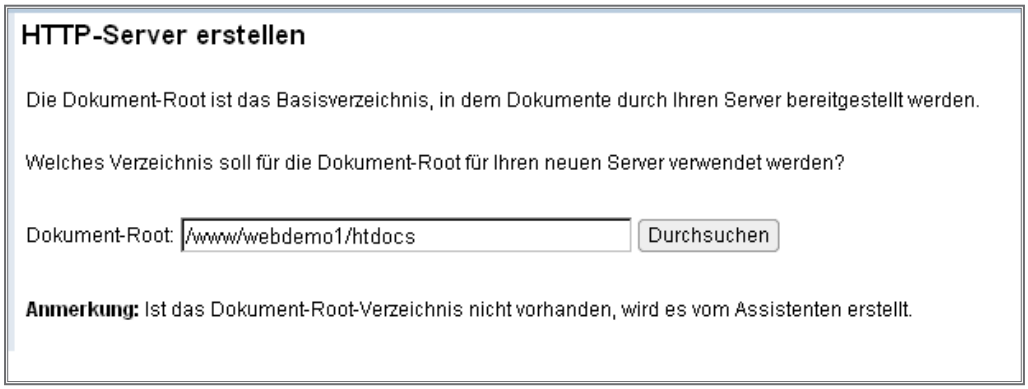
In der folgenden Anzeige definieren wir, welcher Pfad für die Speicherung des HTTP-Servers verwendet werden soll. IBM schlägt als Einstellung grundsätzlich „/www/servername“ vor. Durch unsere Namensvergabe „WEBDEMO1“ wurde der Pfad bereits entsprechend aufbereitet; wir müssen lediglich wieder im unteren Fensterbereich auf die Schaltfläche „Weiter“ klicken.

Sollte es zwingende Gründe geben, so können sie selbstverständlich den Namen des Verzeichnisses individuell anpassen.



Installationsverzeichnis

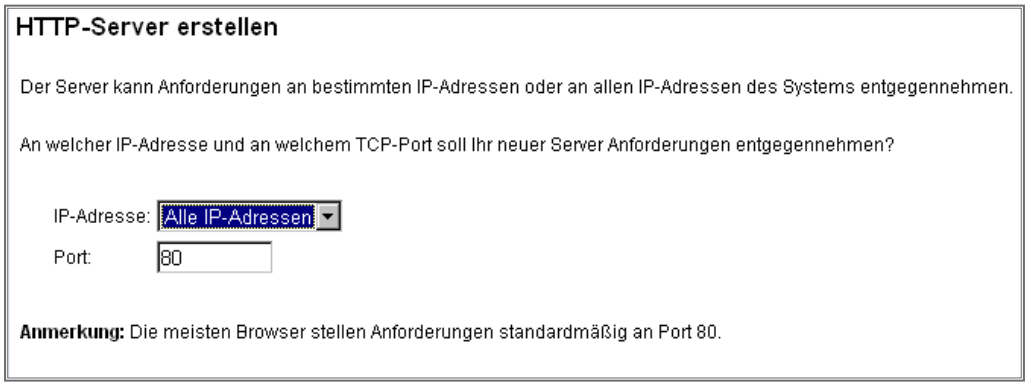
Nachdem die Schaltfläche „Weiter“ angeklickt wurde, müssen wir ein weiteres Verzeichnis angeben: das so genannte Dokument-Root. Dabei handelt es sich um das Stammverzeichnis, in dem die Server-Dokumente bereitgestellt werden. Server-Dokumente sind nichts anderes, als die Web-Seiten, die wir mit dem HTTP-Server verwenden wollen. In dieses Verzeichnis wird während der Server-Konfiguration zum Beispiel der Eintrag „index.html“ gestellt. Dabei handelt es sich um das Startdokument für den Server.



Dokument-Root

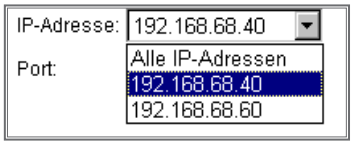
Auch hier schlägt das System wieder einen Namen vor. Wie man unschwer erkennt, besteht dieser wieder aus dem Root-Verzeichnis gefolgt von dem Eintrag „htdocs“. Diese Einstellungen übernehmen wir und setzen die Konfiguration mit einem Klick auf „Weiter“ fort.

Die Kommunikation zwischen dem HTTP-Server und den übrigen Teilnehmern wird auf Basis von TCP/IP-Verbindungen ausgeführt. In der nächsten Anzeige wird definiert, welche IP-Adresse und welcher Port für eingehende Anforderungen verwendet werden sollen.



Port-Informationen

Wahlweise können alle auf der i5 konfigurierten IP-Adressen verwendet werden. Es lässt sich auch nur eine der auf der i5 verwendeten IP-Adressen angeben; damit kann die Administration und Überwachung auf diese eine IP-Adresse beschränkt werden. Diese lässt sich durch einen Klick auf das Erweiterungssymbol rechts neben dem Feld „IP-Adresse“ auswählen.



Spezielle IP-Adresse

5.1.3.1
Seite 4

In unserem Beispiel verwenden wir die IP-Adresse „192.168.68.40“ von der i5.

Neben der IP-Adresse ist die Angabe eines Ports erforderlich, der mit der Voreinstellung „80“ übernommen werden kann. Dieser Port lässt sich auch individuell anpassen. Beim Port „80“ handelt es sich um einen allgemeingültigen Standard, der für die Kommunikation mit einem HTTP-Server verwendet wird. Deshalb sind einige Standardanwendungen oder Standardwerkzeuge (wie zum Beispiel der WebSphere Applikations-Server) auch auf diesen Port ausgelegt und müssen in der Regel nicht speziell in diesem Bereich „HTTP-Port“ angepasst werden.

Sollte der Port an dieser Stelle geändert werden, dann ist die gezielte Verwendung dieses Ports bei dem Zugriff (Zugriff über einen Browser, eine Anwendung oder einen Applikations-Server) auf den Server erforderlich.

In unserem Beispiel verwenden wir bewusst den Standardport „80“, denn diesen benötigen wir unter anderem für die Standardkonfiguration eines WebSphere Servers, ohne diesen entsprechend anpassen zu müssen.

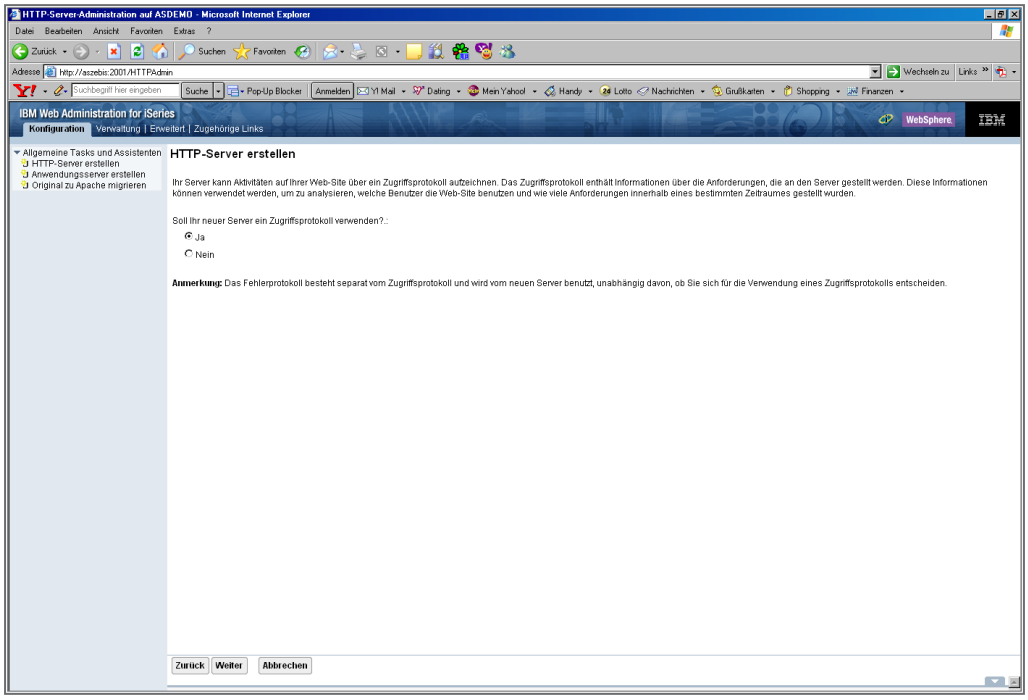
Nachdem die IP-Adresse und der Port ausgewählt wurden, klicken wir auf „Weiter“ und setzen die Konfiguration fort.

IP-Adresse und Port-Angabe

In vielen Fällen besteht ein besonderes Interesse, über die Zugriffe auf bestimmte Bereiche informiert zu werden. Für einen HTTP-Server können wir in der nächsten Anzeige festlegen, ob eine Zugriffsprotokollierung durchgeführt werden soll. Damit lassen sich Auswertungen über die Zugriffe auf den Web-Server auch als spezielle Seiten des Web-Servers analysieren. Unter anderem werden die IP-Adressen aufgezeichnet, mit denen auf den Server zugegriffen wird.

Neben diesem Zugriffsprotokoll wird zusätzlich ein Fehlerprotokoll mitgeschrieben, in dem alle Ausnahmefälle aufgezeichnet werden. Dieses Fehlerprotokoll wird in jedem Fall mitgeschrieben – unabhängig von der Auswahl, ob ein Zugriffsprotokoll geführt werden soll.

Als Empfehlung gilt, das Zugriffsprotokoll zu erstellen. Deshalb übernehmen wir auch die Voreinstellungen in diesem Fenster.

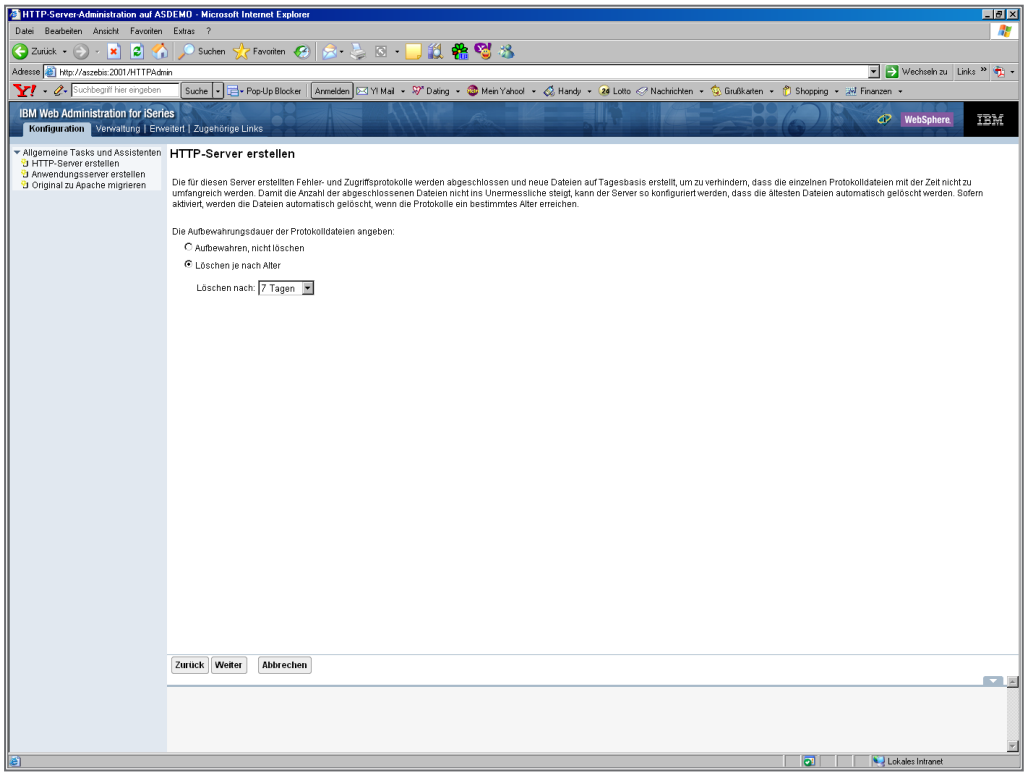


Zugriffsprotokoll verwenden

Da je nach Art und Einsatz des Servers ein solches Zugriffsprotokoll recht ansehnliche Ausmaße annehmen kann, sollte das Protokoll nicht über einen unendlichen Zeitraum geführt werden. Deshalb wird bereits bei der Konfiguration des HTTP-Servers eine Verwaltungsoption angeboten, mit der definiert wird, in welchen Zeitabständen das Zugriffs- und das Fehlerprotokoll reorganisiert werden sollen.

IBM bietet an dieser Stelle zwar auch die Option „Aufbewahren, nicht löschen“ an, allerdings sollte von dieser in der Praxis kein Gebrauch gemacht werden. Denn in diesem Fall müsste eine manuelle Reorganisation der Protokolle erfolgen, was einen zusätzlichen Aufwand bedeutet; deshalb verwenden wir in unserem Beispiel auch die empfohlene Methode der automatischen Bereinigung. Diese lassen wir auf dem Vorschlagswert „7 Tage“ stehen.

5.1.3.1
Seite 6

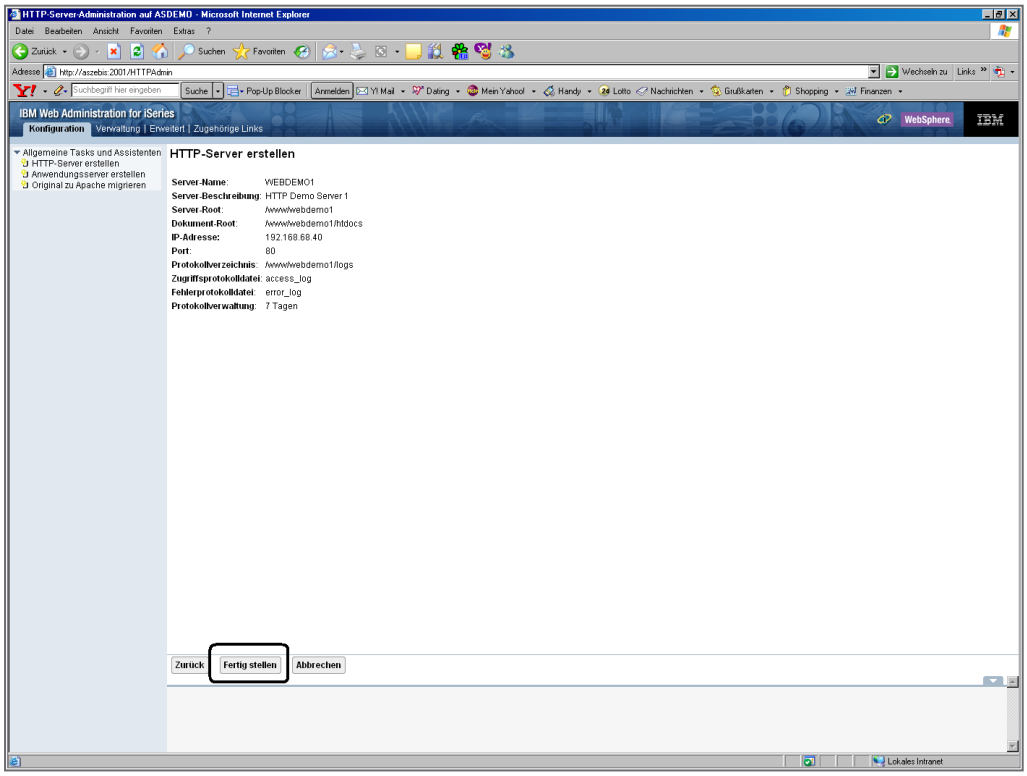


Reorganisation

Was haben wird nun alles angegeben?

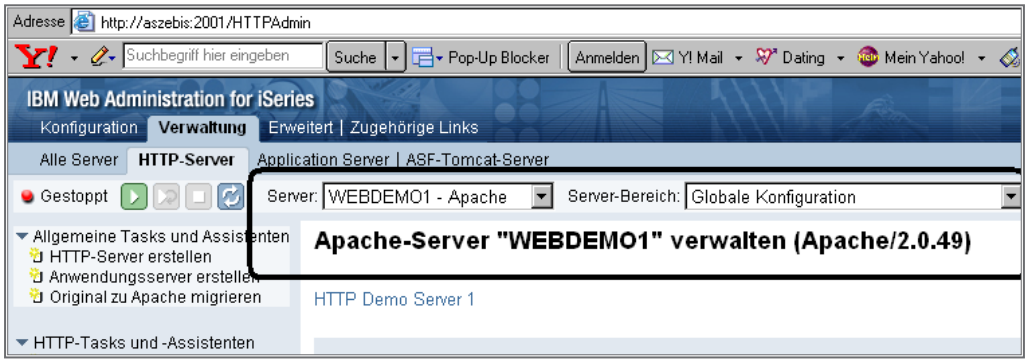
In der letzten Anzeige der Konfiguration eines neuen HTTP-Servers präsentiert das System eine Zusammenfassung aller getroffenen Auswahlen, um diese letztmalig zu kontrollieren. An dieser Stelle besteht noch die Möglichkeit, durch die „Zurück"-Schaltfläche Änderungen an den getroffenen Auswahlen durchzuführen.

Sind die getätigten Auswahlen genau das, was wir wollen, so erstellen wir mit einem Klick auf die Schaltfläche „Fertigstellen" unseren HTTP-Server „WEBDEMO1".



Fertigstellen

Nachdem die Konfiguration beendet wurde, wechselt die Browser-Anzeige in den Bereich der Verwaltung des soeben konfigurierten HTTP-Servers. Wie sie sehen, befinden wir uns in dem Verwaltungsbereich des HTTP-Servers „WEBDEMO1“.



Server verwalten

Damit haben wir die Basiskonfiguration unseres ersten HTTP-Servers auf der i5 abgeschlossen.

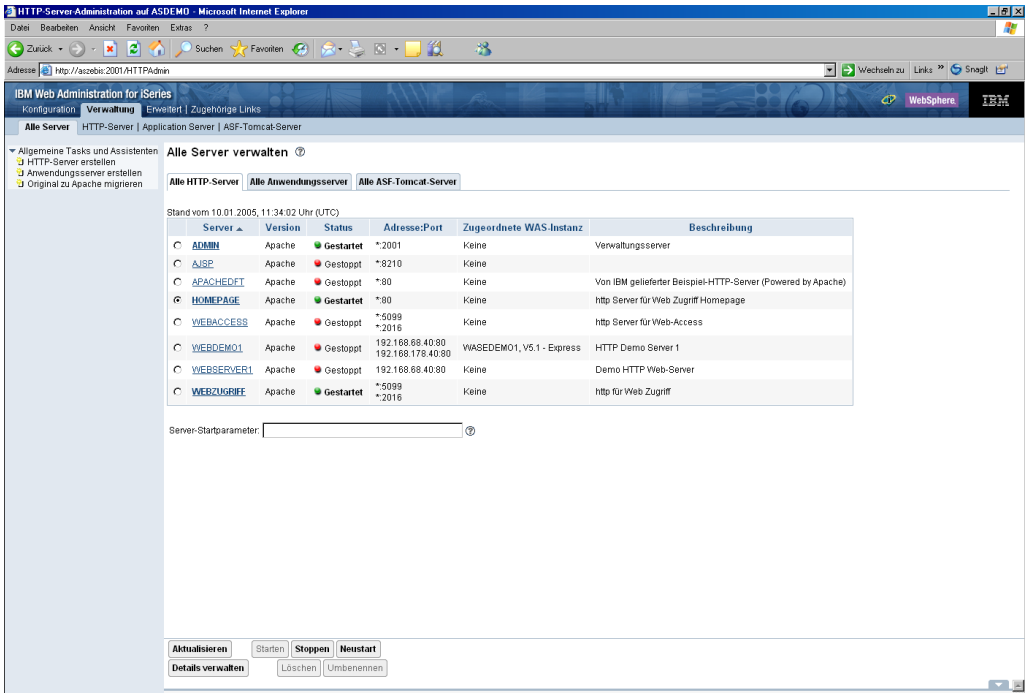
5.1.3.1

Seite 8

Alle http-Server verwalten

Der Administrationsbereich für die auf der i5 installierten unterschiedlichen Server kann entweder direkt für einen einzelnen Server oder unter Verwendung einer Auflistung aller auf dem System installierten Server selektiert werden.

Für die Verwaltung von komplexeren http-Server-Umgebungen bietet IBM den Bereich „alle http-Server verwalten“. Dieser ist in dem Unterabschnitt „alle Server“ enthalten und bietet einen Überblick über alle derzeit auf der i5 installierten http-Server. In einer Default-Umgebung sind bereits einige http-Server in der Übersicht enthalten – so finden wir dort unter anderem auch den Eintrag für die Administrations-Instanz „Admin“ wieder.

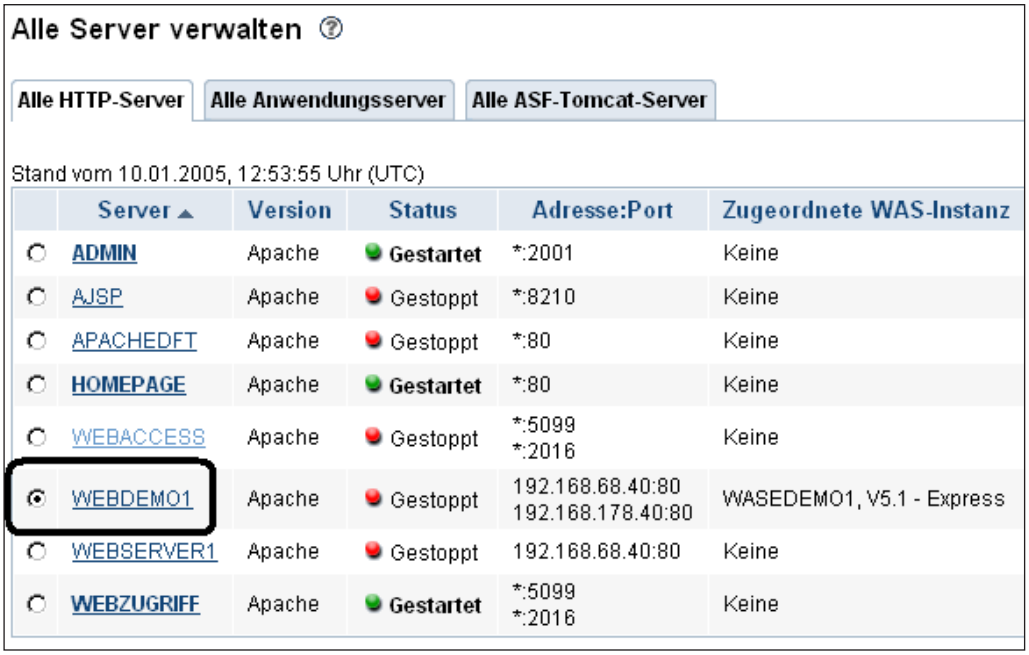


Alle Server verwalten

Die Übersicht aller http-Server ist so aufgebaut, dass diese von hier aus gestartet und gestoppt werden können. Neben dem Servernamen finden wir weitere wichtige Informationen wie den Servertyp, den Status, die IP-Adressen und die optional zugeordnete Applikations-Server-Instanz.

Der aktuell zu verwaltende Server muss in der Auflistung zunächst markiert werden. Dazu klicken wir unseren Beispielserver „WEBDEMO1“ zunächst an. Je nach i5 kann dieser Auswahlprozess einen kleinen Moment dauern – dies wird durch die Anzeige von „Laden“ ausgedrückt.

5.1.3.2
Seite 2



Server auswählen

Leider wirkt sich auch in dieser Anzeige die Eigenart von Browser-basierten Anzeigen aus – indem zum Teil die aktuelle Anzeige nur durch einen Klick auf die Schaltfläche „aktualisieren“ im unteren Bereich der Anzeige auf den letzten Stand gesetzt werden kann.

Grundsätzlich gilt, dass der zu verwaltende Server in der Auflistung der vorhandenen Server selektiert werden muss.

Nun können wir für den ausgewählten Server je nach aktuellem Serverstatus die folgenden Aktionen ausführen:

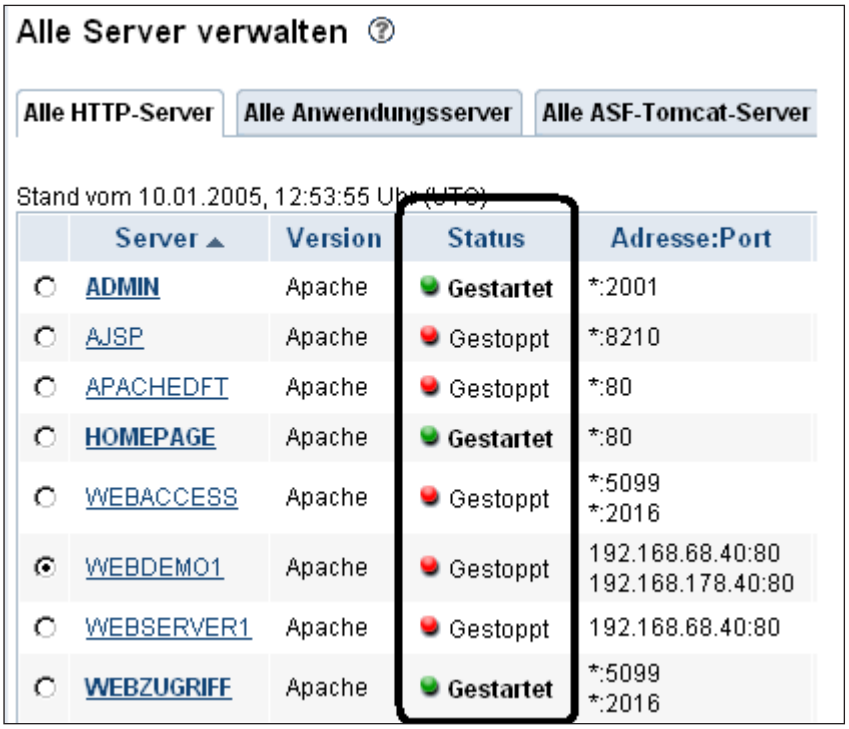
- Starten
- Stoppen
- Umbenennen
- Details verwalten

Von der Auswahl „Umbenennen“ sollte allerdings nur dann Gebrauch gemacht werden, wenn nicht weitergehende Konfigurationen etwa von Anwendungsservern auf diesem http-Server basieren.

Aktivitäten

Die Selektion „Details verwalten“ öffnet den allgemeinen Konfigurationsbereich eines http-Servers, wie wir ihn bereits bei der Einrichtung unseres Demo http-Servers kennen gelernt haben. Hier lassen sich alle Konfigurationsanpassungen an dem Server durchführen.

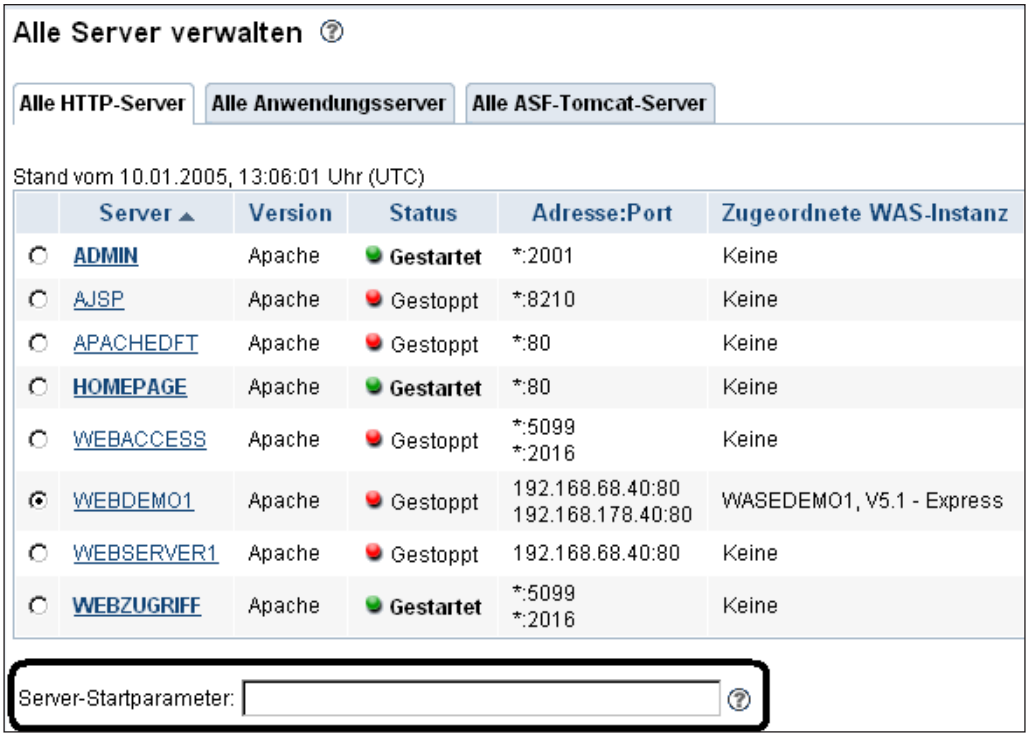
In der Auflistung der Server selbst lässt sich der jeweilige Serverstatus leicht über das entsprechende Symbol erkennen.



Serverstatus

Einen wichtigen Punkt stellt die optionale Auswahl „Server.Startparameter“ unterhalb der Serverauflistung dar.

5.1.3.2
Seite 4



Server Startparameter

Mit Hilfe dieser Startparameter lassen sich verschiedene Startoptionen eingeben:

- *netccsid [nnn]* Überschreibt die Direktive DefaultNetCCSID, die in der Server-Konfiguration angegeben ist.
- *-fsccsid [nnn]* Überschreibt die Standarddirektive DefaultFsCCSID (nicht zu verwechseln mit netccsid!).
- *-d [serverroot]* Ändert den Anfangswert für die Variable ServerRoot in serverroot. Dieser Parameter kann von der Direktive ServerRoot überschrieben werden.
- *-f [configuration]* Bei dem Start sollen die in der Konfiguration enthaltenen Werte verwendet werden.
  Wenn der Konfigurationspfad nicht mit einem Schrägstrich (/) beginnt, wird er als Pfad behandelt, der sich relativ zur Server-Root verhält.
- *-C [directive]* Die angegebene Direktive so verarbeiten, als wäre sie Teil der Konfiguration gewesen.
- *-c [directive]* Die angegebene Direktive nach dem Lesen aller regulären Konfigurationsdateien verarbeiten.
- *-vv* Ausführliche Service-Überwachung aktivieren. Der Parameter -vv wird bei einem Neustart ignoriert und bewirkt kein Ein- oder Ausschalten der Tracefunktion.

- *-vi* Service-Überwachung auf Informationsebene aktivieren.
- *-ve* Service-Überwachung auf Fehlerebene aktivieren.
- *-V* Basisversion des Servers, Erstellungsdatum und eine Liste der Einstellungen für die Kompilierzeit anzeigen, dann Vorgang beenden.
- *-l* Eine Liste aller kompilierten Module anzeigen.
- *-t* Die Syntax der Konfigurationsdatei testen, den Server jedoch nicht starten.
  Mit diesem Befehl wird geprüft, ob alle Einträge der Dokument-Root vorhanden sind und Verzeichnisse darstellen.

Die Verwendung dieser Parameter macht besonders dann Sinn, wenn Fehler auf dem Server gesucht werden müssen. Für den Standardstart eines Servers muss in diesem Feld keine Angabe gemacht werden.

5.1.3.2

Seite 6

Einen http-Server verwalten

Bei der Erstellung des http-Servers haben wir den Konfigurationsbereich für http-Server bereits kennen gelernt. Spätere Anpassungen an einem http-Server lassen sich mit demselben Konfigurations-Bereich durchführen. Dieser ist in den Basisbereich und die komplexere Konfiguration grob untergliedert.

Für einfache Einsatzgebiete genügt in der Regel die Verwendung der Basisverwaltung des http-Servers. Soll dieser allerdings für weitere Einsatzgebiete als zum Beispiel die Standardfunktionen der IBM eingesetzt werden, die in einigen Bereichen der i5/OS-Lizenzprogramme zur Verfügung gestellt werden, dann sind noch weitergehende Konfigurationsanpassungen erforderlich, die ich Ihnen in diesem Abschnitt näher erläutern möchte. Dabei werden die Einzelbereiche, die wir zum Teil bereits im dem Abschnitt „http-Server Basisverwaltung“ kennen gelernt haben, nochmals vertieft.

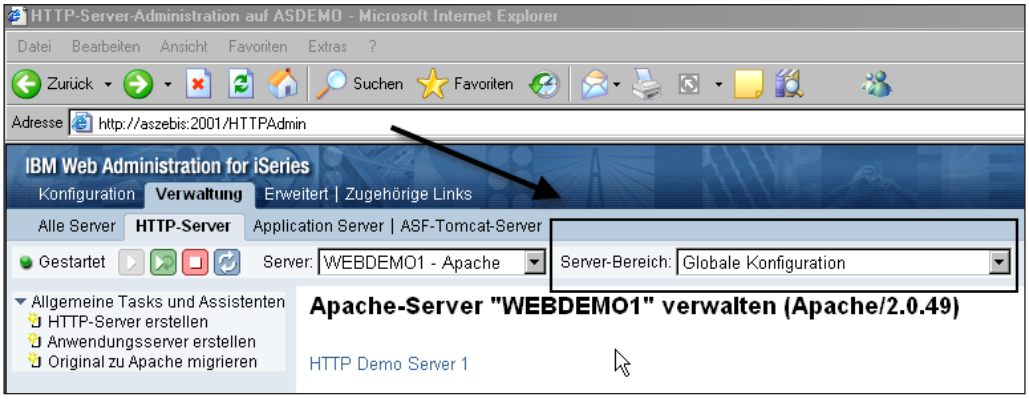
Es gibt verschiedene Ansätze, die Konfiguration eines http-Servers zu erläutern. Zudem ist die Vielfältigkeit der Konfigurationsmöglichkeiten recht komplex – denn die Einsatzbereiche können in komplexen Umgebungen sehr weit reichend sein. Wir wollen in diesem Abschnitt nicht auf alle Details der http-Konfiguration eingehen, sondern uns mit den häufigsten Einstellungen beschäftigen. Dabei geht es sowohl darum, einen http-Server in Bezug auf die Sicherheit zu optimieren, einfache Anwendungen auf diesem lauffähig zu machen und weitere Einstellungen zu erläutern, die im Umgang mit den http-Servern in Verbindung mit Anwendungen und Einsatzmöglichkeiten in einer i5-Umgebung von Bedeutung sind.

Wichtig für die verschiedenen Einsatzbereiche eines http-Servers ist natürlich das Verständnis für die Funktionen und Möglichkeiten, die ein solcher Server bietet. Über die Erläuterung der unterschiedlichen Konfigurationsbereiche lassen sich leicht auch Rückschlüsse auf Funktionen und Lösungsansätze bieten.

Schauen wir uns nun die verschiedenen Bereiche eines http-Servers genauer an.

Dazu öffnen wir zunächst nochmals die Einstiegsseite für die Administration des/der http-Server.

5.1.3.3
Seite 2



Administrationsbereich starten

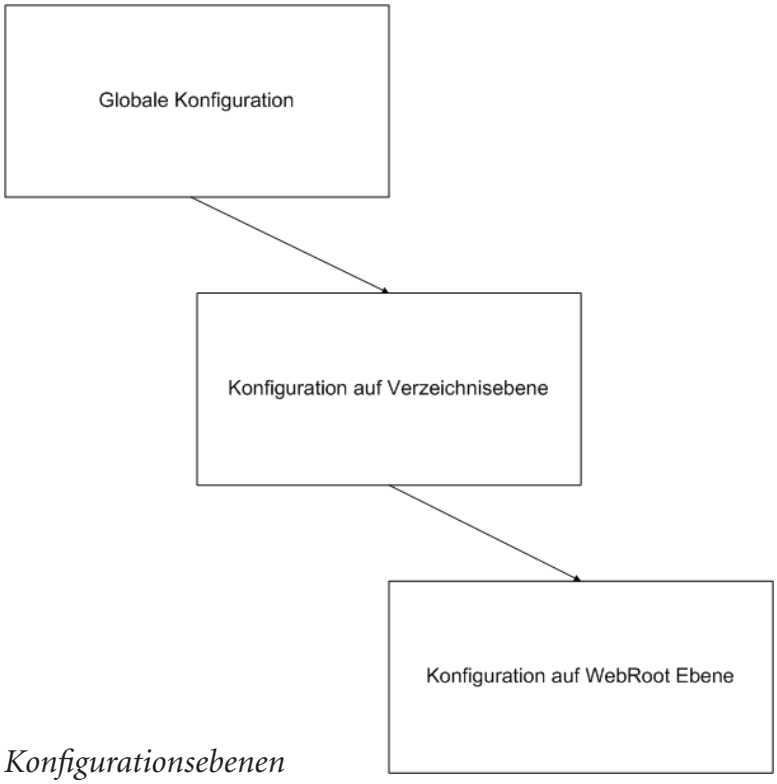
In der Anzeige wählen wir den Bereich „http-Server“ aus und selektieren den gewünschten http-Server – in unserem Beispiel den Server „WEBDEMO1“.

Rechts neben der Auswahl der verschiedenen installierten http-Server befindet sich der „Server-Bereich“.

Dieser ist unterteilt in:

- Globale Konfiguration
- Verzeichnis /
- Verzeichnis /www/servername/htdocs (WebRoot)

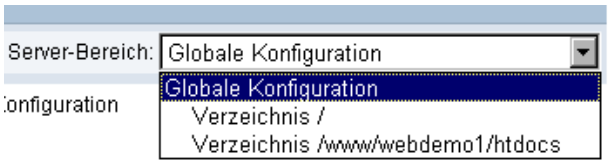
Dies sind die drei unterschiedlichen Teilbereiche des http-Servers, die jeder für sich verwaltet werden können und hierarchisch zu verstehen sind.



Konfigurationsebenen

Je nach dem, welche dieser drei Ebenen ausgewählt wurden, sehen die Verwaltungsanzeigen unterschiedlich aus. Zudem kommt es auf die Art der Änderungen an, die durchgeführt werden, ob diese sich nur auf die aktuelle Ebene oder auch auf die übrigen Ebenen auswirken – doch dazu später mehr.

Diese Teilbereiche des http-Servers lassen sich im oberen Auswahlbereich der Browseranzeige auswählen.



Konfigurationsebenen-Auswahl

In dem sichtbaren Feldbereich wird jeweils die aktuell gewählte Konfigurationsebene angezeigt. Der gewünschte Bereich lässt sich in diesem Feld auswählen. Achten Sie unbedingt bei der späteren Anpassung an den Konfigurationseinstellungen darauf, dass Sie sich in dem korrekten Bereich befinden.

Im linken Navigationsbereich der Anzeige befinden sich verschiedenen Teilbereiche – unter anderem auch die Bereiche für die http-Server-Konfiguration.

Die Konfigurationsbereiche sind bereits optisch in Form von einzelnen Blöcken unterhalb des Auswahlpunktes „Servereigenschaften“ dargestellt:

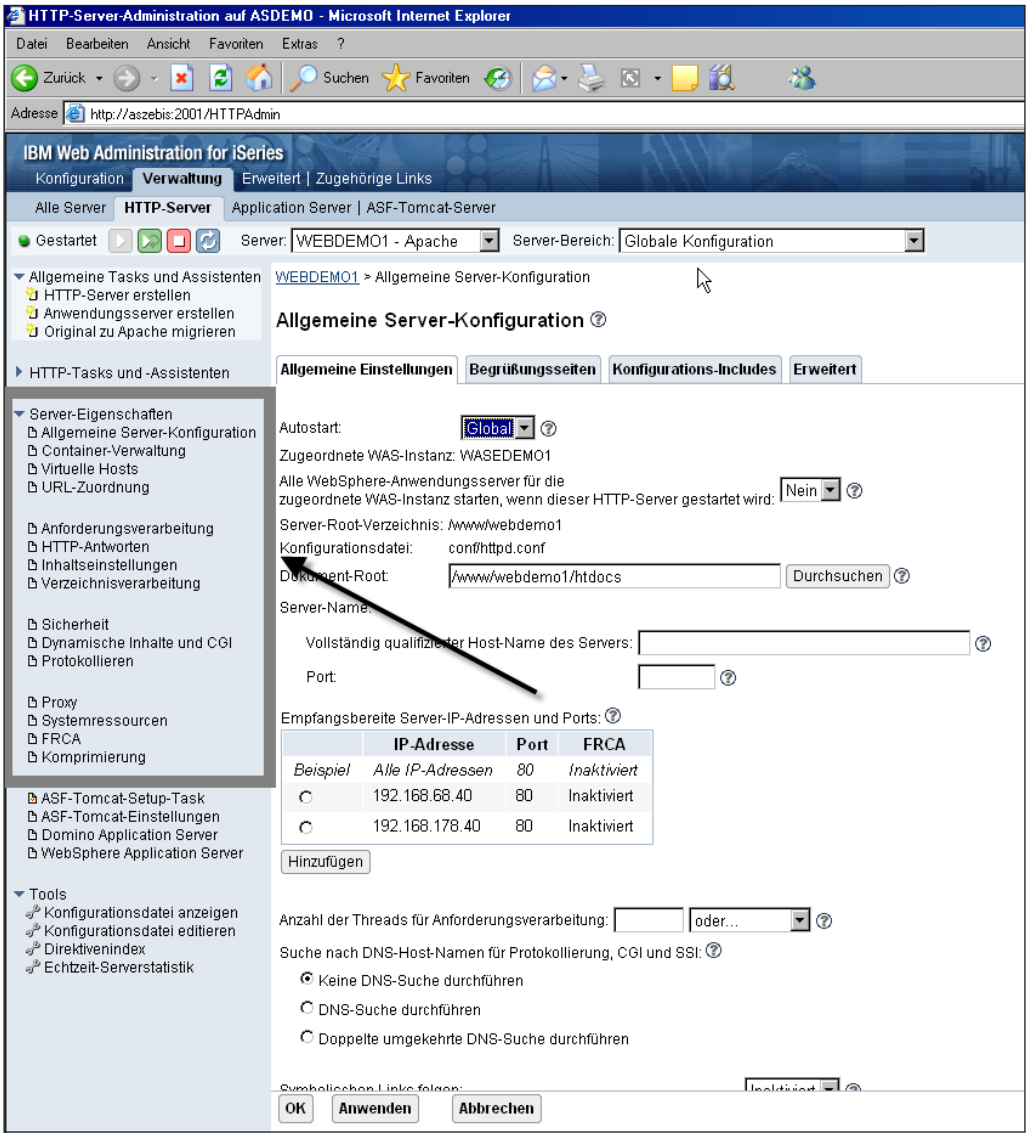
- Allgemeine Server-Konfiguration
- Container Verwaltung
- Virtuelle Hosts
- URL-Zuordnung

- Anforderungsverarbeitung
- http-Antworten
- Inhaltseinstellungen
- Verzeichnisverarbeitung

- Sicherheit
- Dynamische Inhalte und CGI
- Protokollieren

5.1.3.3
Seite 4

- Proxy
- Systemressourcen
- FRCA
- Komprimierung



Konfigurationsbereiche

In den nächsten Abschnitten beschäftigen wir uns mit den unterschiedlichen Einstellungsmöglichkeiten auf den verschiedenen Ebenen.

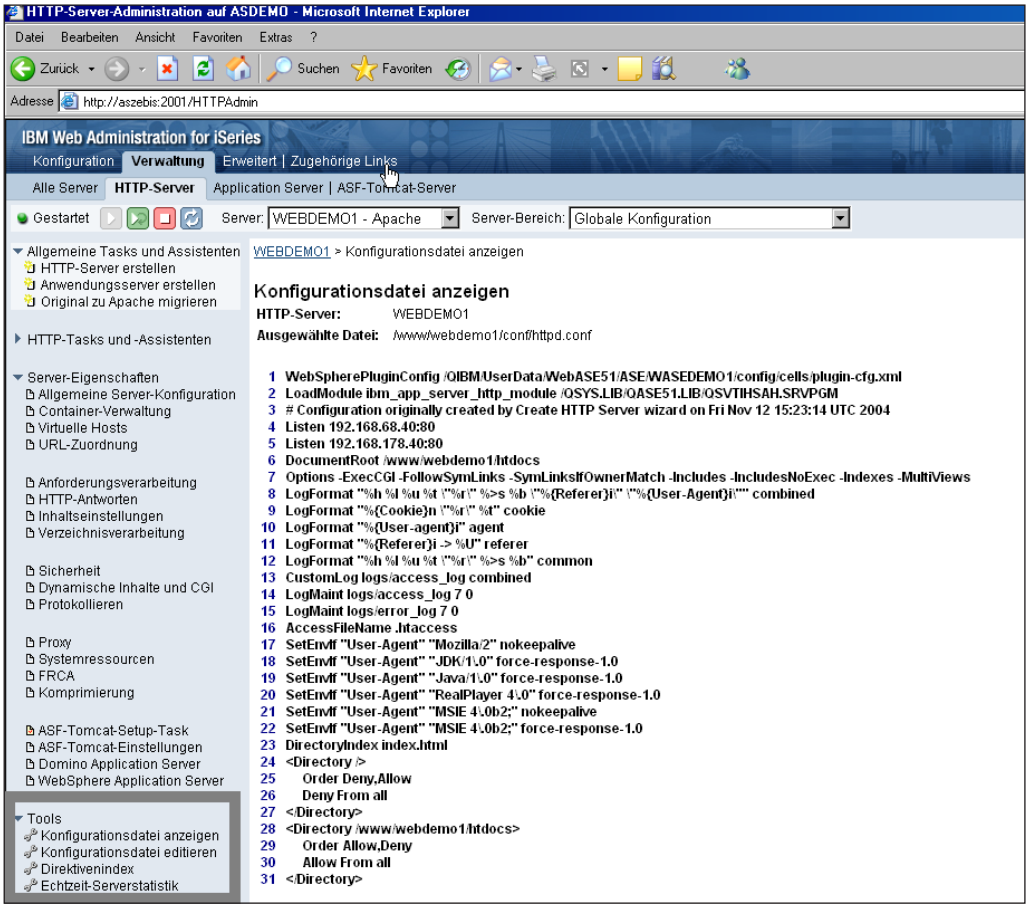
Beachten Sie bitte, dass sich diese Konfigurationseinstellungen jeweils auf einen http-Server beziehen und nicht global für alle http-Server gelten, die auf der i5 installiert sind.

Anmerkung:

Die Konfigurationseinstellungen, die wir nun in diesem Abschnitt vornehmen, werden alle in den Konfigurationsdateien des jeweiligen http-Servers gespeichert. Es ist sicher Geschmacksache, aber erfahrene http-Administratoren verwenden an Stelle der grafischen Konfigurationswerkzeuge, wie die von uns verwendete Administrations-Oberfläche die Direkteingabe der Zeichenstrings in einer Textdatei.

Grundsätzlich werden alle Konfigurationseinstellungen aus dem Browser-Administrations-Bereich heraus in Konfigurations-Dateien in Form von Text-Dateien geschrieben, die bei entsprechender Kenntnis auch nativ editiert und verändert werden können.

Diese Konfigurationsdateien können entweder in einem Editor geöffnet werden oder im unteren Abschnitt des linken Navigationsbereichs unter dem Eintrag „Tools“ aufgerufen werden.



Konfigurationsdateien auswählen

5.1.3.3
Seite 6

Ein Beispiel einer solchen Konfigurationsdatei finden wir bereits auf Basis der zuvor erstellten Grundkonfiguration eines http-Servers. Diese können wir mit Hilfe des Auswahlpunktes „Konfigurationsdatei anzeigen“ in dem Hauptbereich der Browseranzeige anschauen.

Mit etwas Übung lassen sich sicher auch schnell Veränderungen an den Konfigurationseinstellungen mit Hilfe eines Editors realisieren – wir wollen an dieser Stelle allerdings bewusst auf diese Anpassungsmethode verzichten, denn sie hat einen entscheidenden Nachteil: es werden keine Eingabeprüfungen durchgeführt, sondern alle Eintragungen ohne jegliche Plausibilitätsprüfung übernommen.

Die grafische Administrationsoberfläche bietet hingegen in den meisten Bereichen eine gewisse Prüfung in Bezug auf die getätigten Eingaben und ist allein aus diesem Grund für den Beginn der Arbeit mit http-Servern besser geeignet.

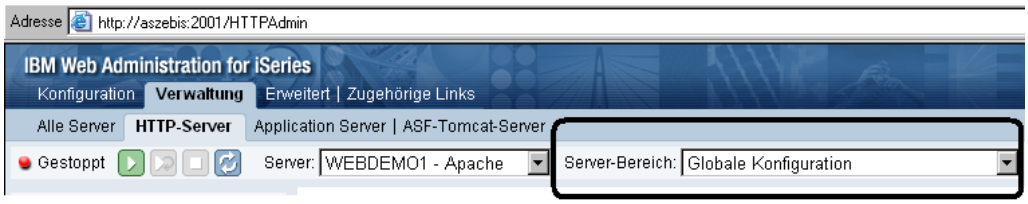
Globale Serverkonfiguration

Ein http-Server besteht, wie bereits erwähnt, aus verschiedenen Ebenen. Die oberste Ebene – auch als „Globale Konfigurationsebene“ bezeichnet, legt generelle Einstellungen für den http-Server fest, die für diesen Server und die darunter befindlichen Ebenen gelten. Zum Teil können diese Einstellungen durch Konfiguration auf den Verzeichnisebenen individuell für die jeweilige Ebene überschrieben werden.

Der Bereich der „Allgemeinen Serverkonfiguration“ kann je nach ausgewähltem Server-Bereich (globale Konfiguration / Verzeichnisebene / Verzeichnisroot) unterschiedlich aussehen. Es ist in jedem Fall darauf zu achten, dass der korrekte Server-Bereich in dem Auswahlfeld oberhalb des Konfigurationsbereichs selektiert wurde.

In diesem Abschnitt werden wir die diversen Teilbereiche jeweils für die einzelnen Server-Bereiche beschreiben. Dabei wollen wir nicht auf alle einzelnen Parameter eingehen, sondern uns auf die Wesentlichen beschränken.

Wir stellen also zunächst sicher, dass wir den korrekten Server-Bereich ausgewählt haben – für unsere nun folgende Beschreibung arbeiten wir in dem Bereich „Globale Konfiguration“.



Globale Konfiguration auswählen

Im linken Navigationsbereich finden wir die verschiedenen Konfigurationsabschnitte unterhalb des Eintrages „Server-Eigenschaften“, die logisch zusammengefasst sind.

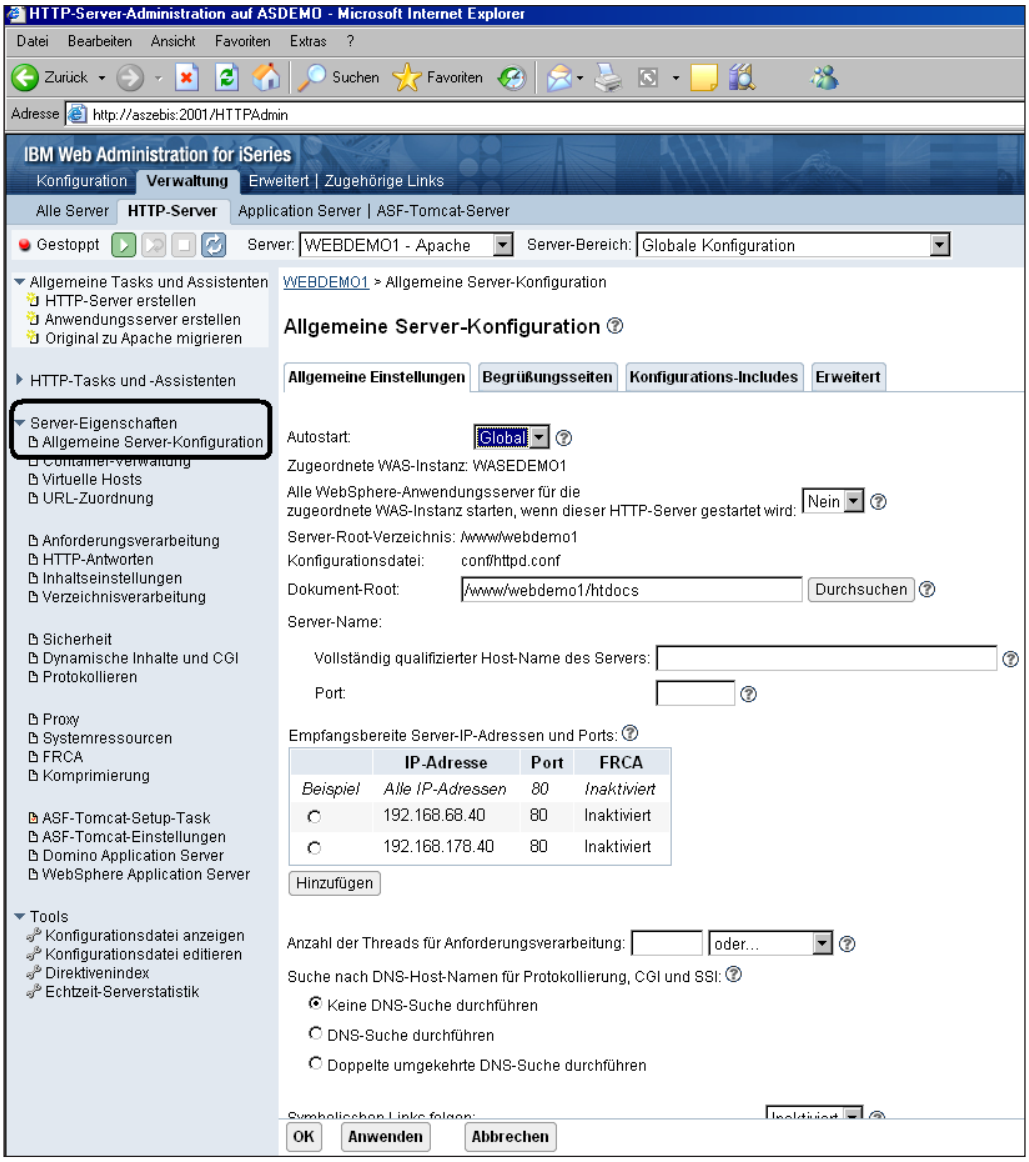
Wir wählen dort nun den Eintrag „Allgemeine Server Konfiguration“ aus. Die neue Anzeige ist in vier Tabellenbereiche unterteilt:

- Allgemeine Einstellungen
- Autostart Einstellungen
- Zugeordnete Applikations-Server
- Dokumenten-Root
- Server-Name
- IP-Bereiche
- DNS-Angaben
- Begrüssungsseite

5.1.3.3.1

Seite 2

- Angabe der Startseite, die bei dem Serverstart erscheinen soll
- Alternativ auch Suchangaben nach Startseite
- Konfiguration-Includes
- Für die Angabe weiterer Konfigurationsdateien, die sich außerhalb des http-Server Bereiches befinden
- Erweitert
- Hier können OS/400 spezifische Angaben hinterlegt werden



Allgemeine Server Konfiguration

Allgemeine Einstellungen

Der erste Tabellenbereich „Allgemeine Einstellungen“ beinhaltet einige wichtige Grundangaben für den http-Server. Diese sind in der Regel bereits durch die Basiskonfiguration eingestellt, müssen aber in der Praxis – und hier dann meist in komplexeren Umgebungen – doch hin und wieder angepasst werden.

Autostart

Zunächst können wir in diesem Tabellenabschnitt angeben, wie und ob der http-Server gestartet werden soll. Für den Serverstart stehen drei Optionen (Ja / Nein / global) zur Auswahl, von denen wohl lediglich die Option „global“ einer Erläuterung bedarf.

Grundsätzlich beziehen sich diese Startangaben auf das Mitstarten des http-Servers bei dem Start von TCP/IP auf dem Server – also im konkreten Fall der i5, ob dieser gerade verwaltete http-Server im Zuge von STRTCP mitgestartet werden soll oder nicht.

„Global“ bedeutet, dass dieser http-Server nach den für alle http-Server gültigen Einstellungen zusammen mit TCP/IP gestartet werden soll.

In der Praxis hat es sich als sinnvoll erwiesen, die Einstellungen für den Autostart für jeden http-Server festzulegen und nicht die Option „global“ zu verwenden.

Zugeordnete WAS-Instanz (optional)

Dieser Bereich wird nur dann angezeigt, wenn dieser http-Server mit einem Anwendungs-/Applikationsserver zusammen arbeitet.

Wie wir in dem Abschnitt „Applikations-Server“ beschrieben haben, arbeiten Applikations-Server in der Regel zusammen mit http-Servern. Bei der Konfiguration eines Applikations-Servers wird der dazugehörige http-Server angegeben. Sollte dem http-Server ein Applikations-Server zugeordnet sein, dann kann an dieser Stelle festgelegt werden, ob die Applikations-Server Instanz automatisch mitgestartet werden soll. Der Standardwert in diesem Bereich ist „nein“ und kann bei Bedarf geändert werden. Dann werden alle Applikations-Server Instanzen nach dem Start dieses http-Servers automatisch mitgestartet.

Dokument-Root

An dieser Stelle wollen wir noch ein wenig tiefer auf die Bedeutung des Dokument-Root eingehen, denn es ist von außerordentlicher Bedeutung – aus diesem Root (zu deutsch: aus diesem Verzeichnis) werden alle Dateien für diesen http-Server geladen und den Anwendern, die auf diesen http-Server zugreifen, verfügbar gemacht.

5.1.3.3.1
Seite 4

Hiermit geben wir also den Speicherort aller HTML-Dokumente an, die für diesen Server verwendet werden sollen.

An dieser Stelle sollte unter keinen Umständen das physische Root-Verzeichnis des Servers angegeben werden – ansonsten werden es Ihnen Hacker danken! Hier sollte das Unterverzeichnis angegeben werden, in dem sich die Daten befinden.

Dies sollte in der Form

/www/servername/htdocs

erfolgen.

Achtung

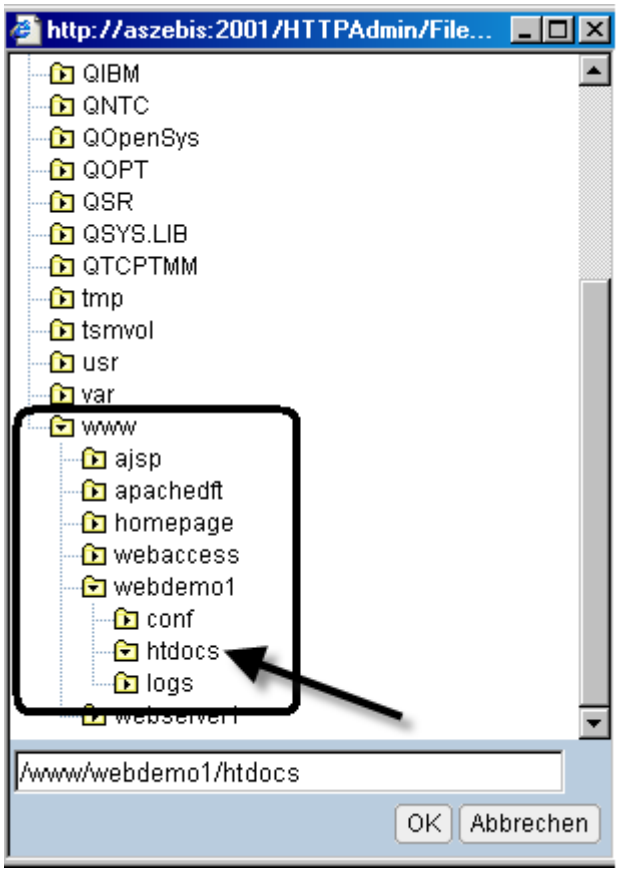
An dieser Stelle ist das einleitende „/“ Zeichen sehr wichtig! Wird dieses vergessen, dann sind Fehler die Folge!

Wie kommen wir nun zu dem Namen des Dokument-Root?

Schauen wir uns dazu nochmals die Beispielkonfiguration unseres http-Servers „Webdemo1“ an.

Dieser wurde wie bereits erläutert, in dem IFS der i5 gespeichert und zwar innerhalb des Verzeichnisses „www“.

Für die Angabe des Dokument-Root können wir in unserem Browser-Fenster auf die Schaltfläche „Durchsuchen“ klicken – entgegen einiger IBM-Dokumentationen wird hier auch nicht auf das Verzeichnis „QIBM/UserData/HTTPA“ verwiesen, sondern auf das korrekte Verzeichnis „www/servername/htdocs“.



Dokument-Root auswählen

Wir merken uns also:

Per Standard wird das Dokument-Root für jeden http-Server in dem IFS-Verzeichnis „www“ und dort unter dem Servernamen des http-Servers gespeichert.

Alternativ dazu kann auch ein anderes Dokument-Root angegeben werden – dieses muss dann in diesem Parameter hinterlegt werden.

5.1.3.3.1
Seite 6

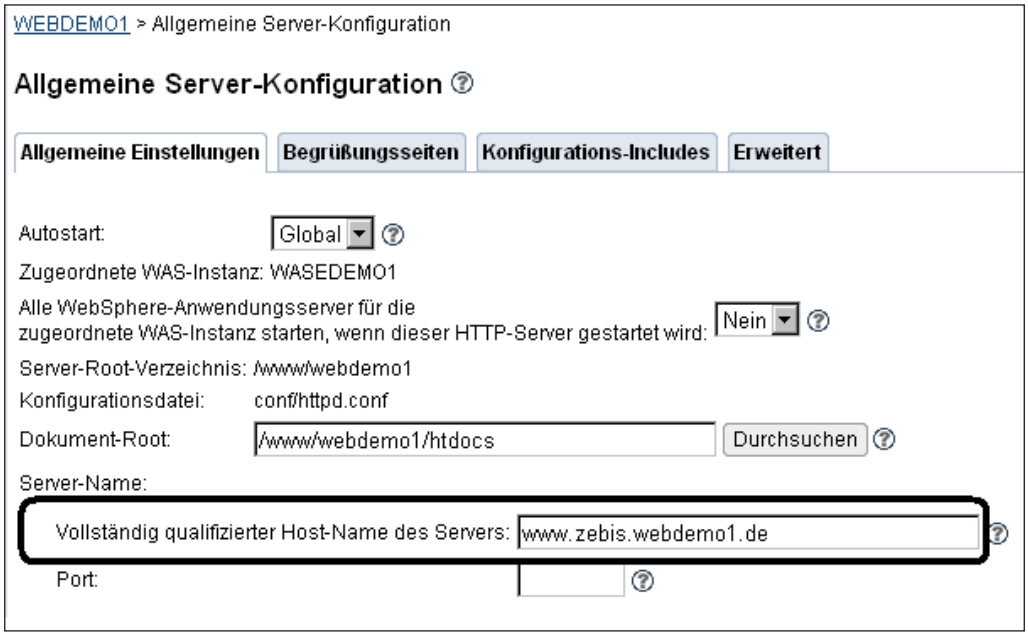
Server-Name

Ein wichtiger Parameter verbirgt sich hinter dem Eintrag „Qualifizierter Hostname des Servers“. Per Standard ist hier kein Eintrag vorhanden – es empfiehlt sich, hier einen qualifizierten Namen einzutragen. Anderenfalls wird – zumindest bei der Verwendung von DNS – versucht, den qualifizierten Namen zu ermitteln.

Unter einem qualifizierten Namen verstehen wir im Zusammenhang mit einem http-Server die Angabe des Servernamens in Verbindung mit der Domäne, für die dieser Server konfiguriert wurde.

In der Regel wird ein http-Server dazu verwendet, Informationen zur Verfügung zu stellen. Nun verbirgt sich hinter einem http-Server zunächst eine Speicherlokation, die durch eine eindeutige IP-Adressierung angesprochen werden kann. Diese Identifikation wird über eine IP-Adresse (und einem dazugehörigen Port) realisiert – nur: wer verwendet schon IP-Nummern in den Weiten des Datennetzwerkes?

Deshalb wird dem Server ein Name zugeordnet der – ergänzt mit den Domänenangaben – eindeutig sein sollte.



Qualifizierter Servername

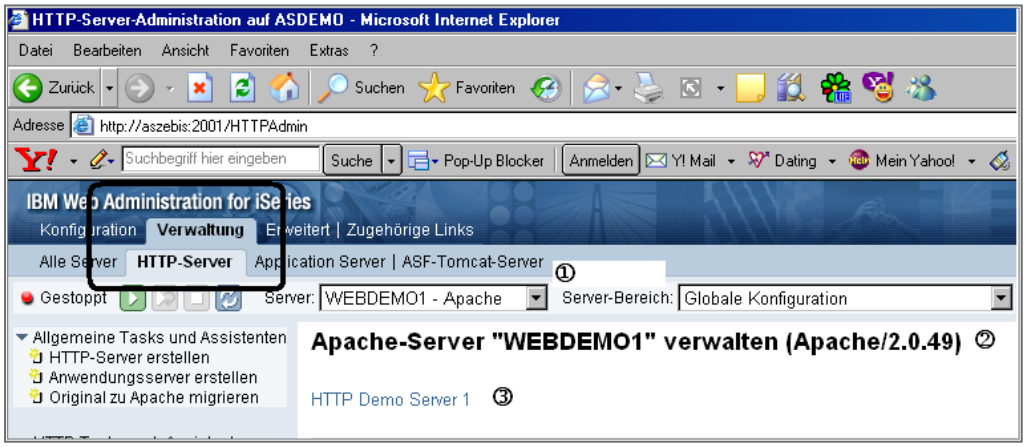
HTTP-Server-Basisverwaltung

Die Administration der HTTP-Server auf der i5 wird zentral über den Admin-Client – also über den Port 2001 (bzw. 2010) – durchgeführt.

In diesen Bereich kann jederzeit während der Arbeit mit dem Administrations-Client gewechselt werden. Wie wir eben gesehen haben, wird der Verwaltungsbereich automatisch nach der Konfiguration eines neuen Servers geöffnet.

Schauen wir uns zunächst den Aufbau der Server-Verwaltungsseite an.

Den oberen Bereich haben wir bereits kennen gelernt. Hier ist nun der Abschnitt „HTTP-Server“ vorausgewählt.



Verwaltungsbereich

Wie wir sehen, ist der soeben konfigurierte Server bereits voreingestellt. Dies lässt sich sowohl an dem Auswahlfeld „Server“ (j) als auch in der Überschrift der mittleren Anzeige (k) erkennen.

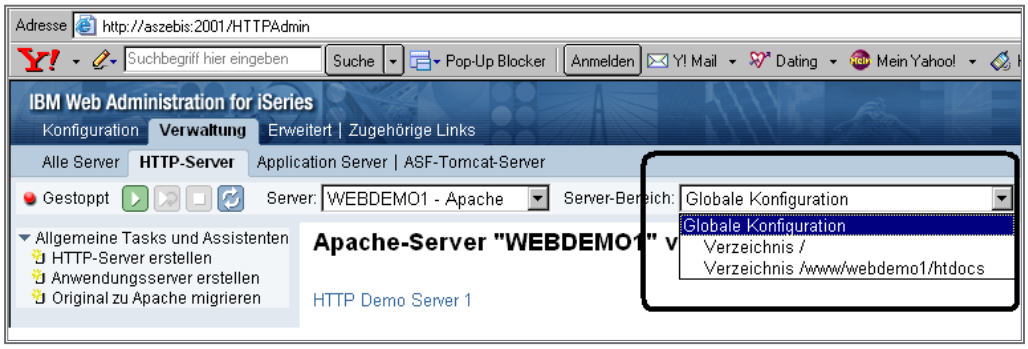
Unterhalb der Überschrift finden wir auch die Bezeichnung (den Langtext) für unseren Server wieder (l).

Rechts neben dem Server in der oberen Browser-Zeile befindet sich ein zusätzliches Auswahlfeld, das derzeit mit „Globale Konfiguration“ vorbelegt ist.

Wir unterscheiden bei der Konfiguration eines HTTP-Servers zwischen den Bereichen:

- Globale Konfiguration
- Stammverzeichnis (/)
- Root-Verzeichnis (/ www/servername/htcos)

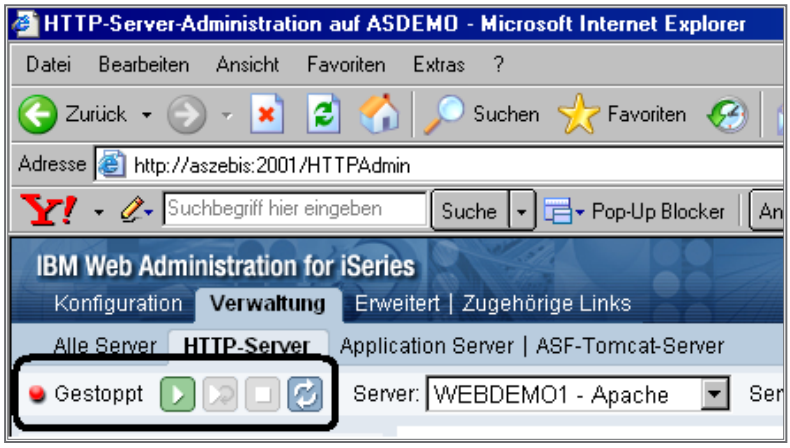
5.1.4
Seite 2



Server-Bereiche

Wir lassen an dieser Stelle die Vorauswahl „Globale Konfiguration“ stehen.

Jeder HTTP-Server kann verschiedene Stati haben. Nach einer Neukonfiguration ist der HTTP-Server gestoppt. Dies erkennen wir sowohl an dem Text „Gestoppt“ als auch an dem roten Kreis links neben dem Statustext.



Server-Status

Wie unschwer vorstellbar ist, kann der Status eines HTTP-Servers natürlich auch „gestartet“ sein.

Neben der Statusanzeige befinden sich vier Symbole, die für die Steuerung des Servers in Bezug auf dessen Status verwendet werden.

Wir wollen an dieser Stelle unseren Server – oder genauer: unsere Server-Instanz – starten. Dazu verwenden wir das entsprechende Symbol neben der Statusanzeige in Form eines Pfeils in einem grünen Symbolbereich.

Server-Instanz starten

Je nach eingesetzter i5 (oder iSeries) kann es einen kleinen Augenblick dauern, bis die Server-Instanz gestartet wird.

Der erfolgreiche Abschluss des Startvorgangs wird mit einer entsprechenden Änderung der Statusanzeige quittiert.

Server-Instanz gestartet

Sollte sich die Statusanzeige nicht ändern, das heißt, erscheint nicht „Gestartet“ als aktueller Status der Server-Instanz, dann hat etwas bei der Konfiguration nicht richtig funktioniert. Oft liegen die Probleme in fehlenden Berechtigungen oder TCP/IP-Einstellungen. Eine detaillierte Fehlerbeschreibung würde diesen Abschnitt sprengen; wir werden auf einige der „beliebtesten“ Fehler später noch genauer eingehen. Meist findet man jedoch im Fehlerprotokoll der i5 entsprechende Hinweise, warum das Starten einer Server-Instanz nicht ausgeführt werden konnte.

Es kann auch bei einigen iSeries etwas länger dauern; sollten sie nicht sicher sein, ob der Start des Servers bereits durchgeführt wurde, können sie jederzeit die Schaltfläche „Anzeigeaktualisierung“ verwenden (siehe weiter unten bei der Erklärung der einzelnen Statussymbole). Auch bei langsamen Maschinen sollte „irgendwann“ die Statusanzeige „Gestartet“ erscheinen.

Wir gehen an dieser Stelle davon aus, dass wir unsere Vorarbeiten fehlerfrei durchgeführt haben und der HTTP-Server den korrekten Status „Gestartet“ hat.

Wie sie sehen, ist der Status „Gestartet“ und die Anzeige der aktiven Symbole hat sich rechts neben der Statusanzeige geändert.

Die übrigen Symbole habe die folgende Bedeutung:

Server-Neustart

Hin und wieder kann es vorkommen, dass ein Server neu gestartet (rebooted) werden muss. Dies wird durch das grüne Symbol mit dem geschwungenen Pfeil durchgeführt und bedeutet zunächst ein Beenden des Servers, welches einen automatischen Neustart zur Folge hat.

Server-Neustart

Server-Stopp

Soll ein Server bzw. eine Server-Instanz beendet werden, dann verwenden wir das rote Symbol.

Server stoppen

5.1.4
Seite 4

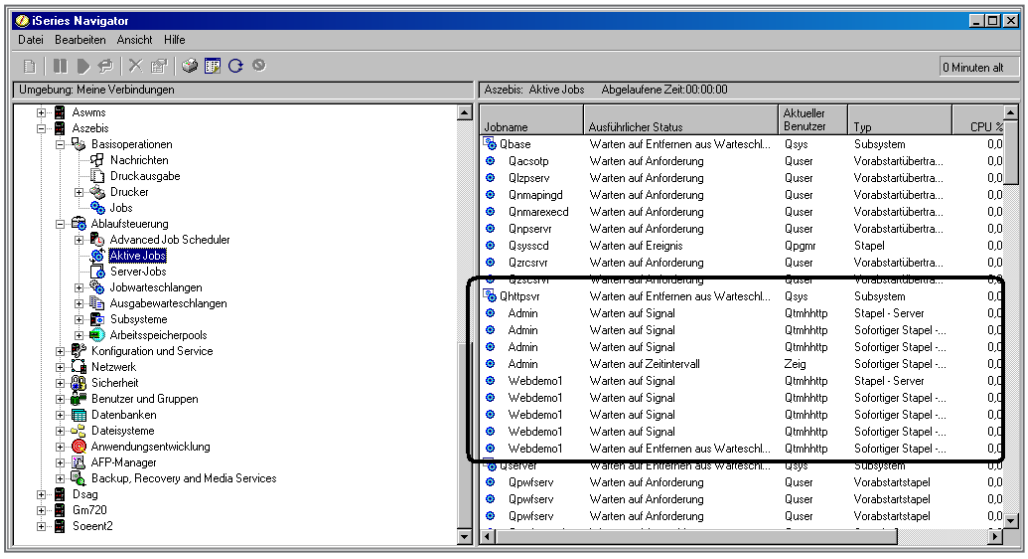
Anzeigeaktualisierung

Hin und wieder muss die Anzeige manuell aktualisiert werden. Dies ist nötig, weil einige Transaktionen keine automatische Aktualisierung der Anzeige ermöglichen. In solchen Fällen klicken wir einfach auf das blaue Symbol mit den beiden geschwungenen Pfeilen. Diese Funktion kommt in etwa der „F5“-Taste der i5 nahe.

Anzeige aktualisieren

Ob ein HTTP-Server wirklich gestartet wurde, lässt sich neben der zuvor genannten Methode auch auf der i5 prüfen.

Dazu verwenden wir einfach den iSeries Navigator und schauen uns die aktiven Jobs an.



Server gestartet

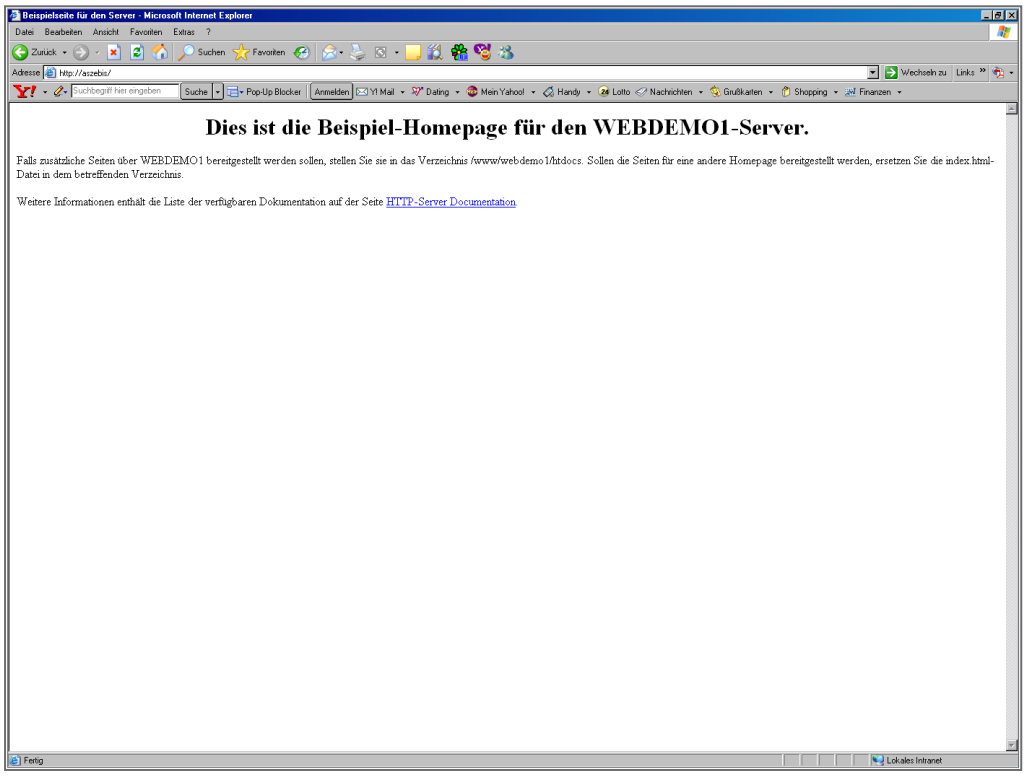
Wie sie sehen, befinden sich unterhalb des Subsystems QHTTPSVR (in dem die HTTP-Server ausgeführt werden) neben dem Eintrag „Admin“ auch einige weitere Einträge von unserem soeben gestarteten Server „Webdemo1“.

Nur der Vollständigkeit halber: Auch wenn wir in dem Bereich HTTP-Server wirklich losgelöst von Green-Screen-Anwendungen arbeiten, erhalten wir die gestarteten Server-Jobs auch in der Anzeige von „WRKACTJOB“.

Abschließend schauen wir uns noch an, ob wir den HTTP-Server auch von einem Browser aus erreichen können. Dazu geben wir den Host-Namen des Servers (der i5) in einem neuen Browser-Fenster ein.

Server-Name

Wenn wir alles richtig angegeben haben, dann sollte nun eine „Begrüßungsseite“ im Browser angezeigt werden.



Willkommensseite

Bei dieser Willkommensseite handelt es sich um das Dokument „index.html“, das während der Konfiguration des Servers automatisch erstellt wurde.

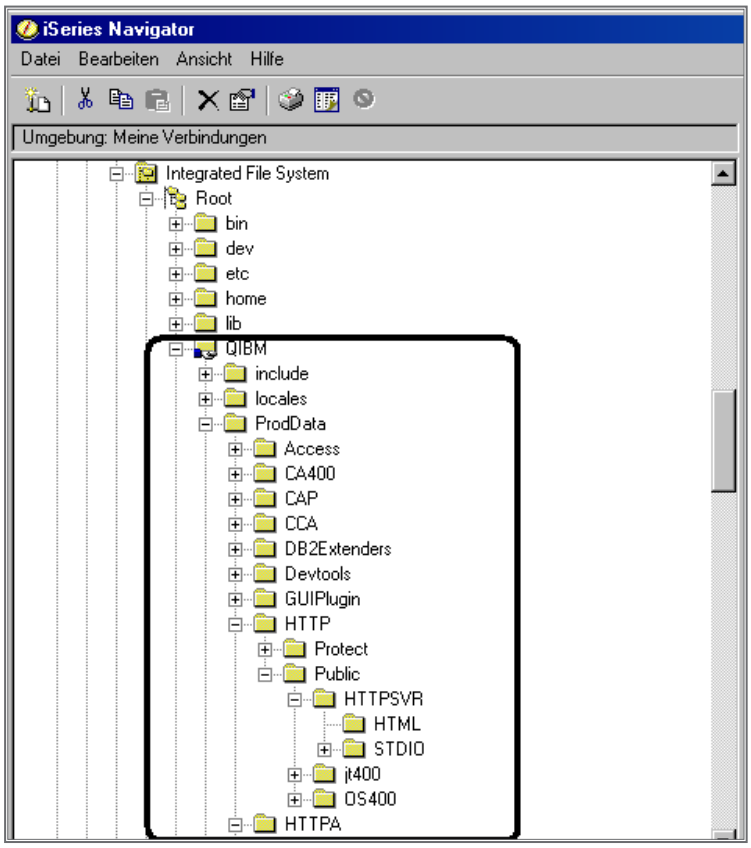
Mit diesem Schritt haben wir zunächst die Grundkonfiguration unseres HTTP-Servers beendet und können diesen nun für unsere Arbeit verwenden.

Schauen wir uns zum Verständnis der Integration des HTTP-Servers auf der i5 noch einmal folgende Punkte an:

- Der HTTP-Server ist als Lizenzprogramm auf der i5 integriert.
- HTTP-Server werden in einem eigenen Subsystem QHTTPSVR ausgeführt.

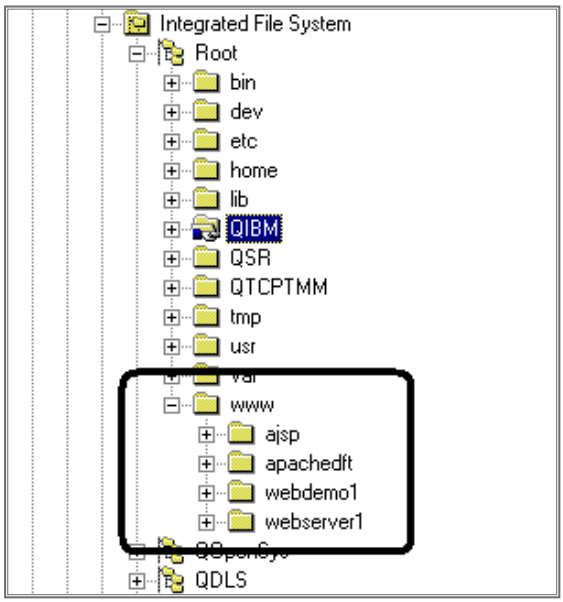
5.1.4
Seite 6

- Die Speicherung der Konfiguration der HTTP-Server und auch der Web-Seiten erfolgt innerhalb des IFS der i5.



IFS Integration HTTP-Server

In dem Unterverzeichnis „www“ befindet sich – wie man sieht – auch unser neu erstellter Server „WEBDEMO1“.



WEBDEMO1

Web-Anwendungen auf dem HTTP-Server

Nachdem wir einen HTTP-Server installiert und auch konfiguriert haben, stellt sich natürlich auch die Frage, was wir alles mit einem solchen Server machen können. Neben dem Einsatz in Verbindung mit einem Applikationsserver lässt sich ein solcher Webserver auch dafür einsetzen, um zum Beispiel einfachere Anwendungen auf Basis eines Browsers auszuführen. IBM bietet u. a. verschiedene Möglichkeiten an, mit denen wir zum Beispiel mit Hilfe eines Webservers auf den Inhalt der Datenbank der i5 in Form der DB2/UDB zugreifen können. Damit lassen sich beispielsweise einfache Anwendungen realisieren, in denen wir auf die Daten der i5 zugreifen und diese bei Bedarf sogar verändern können.

Wie wir bei der Installation des Webservers gesehen haben, besteht unter anderem auch die Möglichkeit, bestimmte Einstellungen für den Bereich CGI vorzunehmen. CGI bedeutet „Common Gateway Interface“ und stellt eigentlich eine standardisierte Schnittstelle für den Datenaustausch zwischen einem Webserver und einem Programm dar. Mit Hilfe dieser Technik sind wir in der Lage, mit HTML Zugriff auf die Daten der i5 zu erlangen und diese Daten in Webseiten einzubinden. In diesem Sinne stellt CGI eine einfache Variante dar, Webseiten mit dynamischen Inhalten zu füllen.

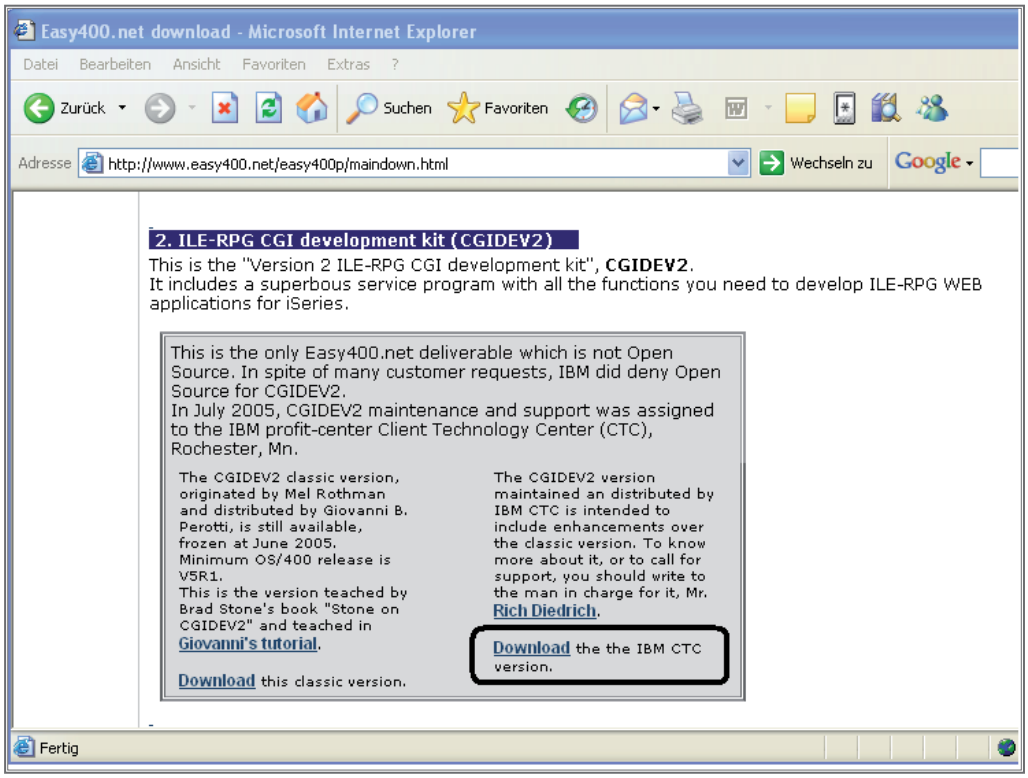
Allerdings ist dieser Ansatz nur für einfachere Webseiten oder auch kleinere Web-Anwendungen gedacht, denn neben dem Vorteil der relativ einfachen Anwendung und Realisierung ist es häufig die Ausführungsgeschwindigkeit, die im Vergleich zu Servlets oder auch PHP deutlich schwächer ist. Trotzdem stellt CGI eine ideale Möglichkeit dar, die ersten Schritte mit der Anbindung von i5 Daten ins Web zu wagen.

Bei CGI handelt es sich, vereinfacht gesagt, um ein Programm, das von einem http-Server aufgerufen werden kann und mit dem die Informationen von einem Datenbanksystem verarbeitet werden können. Das aufgerufene Programm kann in einem solchen Fall in einer beliebigen Sprache erstellt worden sein. Deshalb lassen sich im Zusammenhang mit der i5 zum Beispiel auch RPG- oder COBOL-Programme mit CGI einsetzen. Das aufgerufene Programm liest dann zum Beispiel Eingabefelder aus und führt eventuell erforderliche Berechnungen durch. Als Ergebnis können dann auch Daten aus der DB2/UDB zur Anzeige in einem Web-Browser gebracht werden.

Eines der am häufigsten eingesetzten Tools in dem Bereich von CGI mit i5 ist EASY400 bzw. CGIDEV2. Dabei handelt es sich um eine Ansammlung verschiedener Programme und Funktionen, mit deren Hilfe wir DB2/UDB-Daten in eine Web-Anwendung einbinden können. Dieses Tool wird von IBM kostenlos zum Download über die Adresse http://www.easy400.net/easy400p/maindown.html angeboten.

8. Ergänzung 9/2006

5.1.5
Seite 2

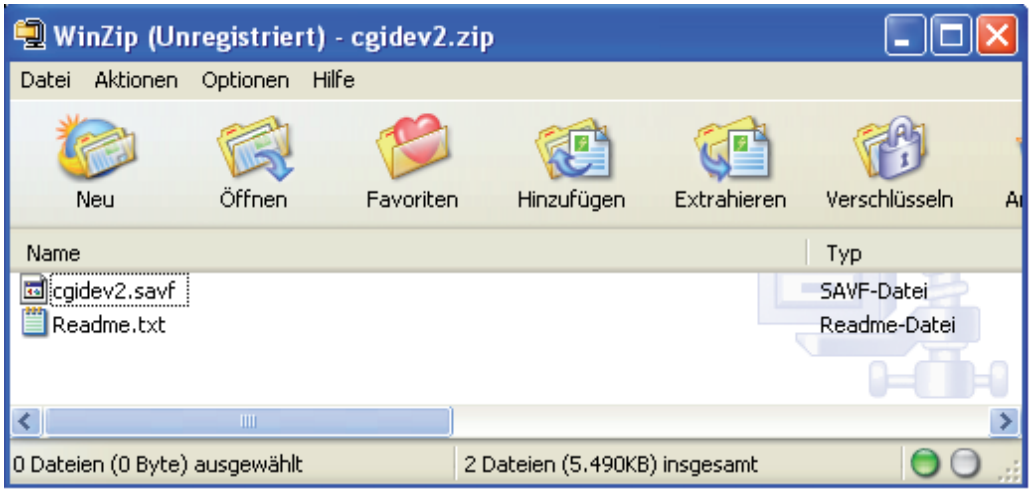


Download CGIDEV2

Auf dieser Webseite stehen neben dem CGIDEV2-Tool noch einige weitere sehr nützliche Tools zum Herunterladen bereit. Wir wollen uns an dieser Stelle aber auf das CGIDEV2-Tool beschränken.

Auf der zuvor genannten Webseite stehen uns zwei verschiedene Versionen zur Verfügung. Die klassische Version, die ab dem OS/400 V5R1 eingesetzt werden kann, ist auch die Basis für unsere Beispiele. Diese sollten Sie sich zunächst vom Internet herunterladen. Ich verzichte an dieser Stelle bewusst auf die einzelnen Schritte, die im Rahmen des Downloads in Form von Standardanzeigen ausgeführt werden müssen und beschränke mich auf die Installation und den weiteren Einsatz dieses Tools.

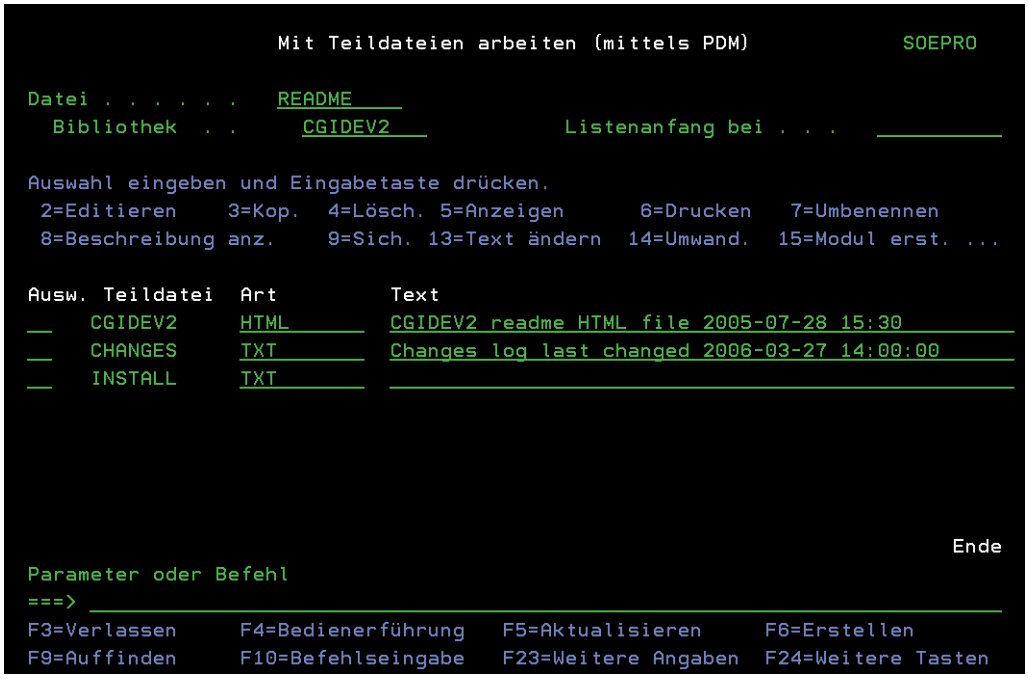
Der Download von CGIDEV2 erfolgt in Form einer Zipdatei, die zwei Einträge enthält: Eine Textdatei, welche weitere Informationen über CGIDEV2 enthält und eine Sicherungsdatei mit den auf der i5 zu installierenden Komponenten.



Inhalt der Zipdatei

Die Sicherungsdatei übertragen wir zunächst nach dem gewohnten Verfahren auf die i5 und speichern den Inhalt dort mit dem Befehl „RSTLIB SAVLIB(CGIDEV2) DEV(*SAVF) SAVF(CGIDEV2)“ zurück.

Um die Installation zu vervollständigen, befolgen Sie bitte die Anweisungen, die sich in der Datei README in der Bibliothek CGIDEV2 befinden:



Installationsanleitung

Nach der Installation des Tools sollten Sie die Anleitungen befolgen. Damit werden dann unter anderem die Voraussetzungen für einen sicheren und auch erfolgreichen Betrieb dieses Tools definiert. Wie bereits erwähnt, ist die Verwendung von CGIDEV2 sicher nur eine von vielen unterschiedlichen Möglichkeiten, Anwendungen für einen Webserver zu erstellen. Es beinhaltet aber bereits eine Fülle von Beispielen, mit denen sich viele der Anforderungen, wie häufig in einem Unternehmen in Bezug auf die Einbindung von i5 Daten in eine Webseite, behandelt werden.

Das Schöne an diesem Tool ist neben dem automatischen Installationsprozess auch die Fähigkeit, die verschiedenen Beispielanwendungen direkt auf dem System ausführen zu können und darüber hinaus auch die Programme, welche als Grundlagen für die Beispiele dienen, einzusehen. In der Regel besteht eine solche Web-Anwendung aus einem HTML-Bereich und einem dazugehörigen i5 Programm wie zum Beispiel ein RPG-Programm. Beide Bereiche sind in dem Lieferumfang von CGIDEV2 enthalten.

Bevor wir CGIDEV2 nutzen können, müssen einige Konfigurationseinstellungen vorgenommen werden. Ich verzichte bewusst an dieser Stelle auf das Abschreiben der Installationsanleitung, die im Internet unter der folgenden Adresse eingesehen werden kann:

http://www.easy400.net/cgidev2o/cgisetup.htm

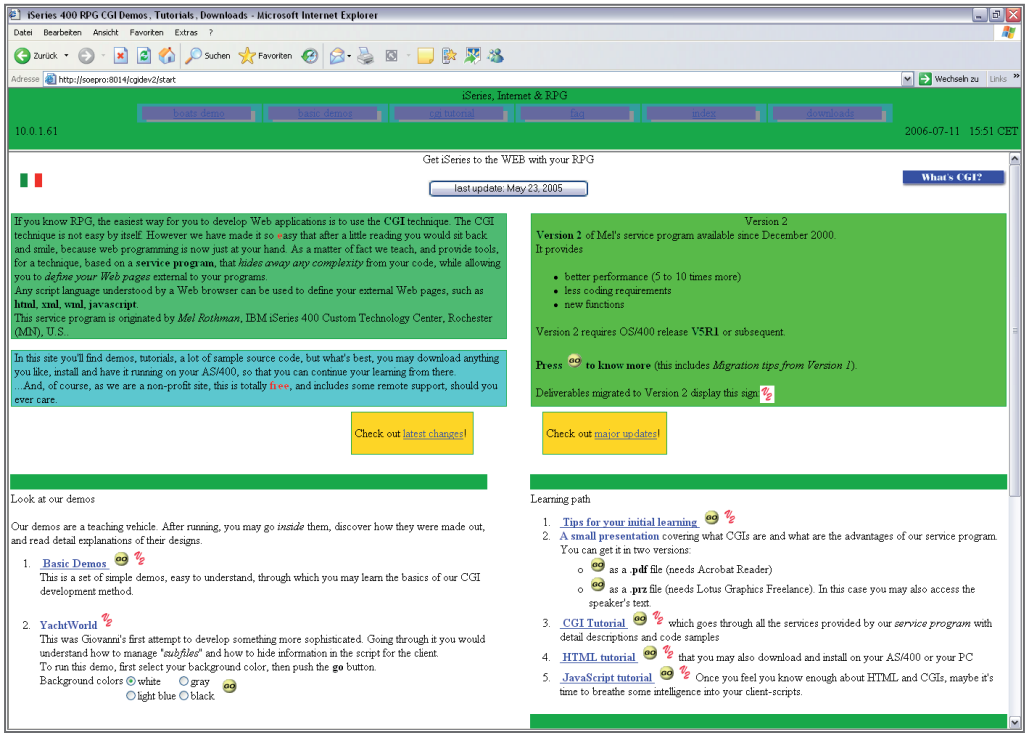
Befolgen Sie die auf dieser Webseite enthaltenen Installations- und Konfigurationsanleitungen. Mit der Ausführung dieser Anleitung wird auf dem Webserver der i5 eine vollständige Anwendung bereitgestellt, welche über eine Vielzahl unterschiedlicher Beispielanwendungen verfügt. Diese Beispiele können wir adaptieren und als Grundlage für eigene Anwendungen nutzen.

Geben Sie für den Aufruf der Web-Anwendung die folgende URL ein:

http://ipadresse:port

Ersetzen Sie in der URL „ipadresse“ mit dem Hostnamen bzw. der IP Adresse der i5 und verwenden Sie den Port, der diesem Webserver zugeordnet wurde.

Das Ergebnis dieser Eingabe sollte dann die folgende Anzeige sein:



Die Startseite des CGIDEV2 Tools auf dem Webserver

Nicht das Sie sich wundern, aber das, was wir jetzt sehen ist eine Webseite, die auf der i5 ausgeführt wird. Es handelt sich dabei um den Willkommensbildschirm des CGIDEV2 Tools. Wie Sie sehen können, besteht diese Anzeige aus verschiedenen Teilbereichen, mit denen wir Zugang zu den unterschiedlichsten Funktionen haben.

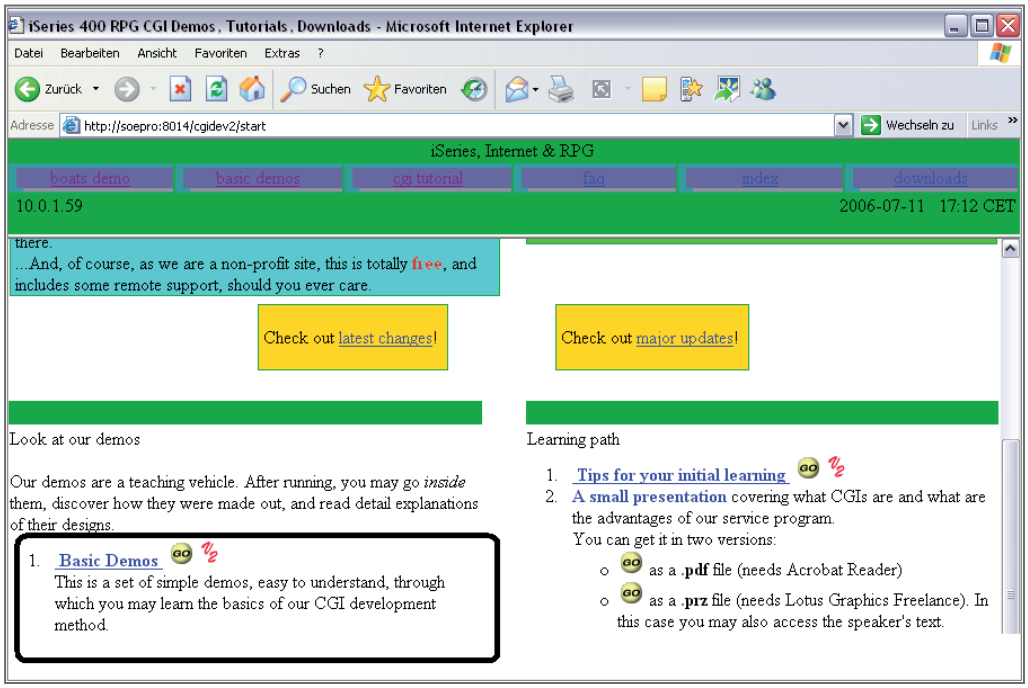
Mit einem Klick auf den Bereich „Basic Demos“ gelangen wir in den Bereich der Beispielprogramme. Das Schöne dabei ist, dass nicht nur die fertigen und lauffähigen Anwendungen von hier aus aufgerufen werden können, sondern dass wir auch sowohl den HTML-Code, der für diese Webseiten benötigt wird, als auch den RPG-Code von dieser Webseite aus aufrufen und einsehen können. Damit fällt ein Zugang in diese neue Welt der Anwendungsentwicklung wirklich leicht.

Ohne, dass wir uns mit jedem der einzelnen Beispiele im Detail beschäftigen, möchte ich an dieser Stelle eine der in CGIDEV2 enthaltenen Beispielanwendungen im Detail behandeln.

5.1.5
Seite 6

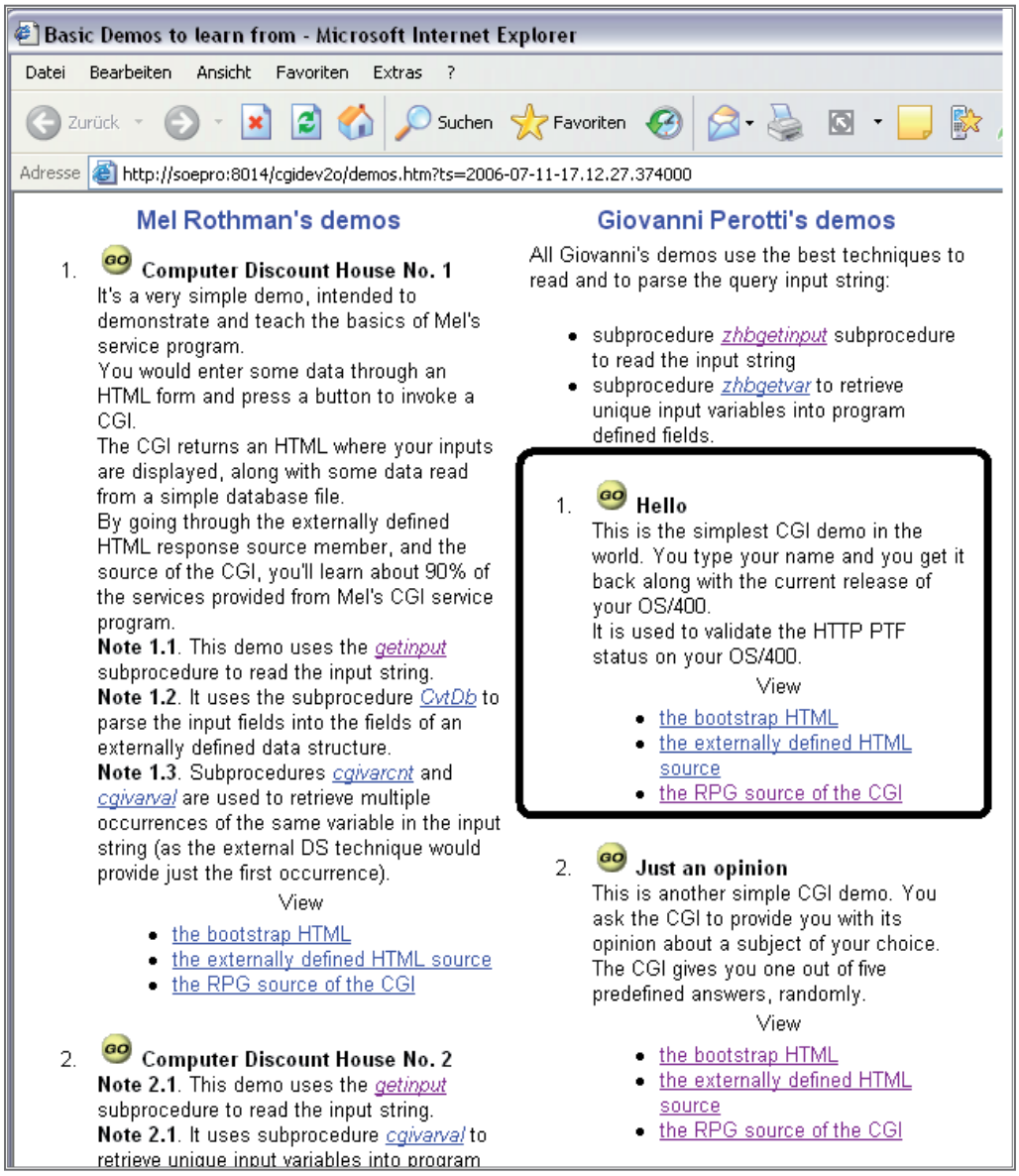
Bei dieser kleinen Anwendung können wir mit Hilfe von zwei Feldern Informationen eingeben, die bei einer erneuten Ausgabe der HTML-Seite enthalten sind. Des Weiteren wird in diesem Programm ein Zugriff auf den Inhalt von Systemdaten realisiert. Mit einem eigenen Programm, das innerhalb der Logik dieser Web-Anwendung ausgeführt wird, erfolgt ein Zugriff auf den Release Stand der Maschine, der als Rückgabewert auf der Webseite ausgegeben wird.

Klicken Sie in der Begrüßungsseite auf den Eintrag „Basic Demos“, wie es die folgende Abbildung zeigt:



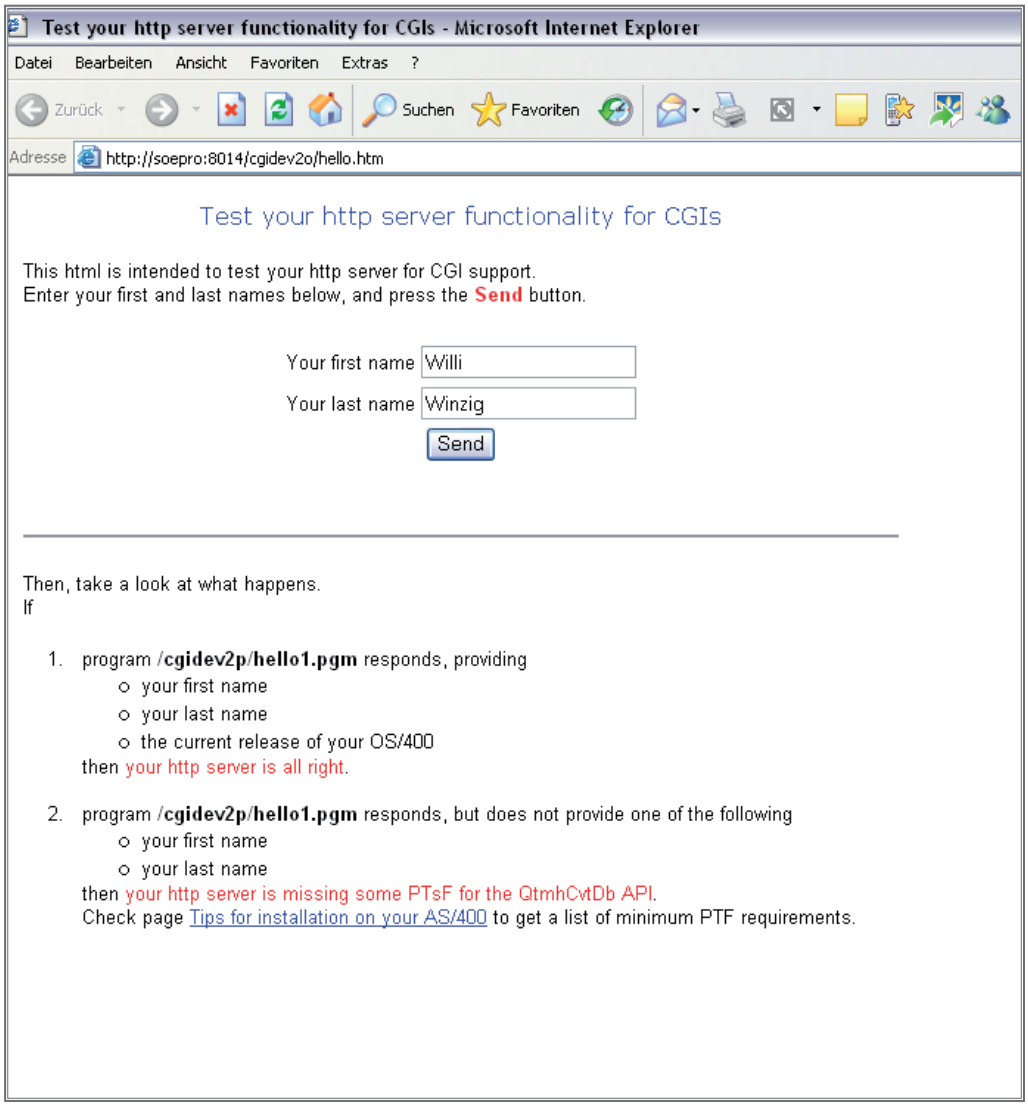
Starten der Basisdemos

In diesen Basic-Demos ist unter anderem das Programm „Hello“ enthalten. Diese können wir mit einem Klick auf das gelbe Symbol „Go“ starten. Schauen Sie sich dazu auch die folgende Abbildung an.



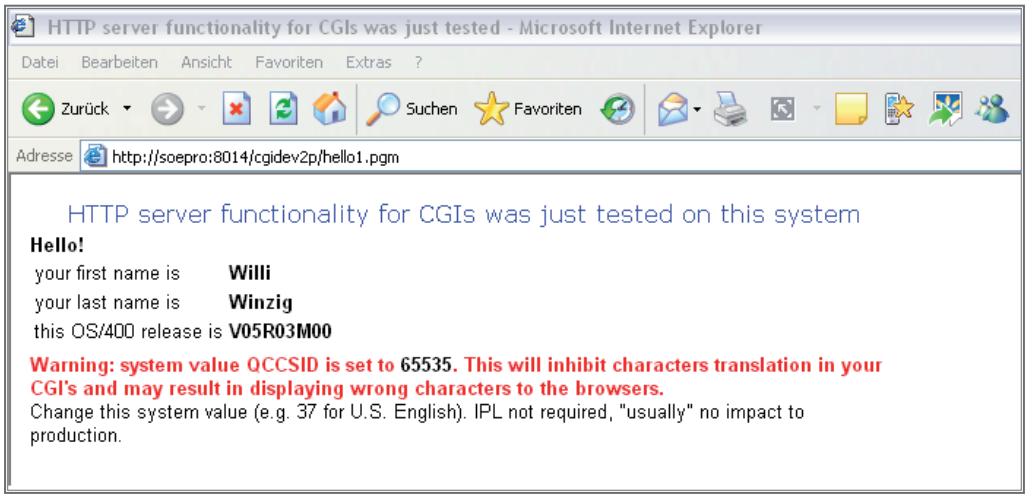
Starten des Beispielprogramms Hello

Dieses einfache Programm erlaubt die Eingabe von zwei Variablen „First Name“ und „Last Name“. Zwar werden diese nicht in einer Datei gespeichert, stehen aber im zweiten Schritt der HTML-Verarbeitung wieder als Ausgabewerte zur Verfügung. In diesen Feldern können Sie einen beliebigen Wert eingeben.



Das Beispiel CGI

Im unteren Bereich der Anzeige sind Hinweise enthalten, welche zum einen Aufschluss über die Arbeitsweise dieser kleinen Anwendung geben, aber auch einige Fehlerhinweise enthalten, die sich auf potentielle Fehler beziehen, die mit dieser Anwendung auftreten können. Mit einem Klick auf die Schaltfläche „Send“ schicken wir die Anforderung an den Webserver auf der i5, der die Eingabefelder verarbeitet und zusätzlich mit Hilfe eines integrierten RPGLE-Aufrufs den Inhalt des Systemwerts QCCSID ausliest und den aktuell auf der i5 installierten i5/OS Betriebssystemstand ausliest. Das Ergebnis wird dann mit einer erneuten HTML-Ausgabe auf dem Bildschirm dargestellt.



Das Ergebnis der Programmausführung

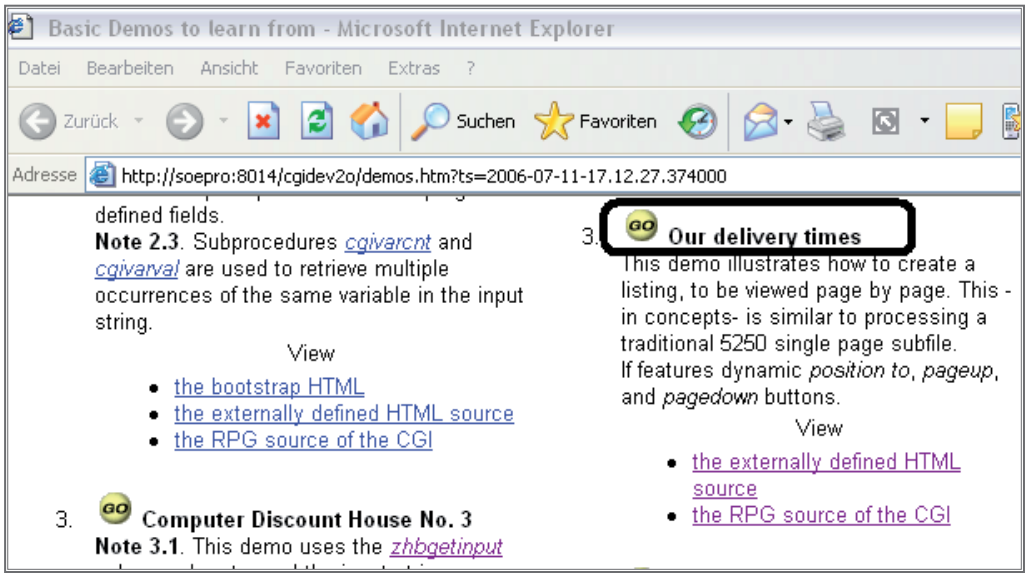
So einfach und trivial dieses Beispiel auch erscheinen mag, es beinhaltet bereits einige der wesentlichen Elemente, die wir in Bezug auf den Zugriff auf i5 Daten benötigen: Die Eingabe von Werten, die wir zum Beispiel als Eingabevariablen verwenden könnten, den Aufruf eines RPG Programms, der natürlich auch in Form eines CL- oder Cobol-Programms ersetzt werden könnte. Grundsätzlich lassen sich alle auf der i5 ausführbaren Programme in einem solchen CGI-Programm einbinden und ausführen. Nachfolgend finden Sie den RPG-Code, der in diesem Programm ausgeführt wird, um die Variablen zu verarbeiten und auch den Systemwert auszulesen. Diesen RPG-Code finden wir, wie auch den erforderlichen HTML-Code direkt in der Beispielwebseite als Link einsehbar.

Das RPG-Beispielprogramm

Schauen wir uns noch ein weiteres Beispiel an, dass Inhalte aus einer i5 Datei ausliest und diese in einem Subfile-ähnlichen Format ausgibt. Auch in diesem Beispiel steht uns erneut ein Eingabefeld zur Verfügung, das als Variable in den eingesetzten HTML- und RPG-Programmen verwendet wird.

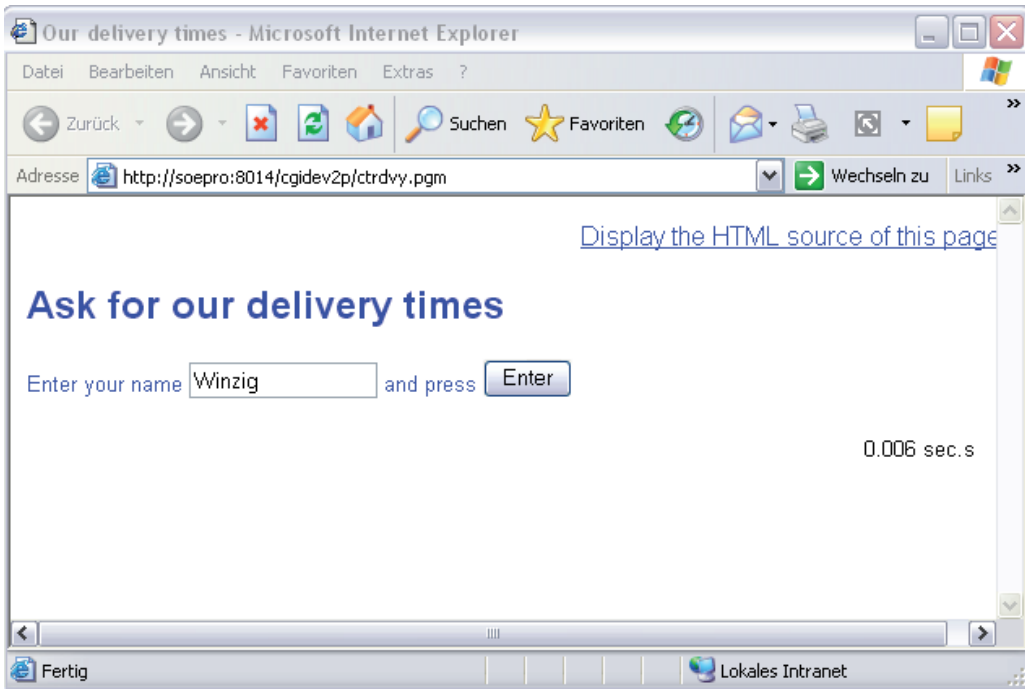
Klicken Sie zum Starten dieser Beispielanwendung auf „GO“ neben „Our Delivery Times“, wie es die folgende Abbildung zeigt.

5.1.5
Seite 10



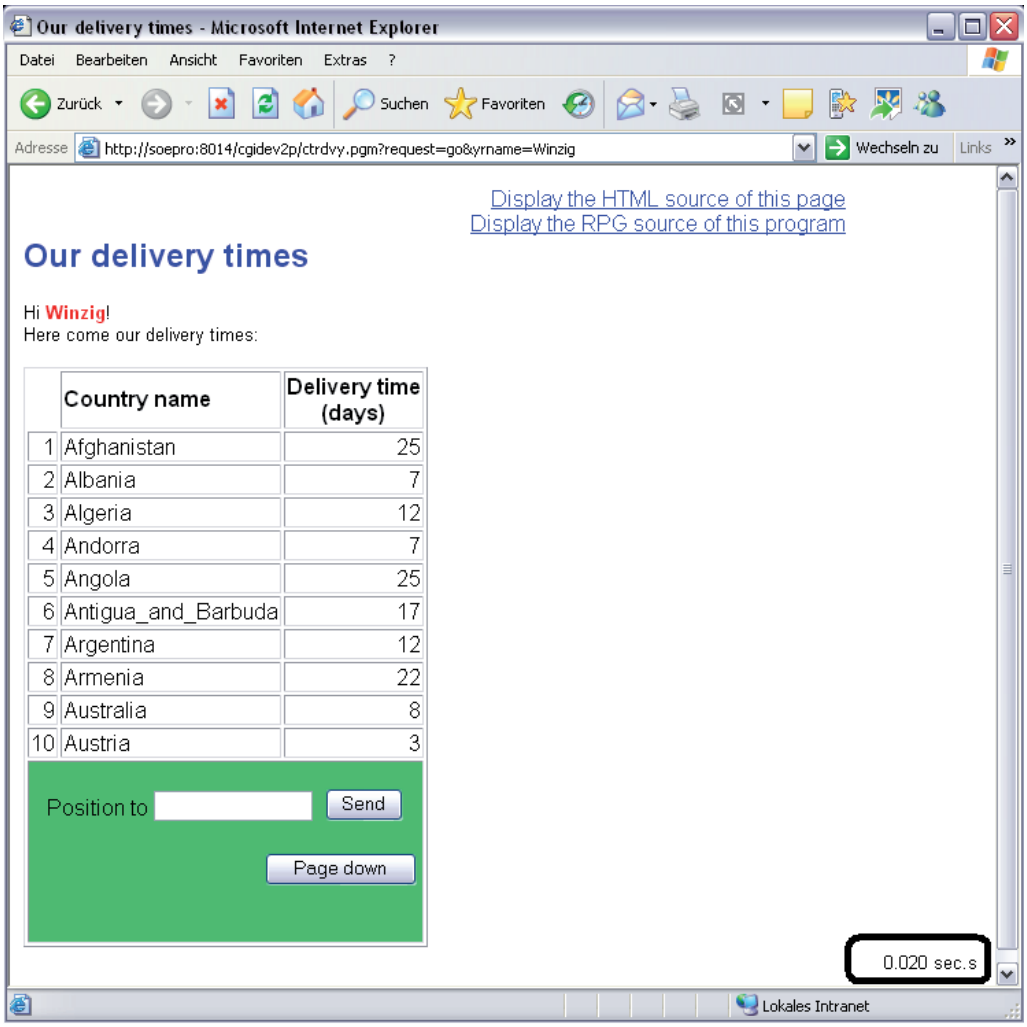
Das Beispiel „Delivery Times“

Nach dem Start der Anwendung werden wir zunächst aufgefordert, einen Namen einzugeben. Dieser wird als Variable für die aufzurufenden Programme verwendet. Nach dem Klick auf die Schaltfläche „Enter“ wird ein RPGLE-Programm aufgerufen, das aus einer i5 Datei die darin befindlichen Datensätze ausliest und diese in Form einer Auflistung in einem Subfile ähnlichen Format ausgibt.



Eingabe einer Variablen

Das Ergebnis dieser Programmausführung sehen wir anschließend in unserem Browser. Neben den einzelnen Datensätzen, die aus der i5 Datei CTRDVY stammen.



Das Ergebnis zeigt die i5 Daten

Achten Sie neben den Informationen, die aus der i5 Datei stammen auch auf den unteren rechten Bereich der Anzeige, der uns Auskunft über die Ausführungsgeschwindigkeit der HTML-Abfrage gibt.

Diese Anzeige wird durch das folgende Codefragment innerhalb der HTML-Codequelle erzeugt:

```
136.00
137.00  /$runtime               **** RESPONSE TIME
138.00  <table width="100%">
139.00  <tr><td align=right>
140.00          <font face="Arial" size="-1">
141.00          /%RUNTIME%/ sec.s
142.00          </td></tr>
143.00  </table>
```

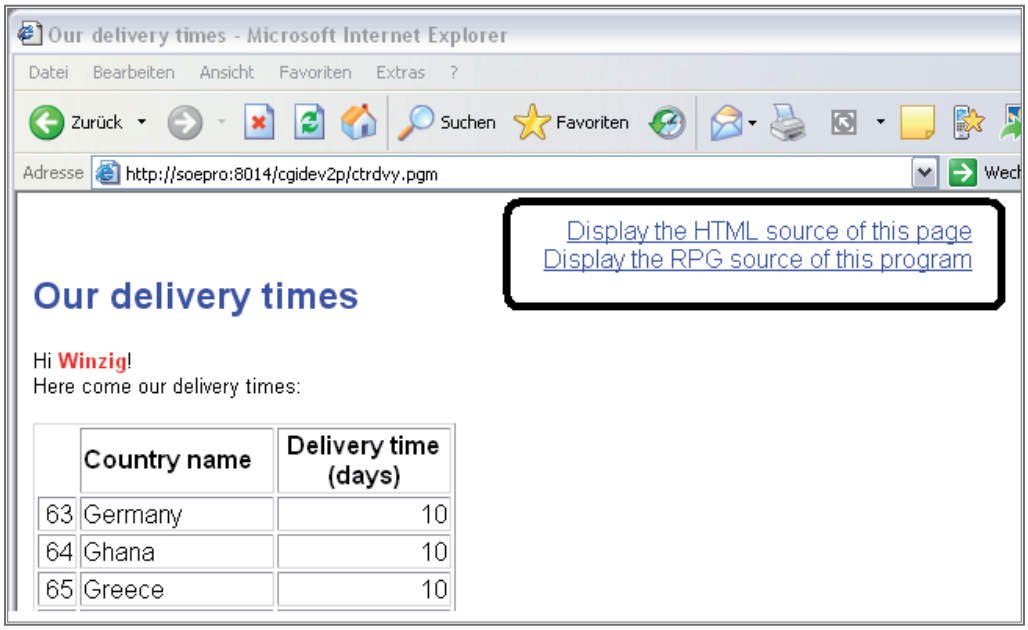
Runtime Codefragment

5.1.5
Seite 12

Wir können in dieser Ergebnisanzeige nun wie gewohnt blättern, indem wir auf die Schaltfläche „Page Down“ klicken. Alternativ dazu lässt sich mit einem weiteren Eingabefeld die Startposition innerhalb der Auflistung festlegen. Wenn wir zum Beispiel mit dem Eintrag „Germany“ in der Liste starten wollen, dann geben wir dazu in dem Feld „Position to“ den Wert „Germany“ ein und klicken auf die Schaltfläche „Send“.

Angabe der Startposition

Mit dieser Eingabevariablen teilen wir dem Programm den Startpunkt innerhalb der Auflistung mit. Auch in diesem Fall können wir mit einem Klick auf „Display HTML Source“ bzw. „Display RPG Source“ die Quellen direkt einsehen, welche mit dieser kleinen Beispielanwendung zum Einsatz kommen.



Einsehen der Programmcodes

Natürlich sind dies zwei abstrakte Beispiele, die aber in ihrem Grundsatz als Vorlage für erste Anwendungen eingesetzt werden können, mit denen Sie i5 Daten in Web-Anwendungen integrieren können. Diese lassen sich dann auf dem Webserver der i5 problemlos ausführen.

Die entsprechenden Sourcen zu diesen Programmen befinden sich in der Bibliothek CGIDEV2. Darin ist unter anderem eine QRPGLESRC enthalten, in der wir auch die RPGLE-Programme der zuvor gezeigten Beispielanwendungen finden. Die HTML-Sourcen befinden sich ebenfalls in dieser Bibliothek und können durch ein Kopieren für eigene Anwendungen verwendet werden.

Ohne zu sehr ins Detail zu gehen wollen wir uns den HTML-Codebereich doch kurz einmal anschauen.

Bei den Anweisungen /$top, $tabrow und $bottom handelt es sich nicht um HTML-Tags, sondern um spezielle Anweisungen, welche das CGIDEV2 Tool benötigt, um die entsprechenden Sektionen in HTML zuzuordnen. Mit /$pageup und $pagedown lassen sich die Blätteraktionen innerhalb der Anzeige realisieren.

5.1.5
Seite 14

Learn from sources

Member CTRDVY in CGIDEV2 / DEMOHTML

```
 1.00  /$top                  **** START HTML
 2.00  Content-type: text/html
 3.00  Expires: Fri Jan 01 00:00:00 1999
 4.00
 5.00  <html>
 6.00  <head>
 7.00  <title>Our delivery times</title>
 8.00  <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
 9.00  <meta name="DOCUMENTCOUNTRYCODE" value="it"/>
10.00  <meta name="DOCUMENTLANGUAGECODE" value="it"/>
11.00  </head>
12.00
13.00  <body text="black" bgcolor="white" link="blue" vlink="blue" alink="blue">
14.00
15.00  /$start
16.00  <table width="600" cellspacing="0" cellpadding="0">
17.00  <tr><td align=right><font face="Helvetica,Helv,Arial,Sans-Serif">
18.00          <a href="/cgidev2p/dspsrc.pgm?cgiinp01=cgidev2&cgiinp02=demohtml&cgiinp03=ctrdvy"
19.00           target="_blank">Display
20.00          the HTML source of this page
21.00          </font>
22.00          </td></tr>
23.00  <tr><td><font face="Helvetica,Helv,Arial,Sans-Serif">
24.00          <br>
25.00          <font color="blue">
26.00          <h2>Ask for our delivery times</h2>
27.00          <font size="-1">
28.00          <form method=get  action="/cgidev2p/ctrdvy.pgm">
29.00          <input type=hidden name=request value=go>
30.00          Enter your name
31.00          <font face="courier" size="3">
32.00          <input type=text name="yrname" size="15"
33.00           maxlength="100"></font>
34.00           and press
35.00          <input type=submit value=" Enter ">
36.00          </form>
37.00          </font></td></tr>
38.00  </table>
```

Codeausschnitt aus dem HTML-Bereich

Die eigentlichen Datenzugriffe erfolgen in einem RPGLE-Programm namens CTRDVY, das zum Beispiel mit dem folgenden HTML-Codebereich aufgerufen wird.

In dem Bereich /$go befinden sich die Tag-Anweisungen, mit denen wir zum Beispiel auch die Codebeispiele einsehen können.

```
40.00  /$go
41.00  <table width="600" cellspacing="0" cellpadding="0">
42.00  <tr><td align=right><font face="Helvetica,Helv,Arial,Sans-Serif">
43.00          <a href="/cgidev2p/dspsrc.pgm?cgiinp01=cgidev2&cgiinp02=demohtml&cgiinp03=ctrdvy"
44.00           target="_blank">Display
45.00          the HTML source of this page
46.00          </font>
47.00          </td></tr>
48.00  <tr><td align=right><font face="Helvetica,Helv,Arial,Sans-Serif">
49.00          <a href="/cgidev2p/dspsrc.pgm?cgiinp01=cgidev2&cgiinp02=qrpglesrc&cgiinp03=ctrdvy"
50.00           target="_blank">Display
51.00          the RPG source of this program
52.00          </font>
53.00          </td></tr>
54.00  <tr><td><font face="Helvetica,Helv,Arial,Sans-Serif" color="blue">
55.00          <h2>Our delivery times</h2>
56.00          </font></td></tr>
57.00  <tr><td><font face="Helvetica,Helv,Arial,Sans-Serif" size="-1">
58.00          Hi <font color="red"><b>/%YOURNAME%/</b></font>!<br>
59.00          Here come our delivery times:<br><br>
60.00          </td></tr>
61.00  </table>
```

Die GO-Anweisungen

Mit den folgenden Anweisungen wird das RPGLE-Programm CRTDVY aufgerufen, welches die Daten aus der i5 Datei ermittelt und diese via Variablen zurück an die HTML-Seite überträgt.

```
/$tabend                **** END   TABLE
<tr><td colspan=3 align=center bgcolor="£EFE4A7">
        <font face="Helvetica,Helv,Arial,Sans-Serif">
        <form name=form1 method=post action="/cgidev2p/ctrdvy.pgm">
        <input type=hidden name=request value=go>
        <input type="hidden" name=yrname value="/%YOURNAME%/">
        <input type="hidden" name=linesinpage value="/%LINESINPAG%/">
        <br>
        Position to
        <font face=courier>
        <input type="text" name=pos_to size=15 maxlength=15>
        </font>
        <input type="submit" value=" Send ">
        </form>
        <table width="100%">
            <tr><td valign=top align=left>
                <font face="Helvetica,Helv,Arial,Sans-Serif">
/$pageup
                <form name=form2 method=post action="/cgidev2p/ctrdvy.pgm">
                <input type=hidden name=request value=go>
                <input type="hidden" name=yrname value="/%YOURNAME%/">
                <input type="hidden" name=lastcountry value="/%LASTCOUNTRY%/">
                <input type="hidden" name=lastseqno value="/%LASTSEQNO%/">
                <input type="hidden" name=linesinpage value="/%LINESINPAG%/">
                <input type="hidden" name=pageupdown value="pageup">
                <input type="submit" value=" Page up ">
                </form>
```

Der Aufruf des RPGLE-Programms

Dies sind die wesentlichen Komponenten, die im Zusammenhang mit der Einbindung von i5 Daten in diesem einfachen HTML-Programm von Bedeutung sind. Sicher gibt es noch eine ganze Reihe von wichtigen Komponenten in diesem Programmablauf. Aber wir können und wollen an dieser Stelle den Code nicht zu sehr vertiefen. Diese beiden Beispiele sollten nur ein wenig verdeutlichen, wie einfach es sein kann, in einer Webseite i5 Daten zu integrieren.

Zwangsläufig kommt in diesem Zusammenhang natürlich auch die Fragestellung auf „Wofür brauche ich dann noch einen Anwendungsserver?“. Diese Frage lässt sich nicht allgemein beantworten. Eines ist aber als grundsätzliche Regel zu beachten: Bei der Ausführung von CGI-Programmen auf einem Webserver lassen sich einfachere Anwendungen realisieren, die ohne Verfügbarkeitsaspekte und zusätzliche Sicherheitsmechanismen eingesetzt werden können. Sobald eine Anwendung aber an Komplexität gewinnt und die mit der Anwendung zu verarbeitenden Informationen auch gegen einen unberechtigten Zugriff geschützt werden sollen, dann wird sich der Einsatz eines zusätzlichen Anwendungsservers auf Dauer nicht vermeiden lassen. Denn wie der Name bereits sagt, ist ein Anwendungsserver dafür ausgelegt, Anwendungen auszuführen und alle damit in Verbindung stehenden Aufgaben und Funktionen bereitzustellen.

5.1.5
Seite 16

HTTP-Performance

Beim Einsatz von Webanwendungen spielt die Performance eine wichtige Rolle. Denken wir einmal an einen alltäglichen Prozess: Wir surfen im Internet und besuchen die Webseite eines Unternehmens, um dort einen Einkauf zu tätigen. Aufgrund der schnellen DSL-Verbindung ist ein rascher Zugriff auf den Inhalt einer Webseite und das Auflisten von Angeboten eines Unternehmens kein Problem mehr – ist es tatsächlich kein Problem?

Betrachtet man die Verarbeitungsprozesse, die beim Einsatz von Webanwendungen durchlaufen werden, dann wird schnell klar, dass beginnend vom Eingabegerät – also beispielsweise dem PC zu Hause – über die Leitung, über eine Firewall, über verschiedene Netzwerkkomponenten etc. der lange Weg bis hin zum Webserver und den darauf laufenden Anwendungen viele mögliche Performanceengpässe verursachen kann.

Neben diesen Performancebereichen wirkt sich natürlich auch die Implementierung des Webservers auf dem jeweiligen Betriebssystem aus. Im Falle des Einsatzes eines „http-Servers powered by Apache“ auf dem System i oder genauer – auf Basis des Betriebssystems i5/OS – bestimmen unterschiedliche Parameter die Performance.

Beim Einsatz von Webservern auf dem System i spielt eine Vielzahl von Parametern eine entscheidende Rolle. Wie wird der einzelne Webserver – oder genauer die Serverinstanz – eingesetzt? Dient der Webserver „lediglich“ als Darstellungsschicht in Kombination mit einem Applikationsserver, der die eigentlichen Anwendungen ausführt oder werden einfache Anwendungen in Form von CGI-Programmen oder PHP-Anwendungen auf dem Webserver direkt ausgeführt? Wie wird der Speicher vom Webserver genutzt?

Dies sind nur einige wenige Fragen, die man sich stellen muss, wenn es um die Leistungsoptimierung bzw. Leistungsplanung eines Webserver Environments im Allgemeinen geht.

Grundsätzlich spielen also mehrere Faktoren eine wesentliche Rolle bei der Leistungsoptimierung einer Webumgebung. Im Fall des Betriebs eines Webservers auf dem System i haben wir Einflussmöglichkeiten auf den Webserver und dessen Leistungsfähigkeit auf

– der nativen Ebene des i5/OS-Betriebssystems

– auf Ebene des Webservers („http-Server powered by Apache“)

15. Ergänzung 6/2008

5.1.6
Seite 2

Leistungsoptimierung i5/OS

Wie bereits erwähnt, bildet die Implementierung des http-Servers auf dem System in Form des Lizenzprogramms 5722-DG1 eine besondere Herausforderung. Es geht darum, die Leistungswerte der http-Server-Instanzen in einer solchen Form zu realisieren, dass sowohl Webanwender als auch die „restlichen" Anwendungen, die neben dem Webserver auf dem System i ausgeführt werden, zufriedenstellende Antwortzeiten erhalten.

Die Besonderheit liegt hier in den unterschiedlichen Charakteristika, welche die verschiedenen Workloads mit sich bringen, die parallel auf dem System ausgeführt werden können.

Deshalb gelten in Bezug auf die Leistungsanpassungen einer Webserver-Installation auch dieselben Grundsätze, die bei allgemeinen Performanceanpassungen auf dem System i Beachtung finden sollten:

- Individuelle Einstellungen sollten die Gesamtheit der Maschine nicht negativ beeinflussen.
- Gleichartige Workloads sollten – in Gruppen zusammengefasst – individuell administrierbar definiert werden. Poolbildungen erleichtern beispielsweise die gezielte Leistungsanpassung für ähnliche Anwendungen.
- Systemwerte als auch andere Grundeinstellungen des Systems sollten unter Berücksichtigung der gesamten Workloads der Maschine vorgenommen werden.
- IBM bietet verschiedene Tools, mit denen Performancemessungen möglich sind. Unter anderem liefert das Lizenzprogramm 5722-PT1 (Performancetools) Informationen zur Auslastung des Systems und gibt in Form eines Advisors auch Empfehlungen für Optimierungen.

Wir wollen an dieser Stelle noch nicht auf diese Details eingehen, sondern uns zunächst einmal diejenigen Bereiche im Betriebssystem i5/OS näher anschauen, die sich auf die Leistung eines Webservers auf diesem System auswirken.

Zunächst sei erwähnt, dass natürlich die grundsätzlichen Performanceeinstellungen des Betriebssystems i5/OS auch beim Betrieb eines Webservers Anwendung finden. Deshalb sollten die Systemwerte, die sich auf das Laufzeitverhalten der unterschiedlichen Anwendungen auswirken können, entsprechend gesetzt werden.

Beachten Sie in jedem Fall, dass sich ein Systemwert auf alle Bereiche der Maschine auswirkt. Sicher spielt die Gesamtauslastung der Maschine mitsamt deren heterogenen Anwendungen eine entscheidende Rolle, was die allgemeinen, aber auch die speziellen Performanceeinstellungen betrifft. Berücksichtigen Sie deshalb bei Leistungsanpassungen Ihrer Maschine, dass

5.1.6.1
Seite 2

die übrigen Anwendungen hinsichtlich der Performancezuteilung nicht in Mitleidenschaft gezogen werden.

Nachfolgend finden Sie die wichtigsten Systemwerte, die im Bereich des http-Servers auf dem System i von Bedeutung sind:

Systemwerte „QDYNPTYADJ“ und „QDYNPTYSCD“
Die beiden Systemwerte „QDYNPTYADJ“ und „QDYNPTYSCD“ werden in Verbindung miteinander eingesetzt.

Der Systemwert QDYNPTYADJ („Dynamische Anpassung der Priorität“) wirkt sich insbesondere bei System-i-Modellen aus, die sowohl eine interaktive als auch eine Batch-Verarbeitung unterstützen. Dieser Systemwert beeinflusst die interaktive Leistung der Maschine zugunsten der Ausführung von Batch-Jobs. Der Systemwert hat nur dann eine Bedeutung, wenn QDNYPTYSCD („Dynamischer Prioritäts-Scheduler) aktiviert ist. Als Standardwert wird QDNYPTYADJ mit der Einstellung „1“ (= aktiviert) ausgeliefert. Soll der Systemwert geändert werden, wird die Änderung erst nach einem IPL wirksam!

Der zweite in diesem Zusammenhang wichtige Systemwert ist der QDYNPTYSCD. Er steuert die Aktivierung bzw. Deaktivierung des dynamischen Prioritäts-Planers. Dieser Systemwert hat als Default-Einstellung den Wert „1“ (=aktiviert). Eine Änderung wird erst nach einem IPL wirksam.

Wenn auf dem System i der http-Server ausgeführt wird, dann sollten diese beiden Systemwerte auf jeden Fall aktiviert sein!

Systemwert „QPRCMLTTSK“ (Prozessor-Multitasking)
Der Systemwert QPRCMLTTSK steuert die Aktivierung der Prozessor-Multi-Tasking-Capability. Dieser Systemwert hat zur Folge, dass ein einzelner System-i-Prozessor sich wie zwei separate Prozessoren verhält. Dieser Systemwert sollte die Einstellung „1“ haben.

Systemwert „QACTJOB“ (Anfängliche Anzahl aktiver Jobs)
Der Systemwert QACTJOB kontrolliert die maximale Anzahl aktiver Jobs im System. Dieser Wert wird am besten nach der folgenden Methode eingestellt:

Mit dem Befehl WRKACTJOB die Auflistung aller aktiven Jobs im System anzeigen. Im oberen rechten Bereich der Anzeige befindet sich die Gesamtanzahl der aktiven Jobs im System. Notieren Sie sich diese Anzahl. Dies sollte zu mehreren Zeitpunkten mit hoher Systemauslastung erfolgen.

Anschließend wird auf die Durchschnittssumme der Werte 10 % aufgeschlagen und das Ergebnis wird im Systemwert QACTJOB festgelegt.

Änderungen an diesem Systemwert erfordern einen IPL zu ihrer Aktivierung.

Systemwert „QADLACTJ“ (zusätzliche Anzahl aktiver Jobs)
Der Systemwert QADLACTJ gibt die maximale Anzahl zusätzlicher Jobs an, wenn die maximale Anzahl der aktiven Jobs (aus dem Systemwert QACTJOB) erreicht ist. Für jeden weiteren Job werden im System mindestens 110 KB Hauptspeicher reserviert.

Der Wert ist in Abhängigkeit vom verfügbaren Hauptspeicher einzustellen.

Poolzuordnung
Ein Pool bildet auf dem System i einen markanten Eckpfeiler, der in Bezug auf Leistungseinstellungen von wesentlicher Bedeutung ist. Mit einem Pool wird beispielsweise die Aufteilung des in einem System verfügbaren Hauptspeichers definiert. Diese kann wahlweise fix vom System-Administrator vorgegeben, oder aber vom System selbständig jeweils an die aktuellen Bedürfnisse angepasst durchgeführt werden.

Basierend auf dem Pool finden wir die so genannten Subsysteme, die als Ausführungsebene der unterschiedlichsten Jobs auf dem System i genutzt werden.

Der http-Server wird auf dem System i in Form verschiedener Jobs ausgeführt, die im Subsystem QHTTPSVR ausgeführt werden:

Aszebis1: Aktive Jobs Abgelaufene Zeit:00:00:00

Jobname	Ausführlicher Status	Aktueller Benutzer	Typ	CPU %	Ausführ...	Anzahl Threads
Qsysscd	Warten auf Ereignis	Qpgmr	Stapel	0,0	10	1
Qhttpsvr	Warten auf Entfernen ...	Qsys	Subsystem	0,0	0	2
Admin	Warten auf Signal	Qtmhhttp	Stapel - Server	0,0	25	1
Admin	Warten auf Signal	Qtmhhttp	Sofortiger Stapel...	0,0	25	1
Admin	Warten auf Java-Prog...	Qtmhhttp	Stapel - Server	0,0	25	24
Admin	Warten auf Signal	Qtmhhttp	Sofortiger Stapel...	0,0	25	24
Ajsp	Warten auf Signal	Qtmhhttp	Stapel - Server	0,0	25	1
Ajsp	Warten auf Signal	Qtmhhttp	Sofortiger Stapel...	0,0	25	1
Ajsp	Warten auf Signal	Qtmhhttp	Sofortiger Stapel...	0,0	25	1
Ajsp	Warten auf Entfernen ...	Qtmhhttp	Sofortiger Stapel...	0,0	25	13
Ajsp	Warten auf Signal	Qtmhhttp	Sofortiger Stapel...	0,0	25	57
Demo1	Warten auf Signal	Qtmhhttp	Stapel - Server	0,0	25	1
Demo1	Warten auf Signal	Qtmhhttp	Sofortiger Stapel...	0,0	25	1
Demo1	Warten auf Signal	Qtmhhttp	Sofortiger Stapel...	0,0	25	1
Demo1	Warten auf Signal	Qtmhhttp	Sofortiger Stapel...	0,0	25	47
Demo1	Warten auf Entfernen ...	Qtmhhttp	Sofortiger Stapel...	0,0	25	3
Qippsvr	Warten auf Signal	Qtmhhttp	Stapel - Server	0,0	25	1
Qippsvr	Warten auf Signal	Qtmhhttp	Sofortiger Stapel...	0,0	25	1
Qippsvr	Warten auf Signal	Qtmhhttp	Sofortiger Stapel...	0,0	25	55
Zendcore	Warten auf Signal	Qtmhhttp	Stapel - Server	0,0	25	1
Zendcore	Warten auf Signal	Qtmhhttp	Sofortiger Stapel...	0,0	25	1
Zendcore	Warten auf Signal	Qtmhhttp	Sofortiger Stapel...	0,0	25	1
Zendcore	Warten auf Signal	Qtmhhttp	Sofortiger Stapel...	0,0	25	47
Zendcore	Warten auf Entfernen ...	Qtmhhttp	Sofortiger Stapel...	0,0	25	3

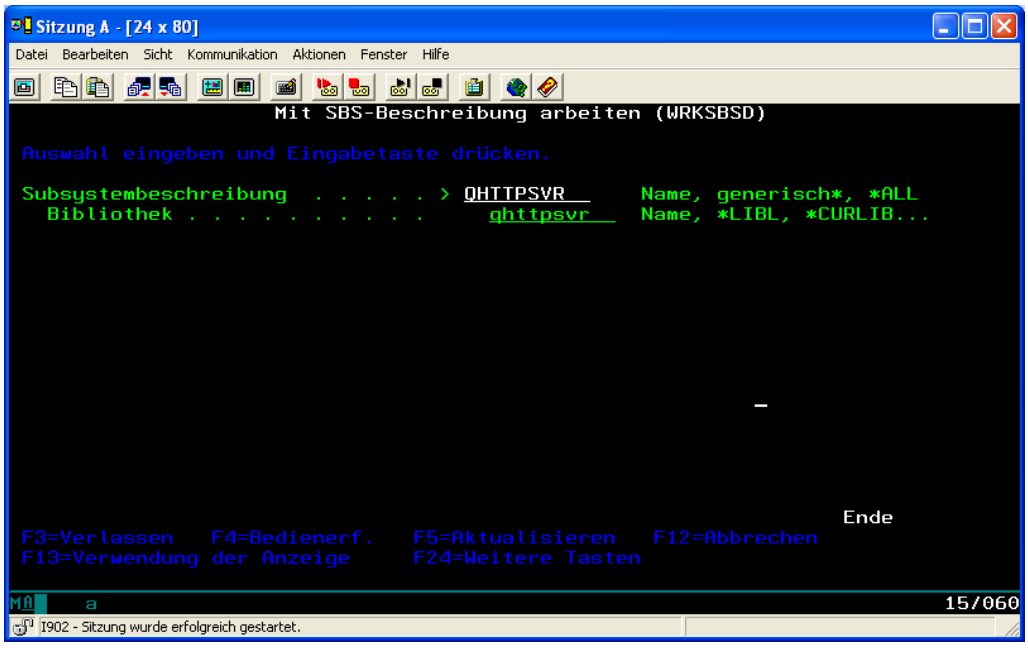
5.1.6.1
Seite 4

Jede Instanz eines Webservers wird in Form von mindestens vier Jobs abgebildet und auf dem System i ausgeführt. Wie bei allen anderen i5/OS-Subsystemen ist auch das QHTTPSVR-Subsystem einem Pool zugeordnet. Dieser Pool verfügt über bestimmte Ressourcen und dient der Ausführung aller Jobs auf dem System. Dabei werden die Systemressourcen in Form von beispielsweise dem Hauptspeicher auf alle im System befindlichen Jobs aufgeteilt. Die nachfolgende Abbildung zeigt eine Auflistung der Standardpools, die von IBM ausgeliefert werden.

Aszebis1: Aktive Pools Abgelaufene Zeit:00:00:01

Pool	Beschreibung	Aktuelle Größe	Aktuelle Threads	Maximal auswählbare	Summe der
Maschine	Von internen Maschinenfunktion...	194,66	85	Kein Maximum	0,0
Basis	Standard-Systempool	1.59...	817	135	0,0
Interaktiv	Für interaktive Arbeit benutzt	201,60	1	51	0,0
Spool	Zum Drucken benutzt	20,16	0	5	0,0

Schauen wir uns einmal die Eigenschaften des http-Subsystems i an. Dazu geben wir in einer 5250-Emulation den Befehl WRKSBSD gefolgt vom Namen des http-Subsystems an. Der qualifizierte Eintrag sieht wie folgt aus:



In der nächsten Anzeige wählen wir nun die Option „5“ und bestätigen diese mit der Eingabetaste. In der Folgeanzeige selektieren wir den Eintrag „2. Pool Definitionen“ – dort sehen wir den zugeordneten Pool – nämlich die Zuordnung des http-Subsystems zu dem Basis-Systempool (auch Basepool oder Pool Nr. 2 genannt).

Ohne an dieser Stelle zu tief ins Detail gehen zu wollen sei erwähnt, dass der Base-Pool für die Ausführung einer ganzen Reihe von Jobs auf dem System i genutzt wird. Dies können klassische RPG-Anwendungen sein, Java-Programme oder auch http-Anwendungen. Anhand dieser beispielhaften Aufzählung kann man bereits erkennen, wie heterogen die in dem Pool ausgeführten Anwendungen sind. Und genauso heterogen sind die individuellen Leistungsanforderungen eines spezifischen Jobs an das System.

i5/OS verfügt in der Grundauslieferung über vier verschiedene Pools:

– Maschinen-Pool
– Base Pool
– Interactive Pool
– Spool Pool

Neben diesen Standard-Pools können zusätzliche gemeinsam genutzte Pools (Shared Pools) verwendet werden. Mit einem solchen Shared Pool lassen sich spezielle Leistungseinstellungen für individuelle Jobarten definieren – häufig eine Grundlage für gute Performancewerte bei der Ausführung von heterogenen Anwendungen auf einem System.

Diese Basiszuordnung kann im einen oder anderen Fall ungünstig sein. Deshalb kann bei genauer vorheriger Analyse ein Anpassen der Standardeinstellungen durchaus spürbare Leistungsverbesserungen bringen. Aber nicht nur für Leistungsoptimierungen, sondern beispielsweise auch für die vereinfachte Analyse von Prozessen im Webserver-Umfeld kann eine individuelle Pool-Zuordnung sinnvoll sein.

Bevor man Änderungen an den Einstellungen der Pool-Zuordnungen vornimmt, sollte man sich im Klaren darüber sein, welche Definitionen für die einzelnen „Shared Pools“ vorgenommen werden müssen. Die Zuordnung von Hauptspeicher und die Festlegung des Bereinigens von Speicher sind nur einige Beispiele für wichtige Bereiche, die sich maßgeblich auf den Betrieb des Webservers auswirken. Auch wenn die im Folgenden beschriebenen Schritte der Anpassung der Subsystembeschreibung des http-Servers auf dem System i durchgeführt werden können während der Server aktiv ist, sollte man diese Änderungen doch nur dann durchführen, wenn Leistungseinbußen – hervorgerufen durch unbedachte Einstellungen der Poolbereiche – kein Problem darstellen. Um also ganz sicher zu gehen, empfiehlt sich die Anpassung der grundlegenden http-Server-Einstellungen auf dem System i erst dann, während kein Anwender mit dem Webserver-Bereich arbeitet.

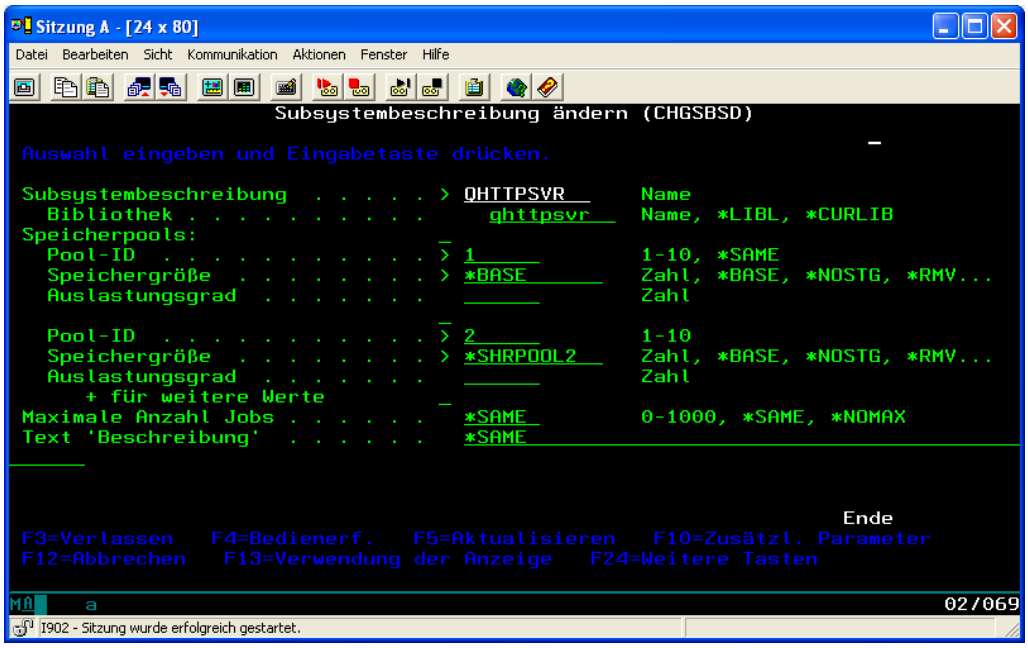
5.1.6.1
Seite 6

Aszebis1: Gemeinsame Pools Abgelaufene Zeit:00:14:22

Pool	Beschreibung	Status	Defini... Größe	Aktuelle Größe	Maximal auswählbare
Maschine	Von internen Maschinenfunktion...	Aktiv	190,77	190,77	Kein Maximum
Basis	Standard-Systempool	Aktiv		1.35...	135
Interaktiv	Für interaktive Arbeit benutzt	Aktiv	453,60	453,60	51
Spool	Zum Drucken benutzt	Aktiv	20,16	20,16	5
Gemeinsam 1		Inaktiv	0,00	0,00	0
Gemeinsam 2		Inaktiv	0,00	0,00	0
Gemeinsam 3		Inaktiv	0,00	0,00	0
Gemeinsam 4		Inaktiv	0,00	0,00	0
Gemeinsam 5		Inaktiv	0,00	0,00	0
Gemeinsam 6		Inaktiv	0,00	0,00	0
Gemeinsam 7		Inaktiv	0,00	0,00	0
Gemeinsam 8		Inaktiv	0,00	0,00	0
Gemeinsam 9		Inaktiv	0,00	0,00	0
Gemeinsam 10		Inaktiv	0,00	0,00	0
Gemeinsam 11		Inaktiv	0,00	0,00	0
Gemeinsam 12		Inaktiv	0,00	0,00	0
Gemeinsam 13		Inaktiv	0,00	0,00	0
Gemeinsam 14		Inaktiv	0,00	0,00	0
Gemeinsam 15		Inaktiv	0,00	0,00	0
Gemeinsam 16		Inaktiv	0,00	0,00	0
Gemeinsam 17		Inaktiv	0,00	0,00	0
Gemeinsam 18		Inaktiv	0,00	0,00	0
Gemeinsam 19		Inaktiv	0,00	0,00	0

Um den http-Server in einem anderen Pool-Bereich auszuführen, muss die Subsystembeschreibung QHTTPSVR angepasst werden.

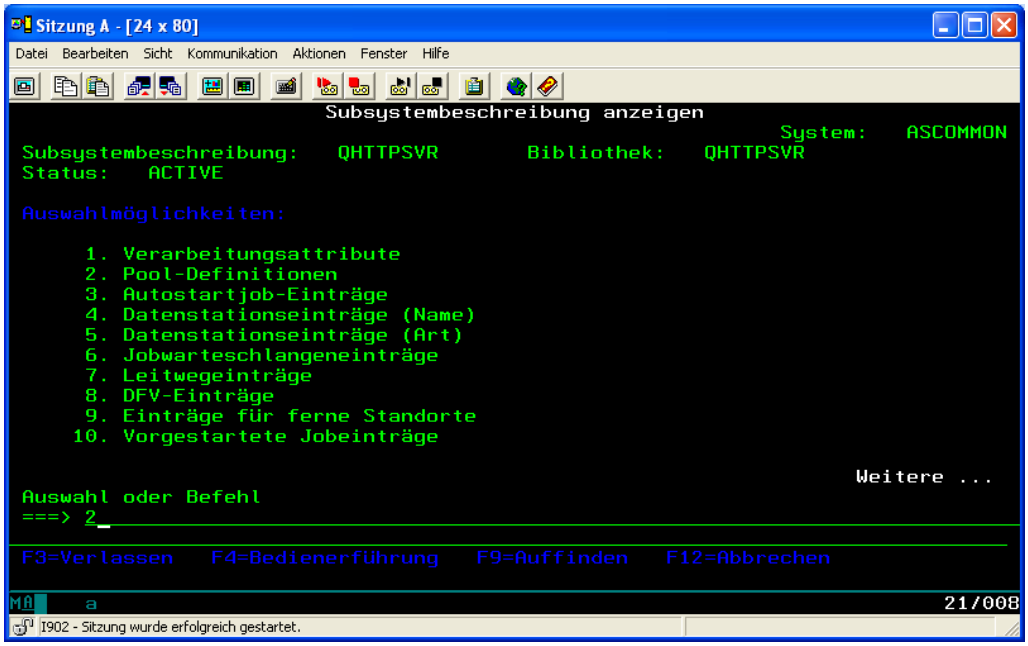
```
CHGSBSD SBSD(QHTTPSVR/QHTTPSVR) POOLS((1 *BASE) (2 *SHRPOOL2)
```



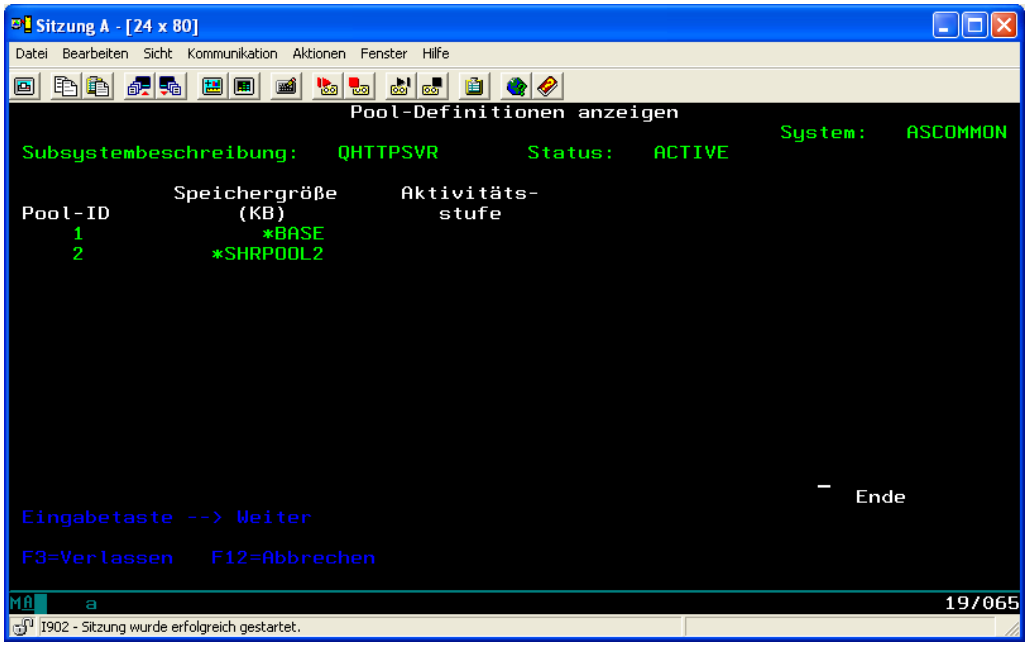
Schauen wir uns die Änderung über die Details zu dem Subsystem nochmals an. Dazu nutzen wir den Befehl „WRKSBSD“ in Verbindung mit dem Namen des Subsystems QHTTPSVR:

```
WRKSBSD SBSD(QHTTPSVR/QHTTPSVR)
```

Wählen Sie in der nächsten Anzeige die Option 5 (Anzeigen) für den Eintrag des Subsystems QHTTPSVR aus und selektieren Sie in der Folgemaske den Eintrag „2. Pool-Definitionen“, wie es die nächste Abbildung zeigt:



Das Ergebnis sehen wir in der Detailanzeige der Pool-Definitonen:

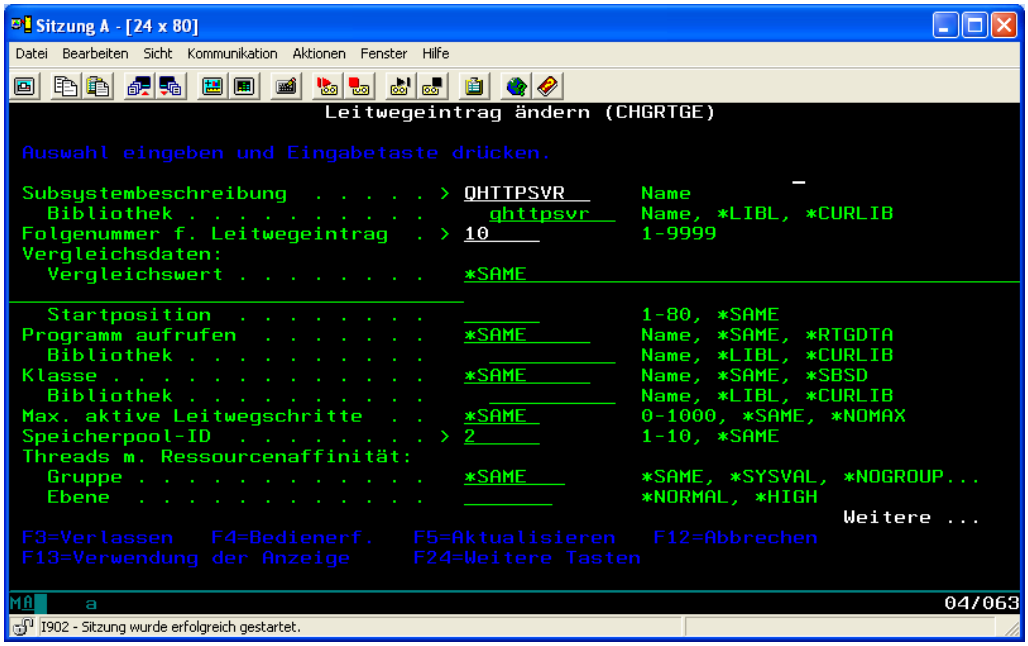


5.1.6.1
Seite 8

Neben der Definition der Poolzuordnung müssen noch die Leitwegeinträge angepasst werden, damit die http-Server-Jobs die neue Pool-Definition berücksichtigen.

Ein solcher Leitweg kann mit Hilfe des Befehls „CHGRTGE“ angepasst werden. Der vollständige Befehl für die Leitweganpassung sieht dann wie folgt aus:

```
CHGRTGE QHTTPSVR SEQNBR(10) POOLID(2)
```



Durch die Differenzierung der http-Server-Jobs auf unterschiedliche Pools lässt sich beispielsweise im Problemfall auch einfacher eine Leistungsanalyse für den Webserver durchführen. Die Belegung von Hauptspeicher, die Anzahl von Fehlerraten (Page Faults) und weitere Parameter sind hier wichtige Eckdaten.

Ein weiterer wichtiger Bereich bei den Leistungseinstellungen einer Webserver-Installation sind mögliche Datenzugriffe. Dynamische Webseiten bedienen sich in der Regel Techniken wie ODBC- oder JDBC-Zugriffen, mit denen die Webanwendung auf den Inhalt der DB2/UDB zugreifen.

Solche Zugriffe sind den System-i-Betreibern sicher durch die QZDASOINIT-Jobs bekannt – denn genau solche Zugriffe verbergen sich hinter diesen Jobs. Die mit ODBC- oder JDBC-Verbindungen generierten Datenzugriffe arbeiten auf Basis von SQL - jeder dieser Datenzugriffe wird auf dem System i in Form eines QZDASOINIT-Job verarbeitet. Die Besonderheit dabei: für jeden Zugriff wird ein solcher QZDASOINIT Job gestartet!

Dass das Starten (und auch das Beenden) eines Jobs auf dem System i eine stark leistungsbeanspruchende Aufgabe ist, dürfte bekannt sein. Dass damit verbunden auch die Handlung der QZDASOINIT-Jobs eine entsprechende Last auf das System bringt, ist eine unveränderliche Tatsache. Die Einstellungen für die Ausführung der QZDASOINIT-Jobs sollte dahingehend angepasst werden, dass diese Jobs in einem separaten Pool ausgeführt werden. Diesem Pool sollten dann die JDBC- und ODBC- Zugriffsjobs zugeordnet werden. Eine solche Zuordnung kann beispielsweise mit dem folgenden Befehl durchgeführt werden:

```
CHGPJE SBSD(QUSRWRK) PGM(QZDASOINIT) POOLID(2)
```

5.1.6.1
Seite 10

Leistungsoptimierung http-Server

Performancewerte lassen sich im Umfeld eines Webservers auf dem System i nicht nur im Umfeld des Betriebssystems i5/OS, sondern auch über spezielle http-Server-interne Einstellungen verwalten. Während wir zuvor die i5/OS-Spezifika in Form der Subsysteme und Pools betrachtet haben, beschäftigen wir uns nun mit einigen weiteren leistungsbeeinflussenden Eckdaten des http-Webservers auf dem System i. Dabei ist u. a. auch die verwendete Version des Webservers von Bedeutung. Diese wird zum Teil während eines Releasewechsels bzw. Wechsels der Version des Betriebssystems, zum Teil aber auch durch das Einspielen neuer PTFs installiert.

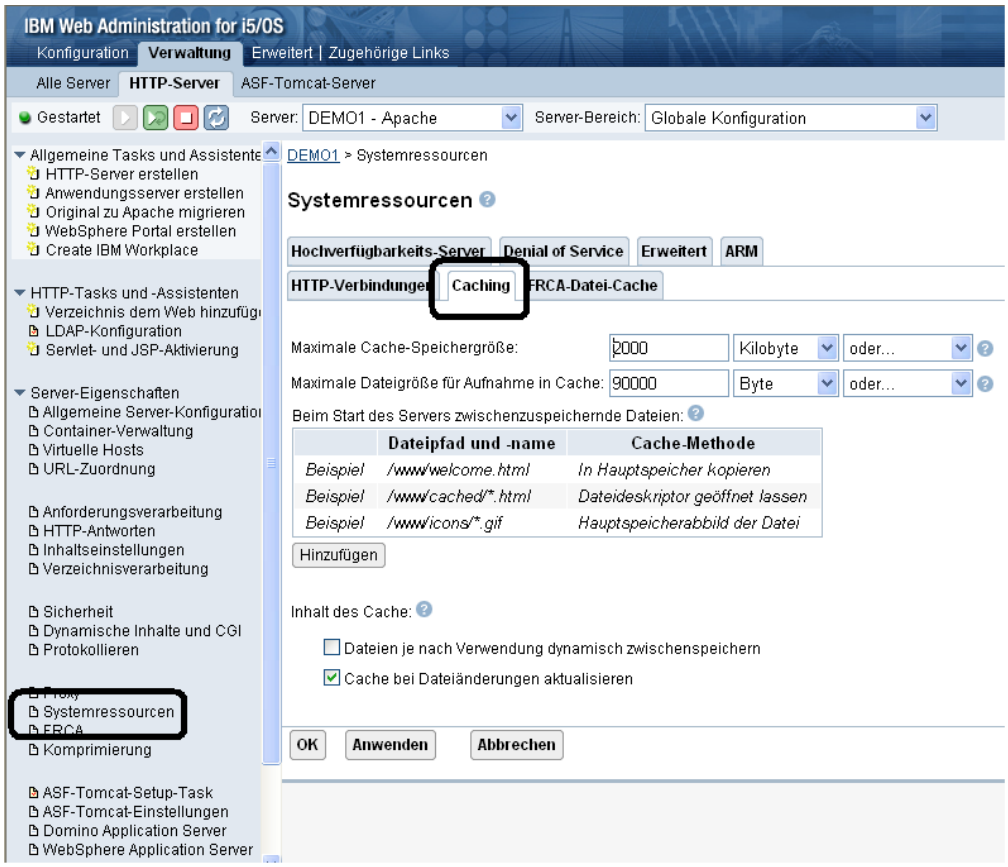
Bis zum Erscheinen der Version V5R4 war der http-Server auf dem System i für einige Anwendungen nur bedingt einsetzbar – fehlte doch die Unterstützung von UTF-8 – einem Standard, der beispielsweise beim Einsatz von XML-Anwendungen sehr wichtig ist. Mit der Version V5R4 unterstützt IBM nun die Versionen ab 2.0.52. Mit der Implementierung der UTF-8-Unterstützung entfällt die bisherige Umsetzung von UTF-8 in EBCDIC, welche zwingend für Daten im UTF-8 Format durchgeführt werden musste. Wer also eine ältere Betriebssystemversion als V5R4 sowie UTF-8-Anwendungen einsetzt, sollte sich allein aufgrund des Wegfalls der Konvertierung Gedanken über einen Upgrade machen – die Leistungsverbesserungen sind spürbar!

Betrachten wir noch einige weitere Besonderheiten bei der Implementierung des Webservers in i5/OS. Die Verarbeitung von Webseiten kann statisch erfolgen. Dabei wird von der Webanwendung der Inhalt einer Seite von einem zentralen Verzeichnis abgerufen. Idealerweise liegt ein solches Verzeichnis innerhalb des IFS des System i. Jeder Aufruf einer solchen Seite veranlasst den Prozess dazu, den Seiteninhalt per Standardeinstellung einzulesen, und beim Verlassen wird dieser „vergessen“. Erfolgt ein erneuter Zugriff auf diese Seite, wird der komplette Inhalt erneut vom fernen Verzeichnis geladen. Eine Besonderheit beim Betrieb des Webservers auf dem System i ist die Option, Inhalte statischer Seiten zwischenzuspeichern. Grundsätzlich sollte man überlegen, ob statische Serverseiten lokal im Cache des http-Servers vorgehalten werden sollen. Die Cachenutzung sollte u. a. auch für häufig wiederverwendete Dateien in Erwägung gezogen werden. Oft ist die Speicherung solcher Dateien im Cache deutlich schneller als das wiederholte Laden der Informationen. Dies gilt besonders auch dann, wenn die Dateien aus dem IFS des System i geladen werden müssen. Das Speichern von allzu großen Dateien im Cache sollte jedoch sorgfältig bedacht werden – nicht selten belegen große Dateien unnötig Cache – und damit auch wertvolle Systemressourcen.

5.1.6.2
Seite 2

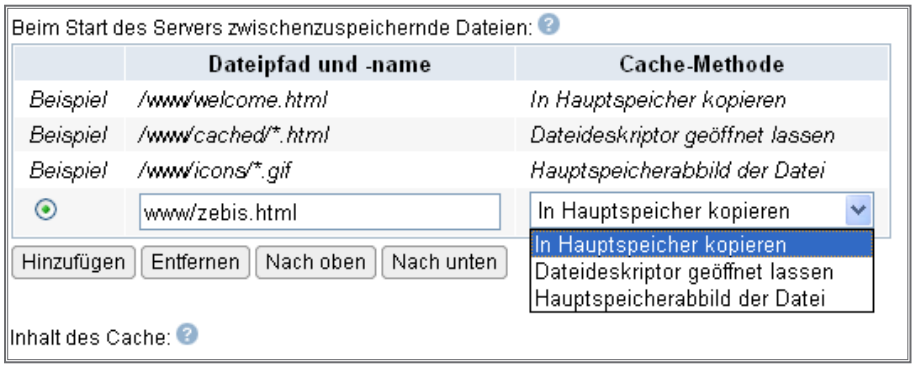
Die Cache-Einstellungen lassen sich pro Serverinstanz im Bereich „Systemressourcen“ definieren. Dabei ist allerdings die Differenzierung zwischen den allgemeinen Cache-Einstellungen und dem FRCA Cache zu beachten! Letzterer ist eine weitere Besonderheit des System i, auf welche wir später noch eingehen werden.

Die folgende Abbildung zeigt den Bereich der Cache-Einstellungen des http-Servers auf dem System i:



Wie Sie anhand der vorhergehenden Abbildung sehen können, lassen sich die Caching-Einstellungen für bestimmte Dateien definieren.

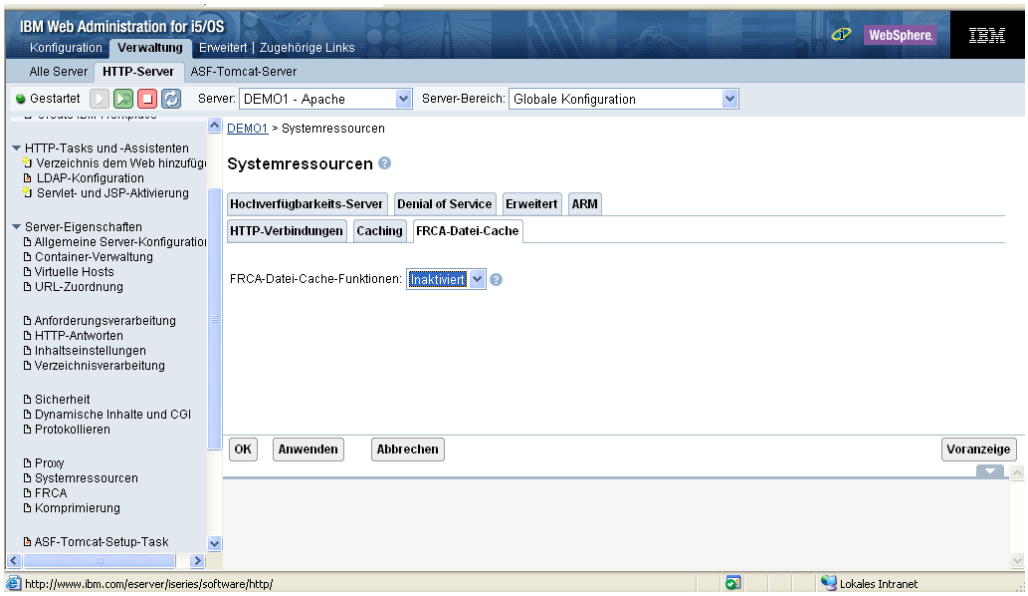
Neue Eintragungen werden in diesem Bereich durch einen Klick auf die Schaltfläche „hinzufügen“ definiert. Damit haben wir dann die Option, neben dem Dateinamen auch die Form der Cache-Methode zu auszuwählen:



Zuvor wurde bereits ein weiterer Leistungsbereich erwähnt: FRCA. FRCA (Fast Response Caching Accelerator) ist die Bezeichnung für eine spezielle System-i http-Server-Technologie, welche ebenfalls Dateien im http-Serverumfeld im Cache zwischenspeichert.

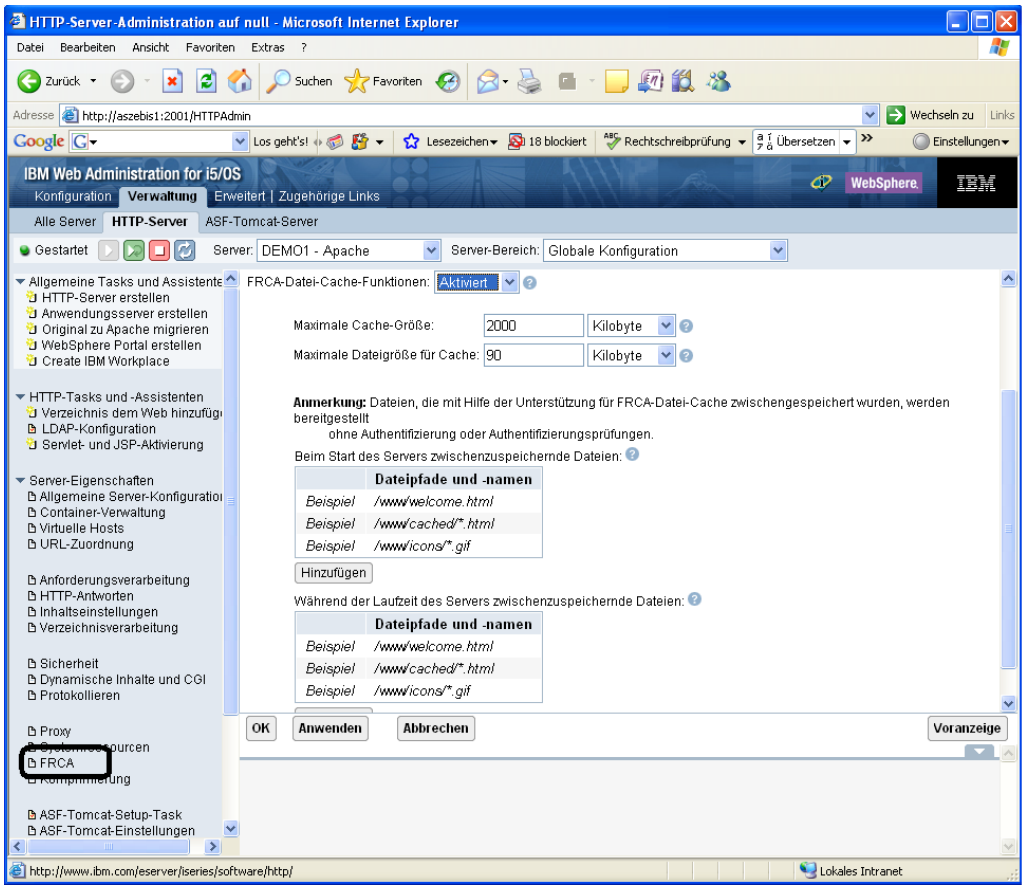
Die FRCA-Einstellungen lassen sich detailliert pro http-Serverinstanz definieren.

FRCA ist in zwei Teilbereichen der Webleistung definierbar. Die allgemeine FRCA-Aktivierung erfolgt im Bereich „Systemressourcen“. Um die FRCA-Funktionen zu aktivieren, muss der Wert in der folgenden Anzeige auf „aktiviert“ gesetzt werden.



5.1.6.2
Seite 4

Nach der Aktivierung werden in derselben Anzeige weitere Parameter zur Definition angeboten. Unter anderem muss dort die gewünschte Größe des Caches angegeben werden. Diese richtet sich nach den Dateien bzw. Anwendungen, die FRCA-Funktionen nutzen sollen.



Achten Sie darauf, dass die Verwendung des FRCA-Caches weiterer Einstellungen auf Basis des i5/OS bedarf. Erst wenn in den TCP/IP-Attributen des System i der Parameter NFC (Netzwerkdatei-Cache) aktiviert wurde, greifen die Einstellungen auf Seiten des Webserver-Administrators.

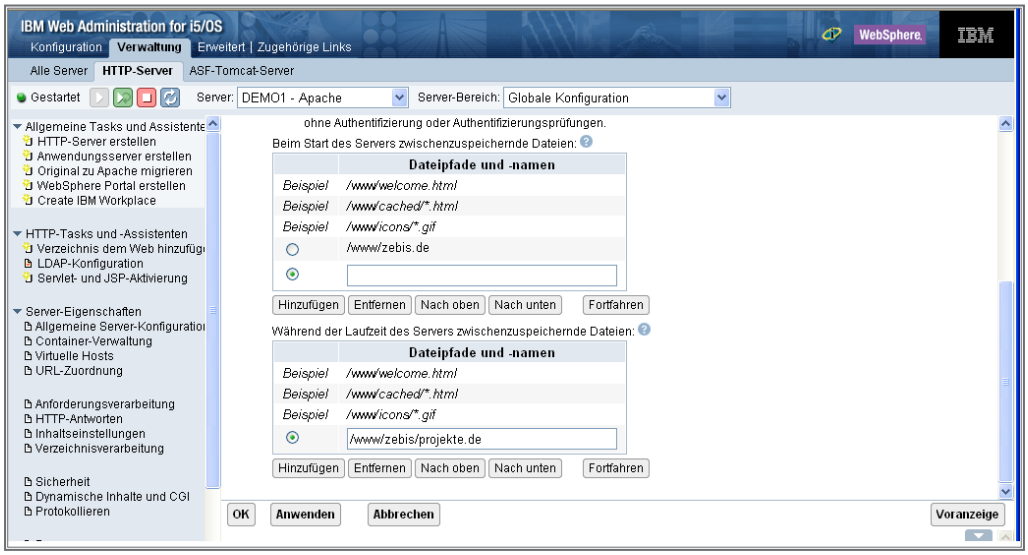
Kontrollieren Sie deshalb die TCP/IP-Einstellungen. Diese können wir mit dem Befehl CHGTCPA anpassen:

```
                    TCP/IP-Attribute ändern (CHGTCPA)

Auswahl eingeben und Eingabetaste drücken.

IP Time to Live (Hop-Limit)  . .  IPTTL          64
IP-QoS-Aktivierung . . . . . .  IPQOSENB       *NO
Inaktiver IP-Gateway:           IPDEADGATE
  Aktivierung  . . . . . .                     *YES
  Intervall  . . . . . .                       2
Zeitüberschreit. bei ARP-Cache  ARPTIMO        15
ECN aktivieren . . . . . .  .   ECN            *NO
Netzwerkdatei-Cache:            NFC
  Aktivierung  . . . . . .                     *YES
  Zeitlimit für Datei im Cache                 300
  Cache-Größe  . . . . . .                     10
Protokollfehler protokollieren  LOGPCLERR      *NO




                                                                      Ende
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F10=Zusätzl. Parameter
F12=Abbrechen   F13=Verwendung der Anzeige   F24=Weitere Tasten
```

Die FRCA-Funktionen unterscheiden in Bezug auf das Speichern von Dateien die beiden folgenden Fälle:

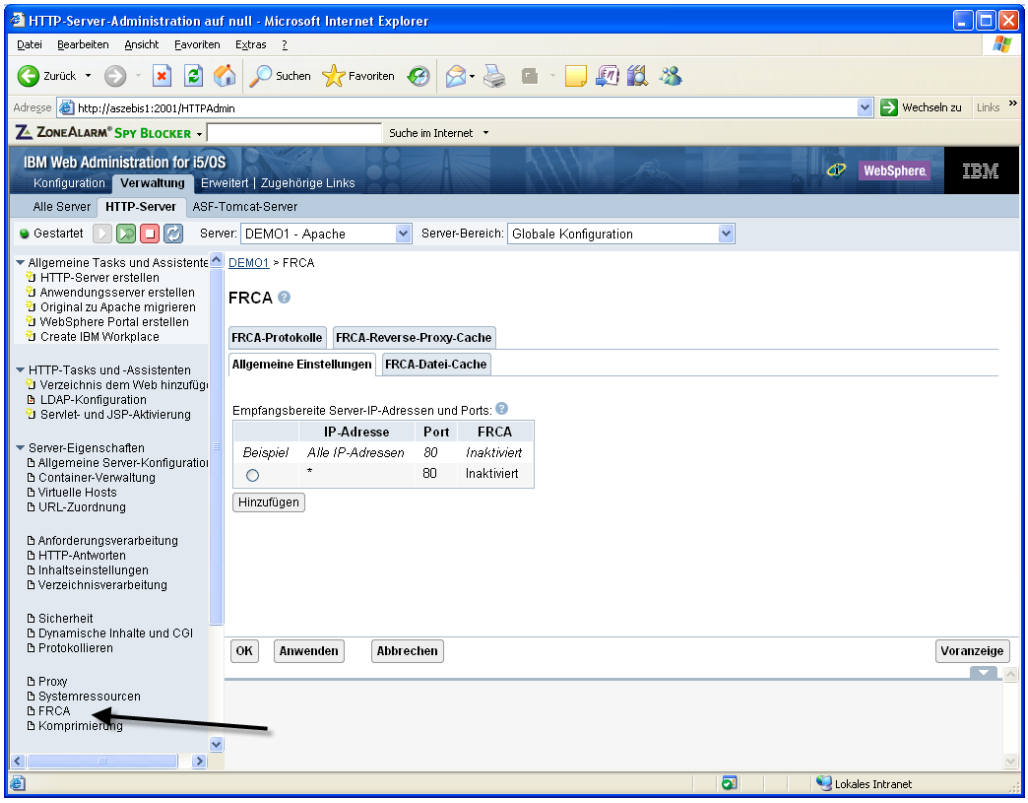
– Dateien, die beim Start des Servers gespeichert werden sollen.
– Dateien, die nach dem Start des Servers gespeichert und somit für eine Wiederverwendung schnell zur Verfügung stehen sollen.

Diese beiden Bereiche lassen sich jeweils individuell verwalten. Fügen Sie in diesen Bereichen nur solche Dateien an, deren Wichtigkeit hinsichtlich Wiederverwendung ein Speichern während des Serverstarts sinnvoll machen.



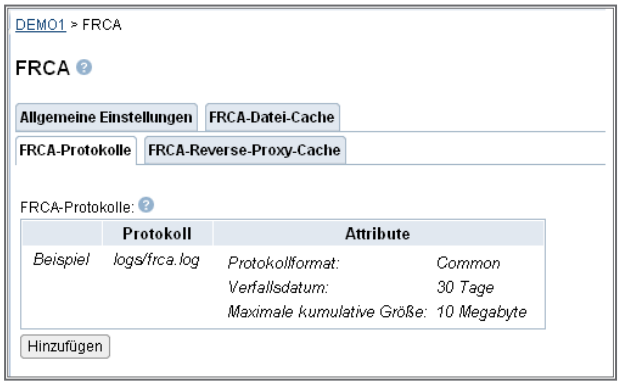
5.1.6.2
Seite 6

Neben diesen allgemeinen FRCA-Konfigurationen steht ein weiterer FRCA-Verwaltungsbereich zur Verfügung, den wir im linken Navigationsabschnitt finden:



Wie in der vorhergehenden Abbildung zu sehen ist, lassen sich für die FRCA-Nutzung separate IP-Adressen mit eigenen Ports definieren. Damit können FRCA-Einstellungen auf diese IP-Bereiche beschränkt und einem speziellen Anwendungs- oder Anwenderbereich zur Verfügung gestellt werden.

Der Abschnitt „FRCA-Protokolle“ bietet die Möglichkeit, die FRCA-Einrichtung zu überprüfen.



Die FRCA-Protokolle enthalten wichtige Informationen zu den FRCA-Aktivitäten. Mit diesen Informationen lassen sich bereits definierte FRCA-Dateien kontrollieren.

Aktivierungsgruppen

Ein weiterer wichtiger Leistungsbereich für Webserveranwendungen sind CGI-Definitionen. Bei CGI handelt es sich um einfache Webanwendungen, die direkt auf dem Webserver ausgeführt werden. Bei der Umwandlung solcher Anwendungen kann zwischen zwei Verfahren gewählt werden:

1. Umwandlung in eine neue Aktivierungsgruppe

2. Umwandlung ein eine bestehende (benannte) Aktivierungsgruppe

Zwischen beiden Verfahren besteht in Bezug auf die Leistungswerte ein gravierender Unterschied. Die Angabe von benannten Aktivierungsgruppen beim Umwandeln eines CGI-Programms kann bis zu fünffache Leistungssteigerungen mit sich bringen. Doch was ist eigentlich eine solche Aktivierungsgruppe?

Beim Bereitstellen eines Programms für die spätere Benutzung wird dieses Programm umgewandelt und aktiviert. Bei der Aktivierung, die besonders seit der ILE-Technologie auf dem System i an Bedeutung gewonnen hat, wird auch der von der Anwendung benötigte Speicher definiert und initialisiert sowie statischer Speicher für das Binden der Programmbereiche zu einem Serviceprogramm zugeordnet. Diese Form der Aktivierung und Zuordnung von Speicher ist keine Besonderheit auf dem System i, sondern eine allgemein gängige Praxis auch für CGI-Anwendungen. Dennoch – das System i wäre nichts besonderes, wenn es nicht auch beim Betrieb von CGI-Anwendungen Besonderheiten geben würde. Die Verwendung sogenannter Aktivierungsgruppen ist eine solche. Eine Aktivierungsgruppe definiert alle Ressourcen, die für die Ausführung eines Programms oder eines Serviceprogramms benötigt werden. Diese Ressourcen definieren sich aus:

- dynamischem Speicher
- Programmvariablen
- Ressourcen für die Speicherverwaltung (z. B. SQL Cursor)
- Exception Handlers

Wenn wir beispielsweise ein ILE RPG-Programm definieren, dann kann die Aktivierungsgruppe über den Parameter ACTGRP festgelegt werden. Auch dort stehen als Parameter „*NEW“ oder „*CALLER“ oder „benutzerdefiniert“ zur Auswahl. Wird letztere Option verwendet, dann generiert das System diese Aktivierungsgruppe automatisch bei der ersten Verwendung. Diese Option sollte für alle persistenten CGI-Programme verwendet werden!

5.1.6.2
Seite 8

Wie im wirklichen Leben, so hat auch dieser massive Performancevorteil einen Nachteil. Bei der Verwendung von benannten Aktivierungsgruppen ist das System selbst nicht in der Lage, diese Aktivierungsgruppen und die damit belegten Ressourcen zu löschen bzw. freizugeben. So sollte ein Anwendungsentwickler beispielsweise sicherstellen, dass die in einer Anwendung genutzten Dateien auch programmgesteuert geschlossen werden. Weiterhin ist sicherzustellen, dass beim Einsatz von Multithreaded-Anwendungen diese Threads von der Anwendung gesteuert werden.

Die Definitionen für den Webserver sind mannigfaltig. Ein wichtiger Faktor im Hinblick auf die Leistungswerte einer Webserver-Installation sind auch die Cache-Einstellungen für die zu verarbeitenden Dateien.

Sicherheit für den Betrieb eines Webservers

Sicherheitseinstellungen sind gerade bei Internetanwendungen immens wichtig.

Je sicherer eine Verbindung oder eine Anwendung, desto mehr wirkt sich dies auf die Leistungsdaten einer Anwendung aus. Verschlüsselungsverfahren, die auch in Verbindung mit einem Webserver auf dem System i eingesetzt werden können, verlangsamen die Anwendungen, die auf dem Webserver ausgeführt werden.

Deshalb sollte von Fall zu Fall entschieden werden, ob eine Anwendung tatsächlich den Sicherheitseinstellungen genügen muss, oder ob gegebenenfalls auf den Einsatz der Verschlüsselung bei einzelnen Anwendungen verzichtet werden kann. Vergleicht man unsichere Verbindungen mit sicheren Verbindungen, und wird bei den sicheren Verbindungen auch noch auf den Einsatz der „KeepAlive“-Option verzichtet, die unter bestimmten Voraussetzungen Verbindungskomponenten weiter nutzt und damit die Geschwindigkeit bei solchen Verbindungen verbessert, dann kann der Einsatz einer solchen Verbindung bis zu fünfmal so viel CPU-Rechenzeit benötigen, wie eine vergleichbare ungesicherte Verbindung.

KeepAlive gestattet die Verarbeitung von verschiedenen Anforderungen über ein und dieselbe TCP/IP-Verbindung. Dies bietet sich besonders dann an, wenn Webseiten über mehrere Bilder verfügen. Wenn KeepAlive auf „aus“ gestellt ist, dann wird für jedes Bild (gegebenenfalls für jede Gruppe von Elementen der Webseite) eine eigene TCP/IP-Verbindung hergestellt. Dieser Prozess ist nicht nur auf dem System i sehr performanceintensiv! Deshalb sollte der Einsatz von KeepAlive abgewägt werden.

Persistente Verbindungen werden in Verbindung mit KeepAlive auch bei Webanwendungen eingesetzt. Bei einer persistenten Verbindung handelt es sich um eine TCP/IP-Verbindung, die für verschiedene Transaktionen gemeinsam genutzt wird. Solche Transaktionen werden in Verbindung mit Webservern als KeepAlive-Transaktionen bezeichnet. Doch warum sollte man persistente Verbindungen nutzen – sind diese nicht etwa widersprüchlich zu einem Sicherheitskonzept, nach dem jede Verbindung unmittelbar nach deren Verwendung getrennt bzw. entfernt werden sollte? Betrachtet man die Eigenschaften eines Verbindungsaufbaus, dann stellt man fest, dass es sich dabei um einen sehr CPU-intensiven Prozess handelt. Nutzt man eine einmal erstellte Verbindung für unterschiedliche Transaktionen, dann kann sich damit das Laufzeitverhalten deutlich verbessern. Dies trifft besonders dann zu, wenn gesicherte Verbindungen genutzt werden.

5.1.6.2
Seite 10

Applikations-Server

Angetrieben durch die neueren Formen der Anwendungsentwicklung sowie deren Anforderungen an die Systeme und die Datenstrukturen, mussten die Hersteller der Server und Betriebssysteme den neuen und modernen Umgebungen dahingehend Rechnung tragen, dass neben der Zur-Verfügung-Stellung der Entwicklungsumgebungen für Programmiersprachen – wie zum Beispiel Java – auch entsprechende Laufzeitumgebungen und nicht zuletzt auch Ausführungsschichten auf den Server-Plattformen verfügbar gemacht werden. Der allgemeine Einzug des e-Business – einem der „Unworte" der letzten Jahre – fordert neben dem im vorhergehenden Kapitel bereits beschriebenen Schichtenmodell auch die entsprechenden Umgebungen. Da eine Integration der Anforderungen in jedes einzelne Betriebssystem kein gangbarer Weg war, hat man sich dazu entschlossen, mittels Middleware, speziellen Werkzeugen und Ausführungsschichten entsprechende Lösungen für diese Anforderungen zu schaffen, die wir heute zum Beispiel auch in Form der Applikations-Server finden.

Neben den allgemeinen Anforderungen an einen Applikations-Server sind es sicher auch die plattformspezifischen Merkmale und Eigenschaften, die bei dem Einsatz eines Applikations-Servers auf oder in Verbindung mit einem Server eine wesentliche Rolle dabei spielen, wieweit die Integration mit diesem System möglich ist und welche Ressourcen-Nutzungen dadurch optimiert werden können.

IBM hat sich im Laufe der letzten Jahre diesen Problematiken und Herausforderungen ebenfalls wie weitere führende Hard- und Software-Hersteller gestellt und in diesem Zuge eine eigene Applikations-Server-Familie – bekannt unter dem Namen „WebSphere" – ins Leben gerufen. Dabei handelt es sich um eine Vielzahl von ehemaligen Einzelprodukten, die zusammengeführt und entsprechend um neue Funktionen und Werkzeuge ergänzt wurden. WebSphere stellt heute eines der strategischen Produkte der IBM dar.

Es gibt auf dem Markt eine Vielzahl von Applikations-Servern. In diesem Kapitel wollen wir uns bewusst nur mit diesen Applikations-Servern beschäftigen, die auf der iSeries direkt ausführbar sind. Weitere Applikations-Server, die zum Beispiel in einer Linux-Partition auf der iSeries ausgeführt werden können, sind zunächst nicht Gegenstand dieses Kapitels.

Bevor wir in die Einzelheiten der WebSphere Applikations-Server gehen, sollte an dieser Stelle klar zwischen den WebSphere Applikations-Servern und der Entwicklungsumgebung unterschieden werden. Letztere trägt zwar auch den Namen „WebSphere", soll hier aber nicht Gegenstand des Interesses sein, obwohl sie sicher in direktem Zusammenhang mit den Servern zu sehen ist. Denn ohne Entwicklungsumgebung gibt es ja schließlich auch keinen

5.2
Seite 2

Nutzen für die Ausführungsebene – in diesem Fall für den WebSphere Applikations-Server.

Der WebSphere Applikations-Server ist seit der Betriebssystem-Version V4R5 auf der iSeries verfügbar und die Integration auf diese Plattform wurde in den letzten Release-Versionen kontinuierlich ausgebaut.

Für den praktischen Einsatz vom WebSphere Applikations-Server auf der iSeries empfiehlt sich mindestens die Verwendung der Betriebssystem-Version V5R1 oder höher.

In den nachfolgend beschriebenen Verfahren und Beispielen bei der Installation und Administration der unterschiedlichen WebSphere Applikations-Server wurde die OS/400-Betriebssystem-Version V5R2 verwendet.

Es ist in jedem Fall darauf zu achten, dass die erforderlichen Gruppen-PTFs, die speziell für die unterschiedlichen Versionen der WebSphere Applikations-Server verfügbar sind, auf dem System installiert werden. Wundern Sie sich nicht, wenn Sie bei der Lieferung der PTF neben den Gruppen-PTF für den WebSphere Applikations-Server auch die Gruppen-PTF für die Bereiche

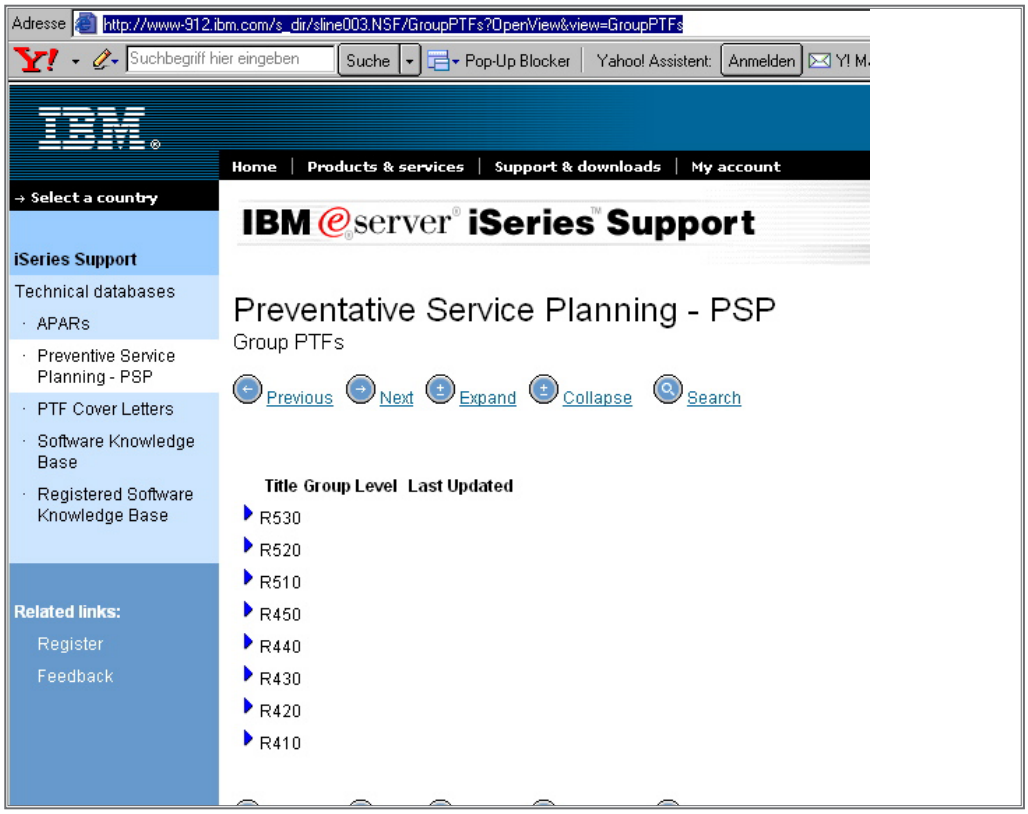
- DB2/400 (SF99502)
- Java (SF99169)
- HTTP Server (SF99098)
- IBM Business Solutions (SF99172)

mitgeliefert bekommen. Der Grund liegt in den engen Vernetzungen der Produkte zueinander. Wie diese aussehen, werden Sie im Verlauf des nächsten Kapitels noch kennen lernen.

Die für die jeweiligen Betriebssystem-Versionen und unterschiedlichen Versionen der Applikations-Server erforderlichen PTFs finden Sie unter der nachfolgenden URL:

HTTP://www-912.ibm.com/s_dir/sline003.NSF/GroupPTFs?OpenView&view=GroupPTFs

Hier bekommen Sie alle verfügbaren PTFs getrennt nach dem jeweiligen Betriebssystem-Release angezeigt.



PTF

Wählen Sie in dieser Übersicht ihr derzeit installiertes OS/400-Release aus. Auf der nächsten Seite bekommen Sie nun eine Übersicht aller verfügbaren Gruppen-PTFs für diese Release-Version. Bereits hier können Sie sehen, wie viele unterschiedliche WebSphere Applikations-Server für das jeweilige OS/400 Release verfügbar sind.

Preventative Service Planning - PSP

Group PTFs

Previous Next Expand Collapse Search

Title	Group Level	Last Updated
R530		
R520		
SF99519: 520 Group Hiper	Level 107	07/20/2004
SF99502: 520 DB2 UDB FOR ISERIES	Level 15	07/15/2004
SF99280: 520 WebSphere Portal Express/Express Plus Service P	Level 5	07/13/2004
SF99279: 520 WAS - ND V5.1.0.4 5733W51	Level 3	06/01/2004
SF99277: 520 WAS - Base V5.1.0.4 5733W51	Level 3	06/01/2004
SF99274: 520 WebSphere App Server - Express V5.1	Level 4	08/06/2004
SF99271: 520 WebSphere App Server - Express V5.0	Level 17	06/24/2004
SF99246: 520 WebSphere App Server ND V5.0	Level 8	06/21/2004
SF99245: 520 WebSphere App Server V5.0 (Base Edition)	Level 10	06/30/2004
SF99232: 520 Websphere Commerce Suite V5.4 for V5R2M0	Level 4	04/15/2004
SF99172: 520 IBM Business Solutions	Level 3	06/04/2004
SF99169: 520 JAVA	Level 17	04/21/2004
SF99166: 520 SF99166 - iSeries Connect 2.0 PTF Group 1 for V	Level 1	03/30/2004
SF99149: 520 WAS - Advanced 4.0.7 5733WS4	Level 9	03/30/2004
SF99148: 520 WAS - Advanced 4.0.7 5733WA4	Level 10	03/30/2004
SF99098: 520 IBM HTTP SERVER FOR ISERIES	Level 15	06/16/2004
SF99085: 520 520 Backup Recovery Solutions	Level 13	06/29/2004
SF99039: 520 Performance Tools for iSeries	Level 2	03/30/2004

WebSphere PTFs

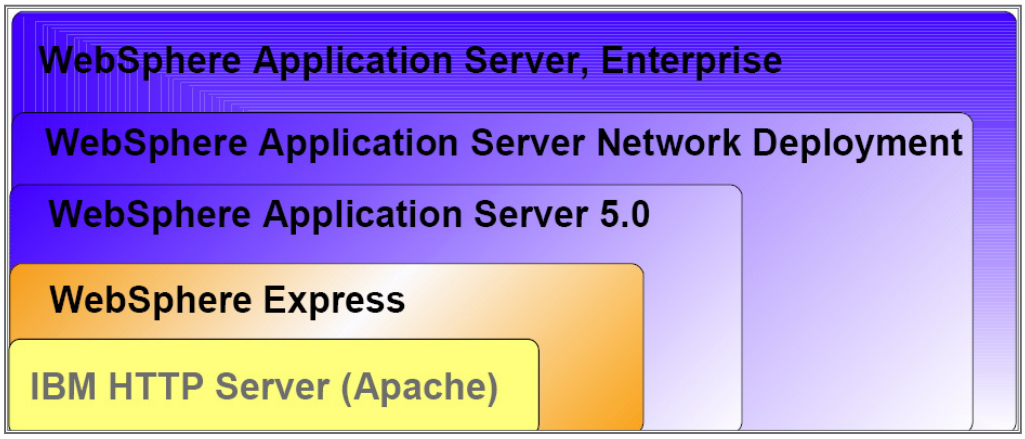
Doch wir wollen nun nicht vorgreifen, denn die unterschiedlichen Versionen der WebSphere Applikations-Server sollen nachfolgend zunächst einmal kurz beschrieben werden.

IBM hat im Bereich von WebSphere zunächst eine ganze Reihe von ehemals eigenständigen Produkten und Funktionen, die für die Integration von „modernen“ Anwendungen (e-Business) erforderlich sind, zusammengefasst. Basis für die Ausführung dieser Anwendungen stellen die WebSphere Applikations-Server dar.

IBM hat für die unterschiedlichen Anforderungen einige Varianten der Applikations-Server entwickelt. Diese wollen wir uns im nächsten Abschnitt in Form einer Übersicht anschauen. Details zu den einzelnen Versionen finden Sie dann in den entsprechenden Unterabschnitten dieses Kapitels.

Die verschiedenen Server-Versionen unterscheiden sich in ihrem Funktionsumfang – und leider auch im Preis!

Die folgende Grafik beinhaltet schematisch die derzeitige aktuelle WebSphere Server-Familie:



WebSphere Familie

IBM HTTP Server (Apache)

Eigentlich ist dieser Eintrag kein WebSphere Applikations-Server, allerdings basieren viele Funktionen der Applikations-Server auf dem HTTP-Server (zum Beispiel die Administration der WebSphere Applikation Server). Aus diesem Grund ist dieser in der Abbildung als Basis für alle Applikations-Server genannt.

Wie man in der Abbildung „WebSphere Familie" sehen kann, sind die einzelnen Server-Varianten stufenweise aufgebaut. Dies ist bewusst auch in der Abbildung so dargestellt, denn die einzelnen Server-Versionen beinhalten die Funktionen der „untergeordneten" Versionen – d.h.: Sie bauen also aufeinander auf.

Entgegen den Server-Versionen 4.x sind alle Server der 5.x-Familie aufeinander aufbauend und verwenden intern jeweils den Code des untergeordneten Servers.

WebSphere Express

Diese Einstiegsvariante der WebSphere Applikations-Server Familie bietet gerade in Verbindung mit den ersten Schritten in diesem Bereich eine sinnvolle Variante für den Einsatz eines Applikations-Servers auf der iSeries. Der WebSphere Applikations-Server Express ist seit der OS/400-Version V5R1 verfügbar, ist unter anderem auch Voraussetzung für einfache e-Business-Anwendungen und kann auch für den Einsatz von WebFacing-Anwendungen verwendet werden. Der wesentliche Unterschied zu den „größeren" Web Sphere Servern liegt in der fehlenden Unterstützung von Enterprise Java Beans (EJB), die für die Ausführung von komplexen e-Business-Logiken erforderlich sind.

In den meisten Fällen reicht – zumindest für den Einstieg – diese Express-Version aus!

Die Installation des WebSphere Applikations-Server Express erfolgt in Form eines auf der iSeries zu installierenden Lizenzprogramms (5722-IWE), das für die Release-Versionen V5R1, V5R2 und V5R3 verfügbar ist.

Derzeit (Stand August 2004) werden die folgenden unterschiedlichen Versionen des Express-Servers unterstützt:

- Version V5.0
- Version V5.1
- Version V5.1.1

Gibt es keine Einschränkungen durch das auf der iSeries installierte Betriebssystem, dann empfiehlt sich der Einsatz der neuesten Version!

Was wird nun von dem WebSphere Applikations-Server Express unterstützt?

Dies sind als Hauptbereiche die folgenden Funktionen:

- Java Server Pages
- Java Servlets
- Web Services
- Spezielle iSeries-Funktionen – wie zum Beispiel WebFacing und HATS

Mit diesem Applikations-Server können unter anderem auch statische und dynamische Web-Seiten verarbeitet werden.

Der wesentliche Unterschied zu den weiteren WebSphere Applikations-Servern liegt in der fehlenden Unterstützung von Enterprise Java Beans (EJB).

WebSphere Application-Server 5.0 (Base Edition) für iSeries

Der WebSphere Applikations-Server in der Version 5.0 (kurz WAS5.0) bietet neben der Express-Version die nächste Einstiegsvariante. Alle Funktionen, die der Express-Server beinhaltet, sind auch in dem WAS5.0 integriert. Darüber hinaus bietet dieser erweiterte Funktionen – wie zum Beispiel die Unterstützung von EJBs – an.

Der WebSphere Applikations-Server unterstützt die folgenden Techniken:

- Java Servlets
- Java Server Pages (JSP)
- Java Message Service (JMS)
- Web Services
- Enterprice Java Beans (EJB)
- J2EE Konnektoren
- …

Der WAS5.0 ist ab der Betriebssystem-Version V5R1 verfügbar und wird auch unter den Release-Ständen V5R2 und V5R3 unterstützt.

Die Funktionen dieser Version werden durch unterschiedliche Modifikationsstufen erweitert. In einem gewissen Rahmen werden auch Fehlerbehebungen innerhalb der Modifikationsstufen durchgeführt. Derzeit sind folgende Modifikationsstufen verfügbar (Stand August 2004):

- WAS5.0
- WAS5.0.1
- WAS5.0.2

WebSphere Application-Server 5.1 (Base Edition) für iSeries

Die Weiterentwicklung des WAS5.0 steht mit dem WebSphere Applikations-Server 5.1 (kurz WAS5.1) zur Verfügung. Dieser bietet die vergleichbaren Funktionen wie sein Vorgänger (der WAS5.0) – ergänzt um neue Funktionen, die in dem Unterkapitel „WebSphere Applikations-Server 5.1“ näher beschrieben sind.

Der WAS5.1 ist für folgende Betriebssystem-Release-Stände verfügbar:

- V5R1
- V5R2
- V5R3

Die Funktionen dieser Version werden durch unterschiedliche Modifikationsstufen erweitert und in einem gewissen Rahmen auch Fehlerbehebungen innerhalb der Modifikationsstufen durchgeführt. Derzeit sind folgende Modifikationsstufen verfügbar (Stand August 2004):

- WAS5.1
- WAS5.1.1

WebSphere Application-Server Network Deployment Version 5.0 for iSeries

Zusätzlich zu den Funktionen der Base-Editionen bietet die Network-Variante erweiterte Funktionen und Werkzeuge in den folgenden Bereichen an:

- Verfügbarkeit
- Skalierbarkeit
- Unterstützung für UDDI Registry und Web Services Gateways
- Erweiterte Fehlerbehandlung
- Network Dispatcher
- …

Der WAS ND5.0 ist für folgende Betriebssystem-Release-Stände verfügbar:

- V5R1
- V5R2
- V5R3

Die Funktionen dieser Version werden durch unterschiedliche Modifikationsstufen erweitert. In einem gewissen Rahmen werden auch Fehlerbehebungen innerhalb der Modifikationsstufen durchgeführt. Derzeit (Stand August 2004) sind folgende Modifikationsstufen verfügbar:

- WAS5.0
- WAS5.0.1
- WAS5.0.2

WebSphere Application Server Network Deployment Version 5.1 for iSeries

Das Nachfolgeprodukt für den WAS ND5.0 stellt die Version 5.1 dar.

Der WAS ND5.1 ist für folgende Betriebssystem-Release-Stände verfügbar:

- V5R1
- V5R2
- V5R3

Bei den zuvor genannten Server-Versionen handelt es sich um die aktuellen Versionen und Release-Stände (Stand August 2004).

Der Vollständigkeit halber werden nachfolgend die Vorgänger-Versionen der V5 Server-Familie erwähnt, die derzeit noch von IBM unterstützt werden:

WebSphere Applikation Server V4.0 Advanced Single Server Edition für iSeries

Der WAS4.0 ist für folgende Betriebssystem-Release-Stände verfügbar:

- V4R5
- V5R1
- V5R2

Der WAS4.0 ist der letzte Applikations-Server, der den originalen IBM HTTP-Server unterstützt.

Wie in dem Kapitel „HTTP-Server“ näher beschrieben, hat IBM diesen IBM-eigenen HTTP-Server zu Gunsten des HTTP-Server „powered by Apache“ ersetzt und die Entwicklung des eigenen Servers eingestellt.

Auch dieser Server ist in verschiedenen Modifikationsstufen verfügbar, die jeweils Erweiterungen und zum Teil auch Fehlerbehebungen enthalten:

- WAS4.0.2
- WAS4.0.3
- WAS4.0.4
- WAS4.0.5
- WAS4.0.6
- WAS4.0.7

Neben diesen „Basisversionen“ der WebSphere Server-Familie gibt es noch weitere Produktkategorien, die sich insbesondere mit Portalen und komplexen Prozessintegrationen beschäftigen. Auf diese werden wir später noch näher eingehen.

5.2
Seite 10

WebSphere Applikations-Server Express

In der Übersicht der Applikations-Server wurden die Grundlagen des Web Sphere Applikations-Servers Express (kurz WASE) bereits kurz beschrieben. In diesem Abschnitt möchte ich Ihnen mehr Details über die Funktionen, die Installation, die Konfiguration und die Administration dieses Servers geben.

Wie bereits erwähnt, handelt es sich beim WASE um die Einstiegsvariante der Applikations-Server; er bietet in den meisten Fällen für den Einstieg in die e-Business-Welt ausreichende Funktionen.

WASE ist seit der Betriebssystem-Version V5R1 verfügbar und wird als Lizenzprogramm (5722-IWE) auf der iSeries installiert. Bevor die Installation vom WASE durchgeführt werden kann, müssen zunächst die Voraussetzungen für eine erfolgreiche Installation erfüllt sein.

Prüfen Sie vor dem Einsatz des WASE, ob diese Voraussetzungen erfüllt sind.

iSeries Hardware-Voraussetzungen:

- Mind. 350 CPW
  - **Achtung!** Einige Funktionen werden auf „alten“ iSeries-Modellen noch im interaktiven Bereich ausgeführt (z.B. WebFacing), die interaktive Leistung muss dann auch entsprechend verfügbar sein!
- Mind. 512 MB Hauptspeicher
- 2 MB L2 Cache
  - Dies gilt als dringende Empfehlung, wobei der Einsatz „älterer“ iSeries-Prozessoren nicht wirklich sinnvoll ist!
- Kommunikationsverbindung (TCP/IP)

Diese Hardware-Voraussetzungen gelten – gerade in den Bereichen CPW und Hauptspeicher – als MINDESTVORAUSSETZUNGEN!

Zudem muss an dieser Stelle dringend darauf hingewiesen werden, dass einige Funktionen, die auf WebSphere Applikations-Server Express ausgeführt werden (zum Beispiel WebFacing), auf „älteren“ iSeries Prozessoren (das sind die, die vor 2003 angekündigt wurden) im interaktiven Bereich ausgeführt werden, was oft zu Lasten der Dialoganwendungen geht!

Für die Installation und Ausführung des WASE sind neben den Hardware-Voraussetzungen auch einige Lizenzprogramme auf der iSeries erforderlich, damit Installation und Ausführung von WASE sichergestellt sind.

5.2.1
Seite 2

iSeries Software-Voraussetzungen:

- Mind. OS/400 Version V5R1
- IBM Developer Kit for Java (5722-JV1), Version 1.4 (Option 6)
- OS/400 Qshell (5722-SS1 Option 30)
- IBM Toolbox for Java (5722-JC1)
- HTTP-Server
  - erforderlich für die Ausführung von Servlets oder Java Server Pages
  - wahlweiser Einsatz von:
    - IBM HTTP-Server powered by Apache (5722-DG1)
    - Lotus Domino (Release 5, oder Release 6, oder Release 6.5)

Diese Einstiegsvariante der WebSphere Applikations-Server-Familie bietet gerade in Verbindung mit den ersten Schritten in diesem Bereich eine sinnvolle Variante für den Einsatz eines Applikations-Servers auf der iSeries. Der WebSphere Applikations-Server Express ist seit der OS/400 Version V5R1 verfügbar, er ist unter anderem auch Voraussetzung für einfache e-Business-Anwendungen und kann unter anderem auch für den Einsatz von WebFacing-Anwendungen verwendet werden. Der wesentliche Unterschied zu den „größeren“ WebSphere Servern liegt in der fehlenden Unterstützung von Enterprise Java Beans (EJB), die für die Ausführung von komplexen e-Business-Logiken erforderlich sind.

In den meisten Fällen reicht – zumindest für den Einstieg – diese Express-Version aus!

Die Installation des WebSphere Applikations-Server Express erfolgt in Form eines auf der iSeries zu installierenden Lizenzprogramms (5722-IWE), das für die Release-Versionen V5R1, V5R2 und V5R3 verfügbar ist.

Derzeit (Stand August 2004) werden die folgenden unterschiedlichen Versionen des Express-Servers unterstützt:

- Version V5.0
- Version V5.1
- Version V5.1.1

Gibt es keine Einschränkungen durch das auf der iSeries installierte Betriebssystem, dann empfiehlt sich der Einsatz der neuesten Version!

Was wird nun von dem WebSphere Applikations-Server Express unterstützt?

- Java Server Pages
- Java Servlets
- Web Services
- Spezielle iSeries-Funktionen – wie zum Beispiel WebFacing und HATS

Wie bereits erwähnt ist der wesentliche Unterschied zu den übrigen WebSphere Applikations-Servern die fehlende Unterstützung von EJBs.

Neben der Ausführung von iSeries-spezifischen Anwendungen –wie das zuvor bereits genannte WebFacing und HATS – können mit dem WebSphere Applikations-Server Express unter anderem statische und dynamische Web-Seiten verarbeitet werden.

Wie Sie WASE installieren, konfigurieren und auf Ihre Bedürfnisse anpassen, wird in den folgenden Unterkapiteln ausführlich behandelt.

5.2.1
Seite 4

WebSphere Applikations-Server Express – Installation

Bevor mit der Installation des WebSphere Applikations-Server Express (kurz: „WASE“) begonnen werden kann, müssen die Hardware- und Software-Voraussetzungen für die Installation erfüllt sein. Diese wurden im vorhergehenden Abschnitt beschrieben.

Zunächst benötigen wir für die Installation und Ausführung ein iSeries-System, das mindestens über 300 CPW Leistung verfügt. Des Weiteren wird der Einsatz von mindestens einem der folgenden Maschinen-Typen empfohlen:

- Modell 270 (ab Prozessor 2250)
- Modell 800 (ab Prozessor 2463)
- Modell 820 (ab Prozessor 2395)

Die Installation ist zwar auch auf älteren Systemen möglich, allerdings ist die Ausführungsgeschwindigkeit auf diesen nicht zufrieden stellend.

Den Prozessortyp der verwendeten iSeries kann wie folgt ermittelt werden:

```
DSPSYSVAL SYSVAL(QPRCFEAT)
```

In der Anzeige wird das Prozessor-Feature aufgelistet.

```
_                              Systemwert anzeigen

 Systemwert . . . . . :   QPRCFEAT
 Beschreibung . . . . . :   Prozessor-Feature


 Prozessor-Feature  . . :   7430
```

Prozessor-Feature anzeigen

Auf dem System sollten mindestens 512 MB Hauptspeicher installiert sein (dies ist die Mindestangabe von IBM – in der Praxis würde ich 1 GB vorschlagen!).

Die Summe des Hauptspeichers kann in der Anzeige „Mit Systemstatus arbeiten“ ermittelt werden. Dazu geben Sie den Befehl „WRKSYSSTS“ ein. Die Summe der Werte, die unter der Spalte „Pool-Größe“ angegeben ist, stellt den installierten Hauptspeicher dar.

5.2.1.1
Seite 2

```
                                Mit Systemstatus arbeiten

  % CPU benutzt  . . . . . :        1,5    Zusatzspei
  % DB-Kapazität . . . . . :        0,0      System-A
  Abgelaufene Zeit . . . . :   00:00:28      % System
  Jobs im System . . . . . :       3435      Gesamtsu
  % perm. Adressen . . . . :      0,007      Unges. P
  % temp. Adressen . . . . :      0,010        belegt
                                             Max. ung
  Änderungen (falls zulässig) eingeben und die Eingab

 System    Pool-      Reserv.     Max.   -DB-Seiten--   -
  Pool   Größe(M)   Größe (M)   Aktiv   fehl.   geles   f
    1     218,28      120,05   +++++     0,0     0,0
    2    3317,87        1,18     112     0,0     0,0
    3     484,99        0,00     185     0,0     0,0
    4       0,25        0,00       3     0,0     0,0
```

Hauptspeicher anzeigen

Für die Installation des WASE sind mindestens 800 MB Plattenspeicher erforderlich. Den freien Plattenplatz kann man ebenfalls in der Anzeige des Befehls „WRKSYSSTS“ ermitteln.

Im oberen rechten Bereich der nachfolgenden Abbildung wird die Summe des gesamten verfügbaren Plattenplatzes angezeigt. Der freie Platz lässt sich über die Prozentangabe „% System ASP benutzt“ dann leicht errechnen:

```
  Mit Systemstatus arbeiten                          ASZEBIS
                                           11.08.04  09:38:15
.  :        1,5    Zusatzspeicher:
.  :        0,0      System-ASP . . . . . .  :     175,8 G
.  :   00:00:28      % System-ASP benutzt .  :     33,6328
.  :       3435      Gesamtsumme  . . . . .  :     175,8 G
```

Verfügbarer Plattenplatz

Sind diese Hardware-Voraussetzungen erfüllt, dann sollten die Software-Voraussetzungen überprüft werden.

Wie bereits erwähnt, benötigen wir mindestens die OS/400-Version V5R1 für die Installation von WebSphere Applikations-Server Express.

Dabei müssen auf dem System die folgenden Lizenzprogramme installiert sein:

- IBM Developer Kit for Java (Mind.Version 1.3)
  - 5722-JV1, Option 5
- OS/400 QShell Interpreter
  - 5722-SS1, Option 30
- TCP/IP
  - 5722-TC1
- HTTP-Server
  - 5722-DG1

  Als HTTP-Server können eingesetzt werden:
  - · HTTP-Server powered by Apache

    verfügbar als PTF für 5722-SS1
  - · Lotus Domino für iSeries

Die installierten Lizenzprogramme können mit dem Befehl „DSPSFWRSC“ angezeigt werden.

```
                        Software-Ressourcen anzeigen
                                                          System:   ASZEBIS

  Ressour.  Option  Feature  Beschreibung
 _5722SS1   16       2929    OS/400 - Ultimedia System Facilities
  5722SS1   18       5103    OS/400 - Media and Storage Extensions
  5722SS1   21       5050    OS/400 - Erw. Unterstützung f. Landessprachen
  5722SS1   22       5050    OS/400 - ObjectConnect
  5722SS1   22       2929    OS/400 - ObjectConnect
  5722SS1   25       5109    OS/400 - NetWare Enhanced Integration
  5722SS1   25       2929    OS/400 - NetWare Enhanced Integration
  5722SS1   30       5050    OS/400 - QShell Interpreter
  5722SS1   30       2929    OS/400 - QShell Interpreter
  5722SS1   31       5050    OS/400 - Domain Name System
  5722SS1   31       2929    OS/400 - Domain Name System
  5722SS1   33       5111    OS/400 - Portable App Solutions Environment
  5722SS1   33       2929    OS/400 - Portable App Solutions Environment
  5722SS1   34       5050    OS/400 - Digital Certificate Manager
                                                                  Weitere ...
  Eingabetaste --> Weiter

  F3=Verlassen   F11=Bibliotheken/Releases anzeigen   F12=Abbrechen
  F19=Marken anzeigen
```

Installierte Software anzeigen

Sind alle Voraussetzungen erfüllt, dann kann mit der Installation begonnen werden.

5.2.1.1
Seite 4

Legen Sie dazu zunächst die Installations-CD in das entsprechende CD/DVD-Laufwerk der iSeries ein.

Die Installation des WASE besteht aus drei einzelnen Optionen:

- 5722-IWE Option *BASE
- 5722-IWE Option 2
- 5722-IWE Option 3

Die Option *BASE ist in jedem Fall zu installieren und stellt die Basis für die übrigen Optionen dar! Diese Basis-Option kann ohne die beiden anderen Optionen eingesetzt werden. Das wäre dann der Fall, wenn der Server nur für HTTP-Routing-Zwecke eingesetzt wird. Allerdings wäre das in der Tat unpraktikabel. Vielmehr macht die Installation in Verbindung mit mindestens der „Option 2“ Sinn:

Die „Option 2“ stellt die Ausführungsumgebung des WASE dar. Diese ist dann erforderlich, wenn Anwendungen auf WASE installiert und ausgeführt werden sollen.

Wahlweise kann als Zusatz die „Option 3“ installiert werden. Diese enthält eine Beispielanwendung: „IBM Telephone Directory“.

Für die Installation ist ein Benutzerprofil erforderlich, dass mindestens über die Sonderberechtigungen *ALLOBJ und *SECADM verfügt. Im Idealfall verwenden Sie für die Installation das Benutzerprofil QSECOFR, um Probleme wegen mangelnder Berechtigung zu vermeiden!

Die Installation des WASE kann im laufenden Betrieb der iSeries durchgeführt werden. Lediglich bei der abschließenden Installation einiger PTFs ist ein IPL erforderlich.

Der Start der Installation erfolgt über den Befehl:

```
RSTLICPGM LICPGM(5722IWE) DEV(OPT01)
```

Mit diesem Befehl wird die *Base-Option auf der iSeries installiert. Vergessen Sie nicht, bei Bedarf – und das ist eigentlich fast immer der Fall(!) – auch die „Option 2“ zu installieren. Dies wird mit dem folgenden Befehl durchgeführt:

```
RSTLICPGM LICPGM(5722IWE) DEV(OPT01) OPTION(2)
```

Achten Sie auf eventuelle Fehlermeldungen als Ergebnis der Installation. Die Installation kann im Menü „Mit Lizenzprogrammen arbeiten“ überprüft werden.

In dieses Menü gelangen Sie über die Eingabe „GO LICPGM“.

```
LICPGM                  Mit Lizenzprogrammen arbeiten

Auswahlmöglichkeiten:

  Manuelle Installation
     1. Alle Lizenzprogramme installieren

  Vorbereitung
     5. Installationsvorbereitung

  Lizenzprogramme
    10. Installierte Lizenzprogramme anzeigen
    11. Lizenzprogramme installieren
    12. Lizenzprogramme löschen
    13. Lizenzprogramme sichern
```

GO LICPGM

Wählen Sie anschließend die Auswahl „10“ = „Installierte Lizenzprogramme anzeigen“ aus. Blättern Sie nun so weit, bis Sie zu dem Eintrag 5722-IWE kommen. Damit die installierten Optionen angezeigt werden, muss in dieser Anzeige die Befehlstaste „F11“ zweifach gedrückt werden. Finden Sie die beiden Einträge

- 5722-IWE *BASE WebSphere Application Server Express
- 5722-IWE 2 IBM WebSphere Application Server Express V5

in der Auflistung? Dann ist die Basis-Installation erfolgreich abgeschlossen!

```
_                  Installierte Lizenzprogramme anzeigen
                                                          Syste
 Lizenz-   Produkt-
 programm   option   Beschreibung
 5639C34    1        WebSphere MQ Classes for Java - samples
 5722DE1    *BASE    DB2 UDB Extenders
 5722DE1    1        DB2 UDB Text Extender
 5722DE1    2        DB2 UDB XML Extender
 5722DE1    3        Text Search Engine
 5722DG1    *BASE    IBM HTTP-Server
 5722DG1    1        Triggered Cache Manager
 5722DP4    *BASE    DB2 DataPropagator
 5722IP1    *BASE    IBM Infoprint Server
 5722IWE    *BASE    WebSphere Application Server - Express
 5722IWE    2        IBM WebSphere Application Server Express V5
 5722JC1    *BASE    Toolbox for Java
 5722JS1    *BASE    Job Scheduler
 5722JV1    *BASE    Developer Kit for Java
```

Installierte Lizenzprogramme anzeigen

Grundwerk 10/2004

5.2.1.1
Seite 6

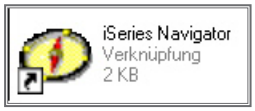
Im Zuge der Installation der Basisoption wird neben dem Lizenzprogramm 5722-IWE Option *BASE auch noch ein entsprechendes Verzeichnis auf der iSeries angelegt, in dem diverse Einstellungen des Applikations-Servers abgelegt werden. Während des Installationsprozesses wird unter anderem das Root-Verzeichnis für WSAE angelegt. Dies hat den folgenden Pfad/Namen:

/QIBM/ProdData/WebASE

Wie bereits erwähnt, werden in diesem Verzeichnis neben Installationseinstellungen auch Konfigurationseinstellungen gespeichert.

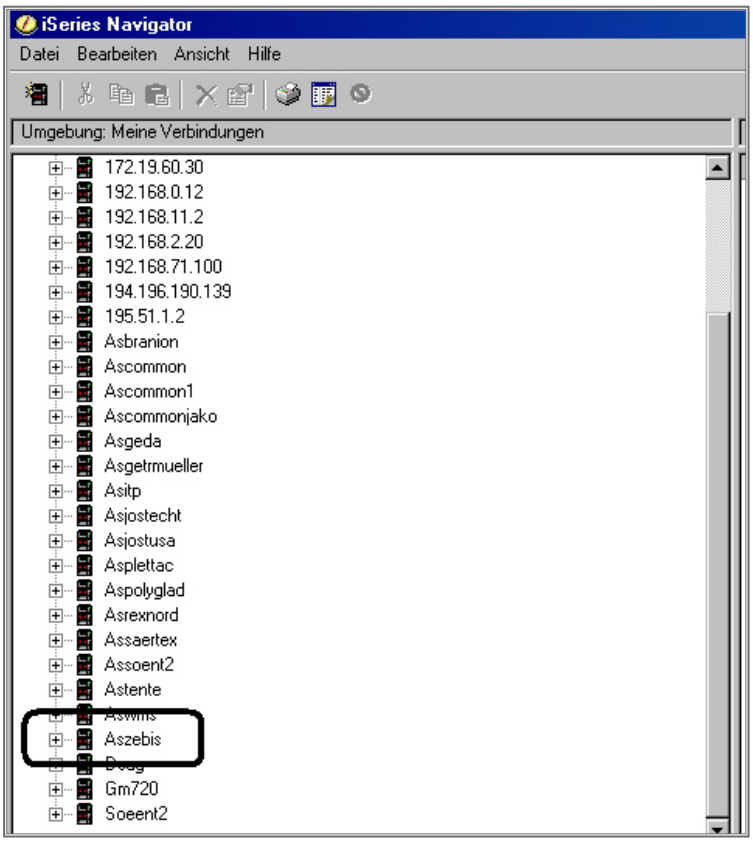
Die Installation dieses Verzeichnisses kann unter dem iSeries Navigator geprüft werden.

Dazu öffnen wir zunächst den iSeries Navigator über das entsprechende Symbol auf dem Desktop oder über den Programmgruppeneintrag „iSeries Access for Windows“



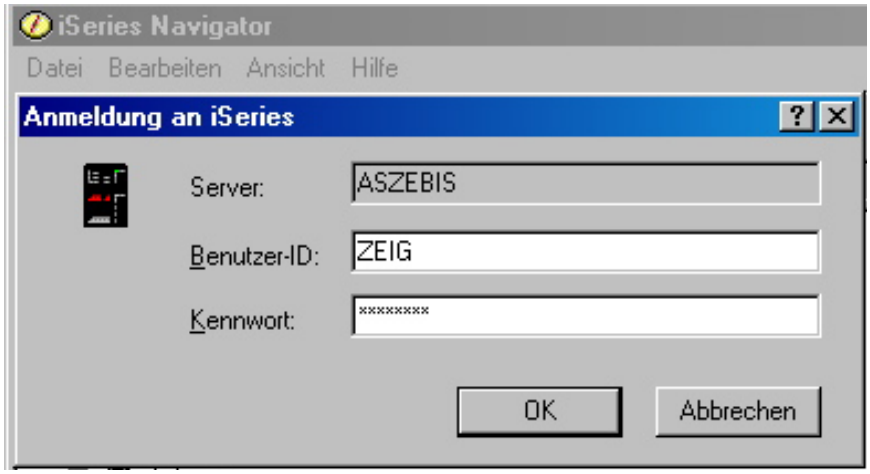
T5-K2-1-1-7 iSeries Navigator starten

Wählen Sie im linken Bereich des iSeries Navigators das zu verwaltende iSeries-System aus.



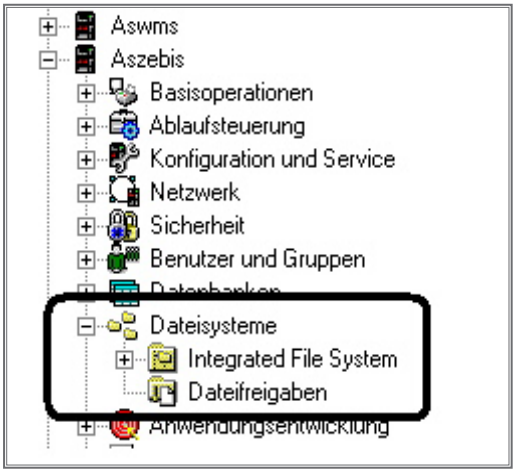
iSeries auswählen

Wenn Sie noch nicht über den iSeries Navigator am iSeries-System angemeldet waren, werden Sie nun zur Anmeldung aufgefordert. Geben Sie im folgenden Fenster den OS/400-Benutzer und das entsprechende Kennwort ein:



iSeries Anmeldung

Wählen Sie unterhalb des iSeries-Systems den Eintrag „Dateisysteme“ aus.



Dateisysteme

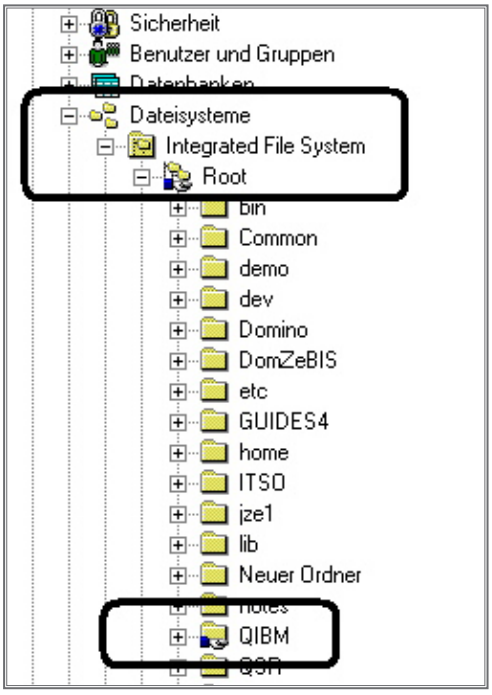
Anmerkung:

Sollten Sie unterhalb des iSeries-Systems keinen Eintrag „Dateisysteme“ haben, dann ist der iSeries Navigator nur fragmentarisch installiert. Installieren Sie in diesem Fall die entsprechenden Komponenten des iSeries Navigators nach. Diese finden Sie auf der entsprechenden CD „iSeries Access“.

Wir gehen hier nun davon aus, dass die erforderlichen Komponenten des iSeries Navigators bereits installiert sind.

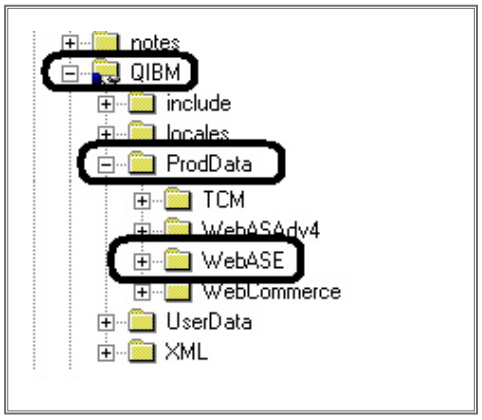
5.2.1.1
Seite 8

Innerhalb des Eintrages „Dateisysteme“ befindet sich der Eintrag „Integrated File System“, den wir durch einen Klick erweitern. Innerhalb des Eintrages „Integrated File System“ finden Sie nun den Eintrag „Root“ – auch diesen erweitern Sie durch einen Klick.



Integrated File System

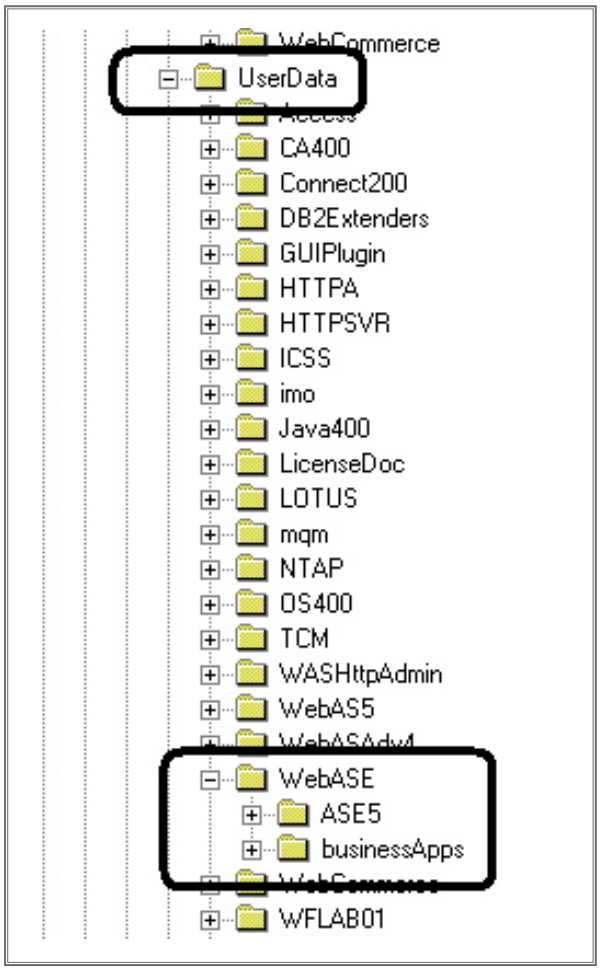
Sie finden nun unter anderem einen Eintrag „QIBM“. In diesem befinden sich auch die zuvor installierten Verzeichnisse des WASE. Öffnen Sie den Eintrag „QIBM“ durch einen Klick und führen Sie diesen Schritt nochmals in dem unter „QIBM“ stehenden Eintrag „ProdData“ aus.



ProdData

Hier finden Sie nun auch das Verzeichnis „WebASE“, das die Informationen des WebSphere Applikations-Servers enthält.

Neben den Installations- und Konfigurationseinstellungen im Verzeichnis „ProdData“ werden in einem zweiten Verzeichnis „UserData“ zusätzlich Benutzerinformationen gespeichert. Dieses Verzeichnis können wir nach dem zuvor beschriebenen Verfahren ansehen und auch verwalten.



UserData

Anmerkung:

Wenn Sie die „Option 3“ des WASE (das IBM Telephone Directory) installieren, dann wird dieses ebenfalls in diese Verzeichnisse installiert. In diesem Fall finden Sie die folgenden Einträge in den jeweiligen Verzeichnissen:

- /QIBM/ProdData/WebASE/businessApps/IBMTelDir
- /QIBM/UserData/WebASE/businessApps/IBMTelDir

Damit ist die Basisinstallation des WebSphere Applikations-Server Express beendet.

5.2.1.1
Seite 10

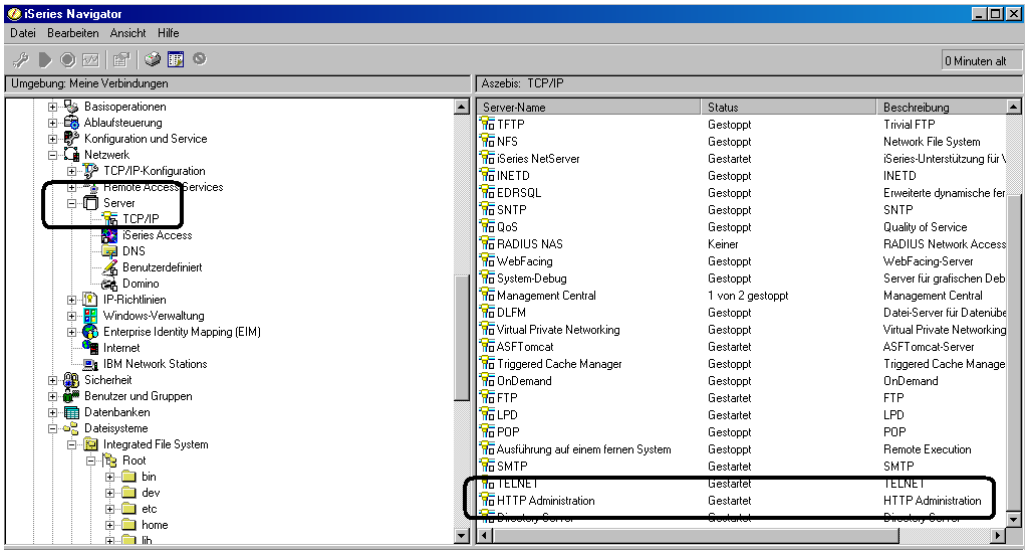
Als weitere Schritte folgen nun noch die Installation der erforderlichen PTFs und die Konfiguration des Applikations-Servers, die ich Ihnen in den nächsten Abschnitten erläutern werde.

WebSphere Express Konfiguration

Die Konfiguration eines WebSphere Express Servers (im nachfolgenden Abschnitt kurz WAS Express genannt) erfolgt über einen Browser-basierten Administrations-Client, der mit der HTTP-Administrationsinstanz der i5 ausgeführt wird.

Bevor wir unseren ersten WebSphere Express Server konfigurieren, müssen wir zunächst sicherstellen, dass die Admin-Instanz des HTTP-Servers der i5 gestartet wurde.

Dazu verwenden wir den iSeries Navigator, wie in der nächsten Abbildung gezeigt wird.



HTTP Admin gestartet

Sollte der Status des HTTP-Administrationseintrages nicht „gestartet“ sein, dann muss dieser vor dem Beginn der Konfiguration des WAS Express gestartet werden:

1. Starten des HTTP-Admin Bereichs unter TCP/IP
   STRTCPSVR SERVER(*HTTP) HTTPSVR(*ADMIN)
2. Starten des Browsers (in unserem Beispiel des Internet Explorers)
3. Angabe der Start-URL für die HTTP-Server Verwaltung (Host-Name und Admin-Port 2001)
   Zum Beispiel http://aszebis:2001 (für eine unsichere Verbindung – Port 2010 für eine sichere Verbindung)

Im Kapitel „Konfiguration HTTP Server“ haben wir diese Anzeige bereits kennen gelernt.

Wie bei der Konfiguration eines HTTP-Servers können auch im Zusammenhang mit einem WAS Express Server während der Konfiguration die Schalt-

5.2.1.2
Seite 2

flächen „Abbrechen“ oder „Weiter“ im unteren Bereich der Browser-Anzeige verwendet werden, um die Konfiguration zu beenden bzw. einen Arbeitsschritt zurückzugehen.

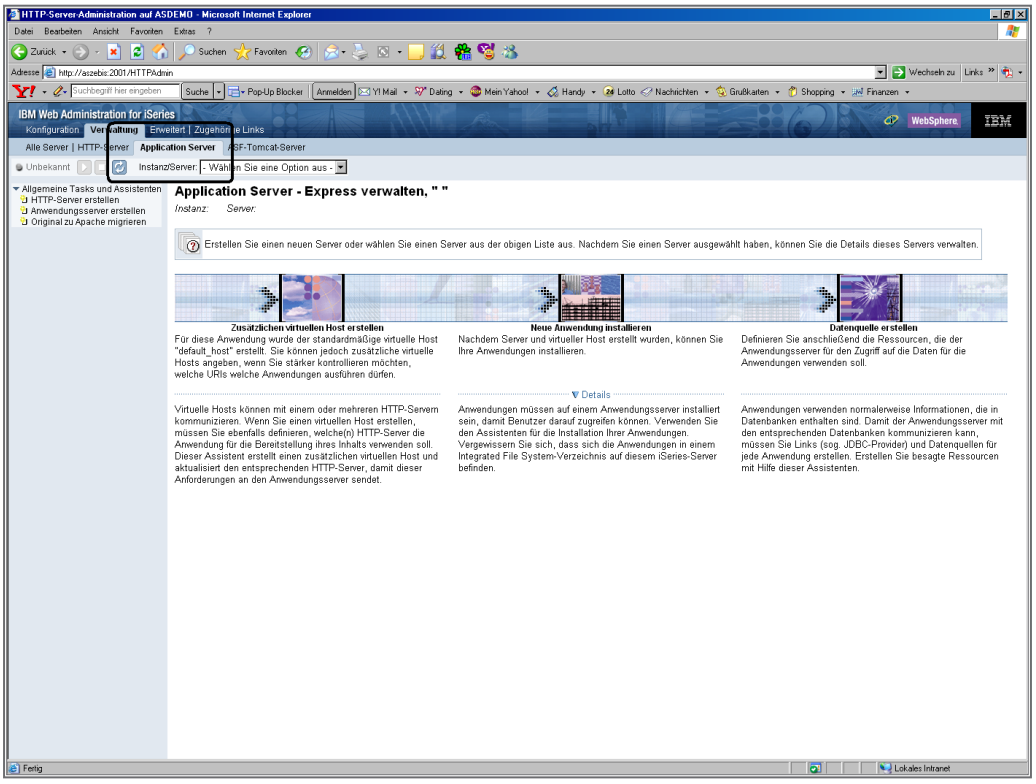
Diese Anzeige bekommen wir bei der Verwendung von V5R3M0 nach dem Aufruf angezeigt. Unter V5R2M0 weicht die Anzeige von unserem Beispiel leicht ab.

Konfiguration auswählen

Wir wählen für die Konfiguration des WAS Express Servers den ersten Eintrag „IBM Web Administration Express for iSeries“ aus.

Die folgende Anzeige wird sowohl für die Konfiguration der HTTP-Server als auch für den WAS Express Server verwendet.

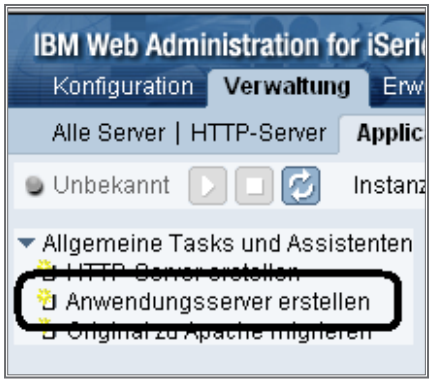
Je nach Voreinstellung bzw. nach dem letzten Verlassen wird die folgende Anzeige des Administrationsfensters variieren. Sollte es nicht bereits der Fall sein, dann wählen wir im oberen Anzeigebereich des Browsers den Eintrag „Application Server“ aus.



Application Server auswählen

Diese Einstiegsseite gibt uns einige Informationen, welche Aufgaben in diesem Bereich abgedeckt werden können.

Da wir einen neuen WAS Express Server erstellen wollen, wählen wir im linken Navigationsbereich den Eintrag „Anwendungsserver erstellen“ aus.

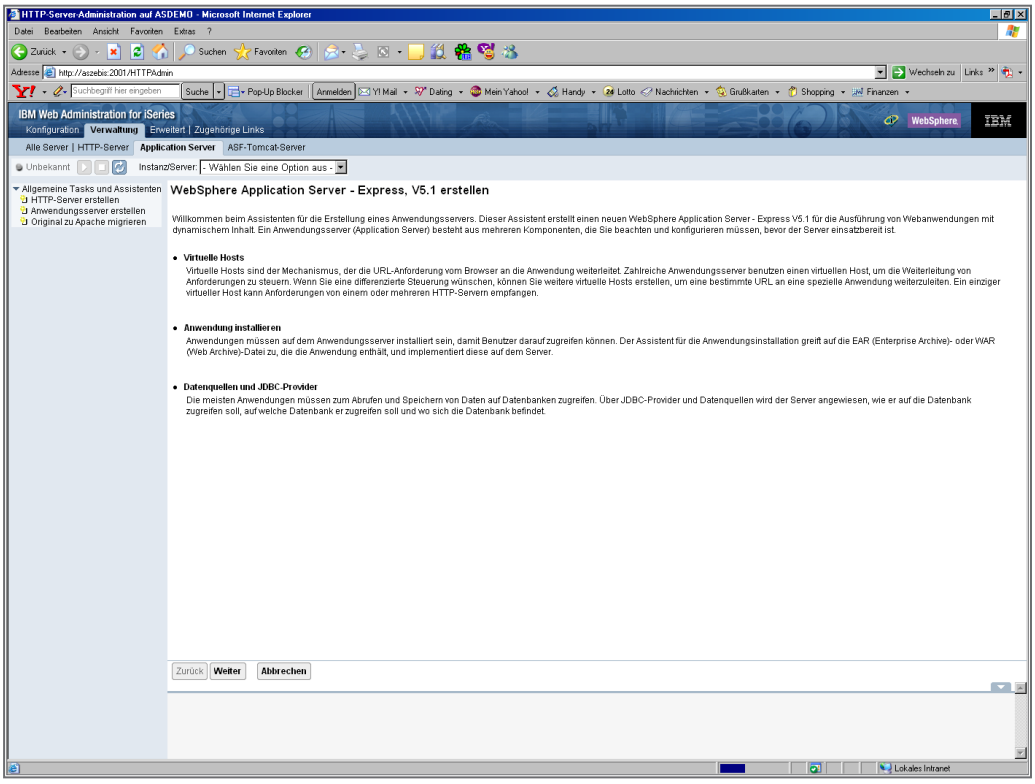


Anwendungsserver erstellen

„Anwendungsserver“ bedeutet in diesem Bereich das Erstellen eines WAS Express Servers.

Die nächste Anzeige gibt uns nun einen Überblick, welche Schritte während der folgenden Konfiguration ausgeführt werden müssen.

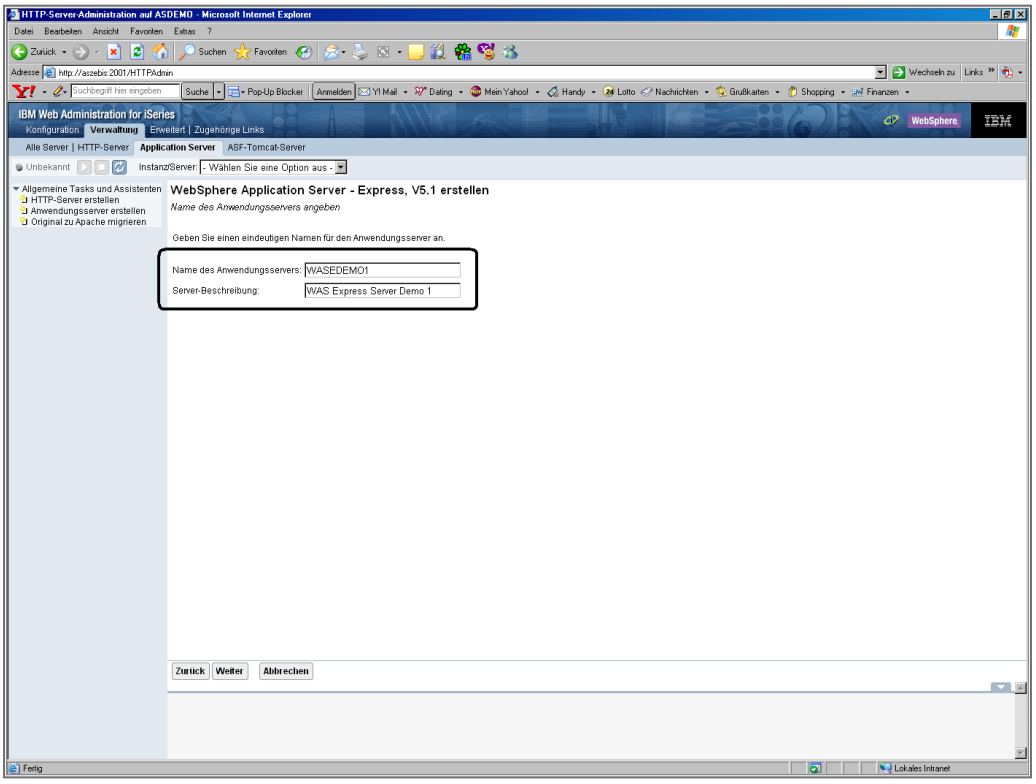
5.2.1.2
Seite 4



Konfigurationsübersicht

Auf die Erklärung der einzelnen Unterabschnitte gehen wir in diesem Kapitel später näher ein und klicken daher auf die Schaltfläche „Weiter“.

Der WAS Express benötigt einen Namen, um mit anderen Servern und Anwendungen zu kommunizieren. Zusätzlich sollte eine sprechende Server-Beschreibung verwendet werden. Es empfiehlt sich keine zu abenteuerliche Namensvergabe, denn in der Regel werden auf einem System mehrere Applikations-Server parallel betrieben. Sprechende Namen erleichtern die Arbeit ungemein!

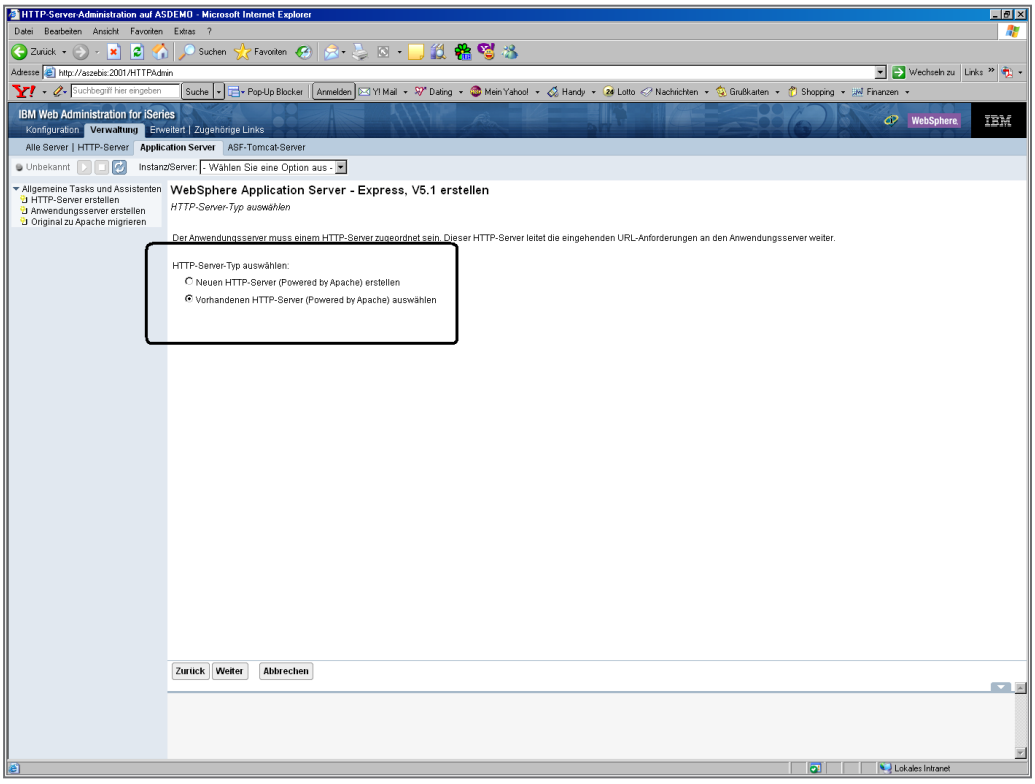


Server-Name

In unserem Beispiel verwenden wir den Namen WASEDEMO1. Nach der Eingabe des Namens und der Beschreibung klicken wir auf die Schaltfläche „Weiter“.

Ein Applikations-Server macht nur in Verbindung mit einem HTTP-Server wirklich Sinn. Denn die Darstellung der Informationen wird über den HTTP-Server realisiert. Diese enge Verbundenheit spiegelt sich in der nächsten Anzeige wieder. Hier können wir auswählen, ob wir für diesen WAS Express Server einen bereits bestehenden HTTP-Server verwenden oder einen neuen HTTP-Server erstellen wollen.

5.2.1.2
Seite 6

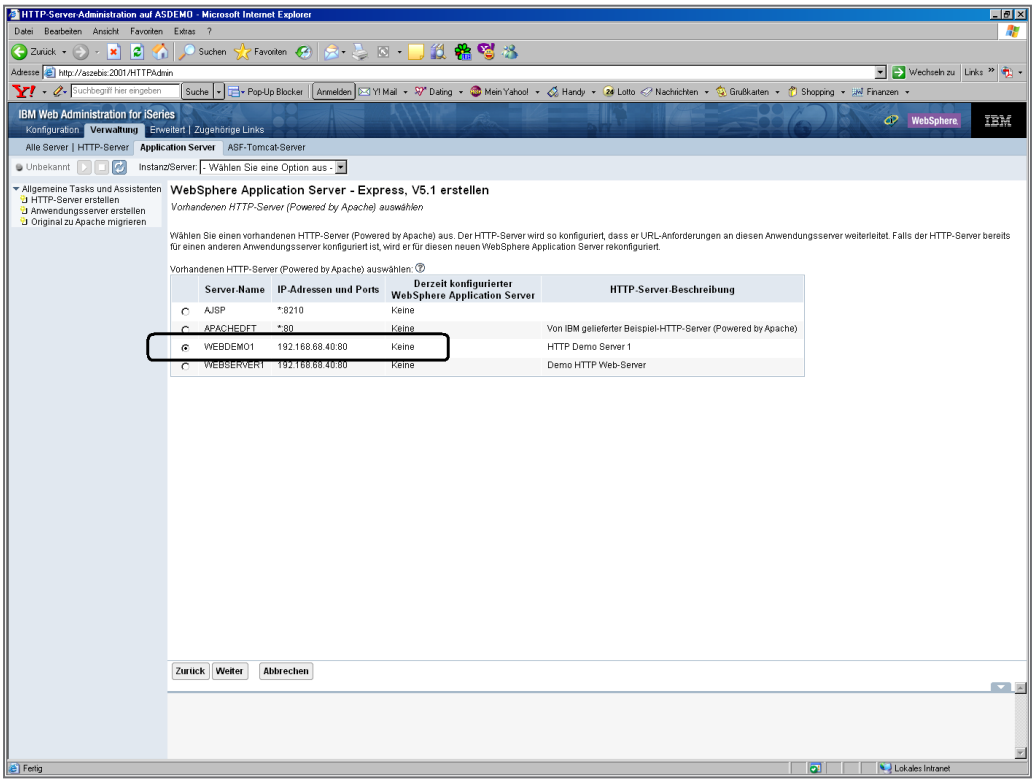


HTTP-Server auswählen

Da wir in dem vorhergehenden Kapitel „Konfiguration HTTP-Server“ bereits einen HTTP-Server auf der i5 konfiguriert haben, verwenden wir diesen auch und ändern die Voreinstellung in der Anzeige entsprechend auf den Eintrag „Vorhandenen HTTP-Server (Powered by Apache) auswählen“.

Sollten sie noch keinen HTTP-Server erstellt haben, dann wählen sie „Neuen HTTP-Server (Powered by Apache) erstellen“ aus. Der Administrationsprozess führt sie dann durch die Konfigurationsschritte des HTTP-Servers bevor WAS Express konfiguriert wird.

Wir nehmen unseren bereits konfigurierten HTTP-Server WEBDEMO1 und wählen diesen in der Übersicht der installierten HTTP-Server aus.

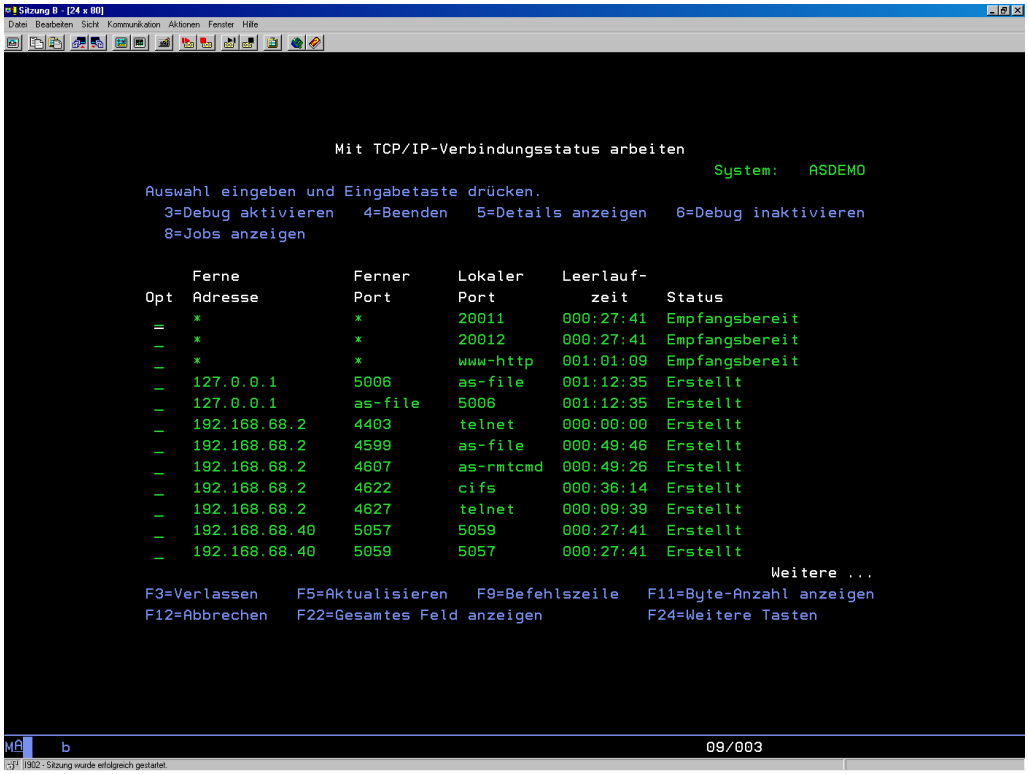


HTTP-Server selektieren

Ein Klick auf „Weiter“ führt wahrscheinlich zu einem „Fehler“, denn für die interne Kommunikation zwischen dem HTTP-Server und dem WAS Express werden 13 Ports verwendet. Diese können zwar in einem beliebigen Bereich zwischen Port 1 und 65523 liegen, müssen aber fortlaufend frei sein (also 13 freie aufeinander folgende Ports).

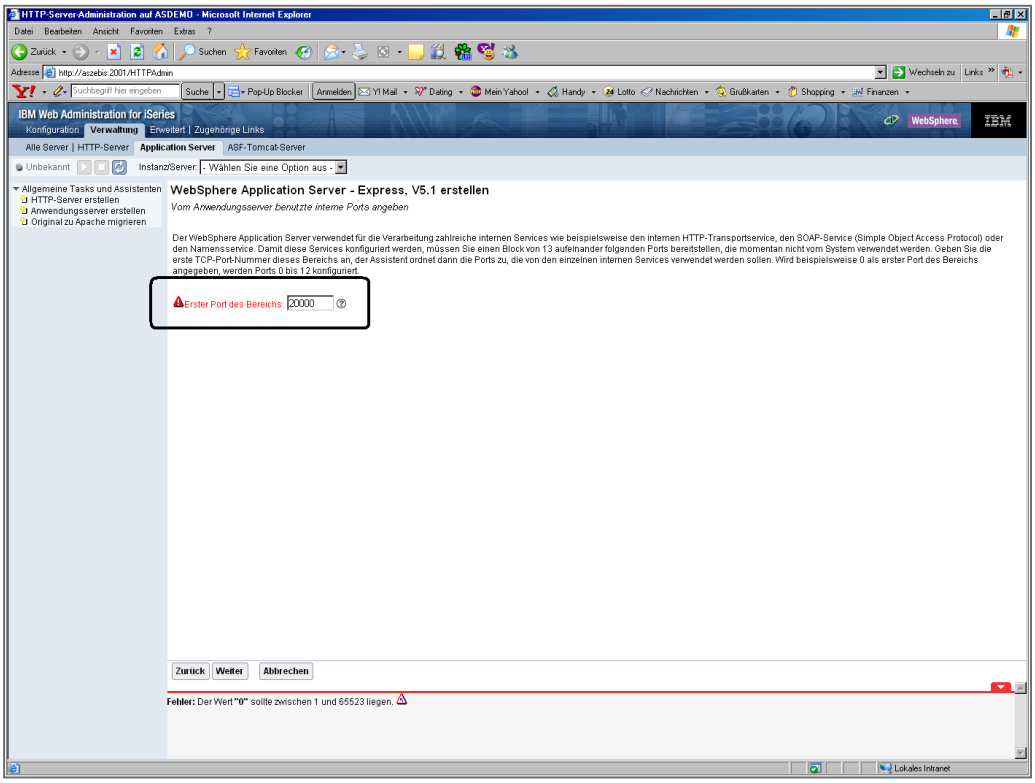
Wissen sie „zufällig“ nicht, welche Ports frei sind, so lässt sich dies leicht mit dem Befehl NETSTAT auf der i5 feststellen. Dazu geben wir auf der i5-Befehlszeile den Befehl NETSTAT ein. In dem angezeigten Menü, das nach dem Drücken der Datenfreigabetaste erscheint, wählen wir Punkt 3 (Mit TCP/IP Verbindungsstatus arbeiten) aus. Jetzt gelangen wir in eine Übersicht aller Verbindungsdefinitionen; in dieser Übersicht befindet sich unter anderem auch die Auflistung der verwendeten TCP/IP-Ports. Nun suchen wir in der erscheinenden Anzeige einen freien Bereich aus, in dem 13 aufeinander folgende Ports vorhanden sind.

5.2.1.2
Seite 8



NETSTAT-Anzeige

In unserem Beispiel verwenden wir einen Port-Bereich beginnend mit „20000“ und tragen diesen Wert in der Anzeige des Browsers ein.



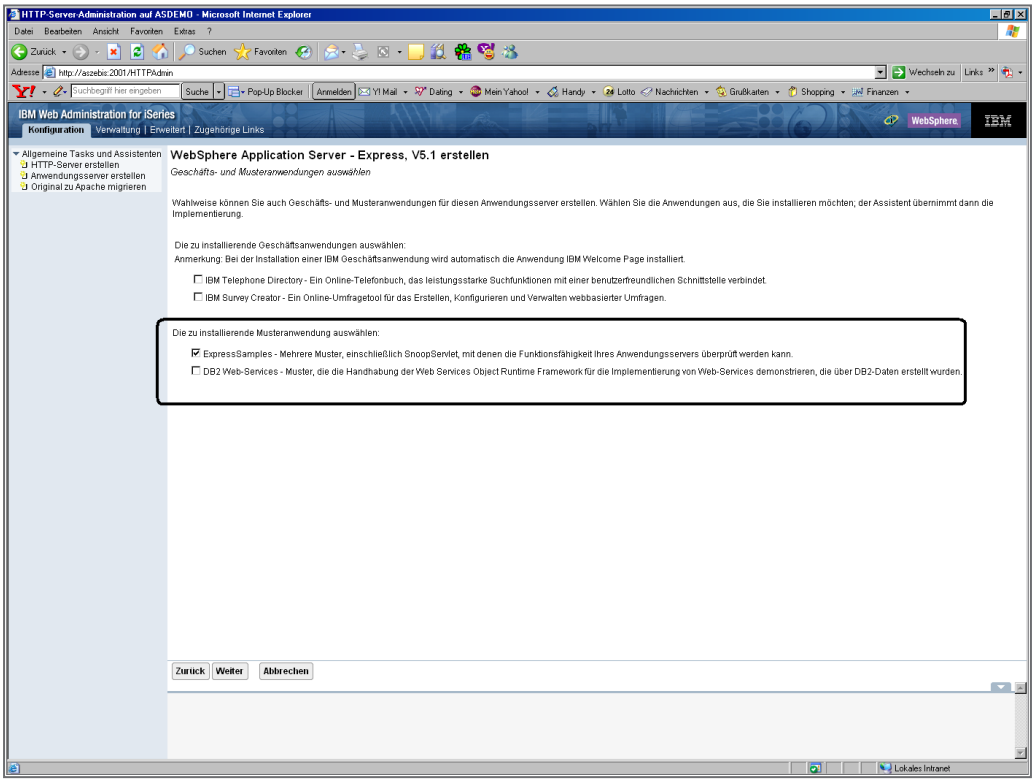
Port-Bereich

Ein Klick auf „Weiter“ lässt uns nun entscheiden, ob wir von IBM ausgelieferte Beispielanwendungen auf diesem WAS Express Server installieren wollen.

Diese Beispielanwendungen sind nicht in der Basisauslieferung des WAS Express enthalten, sie müssen gesondert angefordert werden. Das „IBM Telephone Directory“ ist eine Beispielanwendung, die das Erfassen von Telefonbucheinträgen erlaubt. Diese Anwendung ist auf einer eigenen CD enthalten und befindet sich nicht mit auf der CD des WAS Express.

Da diese häufig als Beispielanwendung verwendet wird, verzichten wir an dieser Stelle auf eine Wiederholung.

Neu bei der Version 5.1des WAS Express ist der so genannte „IBM Survey Creator“ – ebenfalls eine Beispielanwendung, die durch die Installation des PTF SI12985 (für das Lizenzprogramm 5722-BZ1) auf dem System verfügbar gemacht wird. Sollten Sie dieses Beispielprogramm nutzen wollen, dann muss dieses PTF vor der Ausführung der Konfigurationsschritte angelegt werden. Sie können diese Beispielanwendung aber auch jederzeit später nachinstallieren. Wir wählen für unser Beispiel lediglich den Eintrag „Express Samples“ aus.



Beispielanwendungen

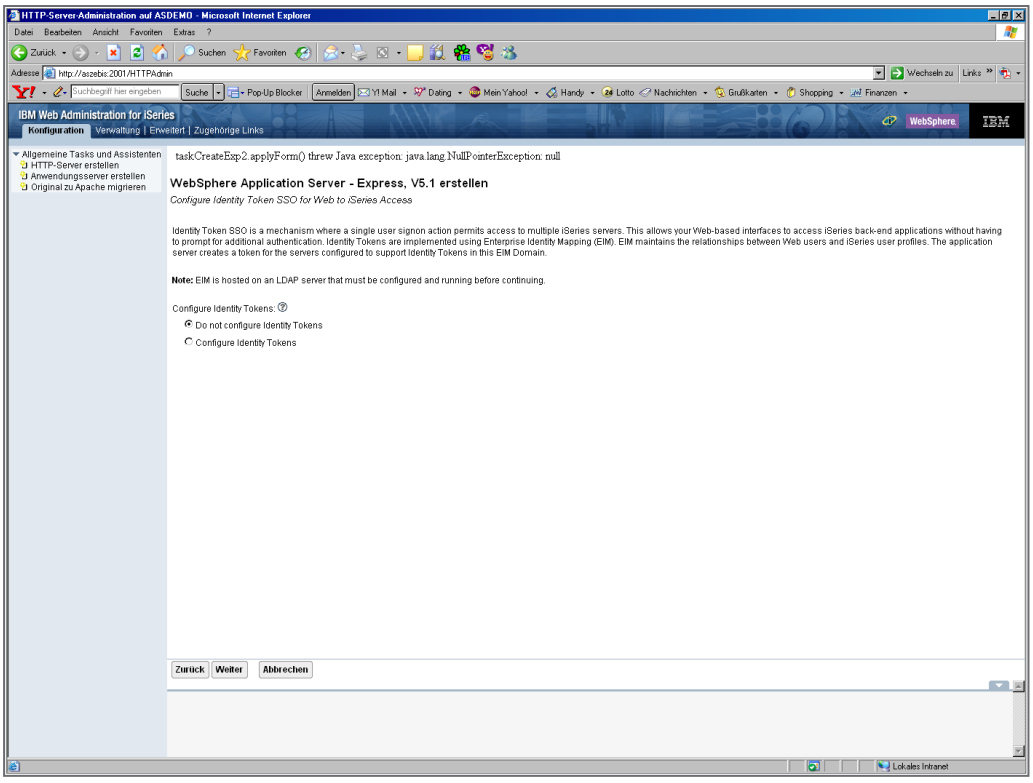
5.2.1.2
Seite 10

Nach der getroffenen Auswahl klicken wir auf die Schaltfläche „Weiter“.

In den letzten Monaten hat das Interesse an SSO (Single Sign-On) stark zugenommen. Um diesen Anforderungen Rechnung zu tragen, hat IBM den Einsatz von SSO auch für den WAS Express realisiert. Wollen sie diese Funktion nutzen, dann wählen Sie den entsprechenden Eintrag aus. In unserem Beispiel arbeiten wir „klassisch“ und verwenden kein SSO.

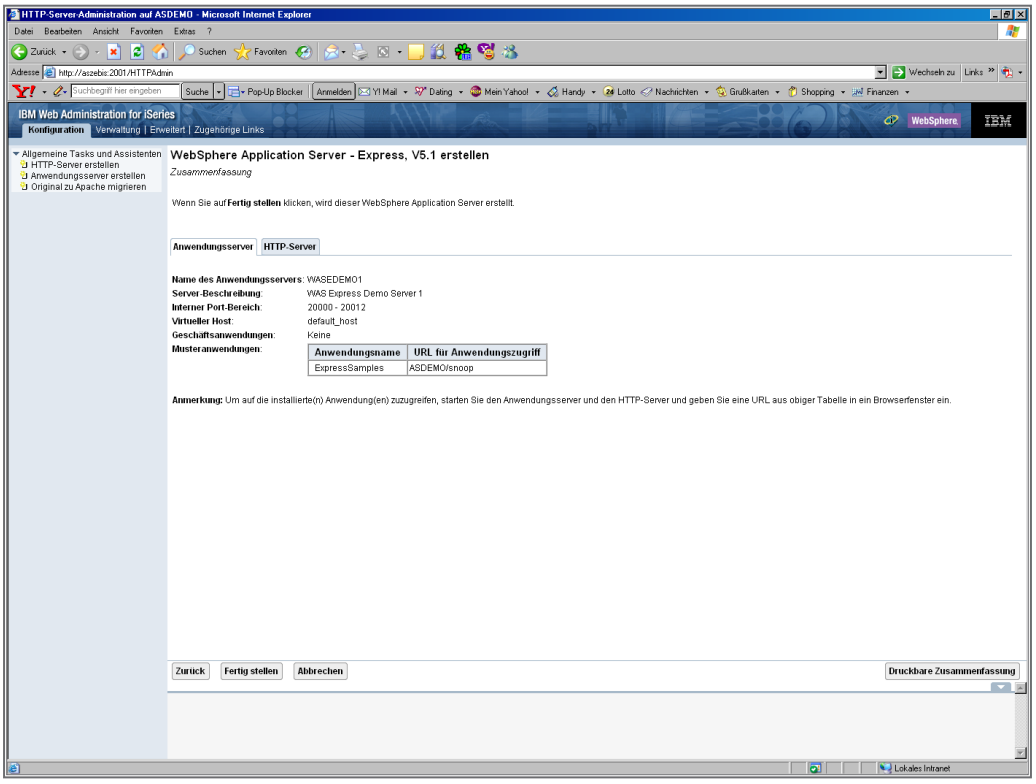
Anmerkung:

Zum Zeitpunkt des Schreibens dieses Kapitels war IBM wohl noch nicht so weit mit den Übersetzungen, wundern Sie sich also nicht, dass die nächste Anzeige auf Englisch erscheint, obwohl alle vorhergehenden Seiten auf Deutsch angezeigt wurden.



SSO

Mit einem erneuten Klick auf „Weiter“ erscheint die Zusammenfassung der bisherigen Konfigurations-Auswahlen. Prüfen Sie diese nochmals auf ihre Richtigkeit, denn mit dem nächsten Mausklick auf „Fertig stellen“ wird der WAS Express-Server konfiguriert.



Fertig Stellen

Wir können neben den Angaben für den WAS Express-Server auch die Angaben zu dem HTTP-Server nochmals prüfen, indem wir den entsprechenden Eintrag auswählen.

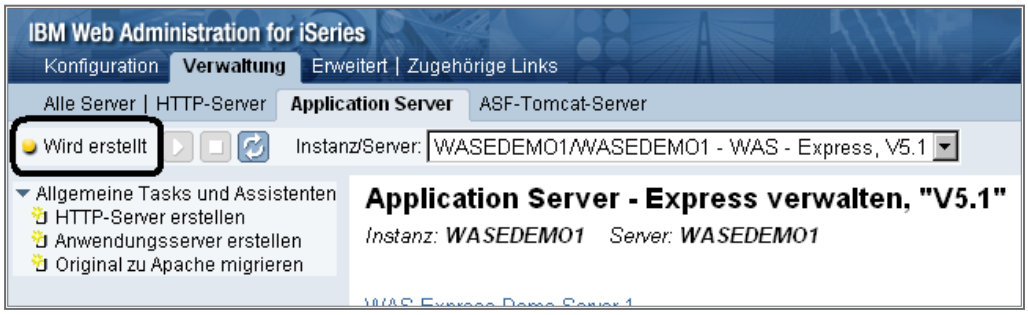
HTTP-Server-Einstellungen

Ist alles in Ordnung? Dann einen Klick auf „Fertig Stellen“ – und der Server wird angelegt.

Beachten Sie in der Folgeanzeige den Bereich, in dem der Status dargestellt wird.

Auch bei schnellen Maschinen erkennen wir hier die nachfolgende Statusanzeige:

5.2.1.2
Seite 12



Wird erstellt

Mit ein wenig Geduld sollte sich diese dann bald ändern und den Status als „Gestoppt“ anzeigen.

Gestoppt

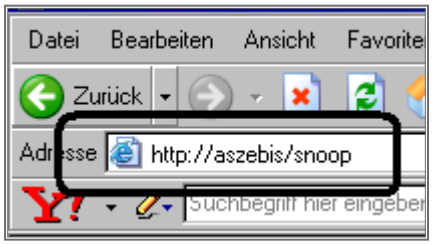
Um mit dem Server arbeiten zu können, muss dieser zunächst gestartet werden. Dieser Prozess ist in dem nachfolgenden Abschnitt näher beschrieben. Wir klicken an dieser Stelle lediglich auf das grüne Symbol rechts neben der Statusanzeige und warten, bis diese (nach einer gewissen Zeit) den Status „Gestartet“ annimmt.

Nun ist der Applikations-Server bereit für den Einsatz. Doch um sicher zu gehen, dass alle Konfigurationseinstellungen auch wirklich korrekt waren, machen wir einen kleinen Test.

Vielleicht ist ihnen im Zuge der Konfiguration die Musteranwendung „Snoop“ aufgefallen? Diese wird automatisch während der Konfiguration für diesen Server mitinstalliert. „Snoop“ liefert eine Anzeige aller für den Server getätigten Einstellungen; sie ist also eine Anzeige der Konfiguration des WAS Express Servers. Um diese „Anwendung“ nun auszuführen und den Server damit testen zu können, öffnen wir ein weiteres Browser-Fenster. Dort geben wir folgende Adresse ein:

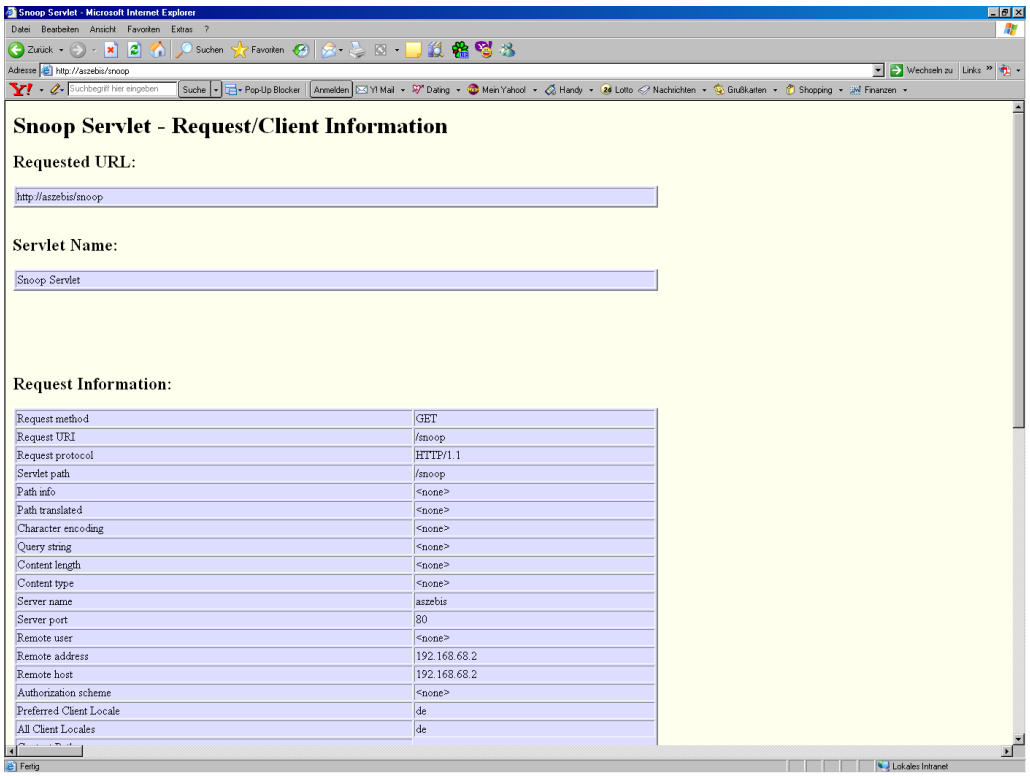
„HTTP//servername/snoop“

In unserem Beispiel sieht das konkret wie folgt aus:



Snoop

Wenn alle Einstellungen korrekt waren, dann liefert „Snoop“ ein Ergebnis – ähnlich dem in der nächsten Abbildung gezeigtem.



Snoop-Ergebnis

Wenn dies funktioniert, dann wurde WAS Express erfolgreich konfiguriert!

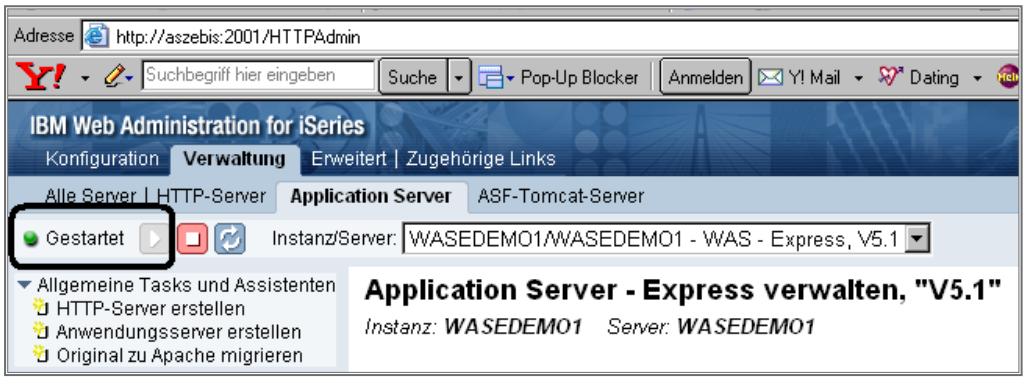
5.2.1.2
Seite 14

WebSphere Express-Instanz starten

Nach der Installation und Konfiguration müssen wir prüfen, ob diese erfolgreich war. Dazu starten wir den Applikations-Server zunächst mit einem Klick auf das grüne Symbol rechts neben der Statusanzeige; auch dieser Schritt kann – je nach Maschine – einen „Augenblick“ Zeit in Anspruch nehmen.

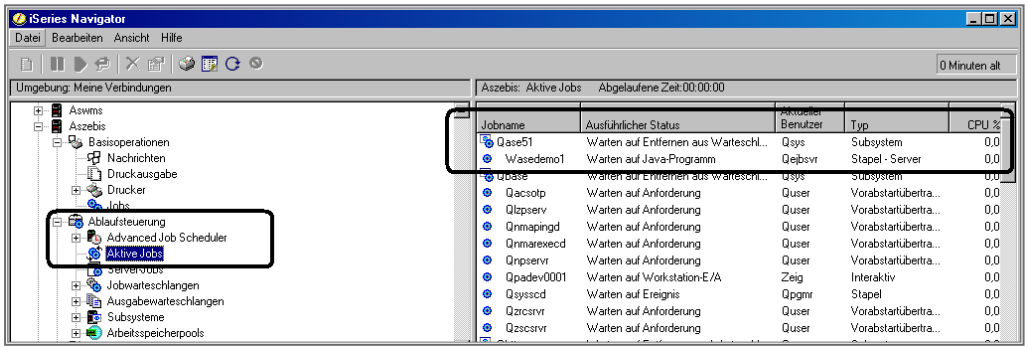
Das Starten des jeweiligen Applikations-Servers ist natürlich unabhängig von dem Installations- und Konfigurationsprozess; es ist Bestandteil der administrativen Arbeit mit WAS Servern.

Nachdem wir auf „Starten“ geklickt haben, ändert sich die Statusanzeige zunächst in „Wird gestartet“, um dann automatisch (ohne Aktualisierung der Browser-Anzeige) in den Status „Gestartet“ zu wechseln.



Server-Instanz gestartet

Das Starten einer WAS Express Server-Instanz führt auf der i5 dazu, dass ein Subsystem (QASE51) gestartet wird. In diesem Subsystem werden dann die verschiedenen Server-Instanzen ausgeführt. Schauen wir uns dies einmal an:



iSeries Navigator und aktive Jobs

Wie sie sehen, befindet sich unterhalb des Eintrages des Subsystems QASE51 die gestartete Server-Instanz WASDEMO1; das ist der zuvor konfigurierte und gestartete WAS Express Server.

5.2.1.2.1
Seite 2

Natürlich können wir uns die aktiven Jobs auch mit WRKACTJOB auf der i5 ansehen:

```
                          Mit aktiven Jobs arbeiten                  ASDEMO
                                                                14.11.04  07:56:47
CPU %:      0,0     Abgelaufene Zeit:   00:00:00     Aktive Jobs:   169

Auswahl eingeben und Eingabetaste drücken.
  2=Ändern   3=Anhalten   4=Beenden   5=Arbeiten mit     6=Freigeben
  7=Nachricht anzeigen    8=Mit Spool-Dateien arbeiten   13=Unterbrechen

Ausw.  Subsystem/Job  Benutzer    Art  CPU %  Funktion         Status
 __    QASE51         QSYS        SBS    0,0                   DEQW
 __      WASEDEMO1    QEJBSVR     BCH    0,0  PGM-QASESTRSVR   JVAW
 __    QBASE          QSYS        SBS    0,0                   DEQW
 __      QPADEV0001   ZEIG        INT    0,0  CMD-WA           RUN
 __      QSYSSCD      QPGMR       BCH    0,0  PGM-QEZSCNEP     EVTW
 __    QHTTPSVR       QSYS        SBS    0,0                   DEQW
 __      ADMIN        QTMHHTTP    BCH    0,0  PGM-QZHBMAIN     SIGW
 __      ADMIN        QTMHHTTP    BCI    0,0  PGM-QZSRLOG      SIGW
 __      ADMIN        QTMHHTTP    BCI    0,0  PGM-QZSRHTTP     SIGW
                                                                Weitere ...
Parameter oder Befehl
===>
F3=Verlassen    F5=Aktualisieren      F10=Neubeginn   F11=Aufgelaufene Daten
F12=Abbrechen   F23=Weitere Angaben   F24=Weitere Tasten
```

WRKACTJOB

Wir haben gesehen, wie wir einen Applikations-Server oder dessen Instanz starten können – oder besser gesagt starten müssen, denn nach jedem Beenden der i5 oder aller Subsysteme werden auch alle Applikations-Server beendet.

Damit diese automatisch nach einem i5-Neustart oder dem Starten der Subsysteme mit gestartet werden, können unterschiedliche Verfahren eingesetzt werden:

- Starten mit QSHELL
- Starten mit i5-Befehlen

Das Starten eines WAS Express lässt sich unter Verwendung von QSHELL ausführen.

In der QShell-Befehlszeile geben wir für den Start unseres WAS Express Servers den folgenden Befehl gefolgt von der Eingabetaste ein:

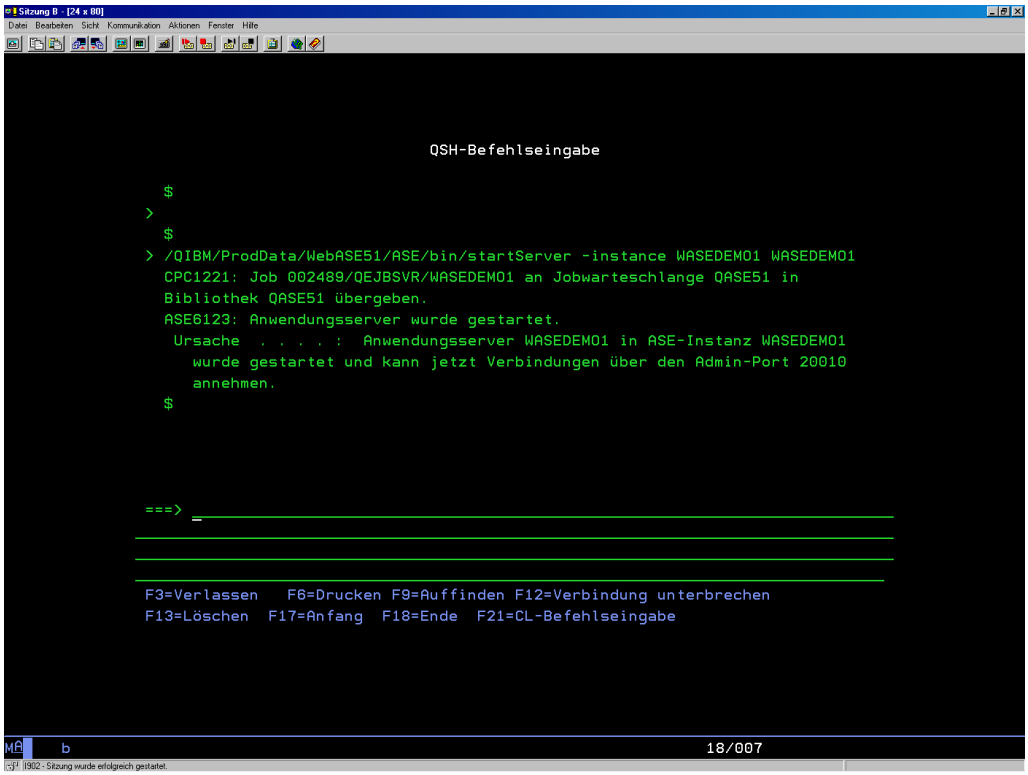
/QIBM/ProdData/WebASE51/ASE/bin/startServer -instance WASEDEMO1 WASEDEMO1

Bei der Angabe dieses Pfades handelt es sich um den Speicherort der Installations- und Konfigurationsangaben innerhalb des IFS der i5. In der Regel befindet sich der WAS Express 5.1 in dem oben gezeigten Verzeichnis: /QIBM/ProdData/WebASE51/bin.

In dem Verzeichnis „bin“ befindet sich eine Reihe von verschiedenen Einträgen (Befehle oder Programme), die für die Ausführung des Servers von Bedeutung sind. Unter anderem finden wir darin auch den Eintrag „startServer“; das ist genau den Befehl, der für das Starten des WAS Express benötigt wird.

Dieser Befehl „startServer“ muss mit den Angaben „-instance“ gefolgt von den Namen der Instanz und des Applikations-Servers eingegeben werden.

Mit dieser Kombination werden die Instanz „WASDEMO1“ und der Server „WASDEMO1“ gestartet.



QSHELL-Eingabe

Das Ergebnis ist genau dasselbe, als ob wir in dem Browser-Fenster den Server starten.

Dieses „startServer“-Skript verfügt noch über einige Parameter:

```
startServer server -instance instance
 [ -nowait ] [ -timeout seconds ] [ -trace ] [ -help | -? ]
```

5.2.1.2.1

Seite 4

Starten WAS Express mit CL-Befehlen

Wir können selbstverständlich den WAS Express auch mittels CL-Befehlen starten. So lässt sich zum Beispiel der Start des WAS Express in das Startprogramm der i5 (QSTRUP) integrieren.

Die für den Start erforderlichen Befehle sind:

- Starten von TCPIP
  STRTCP
- Optional zum Starten des Administrations-Bereiches
  STRTCPSVR SERVER(*HTTP) HTTPSVR(*ADMIN)
- Starten des Subsystems für den WAS Express Server
  STRSBS QASE51/QASE51
- Starten des WAS Express-Servers

```
SBMJOB CMD(CALL PGM(QASE51/QASESTRSVR) PARM('-instance'
    '/QIBM/UserData/WebASE51/ASE/instance' '-server' 'server'))
    JOB(server) JOBD(QASE51/QASE51) JOBQ(QASE51/QASE51)
USER(QEJBSVR)
    LANGID(*USRPRF) CNTRYID(*USRPRF) CCSID(*USRPRF)
```

Diese Befehle können nacheinander auf der Befehlszeile der i5 eingegeben werden und führen zum Starten des WAS Express Servers auf der i5.

Für das Starten des WAS Express Servers muss das ausführende Benutzerprofil mindestens über die Berechtigung *USE für das Benutzerprofil QEJBSVR verfügen.

5.2.1.2.1.2

Seite 2

Starten WAS Express automatisch mit Subsystem

Wie heißt es doch so schön: „Wer die Arbeit kennt und sich nicht drückt, der ist verrückt."

Diesem Motto folgend sollte jede zusätzliche Arbeit vermieden werden – so auch das manuelle Starten eines WAS Express Servers. Deshalb empfiehlt sich in der Praxis die Verwendung des nachstehend erläuterten Verfahrens, bei dem der WAS Express Server automatisch beim Starten des QASE51-Subsystems mitgestartet wird.

Ausführungsschritte

① ▷ ② ▷ ③ ▷ ▷
Ausführungsschritte

1. Zunächst erstellen wir eine Kopie der Standard-Jobbeschreibung für den QASE51.

```
CRTDUPOBJ OBJ(QASE51) FROMLIB(QASE51) OBJTYPE(*JOBD) TOLIB(WASLIB)
NEWOBJ(WASDEMO1)
```

   Damit stellen wir sicher, dass keine Veränderungen an der Standard-Einstellung von IBM durchgeführt werden.

2. Nun passen wir die Jobbeschreibung so an, dass der WAS Express Server automatisch mitgestartet wird, wenn das Subsystem QASE51 startet. Dazu verwenden wir den Befehl CHGJOBD und geben in dem Feld RQSDTA den Befehl für den Start des Servers an:

```
'QSYS/CALL PGM(QASE51/QASESTRSVR) PARM(''-instance''
''/QIBM/UserData/WebASE51/ASE/myinst'' ''-server'' ''myserv'')'
```

   Für unser Beispiel würde das konkret wie folgt aussehen:

```
'QSYS/CALL PGM(QASE51/QASESTRSVR) PARM(''-instance''
''/QIBM/UserData/WebASE51/ASE/WEBDEMO1'' ''-server'' 'WEBDEMO1'')'
```

3. Anschließend fügen wir dem Subsystem QASE51 einen Autostart-Eintrag hinzu. Dieser hat den allgemeinen Aufbau:

```
ADDAJE SBSD(QASE5/QASE51) JOB(myserv) JOBD(mywasjobd/myserv)
```

   Oder speziell für unser Beispiel angepasst:

```
ADDAJE SBSD(QASE5/QASE51) JOB(WASDEMO1) JOBD(WASLIB/WASDEMO1)
```

4. Nun können wir in dem Startprogramm (QSTRUP) den Start des Subsystems QASE51 einbauen.

```
STRSBS QASE51/QASE51
```

   Achten sie darauf, dass vor dem Start des Subsystems QASE51 TCPIP gestartet sein muss!

5.2.1.2.1.3

Seite 2

WebSphere Express-Anwendung installieren

Die Aufgabe eines Applikations-Servers (oder Anwendungs-Servers) liegt darin, Anwendungen unterschiedlicher Arten auszuführen. Dabei können auf einem Applikations-Server gleichzeitig mehrere Anwendungen ausgeführt werden. Es kommt in der Regel immer auf den Individualfall an, welche und wie viele Anwendungen parallel auf einem einzelnen Applikations-Server – oder besser einer Applikations-Server-Instanz aufgeführt werden – denn nicht selten wird die Administration deutlich komplizierter und auch das Laufzeitverhalten eines Server kann negativ beeinflusst werden, je mehr Anwendungen auf einem Server ausgeführt werden. Zudem sollten bei einem eventuell erforderlichen Server-Neustart möglichst wenig Anwendungen beeinflusst werden.

Bevor eine Anwendung verwendet werden kann, muss diese zunächst auf dem Applikations-Server installiert werden (manche sprechen hier auch von „deployen“ der Anwendung).

Es versteht sich von selbst, dass die zu installierende Anwendung für den Server zugänglich gespeichert, vorhanden sein muss. Dabei müssen die Anwendungen innerhalb des integrierten File Systems (IFS) der i5 gespeichert sein.

Die zu installierende Web-Anwendung kann in zwei unterschiedlichen „Formaten“ vorhanden sein:

- EAR-Installationsdatei
  - „Enterprise Archive Ressource“ oder zu Deutsch: Enterprise Archiv-Datei
- WAR-Installationsdatei
  - Web-Anwendungsarchivdatei

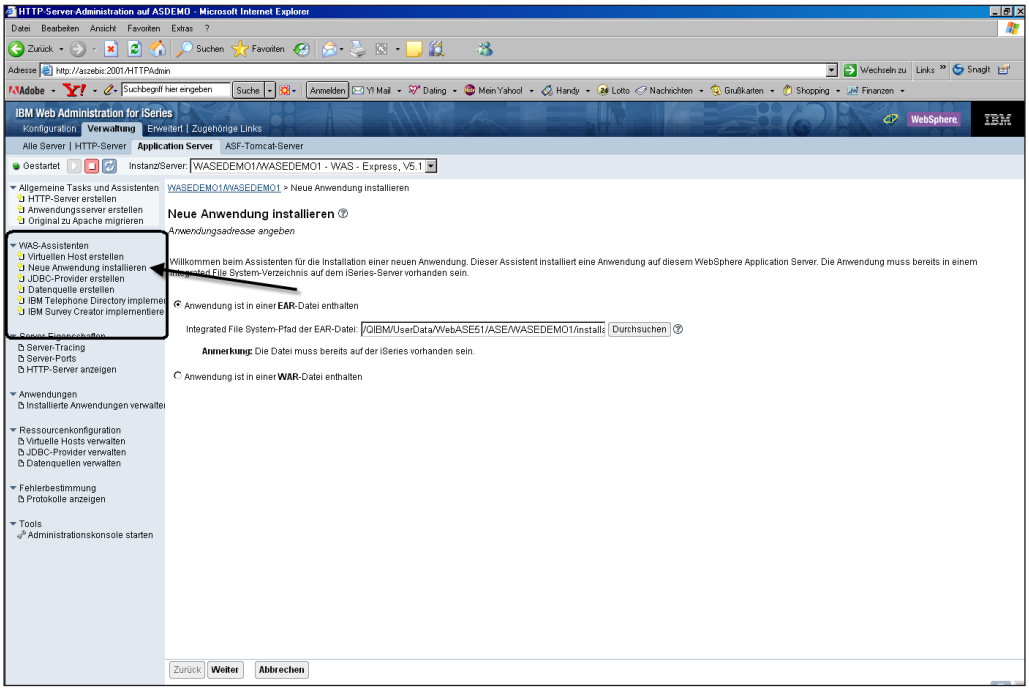
Es kommt auf die Anwendung beziehungsweise die Umsetzung an, in welchem dieser beiden Dateiformate die Anwendung gespeichert wurde.

5.2.1.3
Seite 2

Die Installation einer Anwendung selbst wird, wie alle administrativen Tätigkeiten an dem Applikations-Server selbst, über den Administrations-Client durchgeführt. Details zu dem Start des Administrations-Clients enthält das Kapitel 5.2.

Bevor wir jedoch eine Anwendung installieren können, muss sichergestellt sein, dass der Applikations-Server gestartet ist – anderenfalls kann es – je nach PTF-Stand – zu einem Fehler in dem Browser-Fenster kommen.

Nachdem sichergestellt ist, dass der Applikations-Server, auf dem wir die Anwendung installieren wollen, auch gestartet ist, wählen wir im linken Navigationsbereich den Punkt „Neue Anwendung installieren“ aus. Dieser befindet sich unterhalb des Eintrages „WAS-Assistenten“.



Anwendung installieren

Jede Anwendung die installiert werden soll, muss sich in einem Pfad innerhalb des IFS befinden.

Dieser muss in dem entsprechenden Feld „Integrated File System EAR – Datei“ angegeben werden.

Als Standardeinstellung wird in diesem Feld der Installationspfad des Benutzerbereiches des Applikations-Servers vorgeschlagen. Dieser befindet sich innerhalb des Verzeichnisses „/QIBM/UserData/WebASE51/ASE/“ und darin in dem Unterverzeichnis für den jeweiligen Applikations-Server – in unserem Beispiel ist dies das Verzeichnis „WASEDEMO1“. Die installierbaren Anwendungen befinden sich innerhalb des Unterverzeichnisses „installableApps“.

Als Empfehlung gilt, dieses Verzeichnis als Installationspfad für neu zu installierende Anwendungen einzustellen und die Installationsdaten dort zu hinterlegen.

Installationspfad

Wie sich über die Option „Durchsuchen“ erahnen lässt, kann natürlich auch jedes andere beliebige Verzeichnis für die Installation neuer Anwendungen angegeben werden. Wir verwenden bewusst in diesem Abschnitt die Standardeinstellungen.

5.2.1.3

Seite 4

Telephone Directory installieren

Als Beispiel für die Installation einer Anwendung auf einem Applikations-Server dient an dieser Stelle eine von IBM mitgelieferte Anwendung – das Telephone Directory.

Dabei handelt es sich um eine vollwertige Version eines Telefon-Verzeichnisses mit hochwertigen Suchfunktionen – allerdings leider auf maximal 1000 Einträge beschränkt. Trotzdem ist es gerade als Einstieg in die Welt der „modernen" Anwendungen und als Grundlage für weitere eigene Anwendungen gut geeignet.

Das „Telephone Directory" ist für alle Versionen des WebSphere Applikations-Server Express verfügbar. Dabei ist zu beachten. Dass es für jede Version des WebSphere Applikations-Server Express (WASE) eine „passende" Version des „Telephone Directory" gibt. In unserem Beispiel gehen wir von dem WASE in der Version 5.1 aus. Das zu diesem Server passende „Telephone Directory" muss selektiv installiert werden. Dazu liefert IBM eine eigene Installations-CD „Telephone Directory Version 5.1" aus. Auf dieser CD ist ein Lizenzprogramm „5722-IWE" enthalten, das vor der Aktivierung auf dem Applikationsserver zunächst mit dem Befehl „RSTLICPGM" auf der i5 installiert werden muss.

Die Installation ist recht einfach gehalten und besteht aus zwei Bereichen „*BASE" und der Option „3". Diese beiden Optionen speichern wir mit den folgenden Befehlen auf der i5 zurück:

```
RSTLICPGM LICPGM(5722IWE) DEV(OPT01) OPTION(*BASE)
```

und

```
RSTLICPGM LICPGM(5722IWE) DEV(OPT01) OPTION(3)
```

Bevor die Anwendung „Telephone Directory" auf dem Applikations-Server installiert und ausgeführt werden kann, müssen die Restore-Befehle für die beiden Optionen ausgeführt worden sein.

(In dem Installationsbereich des WebSphere Applikations-Server steht dann zwar die Installation als Auswahlpunkt zur Verfügung – diese Installation schlägt dann aber fehl, wenn die erforderlichen Lizenzprogramme nicht vorhanden sind.)

Doch bevor wir uns die Installation einer Anwendung ansehen, bedarf es noch einiger vorausgehender Schritte:

5.2.1.3.1
Seite 2

LDAP

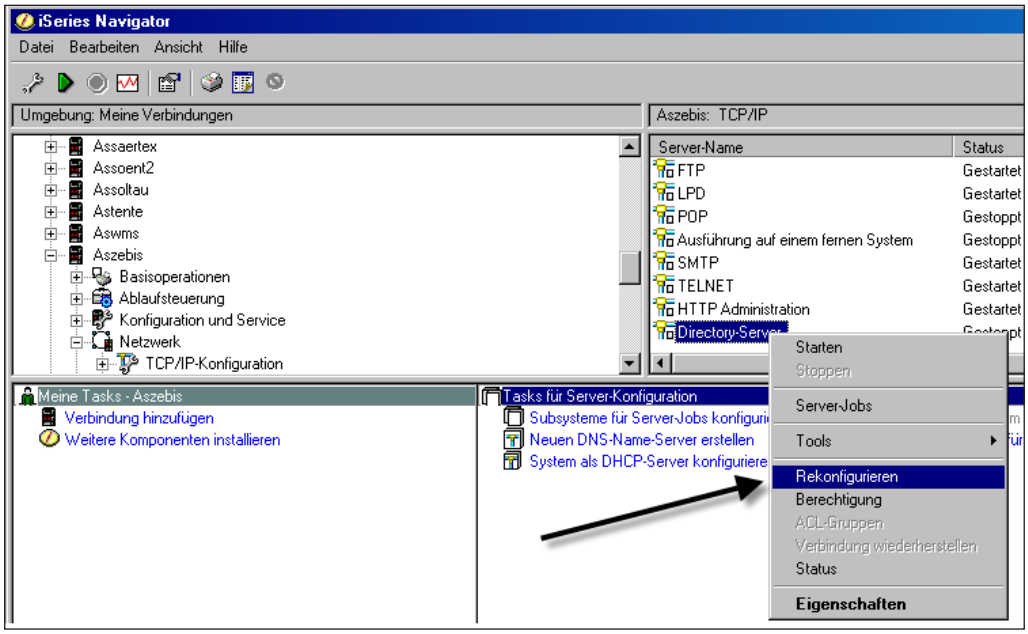
Eine der wesentlichen Voraussetzungen für Web-Anwendungen stellt in vielen Bereichen das sogenannte Directory dar. In der Fachsprache begegnet uns dieses oft als LDAP (Lightweight Directory Access Protocol) – die i5 bietet entsprechende Möglichkeiten, unter i5/OS die notwendigen LDAP- Funktionen abzubilden. Diese sind seit der Version V5R1 in die Basis des Betriebssystems integriert – müssen allerdings separat konfiguriert werden.

Da wir für unsere Beispielanwendungen auf die Funktionen des LDAP zurückgreifen, wollen wir an dieser Stelle die LDAP-Komponenten der i5 konfigurieren. Dabei bedienen wir uns – wie kann es mittlerweile schon anders sein – dem iSeries Navigator.

LDAP wird unterhalb des Eintrages „Netzwerk / Server / TCPIP“ in Form des Eintrages „Directory Server“ angezeigt. Ist dieser LDAP-Eintrag bereits komplett konfiguriert, dann können Sie diese folgenden Ausführungsschritte überspringen – anderenfalls führen Sie bitte diese Schritte aus.

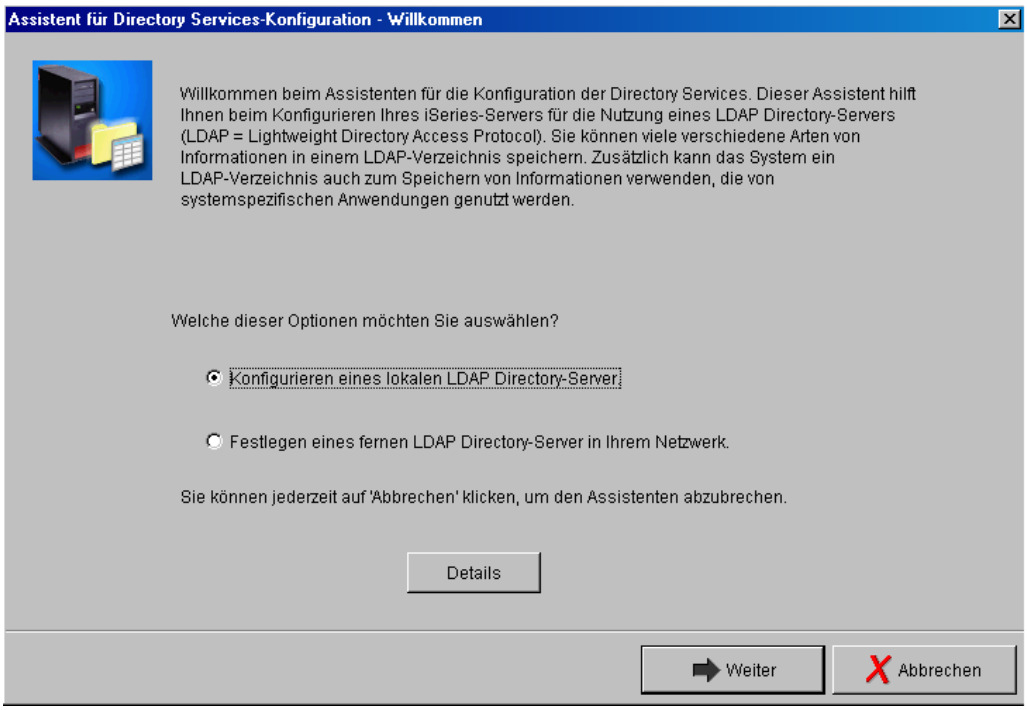
Ausführungsschritte:

- Öffnen des iSeries Navigators
- Auswahl des Eintrages
  - Netzwerk / Server / TCPIP / Directory-Server
- Für Anpassungen der Konfiguration sollte der Dienst kurzfristig gestoppt werden (Achtung! Dabei sollte natürlich keine andere Anwendung beeinträchtigt werden) – allerdings ist die Konfiguration nur einmalig erforderlich.
- Klick der rechten Maustaste auf den Eintrag „Directory Server“
- Auswahl „Rekonfigurieren“



LDAP konfigurieren

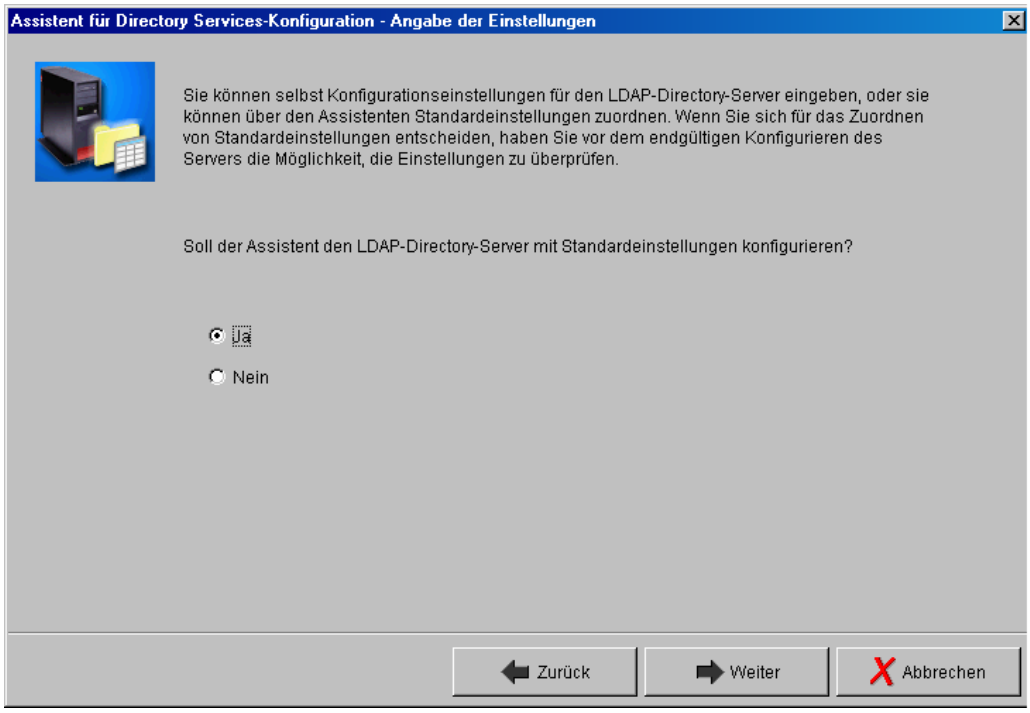
- Die Auswahl in der nächsten Anzeige kann übernommen werden.



Directory Informationen

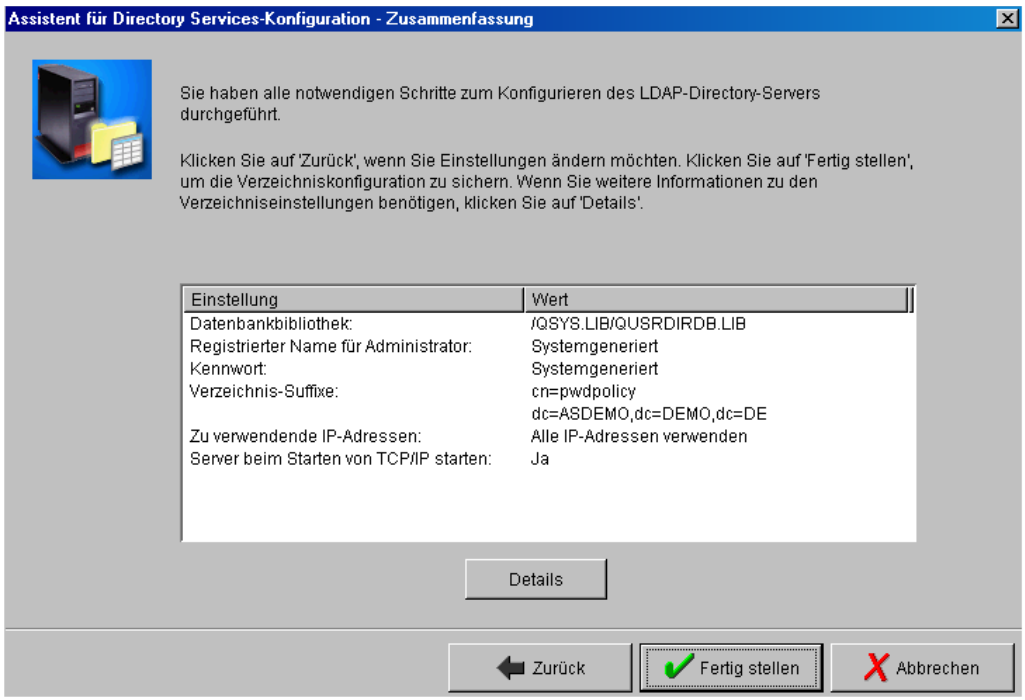
5.2.1.3.1
Seite 4

- Nach einem Klick auf die Schaltfläche „Weiter“ erscheint die nächste Anzeige. Auch hier können wir alle Standardeinstellungen akzeptieren.



LDAP-Standardeinstellungen

- Nach einem Klick auf die Schaltfläche „Weiter“ gelangen wir bereits in den Abschlussbildschirm der Installation. Hier können die Details angesehen werden.



LDAP-Zusammenfassung

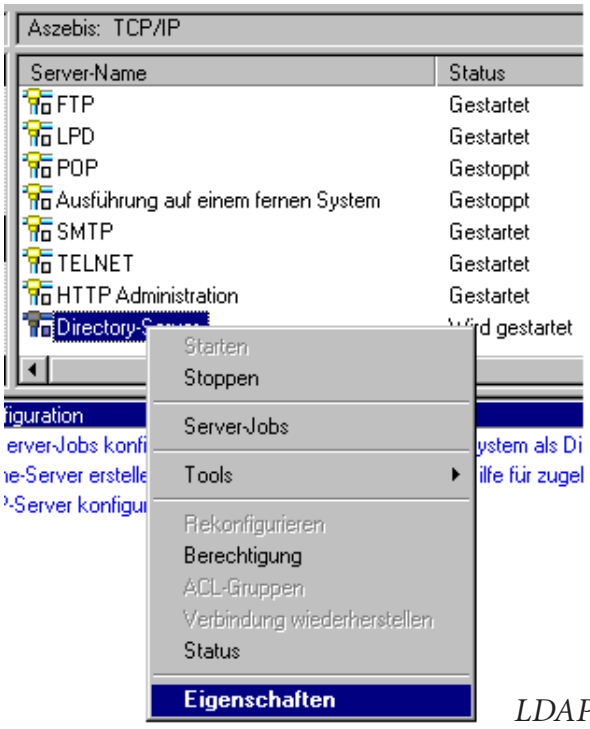
Mit diesen Ausführungsschritten wurde eine Grundkonfiguration LDAP auf der i5 erstellt.

5.2.1.3.1
Seite 6

Nur wäre es natürlich zu einfach, wenn wir alle Grundeinstellungen ohne Einschränkungen übernehmen könnten. Wir müssen nun noch einige kleinere Anpassungen vornehmen, damit die Anwendungen im Bereich der Applikations-Server auch befriedigend ausgeführt werden können.

Dafür starten wir zunächst den Directory-Server durch einen Klick mit der rechten Maustaste auf den Eintrag in dem iSeries Navigator und der anschließenden Auswahl „Starten“. Dies ist natürlich nur dann erforderlich, wenn der Dienst nicht bereits gestartet sein sollte.

Nun wählen wir den Eintrag erneut mit der rechten Maustaste aus und selektieren den Eintrag „Eigenschaften“.



LDAP-Eigenschaften

In diesem Bereich können wir alle Einstellungen der LDAP-Konfiguration anpassen – aber Vorsicht – ein Klick an der richtigen Stelle und nichts funktioniert mehr (zumindest nicht in dem LDAP Bereich).

Wir wählen nun den Tabellenabschnitt „Leistung“ aus – denn hier müssen wir eine Änderung vornehmen.

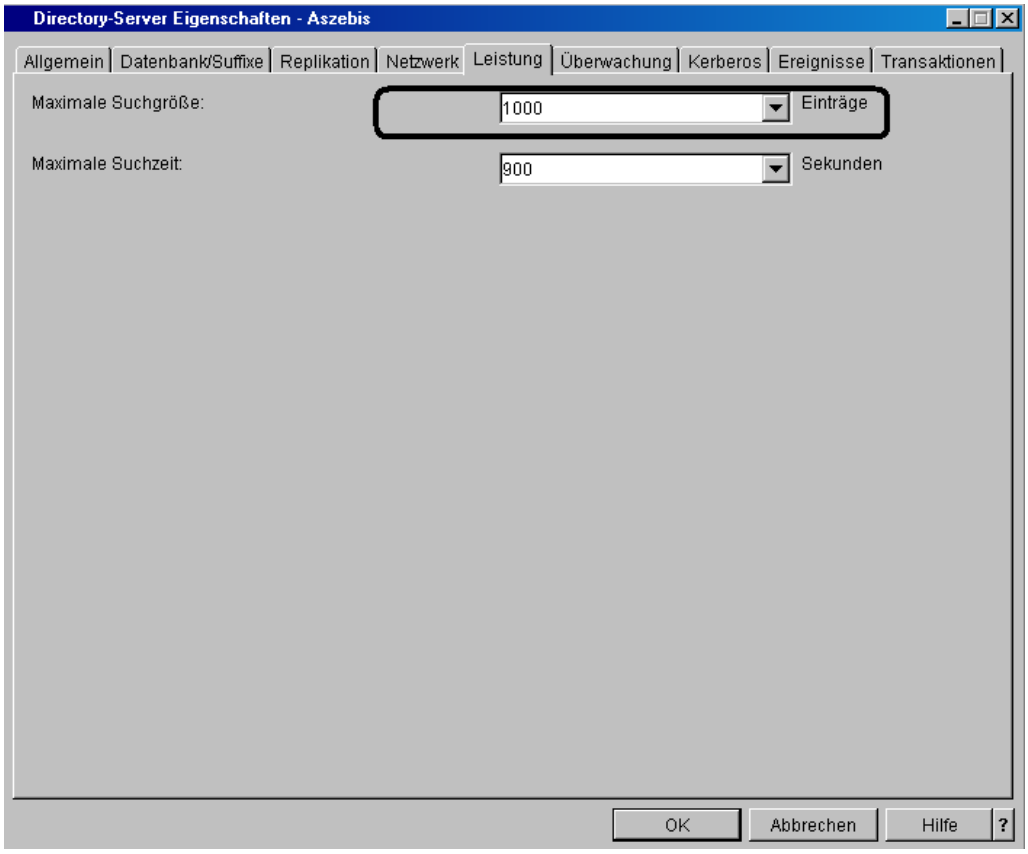
(es wäre schön, wenn IBM dort bereits die Einstellungen per Standard ausliefern würde, wie wir diese mit dem folgenden Schritt modifizieren). LDAP bietet die Möglichkeit, Suchanfragen recht elegant zu verarbeiten. Es lässt sich sicher leicht nachvollziehen, dass die Ausführungsgeschwindigkeit solcher Suchanfragen von der Anzahl der zu durchsuchenden Datensätze abhängt. In der Praxis werden häufig „größere“ Anzahlen von Datensätzen verwendet. Nun kann in den Leistungseinstellungen von LDAP festgelegt werden, wie

diese Suchanfragen in Bezug auf die Suchgröße beziehungsweise die Suchzeit behandelt werden soll. In der Literatur findet man zu diesem Thema unterschiedliche Meinungen, in der Praxis haben sich Anfangseinstellungen von 1000 durchaus bewährt. (Einige Quellen raten zu der Einstellung „nomax“ bzw. kein Maximum, was allerdings bei wirklich großen Suchanfragen durchaus zu Lasten der Performance gehen kann!)

Wir ändern an dieser Stelle den Wert für die maximale Suchgröße deshalb auf „1000“.

Anmerkung:

Diese Änderung ist deshalb im Zusammenhang mit dem „Telephone Directory“ sinnvoll, weil IBM in diesem Beispielprogramm bis zu 1000 Telefonbucheinträge zulässt – also macht es durchaus auch Sinn, diese Anzahl als Suchgröße für LDAP auszuwählen – allerdings auch nur, wenn diese Anzahl der Sätze annähernd erreicht wird.



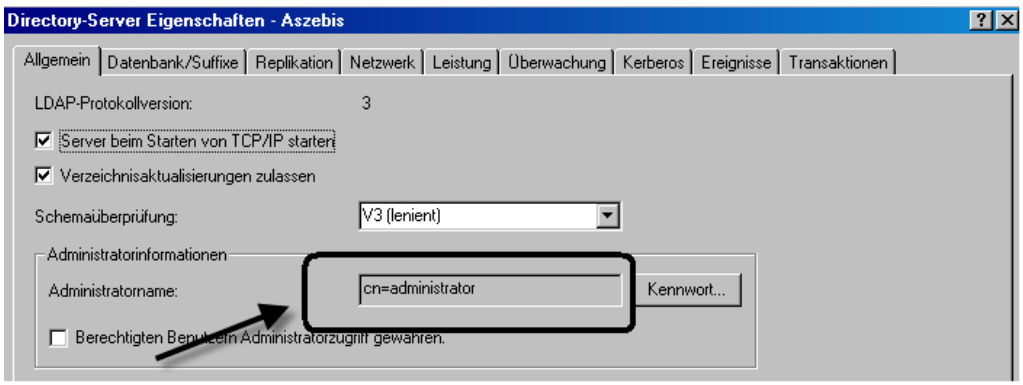
LDAP-Suchgröße

5.2.1.3.1
Seite 8

Tipp!

Wir wollen an dieser Stelle die Details der LDAP-Konfiguration nicht vertiefen – nur einen kleinen Tipp geben – denn allzu oft kommen Fragen nach dem Administrator und dem dazugehörigen Kennwort – beide Informationen werden auch später bei der Installation der Anwendung auf dem Applikations-Server benötigt. Diese beiden Informationen können wir leicht aus den Eigenschaften des Directory-Servers ermitteln – hier fehlen vielerorts die entsprechenden Hinweise in der IBM-Literatur:

Wir öffnen mit einem Klick der rechten Maustaste auf dem Eintrag „Directory Server“ das Auswahlfenster und selektieren den Eintrag „Eigenschaften“. Hier befindet sich in dem Tabellenabschnitt „Allgemein“ unter anderem die Information, welcher Administrator für den LDAP-Bereich angegeben wurde. Beachten Sie die Schreibweise, die mit einem führenden „cn=“ beginnt – denn der komplette Eintrag wird später als Angabe benötigt. In unserem Beispiel trägt der Administrator den Namen „cn=administrator“. Das erforderliche Kennwort kann durch einen Klick auf „Kennwort“ rechts neben dem Administratoreintrag angepasst werden.



LDAP-Administrator

Mit diesen Einstellungen haben wir nun die Voraussetzungen geschaffen, LDAP-Dienste auf der i5 zu verwenden. Diese sind Voraussetzung für die nun zu installierende Beispielanwendung „Telephone Directory“.

Nun starten wir die QSHELL mit Eingabe des Befehls „STRQSH“ auf der i5/OS-Befehlszeile und geben dort den folgenden Befehl ein, um das Installationsskript für das Telephone Directory zu starten. Hier benötigen wir wieder den LDAP-Administrator und dessen Kennwort. Wie wir diese Informationen erhalten, ist weiter oben beschrieben.

itdsetup -w (LDAP Kennwort) -a (Kennwort für IBM_Telephone_Directory) administrator_password -u

In unserem Beispiel geben wir dazu an:

itdsetup -w abcdefgh -a telefon –u

(“abcdefgh” ist das Kennwort für LDAP und “telefon” das neue Kennwort, das wir der Anwendung zuweisen).

Waren die LDAP-Konfiguration und die übrigen Einstellungen korrekt, dann sollte der Initialisierungsprozess mit einer Erfolgsmeldung enden:

```
  Creating IBM Telephone Directory administrator....
  adding new entry cn=iweAdministrator,cn=users,dc=asdemo.demo.de

  Creating IBM Telephone Directory administrator group....
  adding new entry cn=iweAdministrators,cn=users,dc=asdemo.demo.de

  Deleting IBM Telephone Directory create administrator....
  Modifying access rights to cn=users,dc=asdemo.demo.de....
  modifying entry cn=users,dc=asdemo.demo.de

  Creating ITD configuration file....
  Creating ITD LDAP file....
  IBM Telephone Directory configuration complete!
  $

===> itdsetup -w abcdefgh -a telefon -u
```

Telephone Directory Initialisierung

In diesem Zusammenhang schlägt IBM zwar noch einige weitere Einstellungen vor, diese sind für eine Grundinstallation allerdings nicht erforderlich.

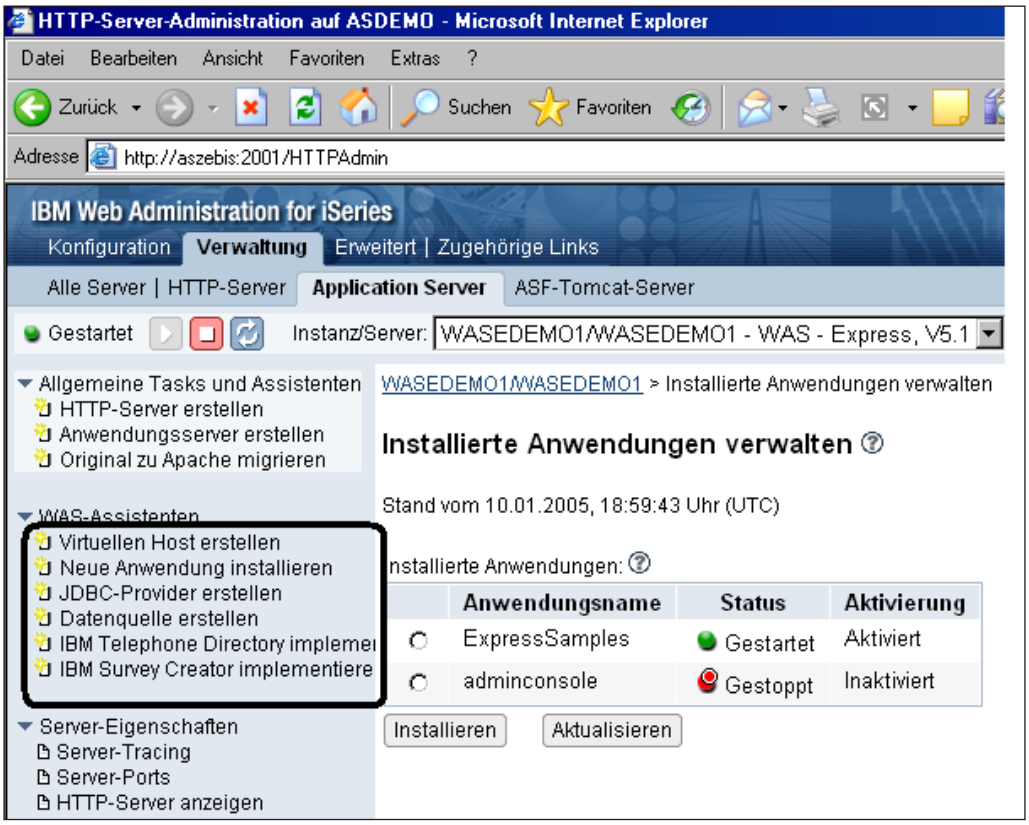
Zwar sind die Funktionalitäten des Telephone Directory recht schlank gehalten, dennoch lässt sich mit dieser einfachen Beispielanwendung recht leicht die Funktion einer Web-Anwendung im Zusammenspiel mit den verschiedenen Ebenen darstellen.

Die Bereitstellung einer Anwendung auf einem Applikationsserver verläuft immer nach einem gleichartigen Verfahren. Hier gilt also: Waren die vorhergehenden Installationsschritte auf die Spezifika der Anwendung „Telephone Directory“ abgestimmt (nicht jede Anwendung benötigt die LDAP-Funktionalitäten) so sind die Installationsschritte, wie wir Sie nun exemplarisch für das „Telephone Directory“ beschreiben weitestgehend für alle übrigen Anwendungen verwendbar.

5.2.1.3.1
Seite 10

Zunächst wählen wir in dem Verwaltungsbereich des Applikationsservers einen der Bereiche aus, von dem die Installation einer Anwendung möglich ist. IBM differenziert hier noch zwischen den Menüeinträgen „Neue Anwendung installieren“, mit der grundsätzliche jede Art von Anwendung auf dem Applikationsserver installiert werden kann und den von IBM vorgefertigten Anwendungen in Form des bereits genannten „Telephone Directory“ sowie des „IBM Survey Creator“.

Diese Installationspunkte finden wir im linken Navigationsbereich unterhalb des Eintrages „WAS- Assistent“.

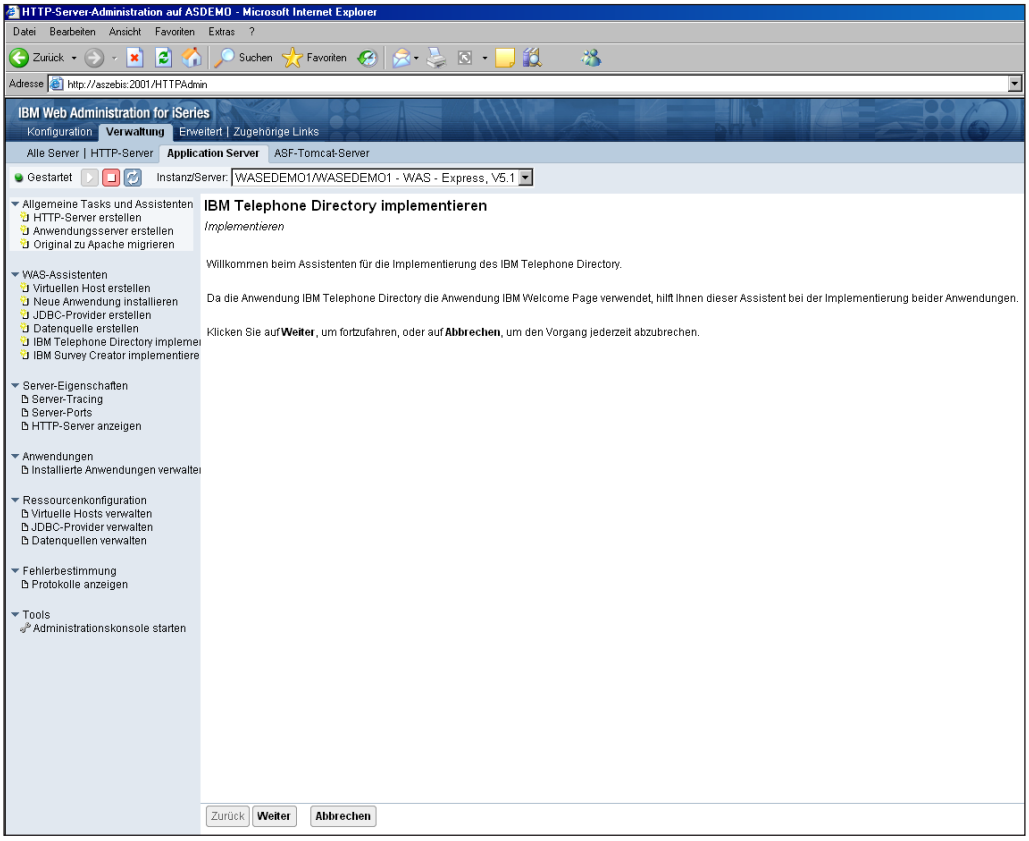


Navigationseinträge Installationen

Sobald wir eine Anwendung installiert haben, wird diese in der Übersicht „Installierte Anwendungen verwalten“ angezeigt, in die wir ebenfalls durch einen Klick in dem Navigationsbereich verzweigen können – auf die Verwaltung mehrerer installierter Anwendungen gehen wir später noch ein.

Wir wollen für unsere Beispielinstallation nun das Telephone Directory installieren. Dabei gehen wir davon aus, dass die dazu erforderliche Software als auch die übrigen Voraussetzungen (PTF, Applikationsserver, http-Server) erfüllt sind. Da IBM für jede Version gerade die Voraussetzungen in Bezug auf die PTF ständig anpasst, hier lediglich der Verweis auf die Web-Seiten der IBM, auf denen die jeweils aktuellen Informationen enthalten sind.

Wir wählen nun in dem linken Navigationsbereich des Administrationsclients den Eintrag „IBM Telephone Directory implementieren“ aus. Der Installationsassistent bietet eine gute Hilfe bei der Installation.

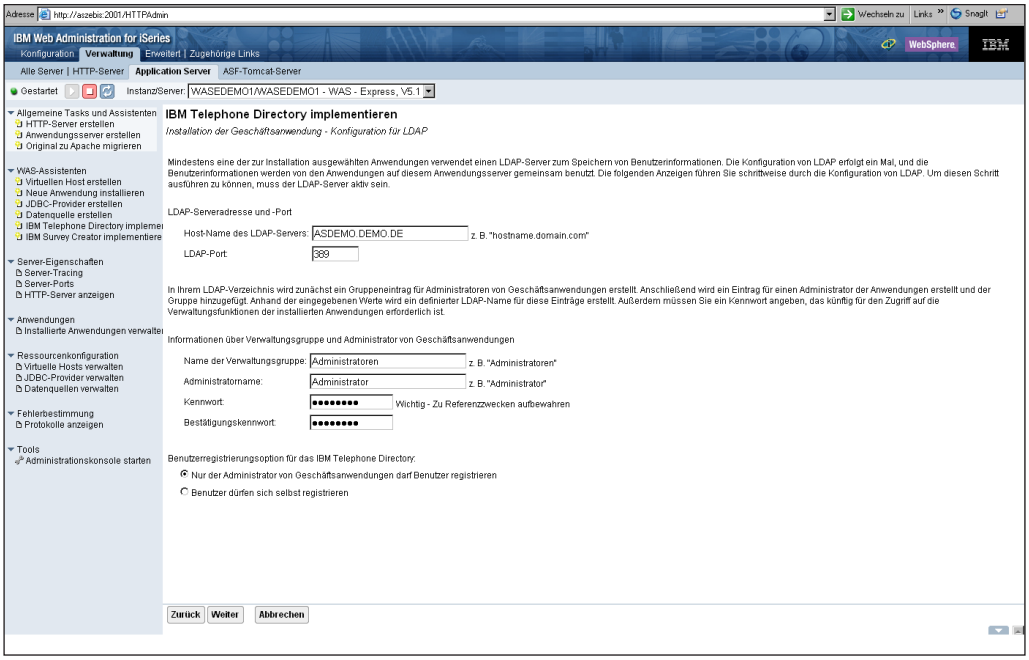


Telephone Directory installieren

Nachdem die Einführungsseite erscheint, klicken wir auf die Schaltfläche „Weiter“ im unteren Bereich der Anzeige.

In Abhängigkeit der Konfigurationseinstellungen des Applikations-Servers werden diverse Voreinstellungen getroffen. Unter anderem wird der Host-Name des LDAP-Server voreingestellt. Dieser ist für die Ausführung der Anwendung erforderlich.

5.2.1.3.1
Seite 12

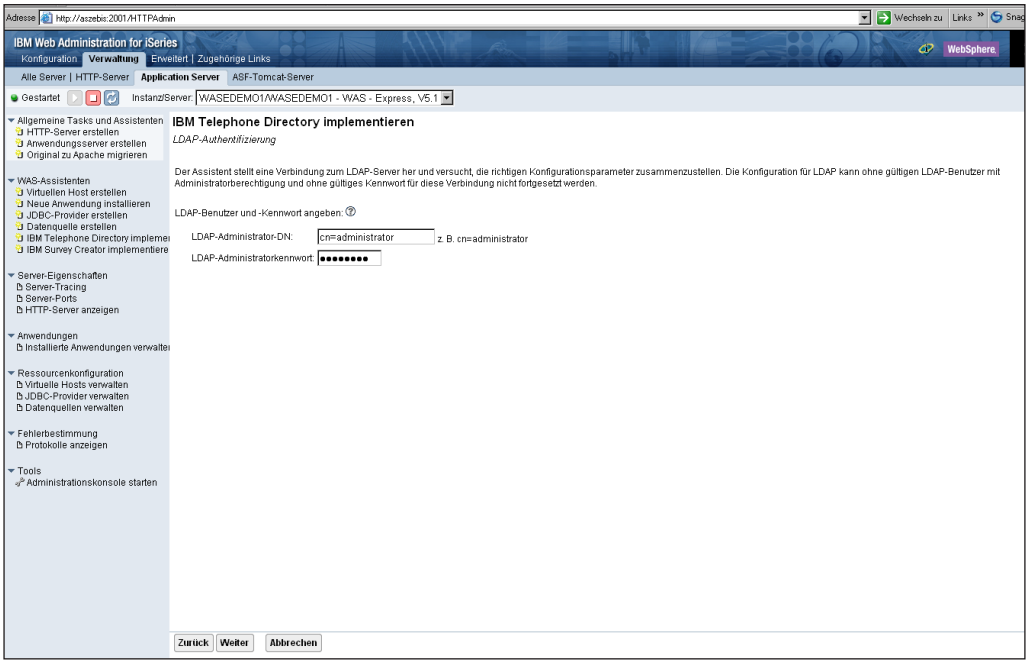


Telephone Directory implementieren

Für die spätere Administration der Anwendung ist ein eigener Administrator erforderlich, der für spezielle Aufgaben in Bezug auf das Telephone Directory verwendet wird. Der Name des Administrators als auch der Gruppe, der dieser hinzugefügt wird, kann an dieser Stelle frei gewählt werden. Der Installationsprozess schlägt Mustereinstellungen vor, die wir an dieser Stelle übernehmen. Diese Einstellungen ergänzen wir um ein Administrationskennwort, das wie auch von anderen Installationen gewohnt, doppelt einzugeben ist.

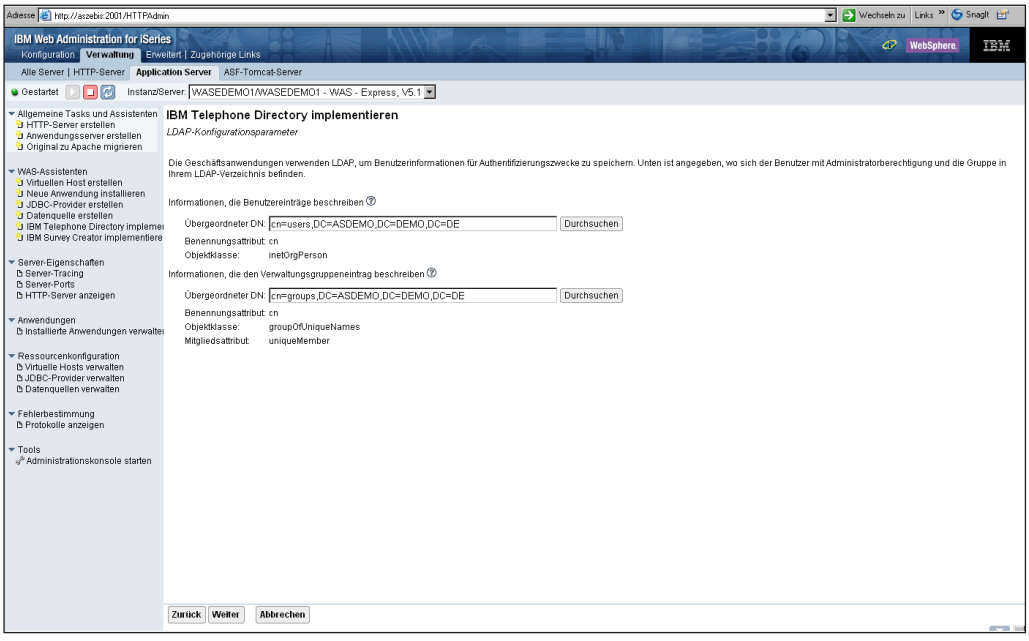
Ein Klick auf die Schaltfläche „Weiter“ führt uns zur Eingabeseite für die LDAP-Authentifizierung.

Auch hier finden wir wieder die bekannten Angaben in Bezug auf den Administrator dieses Bereiches.



LDAP-Angaben

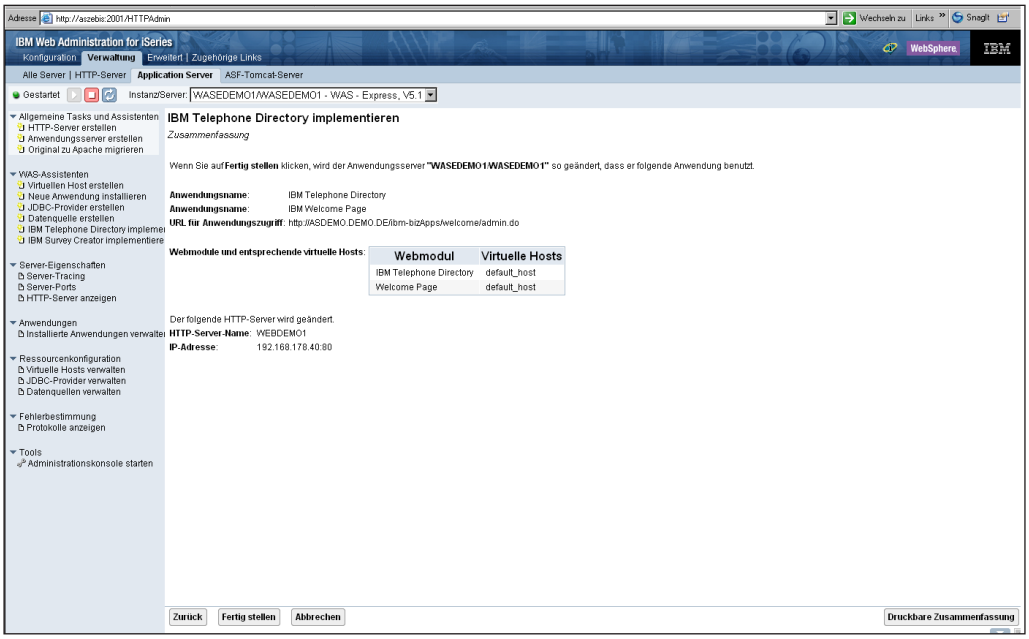
Nach einem Klick auf „Weiter“ werden in der nächsten Anzeige detaillierte Informationen für die Verwendung von LDAP abgefragt. Diese übernehmen wir, wie von dem System vorgeschlagen.



LDAP-Konfigurationsparameter

5.2.1.3.1
Seite 14

Nach einem erneuten Klick auf „Weiter“ erscheint bereits die Abschlussanzeige des Installationsprozesses der Anwendung.



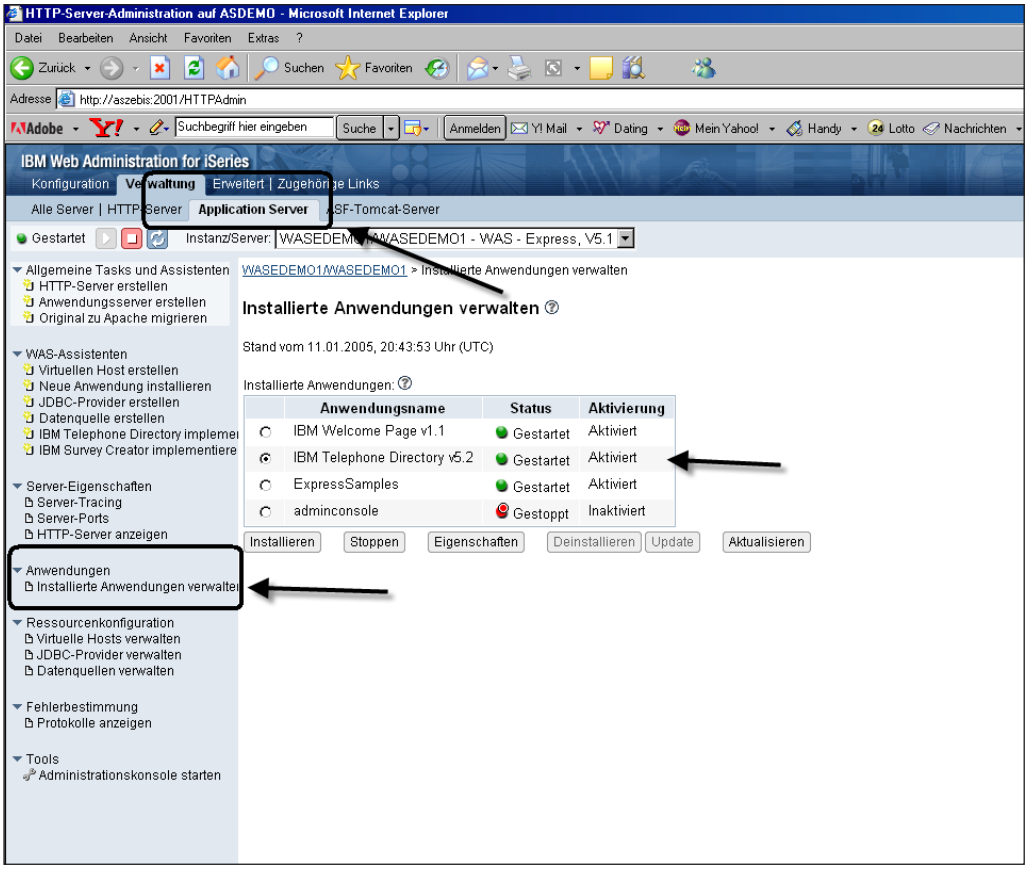
Telephone Directory Zusammenfassung

Für die spätere Arbeit mit der Anwendung ist die URL erforderlich – also die Adressierung, über die diese Anwendung direkt aufgerufen werden kann. Diese finden wir in der Abbildung unter dem Eintrag „URL für Anwendungszugriff“.

Beachten Sie auch den Eintrag für den http-Server, der diesem Applikations-Server zugeordnet ist. Über diesen Server findet bei der Ausführung der Anwendung die Kommunikation mit dem Browser statt.

Mit einem Klick auf die Schaltfläche „Fertig stellen“ wird die Anwendung auf der Applikations-Server-Instanz installiert. Das Deployen oder auch Installieren einer Anwendung auf einem Applikations-Server kann je nach Komplexität der Anwendung und der verwendeten Hardware einige Zeit in Anspruch nehmen – hier heißt es ein wenig Geduld mitbringen.

Den aktuellen Stand der Installation können wir jederzeit in der Übersicht der installierten Anwendungen (dieser Auswahlpunkt befindet sich in der linken Navigationsseite im Bereich der Applikations-Server) ansehen. Wählen wir diesen Eintrag aus, dann erscheint eine Übersicht aller auf diesem Server installierten Anwendungen und deren Ausführungsstatus.



Installierte Anwendungen verwalten

Hier finden wir auch den Eintrag für das soeben installierte „Telephone Directory“. Damit wir damit arbeiten können, müssen wir diese Anwendung, wie jede andere Anwendung auch, zunächst starten.

Sollte der Status also nicht bereits auf „gestartet“ stehen, dann markieren wir den Eintrag und wählen anschließend den Button „Starten“ aus. Auch hier müssen wir dem Server einige Momente Zeit geben – dann betätigen wir die Schaltfläche „Aktualisieren“, um den Browser mit dem aktuellen Stand der Daten zu versorgen – der Status der Anwendung sollte dann auf „gestartet“ stehen.

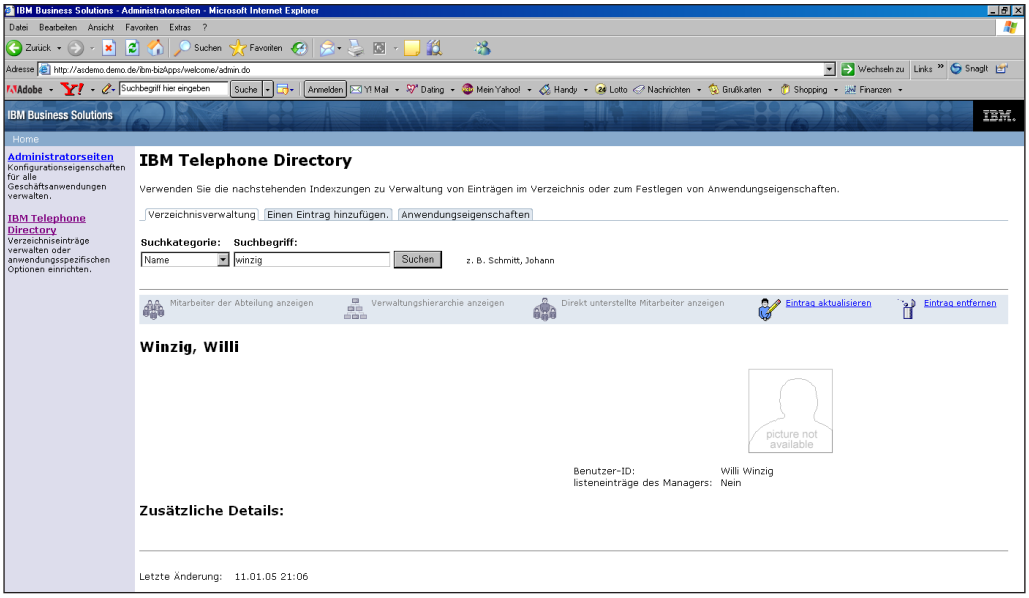
5.2.1.3.1
Seite 16

Anwendung starten

Sobald der Status „gestartet“ lautet, steht einem Arbeiten mit der Anwendung nichts mehr im Wege – allerdings: ein oft gemachter Fehler liegt darin, lediglich den Anwendungs-Server zu starten. Stellen Sie in jedem Fall sicher, dass der erforderliche http-Server – denn dieser übernimmt die Eingabe-/Ausgabeoperationen von/zu dem Browser – ebenfalls gestartet ist! Informationen zum Starten des http-Servers finden Sie in dem Kapitel 5.1.

Wir können die Anwendung nun durch eine Direkteingabe in einem neuen Browserfenster durch Angabe der Start-URL (diese hatten wir uns zuvor notiert) starten und mit der Anwendung arbeiten. Diese ermöglicht neben dem Erfassen von Telefonbucheinträgen auch elegante Suchfunktionen.

Sicher keine bahnbrechende Anwendung – für erste Schritte in diesem Bereich sollte sie aber durchaus eine Installation wert sein und eignet sich gut für das Vertrautmachen mit den Technologien rund um Applikations-Server, Web-Server & Co.



IBM Telephone Directory

5.2.1.3.1

Seite 18

iSeries Access for Web

Ich möchte nicht zu viele Beispiele aufzählen, die in Verbindung mit dem WebSphere Express Server eingesetzt werden können. Das nachfolgende Beispiel halte ich für durchaus nützlich, deshalb möchte ich es an dieser Stelle auch vorstellen. Außerdem verfügt jeder iSeries-Betreiber, der das Lizenzprogramm „iSeries Access Family" sein Eigen nennt, automatisch über eine Lizenz für die Funktionen, die ich Ihnen in diesem Abschnitt gerne vorstellen möchte. Die Rede ist hier von „iSeries Access for Web", einer Möglichkeit des Zugriffs auf die iSeries ganz ohne Client-gebundene Zusatzsoftware wie z.B. iSeries Access. Auf dem Client wird für einen Zugriff auf die iSeries lediglich ein gängiger Browser benötigt und schon kann in Verbindung mit einem WebSphere Application Server Express über einen Browser eine 5250-Emulation ausgeführt werden, Jobsteuerungen, Drucksteuerungen, Nachrichtensteuerungen etc. verwaltet werden und die wichtigsten administrativen Aufgaben auf der iSeries ohne spezielle Client-Software aus einem Browser heraus ausgeführt werden.

Weiterhin unterscheidet sich die Installation dieser IBM-Standardanwendung von dem zuvor gezeigten Beispiel in einigen wesentlichen Punkten.

Ich halte die Funktionen, die in iSeries Access for Web integriert sind wirklich für ausgezeichnet, wenn es darum geht, „mal eben schnell" von irgendwo auf das System zuzugreifen und administrative Tätigkeiten auszuführen. Als Einschränkung des Einsatzes von iSeries Access for Web sind die zum Teil nicht unterstützten 5250-Funktionstasten und zum Teil die Geschwindigkeit zu nennen. Diese Nachteile lassen dieses Produkt wenig geeignet für den so genannten Standard-Benutzer erscheinen.

Meine Empfehlung lautet klar: iSeries Access for Web ist ein hervorragendes Werkzeug für einen unkomplizierten, schnellen Zugriff auf die iSeries, um dort administrative Aufgaben durchzuführen. So lassen sich schnell „hängende" Jobs freigeben, Nachrichten beantworten oder auch einmal zügig die Umsatzzahlen eines Kunden anschauen.

Der Vorteil dieser Software:
Sie benötigen lediglich einen Internetzugang und ein Benutzer-Login auf die iSeries und schon haben Sie einen Zugriff auf Ihr System, ohne jegliche Emulations-Software. Damit besteht theoretisch die Möglichkeit des Zugriffes auf Ihr System von jedem mit dem Internet verbundenen Rechner. Voraussetzung ist lediglich ein erforderliches Lizenzprogramm auf Seiten der iSeries, das optional installiert werden kann – und natürlich die erforderliche Berechtigung, auf die Daten der iSeries zugreifen zu können.

5.2.1.3.2
Seite 2

iSeries Access for Web ist ein Bestandteil der iSeries-Access-Familie (5722-XW1), das als zusätzliches Lizenzprogramm (5722-XH2) installiert werden muss. Für die Installation benötigen wir QSHELL, das ebenfalls als zusätzliches Lizenzprogramm (5722-SS1 Option 30) verfügbar ist.

Die Basis von iSeries Access for Web bildet verschiedene Standard-Technologien wie Java-Servlets, HTTP, HTTPs und HTML. In diesem Zusammenhang ist auch der Einsatz dieses Tools in Verbindung mit einem Anwendungs-Server zu sehen. In der neuesten Version V5R3M0 besteht außerdem die Möglichkeit, iSeries Access for Web mit Hilfe einer Portlet-Version in einen WebSphere Portal Server einzubauen und dort einen Zugriff auf die iSeries-Informationen zu erhalten. Dabei unterstützen sowohl der „WebSphere Portal Server" als auch die Express-Variante, der „WebSphere Portal Server Express" die Ausführung des iSeries Access for Web Portlets. Mit dieser Technologie sollen dann Systemintegration und auf den Benutzer abgestimmte Oberflächen eine Optimierung der Arbeit bedeuten. Da ist dann auch der direkte Zugriff auf die iSeries-Anwendungen sinnvoll.

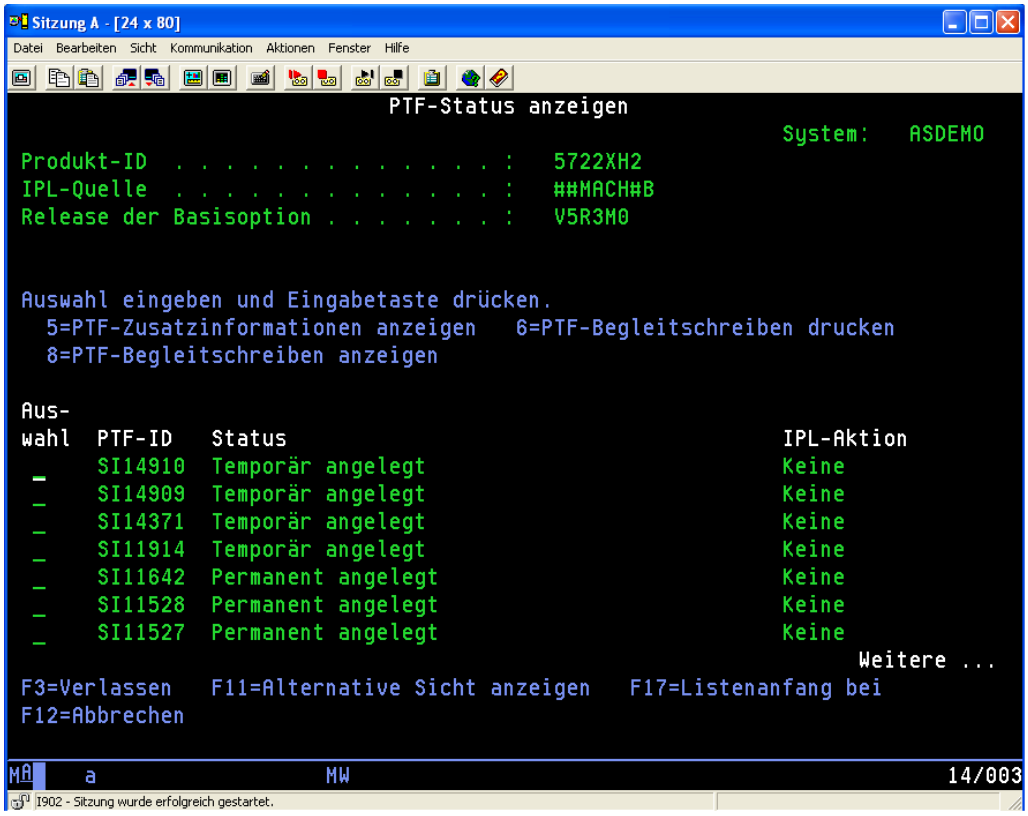
IBM bietet grundsätzlich verschiedene Möglichkeiten, iSeries Access for Web auf einer iSeries auszuführen. Für die Ausführung des Server-basierten Teils ist ein Anwendungs-Server wie zum Beispiel der WebSphere Application Server Express erforderlich, der ebenfalls auf der iSeries installiert wird. Alternativ dazu kann die Software auch in Verbindung mit einem ASF-Tomcat ausgeführt werden.

In unserem Beispiel verwenden wir für die Installation den WebSphere Application Server in der Version 5.1.

Neben dem installierten WAS Express ist ein dazugehöriger Web-Server erforderlich, der in der Regel bereits während der Installation des WAS mitinstalliert wurde. Bei Bedarf können Sie aber auch einen dedizierten HTTP-Server nur für die Unterstützung des iSeries Access for Web verwenden – diese Option würde ich Ihnen für eine Echtinstallation auf jeden Fall empfehlen, denn dann besteht die Möglichkeit der individuellen Anpassung der Einstellungen des WAS als auch des Web-Server für die Anforderungen des iSeries Access for Web.

Des Weiteren benötigen Sie die Lizenz für „iSeries Access" (5722-XW1), mit der auch iSeries Access for Web lizenziert ist. Aber Achtung! iSeries Access for Web wird nicht in jedem Fall automatisch mit der Installation von iSeries Access auf dem System als Lizenzprogramm installiert! Sie müssen dieses in der Regel zusätzlich installieren (Lizenzprogramm Nr. 5722-XH2).

Wie bei allen anderen Lizenzprogrammen, so gilt auch hier, dass bei Bedarf die erforderlichen PTF nach der Installation des Lizenzprogramms auf dem System eingespielt werden müssen. Die aktuellen Informationen hierzu finden Sie wie immer auf den Webseiten der IBM.



PTF Stand

Ich möchte an dieser Stelle auf die Erläuterung der Installation des Lizenzprogramms und der erforderlichen PTF bewusst verzichten – diese unterscheiden sich nicht von der Installation anderer Lizenzprogramme.

Durch den modularen Aufbau eignet sich iSeries Access for Web auch für verschiedene Web-Anwendungen. Primäre Einsatzbereiche sind allerdings die klassischen iSeries-Anforderungen wie zum Beispiel:

- Der Aufbau einer 5250-ähnlichen Dialogsitzung
- Das Ausführen von Batch-Befehlen
- Der Zugriff auf Dateien
- Der Zugriff auf das IFS
- Verwendung von Ausgabe-Warteschlangen
- Die Verwendung von Nachrichten-Warteschlangen
- etc.

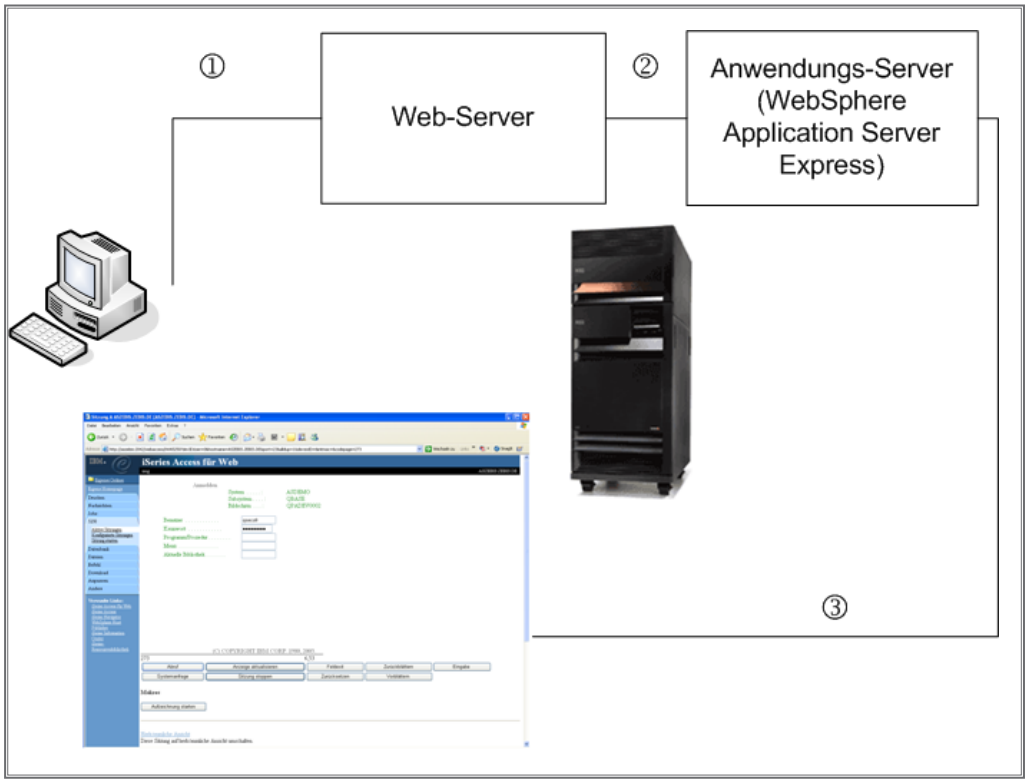
Leider werden in iSeries Access for Web nicht alle gängigen Tasten der 5250-Emulation von iSeries Access unterstützt – eine Einschränkung, die wohl auch in absehbarer Zeit von IBM nicht verändert wird!

5.2.1.3.2
Seite 4

Für den Arbeitsplatz, über den auf die iSeries zugegriffen werden soll, gelten als Mindestvoraussetzungen:

- Ein Browser – idealer Weise Microsoft Internet Explorer (in der Praxis habe ich auch andere Browser wie zum Beispiel „Opera“, „Firefox“ getestet – beide wurden einwandfrei unterstützt).
- Eine möglichst performante Verbindung in das Internet (oder eine Netzverbindung zur iSeries).

Die nachfolgende Abbildung skizziert diesen Ablauf schematisch:



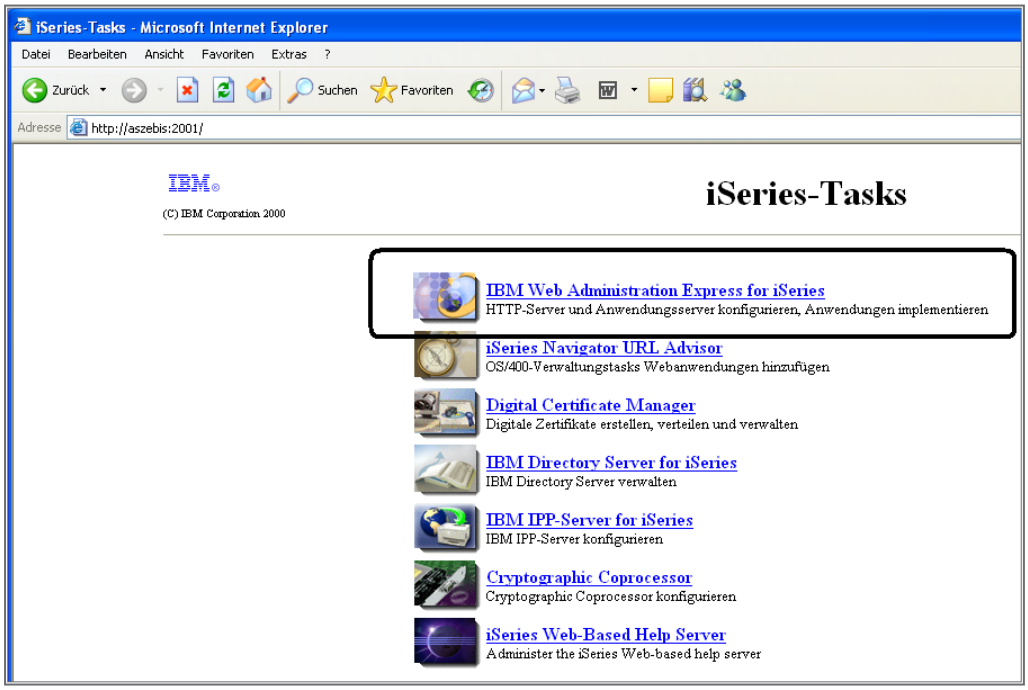
iSeries Access for Web – der Ablauf

1. Ein Client greift über einen Browser auf den Web-Server zu – dabei handelt es sich dann um den HTTP-Server, den wir für die Arbeit mit iSeries Access for Web konfiguriert haben.
2. Die Anforderung (z.B. Ausgabe einer 5250-Emulation in einem Browser) wird an den Anwendungs-Server weiter geleitet. Dieser Anwendungs-Server ist in unserem Beispiel der WebSphere Application Server Express in der Version 5.1.
3. Nachdem die Anforderung verarbeitet wurde, wird die Ausgabe – wieder gesteuert über den Web-Server, an den Browser übergeben. In unserem Beispiel sieht der Anwender jetzt die 5250-Anzeige.

Doch zurück zu der Installation!

Nachdem das Lizenzprogramm „iSeries Access for Web“ – 5722-XH2 und die dazugehörigen PTF installiert worden sind, kann mit der Einrichtung begonnen werden.

Wie bereits in dem vorhergehenden Kapitel beschrieben, verwenden wir für die Konfiguration des iSeries Access for Web in Verbindung mit dem WebSphere Application Server Express (kurz: WASE) die Administrations-Oberfläche des WASE, die wir in einem Browser starten. Die nachfolgende Abbildung enthält das Beispiel des Starts des Administrationsbereiches des WASE auf einem nicht gesicherten Port „2001“.



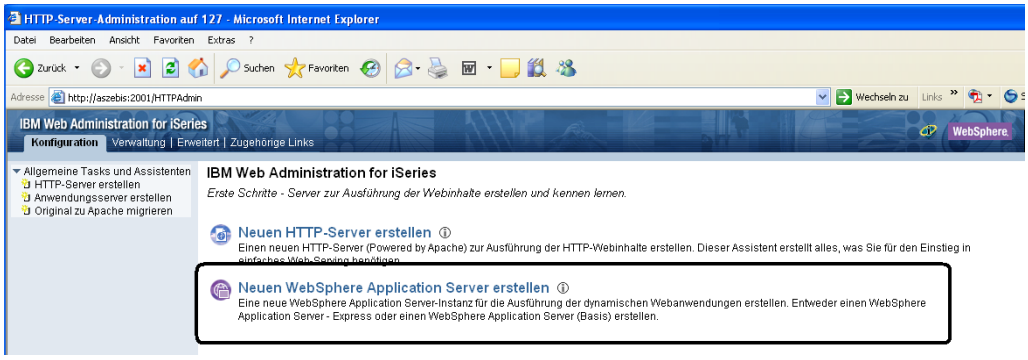
Admin starten

Auf jedem physisch auf einer Maschine installierten WASE können wir mehrere Instanzen erstellen, die jede für sich eine eigene administrative Einheit bildet. Das empfehle ich auch für den Einsatz von iSeries Access for Web.

In diesem Fall erstellen wir eine eigene Instanz des WASE und einen eigenen Web-Server, der ebenfalls exklusiv für den Einsatz von iSeries Access for Web verwendet wird.

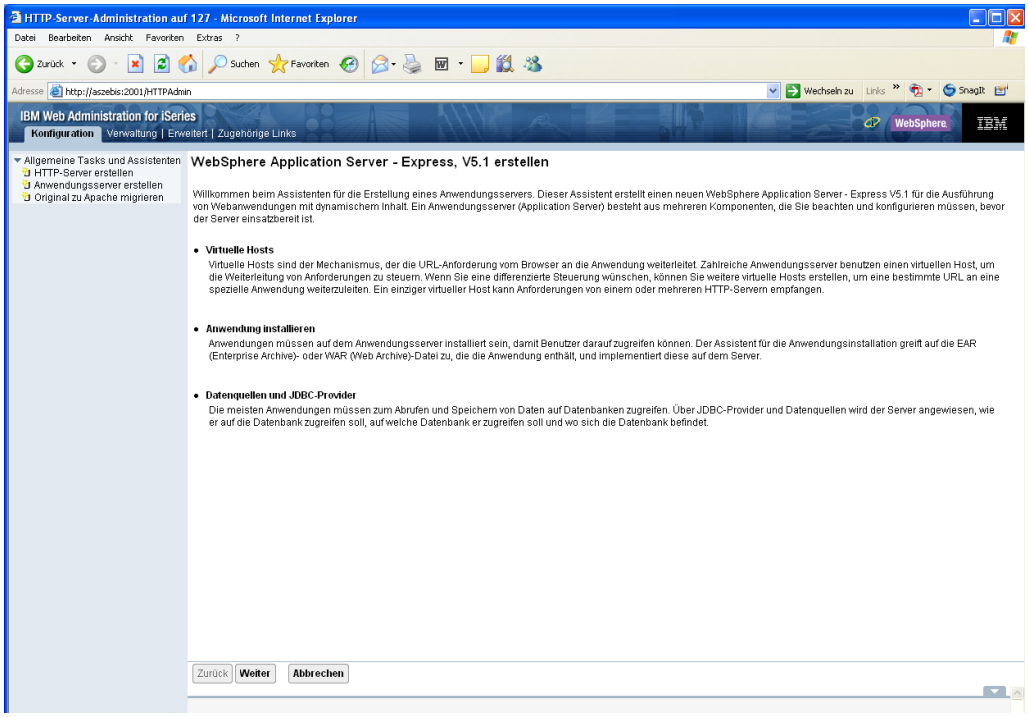
Dazu wählen wir den Tabellenbereich „Konfiguration“ aus, von wo aus wir sowohl den HTTP-Server, als auch den WASE erstellen können. Hier selektieren wir zunächst „Neuen WebSphere Application Server erstellen“.

5.2.1.3.2
Seite 6



Application Server erstellen

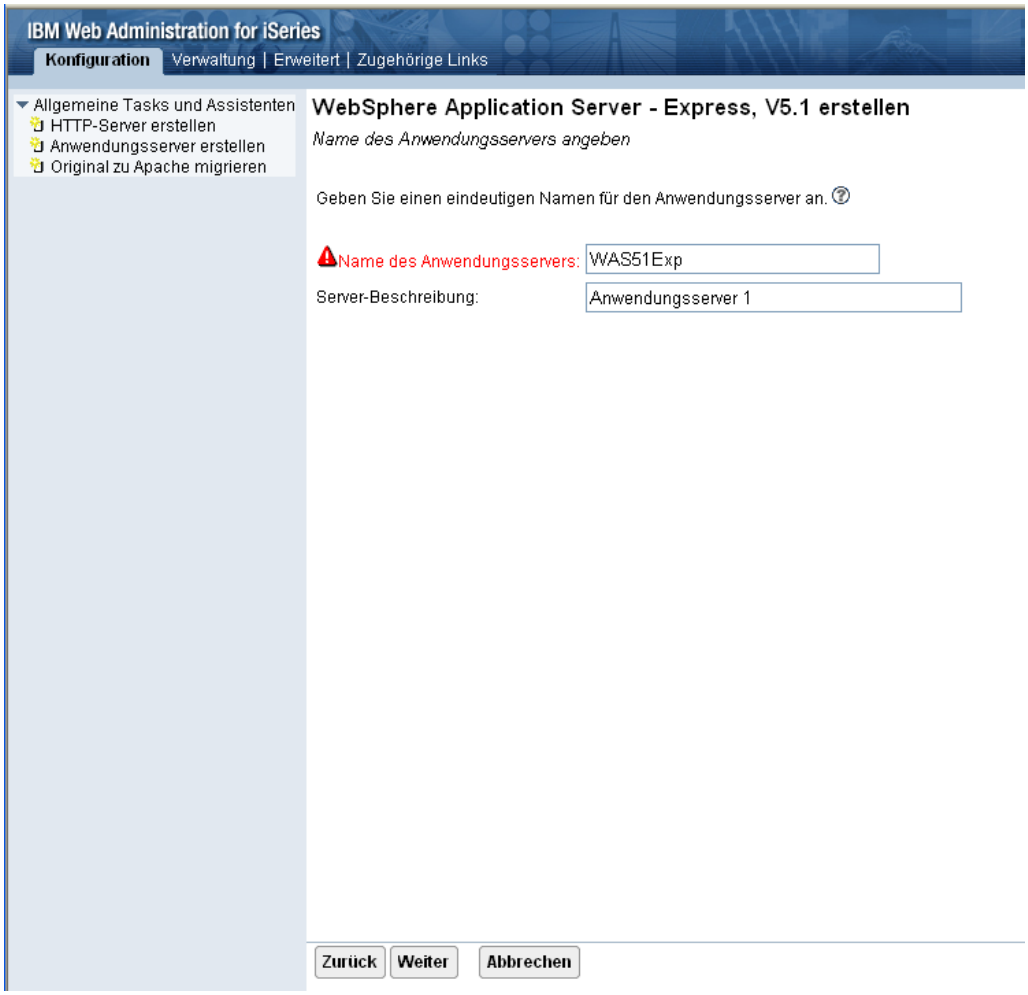
Die nächste Anzeige enthält eine Beschreibung aller anstehenden Einzelschritte für die Erstellung der WASE Instanz.



Beschreibung

Nach einem Klick auf die Schaltfläche „Weiter“ beginnt die eigentliche Konfiguration.

Mit dieser Installation bzw. Konfiguration erstellen wir einen neuen WASE Server. Diesem geben wir einen eindeutigen Namen – in unserem Beispiel nennen wir den Server „WAS51EXP“. Dieser Name ist später bei der Administration des Server von Bedeutung. Die Server-Beschreibung enthält einen Freitext.

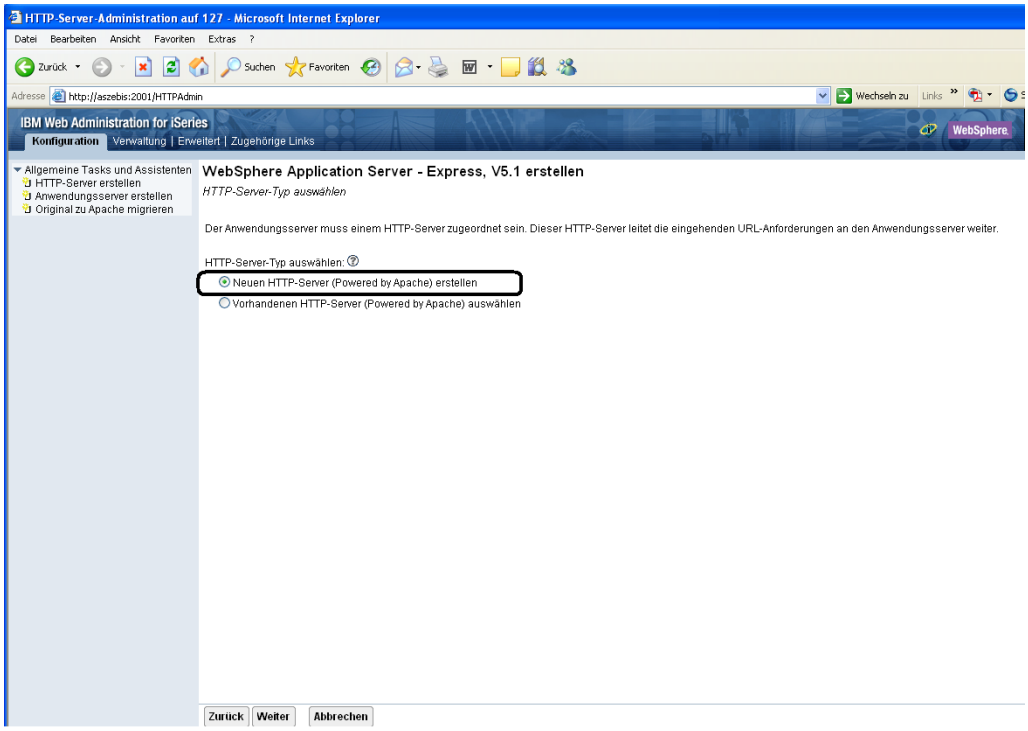


Name des Server

Nachdem wir sowohl den Namen des Anwendungsserver als auch die Server-Beschreibung angegeben haben, klicken wir auf „Weiter“.

Der Installationsprozess erkennt jetzt, dass ein neuer Anwendungsserver erstellt werden soll. Da für diesen auch ein zugeordneter Web-Server erforderlich ist, werden wir in der nächsten Anzeige gefragt, ob wir einen neuen Web-Server erstellen, oder auf einen bereits bestehenden Web-Server zugreifen wollen.

5.2.1.3.2
Seite 8

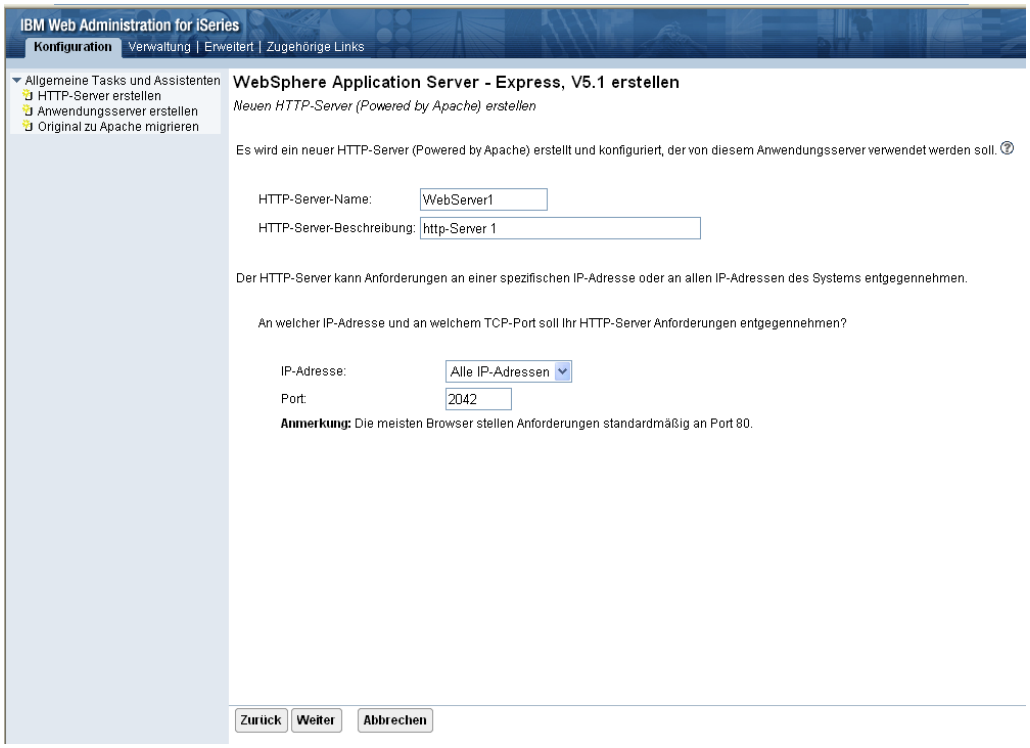


Web-Server auswählen

Um eine strikte Trennung und spätere individuelle Administration des iSeries Access for Web-Zugriffs garantieren zu können, empfehle ich an dieser Stelle das Erstellen eines neuen Web-Server.

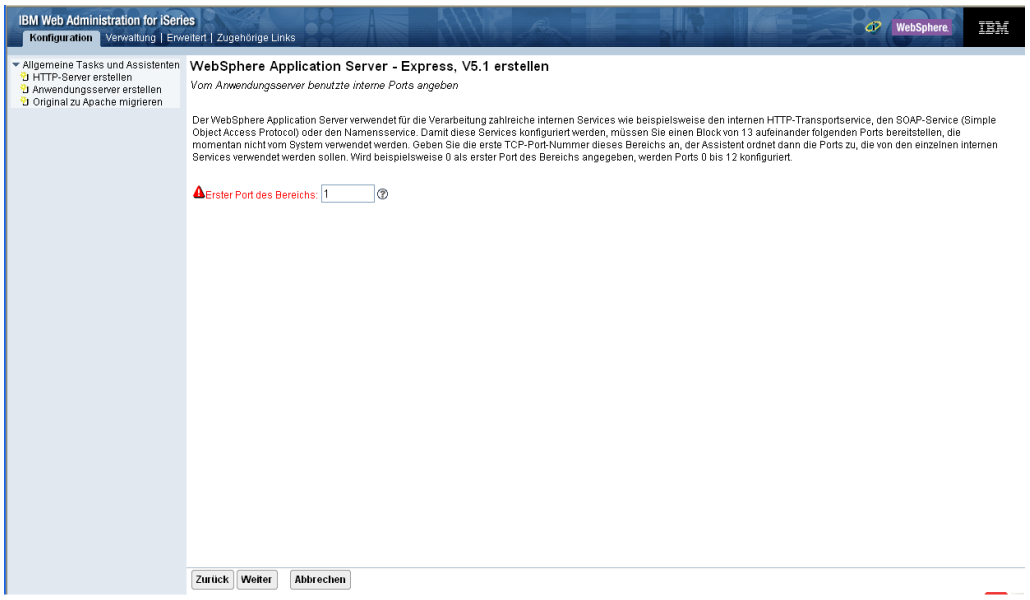
Diese Option wählen wir aus und klicken auf die Schaltfläche „Weiter“.

Der nächste Schritt ist ebenfalls schon bekannt – hier geht es um die Namensvergabe für den Web-Server sowie einige globale Einstellungen. Unser Web-Server erhält den Namen „WebServer1“. Für die Kommunikation geben wir den Port 2042 an. Dieser wird als Standard für iSeries Access for Web verwendet. Sollten Sie unsicher sein, ob dieser Port auf Ihrem System noch frei ist, dann prüfen Sie dies zunächst mit dem Befehl „NETSTAT *CNN“. Wird der Port bereits von einer anderen Anwendung verwendet werden, dann müssen wir an dieser Stelle einen anderen Port angeben. Damit verändert sich dann später auch der Zugriff auf iSeries Access for Web, wo neben der IP-Adresse (oder dem Hostnamen), auch der Port angegeben werden muss, über den der Zugriff durchgeführt werden soll.



Web-Server Angaben

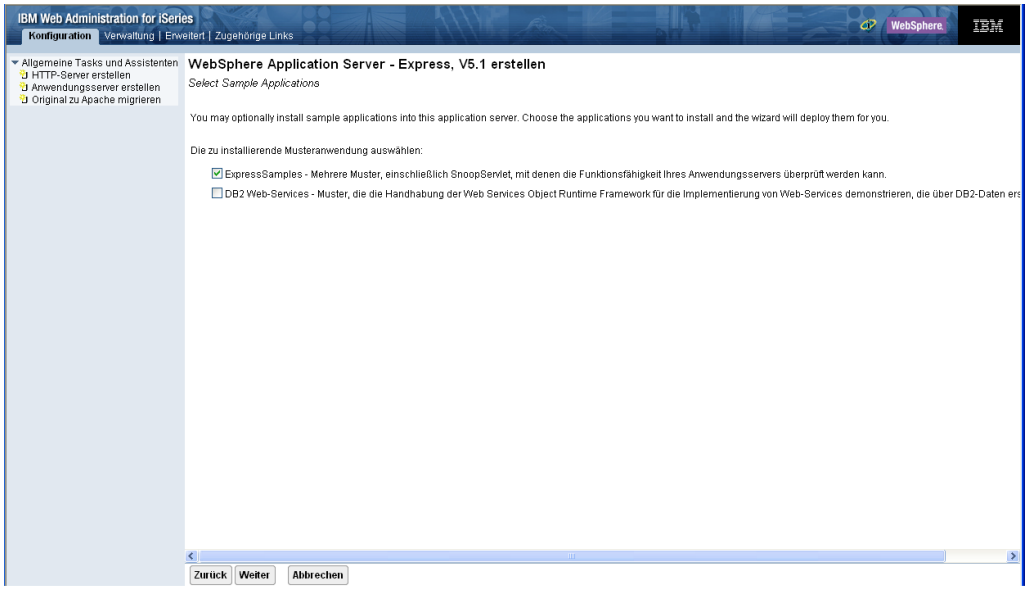
Die Portverwendung spielt auch bei dem Einsatz von iSeries Access for Web eine wesentliche Rolle. Neben dem zuvor genannten Standardport 2042 werden noch einige weitere Ports verwendet, die für das Ausführen des WASE benötigt werden. Dabei müssen 13 aufeinander folgende Ports frei sein.



Portfolgen

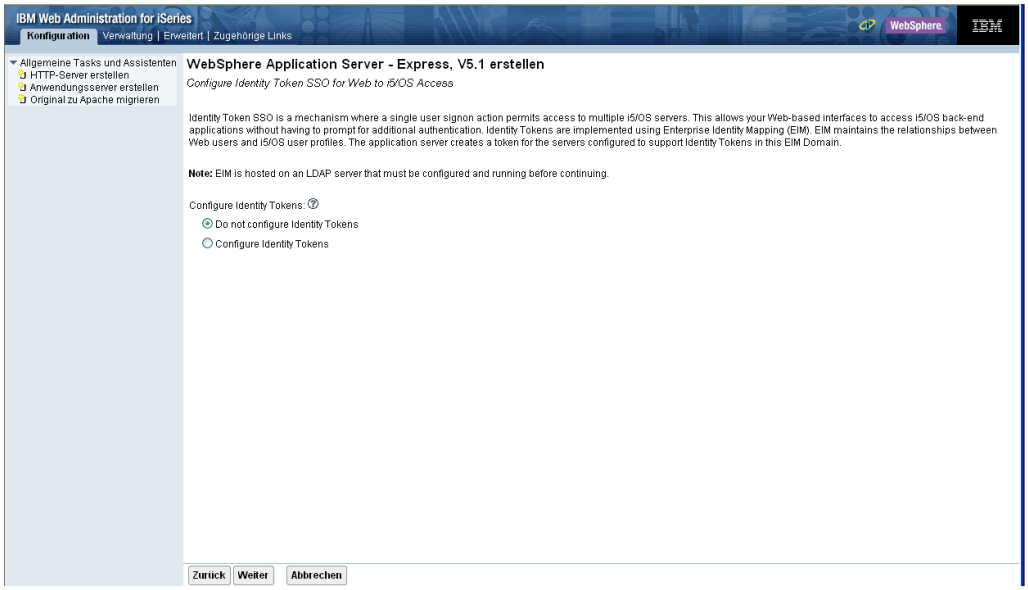
5.2.1.3.2
Seite 10

Es ist nicht erforderlich, aber wir führen es für unsere Beispielinstallation trotzdem aus: die Installation von mitgelieferten Musteranwendungen. Für eine Minimal-Installation dieses WASE können Sie in der nächsten Anzeige auch beide Auswahlfelder leer lassen und einfach auf die Schaltfläche „Weiter“ klicken.



Beispielanwendungen

Bedingt durch die vielfältigen Anwendungen und deren fortschreitende Integration ist die ehemals hoch gelobte Möglichkeit der Absicherung jeder einzelnen Anwendung mittlerweile zu einer Last geworden. „Single Signon“ (kurz: SSO) ist hier das „Zauberwort“ – die Möglichkeit des gesicherten, zentral gesteuerten Kennwort-Zugriffs auf verschiedenste Anwendungen. Dieser Möglichkeit Rechnung tragend, können wir für diesen WASE die SSO-Features installieren und somit den Zugriff auf verschiedene Anwendungen ohne erneute Authentifizierung ermöglichen.

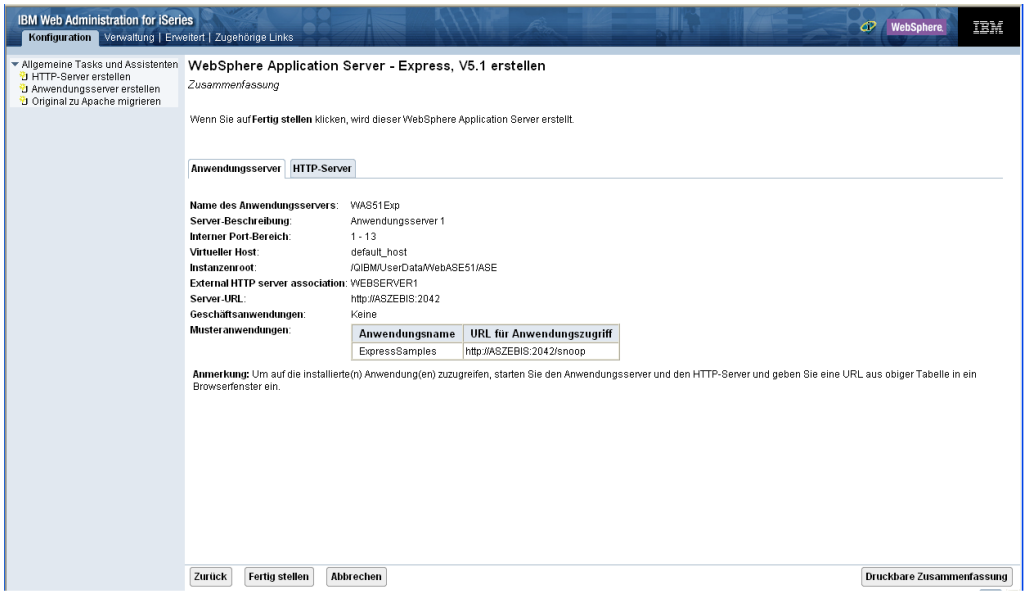


SSO ermöglichen

Sollten Sie in Ihrem Umfeld SSO realisiert haben, dann empfehle ich die Auswahl „Configure Identity Tokens“. Für unsere Beispielinstallation verzichten wir jetzt auf die Verwendung der SSO-Features und klicken auf die Schaltfläche „Weiter“.

Das war es auch schon. Diese wenigen Angaben genügen, um sowohl einen WASE als auch einen dazugehörigen Web-Server zu konfigurieren. Die nachfolgende Abbildung zeigt nochmals eine Zusammenfassung der getätigten Angaben. Prüfen Sie diese. Sollten Sie noch Änderungen vornehmen wollen, dann klicken Sie einfach auf die Schaltfläche „Zurück“ und gehen soweit zurück, bis Sie die zu ändernden Angaben korrigieren können. Sind alle getätigten Angaben korrekt, dann können wir die beiden Server (den Anwendungs-Server und den Web-Server) mit einem Klick auf die Schaltfläche „Fertig stellen“ generieren.

5.2.1.3.2
Seite 12



Installations-Zusammenfassung

Nachdem die Installation abgeschlossen ist, sollten sowohl der WASE als auch der zugehörige Web-Server gestartet sein. Prüfen Sie deren Status und starten Sie beide Server, wenn diese nicht bereits aktiv sein sollten.

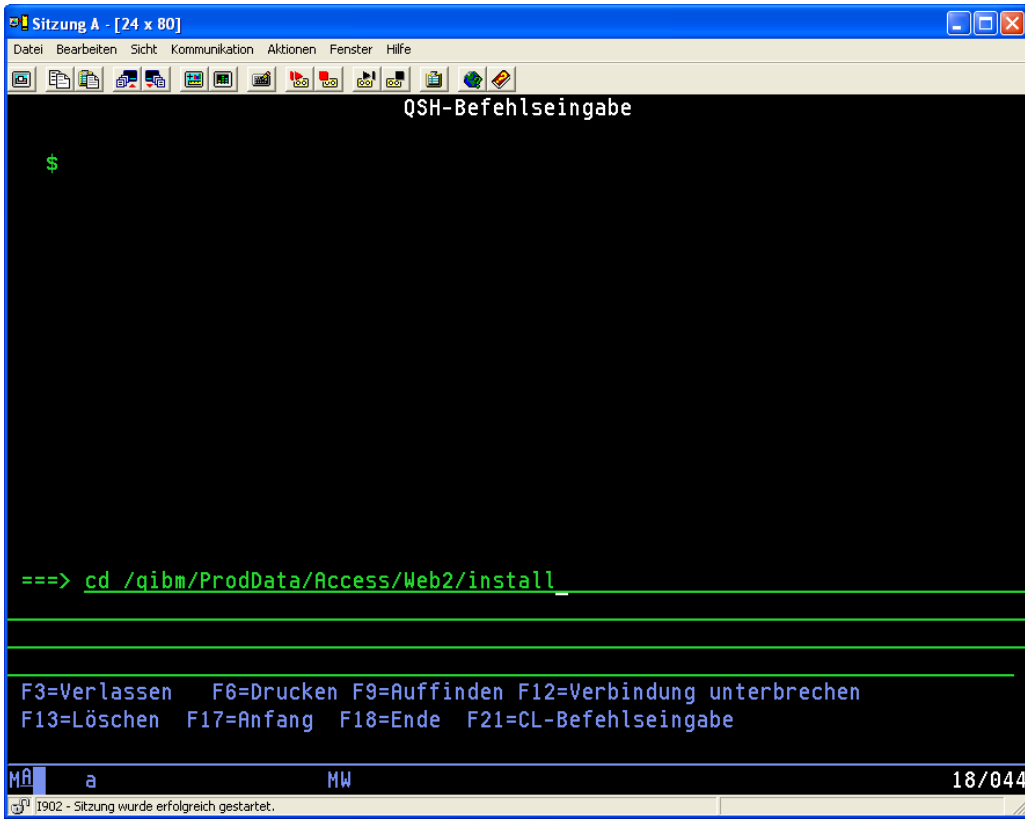
Damit haben wir die Voraussetzung für die Installation von iSeries Access geschaffen.

Für die Installation von iSeries Access for Web verwenden wir in unserem Beispiel eine QSHELL-Eingabe und starten diesen Bereich durch die Eingabe „QSH“ in der Befehlszeile einer 5250-Sitzung.

Jetzt wird das QSHELL-Eingabefenster geöffnet, das wir für die Installation der Software benötigen.

Die Installations-Angaben für iSeries Access for Web wurden mit der Installation des Lizenzprogramms 5722-XH2 in das Verzeichnis „QIBM/ProdData/Access/Web2/install“ gestellt.

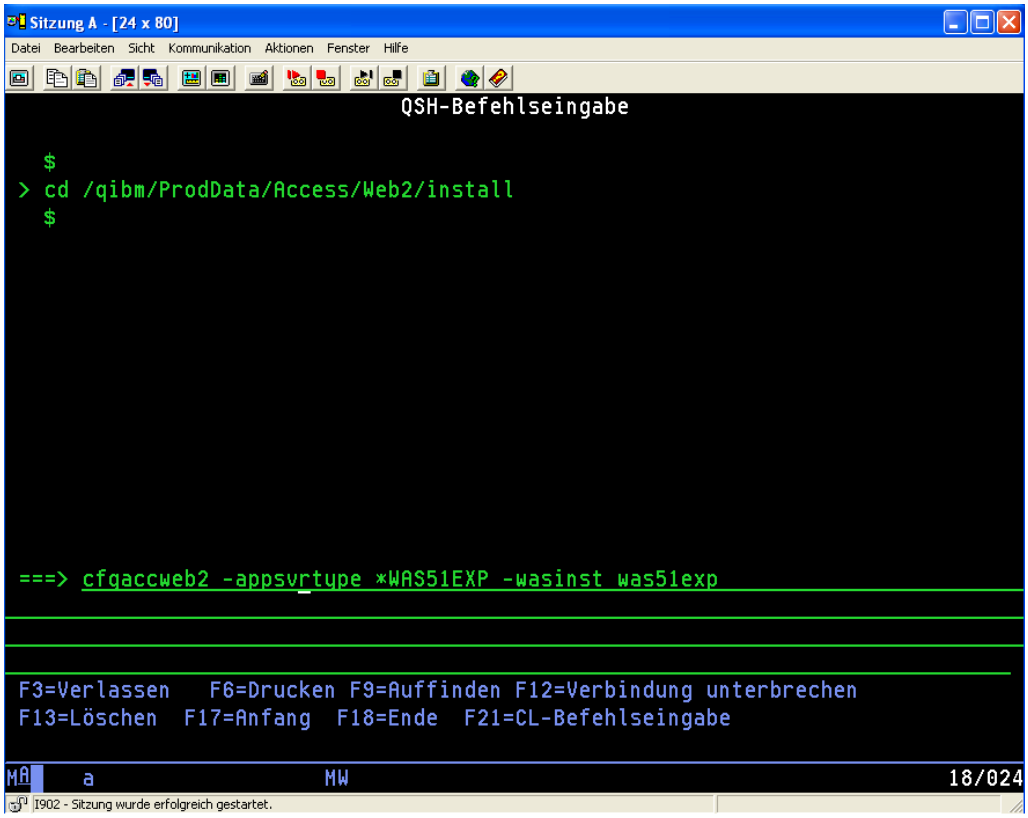
Mit dem in der nachfolgenden Abbildung gezeigten Befehl „CD“ wechseln wir in dieses Installations-Verzeichnis.



QSHELL Eingabe

In diesem Verzeichnis befindet sich auch das Startprogramm für die Installation, das wir mit einem speziellen Befehl aufrufen, wie in der nachfolgenden Abbildung gezeigt wird. Sollten Sie für Ihre Installation einen anderen Namen für den WASE-Server verwendet haben als „WAS51EXP“, dann müssen Sie den letzten Parameter in dem Befehl „cfgaccweb2“ durch den von Ihnen verwendeten Server-Namen ersetzen.

5.2.1.3.2
Seite 14



Aufruf der Konfiguration

Jetzt heißt es in Abhängigkeit der Maschine und deren Auslastung, ein wenig Geduld zu haben. Nach der Eingabe des oben gezeigten Befehls wird iSeries Access for Web auf dem angegebenen WASE-Server – in unserem Fall ist das der Server „WAS51EXP“ installiert. In der QSHELL-Anzeige erscheinen verschiedene Meldungen, die uns über den Status der Installation informieren. Zum Abschluss sollte eine Erfolgsmeldung erscheinen.

Damit ist die Installation von iSeries Access for Web abgeschlossen – jetzt steht für einen Zugriff auf die iSeries via Browser nichts mehr im Wege.

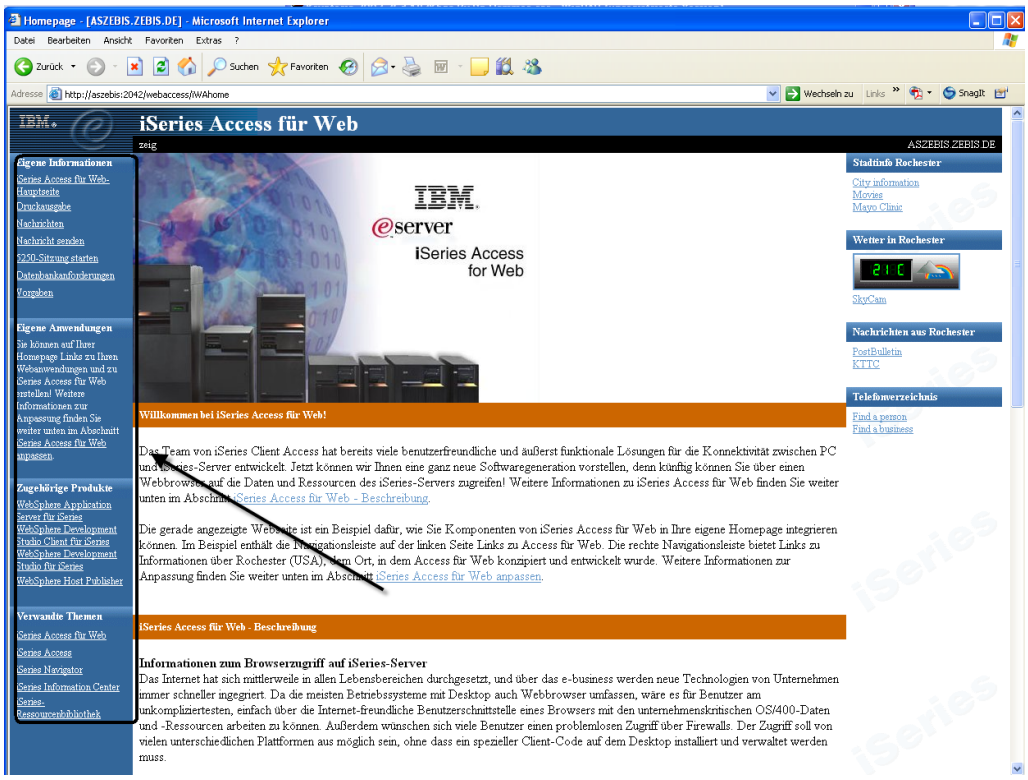
Hat alles funktioniert? Ein Test wird es zeigen:

Nachdem sowohl der Anwendungs-Server (WAS51EXP) und der dazugehörige Web-Server gestartet wurden, öffnen wir einen Browser. IBM gibt in der Literatur meist den Microsoft Internet Explorer an – Gozilla und Firefox eignen sich ebenso. Als Pfad geben wir die IP-Adresse (oder den Hostnamen) der iSeries, gefolgt von einem Doppelpunkt und dem bei der Konfiguration angegebenen Port (bei einer Standard-Installation dem Port 2042) und der Startseite für iSeries Access for Web an.

Eine solche Startseite kann beispielsweise so aussehen:

```
http://servername:2042/webaccess/iWAHome
```

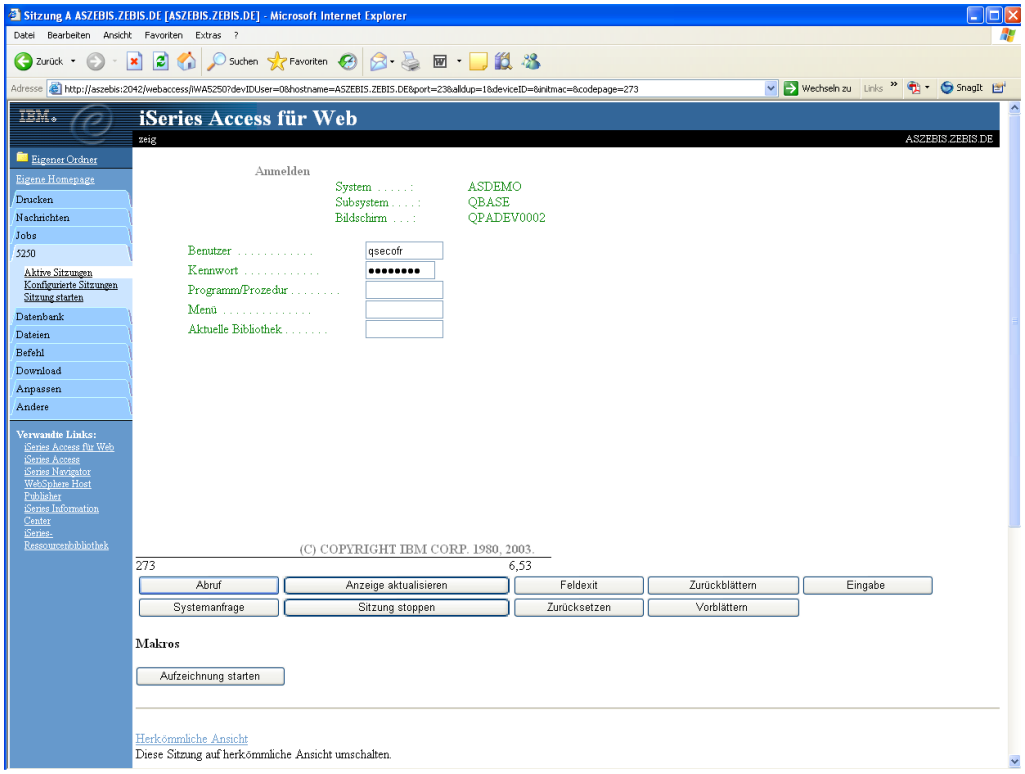
War die Installation erfolgreich, dann sollten wir jetzt eine ähnliche Anzeige erhalten, wie sie in der nächsten Abbildung zu sehen ist.



Erster Zugriff

Von dieser Startseite aus können Sie auf alle Funktionen des iSeries Access for Web zugreifen. Das sind neben der 5250-Emulation der Zugriff auf Drucker und Nachrichten, um nur einige zu nennen.

5.2.1.3.2
Seite 16



5250 Signon Bildschirm

Wie Sie sehen – stellt der WebSphere Application Server Express die Basis für verschiedenste Anwendungen, sei es als Basis für die Ausführung eigener Anwendungen oder die Ausführung von Standard-Anwendungen, wie sie auch von IBM angeboten werden.

Dieses Musterbeispiel einer WASE-Anwendung ist keine Demo-Version, sondern es handelt sich um eine produktiv einsetzbare Vollversion eines iSeries-Administrations-Zugriffs auf die iSeries mit Hilfe eines Browsers.

Verwalten installierter Anwendungen

Nachdem eine Anwendung auf dem Applikationsserver installiert wurde, kann es unter bestimmten Umständen erforderlich sein, dass wir gewisse Änderungen an der Anwendung durchführen. Dazu benötigen wir einige Informationen über die Anwendung und müssen bestimmte Regeln beachten, damit in Ausführung befindliche Anwendungen nicht etwa abstürzen.

In diesem Kapitel möchte ich Ihnen vorstellen, wie wir bereits installierte Anwendungen verwalten und ändern können. Dabei habe ich mich in diesem Kapitel bewusst auf die Verwendung der Standard-Oberfläche in Form des mit der i5 angebotenen Web-Administrators beschränkt und auf die Verwendung der Möglichkeiten mit der Administrationskonsole an dieser Stelle verzichtet. Informationen über deren Verwendung finden Sie in dem Kapitel 5.2.4 ff.

URL einer Anwendung ermitteln

Eine der wesentlichen Eigenschaften einer Anwendung ist deren URL – also die Startangabe, mittels deren wir eine Web-Seite – oder eben eine Web-Anwendung – aufrufen können. Unter einer URL verstehen wir einen Aufruf für eine Anwendung, die in einem bestimmten Bereich des Anwendungsservers gespeichert ist und von dort aus aufgerufen werden kann.

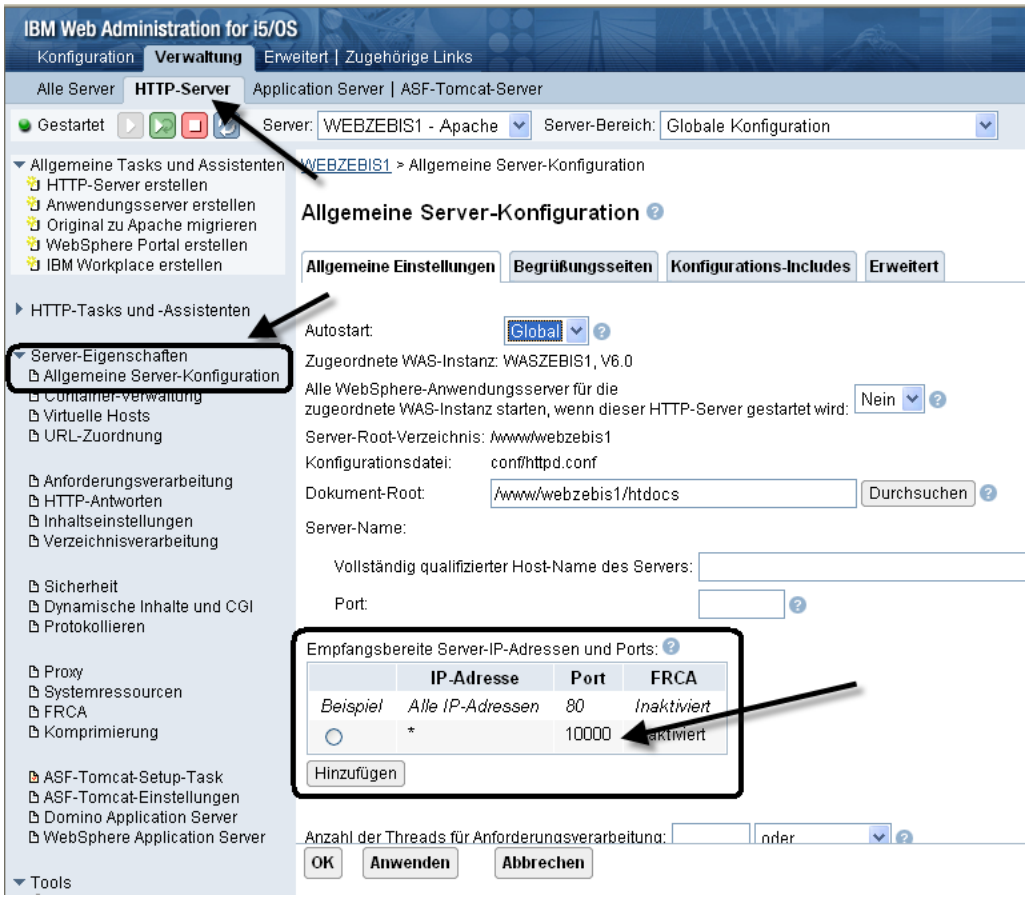
Ein Beispiel für eine solche URL:

http://Systemname:Portnummer/ContextRoot/Servletname

Der Systemname ist in unserem Fall der Hostname des Rechners, auf dem der Server läuft. Alternativ zu dem Hostnamen kann auch dessen TCP/IP-Adresse verwendet werden. Neben der IP-Adresse benötigen wir noch den Port, über den die Kommunikation für diese Anwendung ausgeführt werden soll. Diesen Port haben Sie bei der Einrichtung des zu dem Anwendungsserver gehörenden Web-Servers definiert.

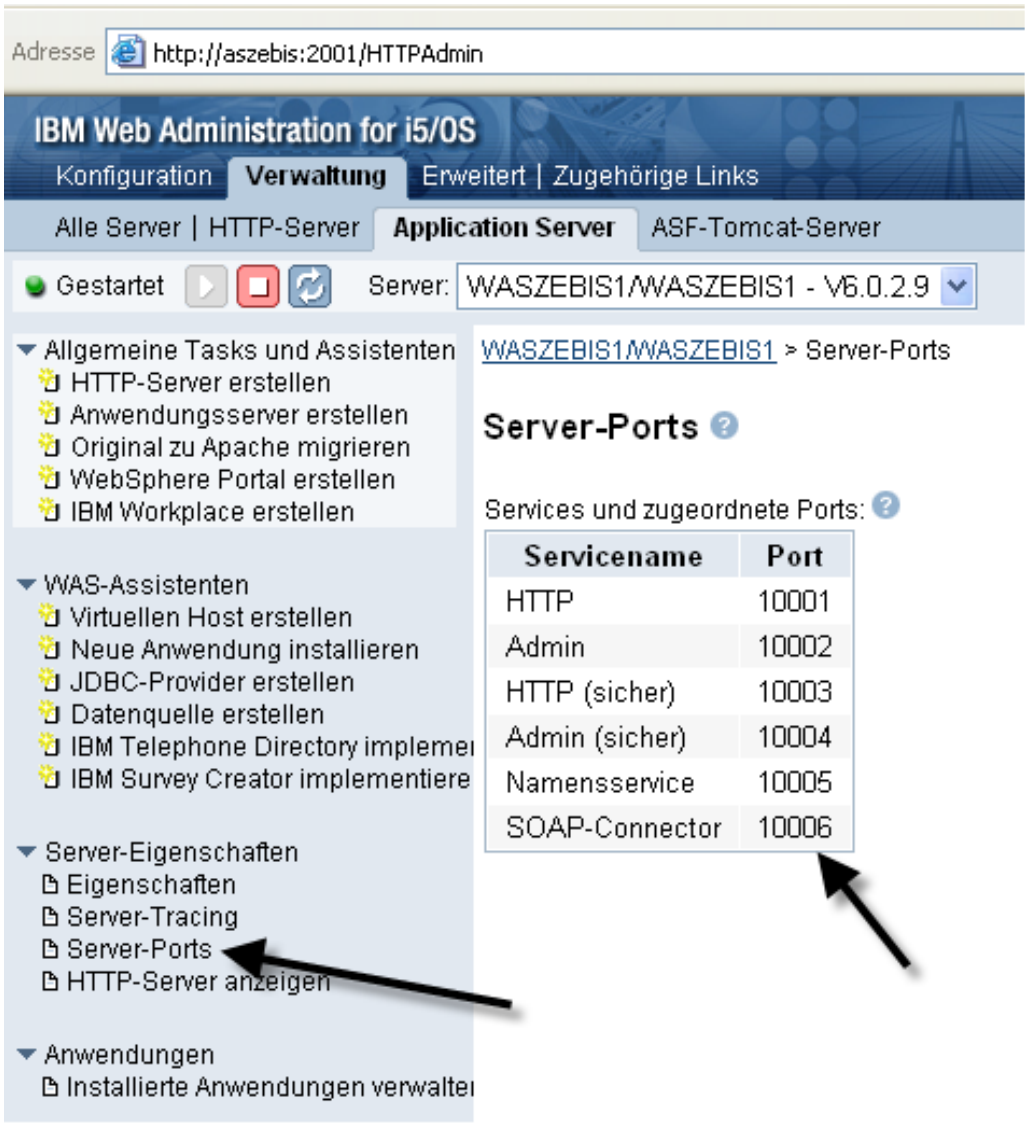
Falls Sie diese Werte nicht mehr wissen sollten: Sie lassen sich recht einfach mit Hilfe des WebSphere Administrators ermitteln. In dem Bereich der Verwaltung des dem WAS zugeordneten Web-Servers finden wir die Ports, über welche die Kommunikation ausgeführt wird.

5.2.1.4
Seite 2



Port-Angaben

Eine weitere Möglichkeit, die verwendeten Ports zu ermitteln, basiert auf dem Administrationsbereich des WebSphere Application Servers. Dort finden wir in dem linken Navigationsbereich den Abschnitt „Server Ports“, in dem die bei der Erstellung definierten Ports bearbeitet werden können.



Port-Angaben im WebSphere Bereich

Tipp

Wir finden in der Port-Auflistung unter anderem auch sichere Ports. Mit diesen Ports lassen sich sichere Verbindungen aufbauen. In unserem Beispiel kann also über den Port 10003 eine sichere http-Verbindung aufgebaut werden!

Neben der IP-Adresse und dem Port, über den die Kommunikation ausgeführt werden soll, brauchen wir als weitere Komponente das Verzeichnis, in dem wir die Anwendung ehemals installiert haben. Um dieses zu ermitteln, starten wir den WebSphere Administrationsbereich und klicken in dem linken Navigationsbereich auf den Eintrag „Installierte Anwendungen verwalten“ wie es die folgende Abbildung zeigt:

5.2.1.4
Seite 4

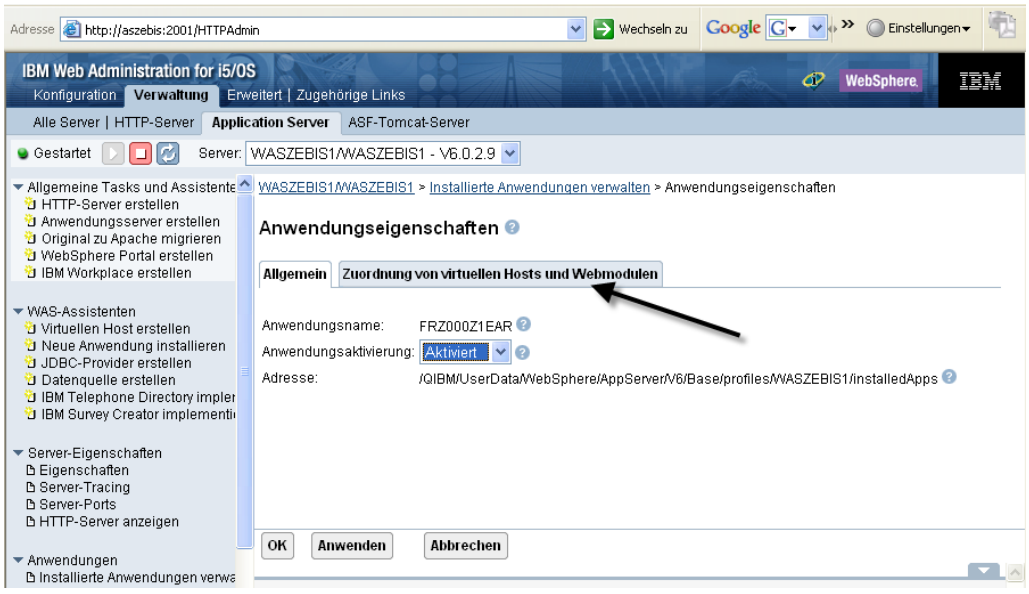
Bereich installierte Anwendungen

Nach der Auswahl erhalten wir eine Auflistung aller auf dem Anwendungsserver installierten Anwendungen.

Liste der installierten Anwendungen

In der Auflistung wählen wir die gewünschte Anwendung mit einem Mausklick aus und selektieren anschließend die unterhalb der Auflistung befindliche Schaltfläche „Eigenschaften“.

Mit dieser Aktion gelangen wir in die „Eigenschaften der Anwendung“.



Eigenschaften der Anwendung

Auch diese Anzeige ist, wie viele andere Bereiche innerhalb des WebSphere Administrators auch, in Tabellenabschnitte unterteilt. Wir wählen den Tabellenbereich „Zuordnung von virtuellen Hosts und Webmodulen“ aus. Darin befindet sich unter anderem der Eintrag „Kontextroot“ – die Information, welche wir benötigen!

5.2.1.4
Seite 6

Kontextroot

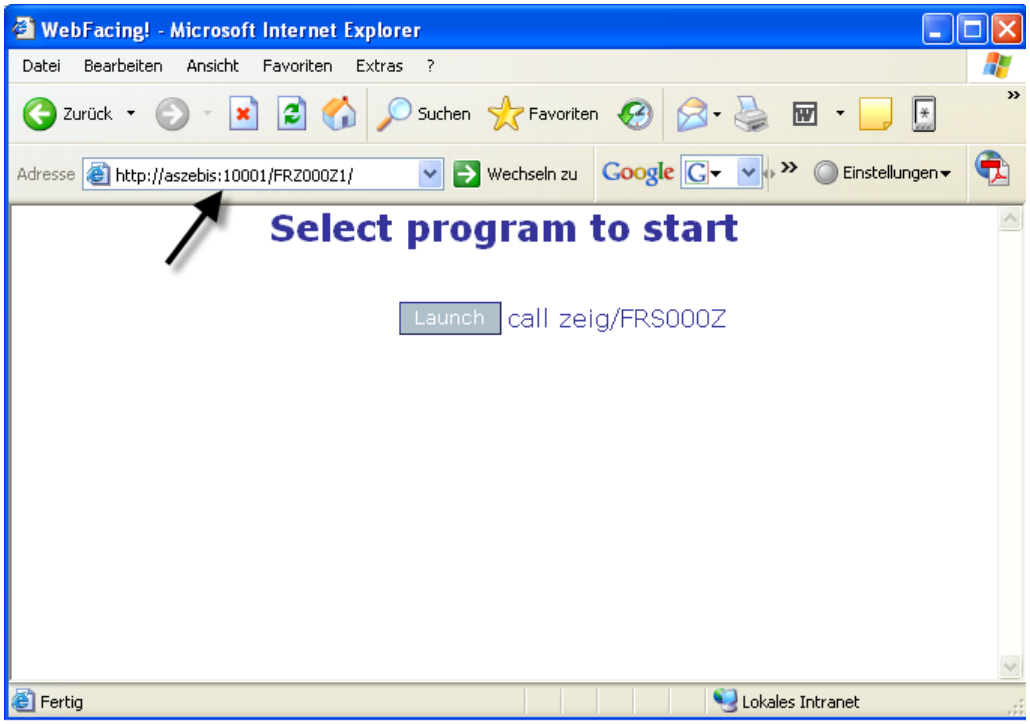
Neben der Angabe der URL finden wir in dieser Anzeige noch einige weitere, wichtige Informationen, wie das Webmodul, den virtuellen Host, die unter Umständen auch für die weitere Administration von Bedeutung sind.

Für unsere Zwecke genügt aber an dieser Stelle die Information des Kontextroots.

Diese Fragmente in Form der Hostangaben, der Portnummer und der Kontextroot sind die Bestandteile, mit denen wir die URL einer Anwendung bestimmen bzw. erstellen. Für unsere Beispielanwendung lautet die URL folglich:

http://hostname:10001/FRZ000Z1

Testen wir doch einfach einmal diese ermittelten Informationen, indem wir diese in einem Browser in dem Eingabebereich „Adresse“ eingeben und die Eingabe mit der Entertaste bestätigen – das Ergebnis sollte die Startseite der gewünschten Anwendung sein!



Startseite der Anwendung

Damit haben wir die URL-Informationen zu einer Anwendung erfolgreich ermittelt.

Sollten Sie sich jetzt fragen, wozu ich Ihnen diese „Technik“ gezeigt habe, dann lautet die Antwort: Für den Alltag eines Anwendungsentwicklers und sicher auch für die Arbeit des Administrators, der sich um die WebSphere Installation kümmern muss. In der Praxis brauchen wir diese Arbeitsschritte zum Beispiel wenn es darum geht, installierte Anwendungen zu testen, zu debuggen oder in Bezug auf deren Installation anzupassen.

Bei der Anwendung, die wir in dem vorhergehenden Beispiel verwendet haben, handelt es sich um eine WebFacing-Anwendung, die auf einem WAS installiert wurde. Diese WebFacing-Oberfläche ist, Sie wissen es sicher, eine Möglichkeit, bestehende 5250-Anwendungen mit einer grafischen Oberfläche zu versehen und somit einen Ersatz für die grüne Oberfläche zu bieten. Das Interessante bei WebFacing: es ist einfach anzuwenden; die Anwendungslogik (also das RPG oder Cobol Programm) wird NICHT verändert; der Parallelbetrieb der bisherigen 5250-Oberfläche und der Browseroberfläche ist problemlos möglich!

5.2.1.4
Seite 8

WebFacing-Oberfläche

WebSphere Application Server

IBM hat eine Vielzahl von unterschiedlichen Paketen im Umfeld der Anwendungs-Server im Portfolio.

Neben dem in den vorhergehenden Kapiteln vorgestellten WebSphere Application Server Express, der als Einstiegsversion in die Welt der Application-Server zu sehen ist, bietet der „große" Application Server in Form des WebSphere Application Server in dessen unterschiedlichen Varianten die breiteste Basis für mögliche Anwendungen.

War die Installation eines WebSphere Application Server (kurz: WAS) noch bis vor wenigen Jahren ein äußerst komplizierter Prozess, so hat sich das in den letzten Versionen stark geändert.

IBM hat viel an der Vereinfachung des Installations-Verfahrens getan. Unter anderem wird der Installationsprozess wesentlich durch die Verfügbarkeit des weitestgehend automatisierten, grafisch gestützten Launchpad geprägt.

Das Launchpad ist eine Java-basierte Anwendung, mit der weitestgehend plattformunabhängig die Basisinstallation des WebSphere Application Server und aller damit unmittelbar zusammenhängenden Schritte, wie zum Beispiel das Erstellen eines Web-Server oder des für die Ausführung des Anwendungs-Server erforderlichen Profils möglich ist.

Neben den Installationsroutinen beinhaltet das Launchpad außerdem die Basis-Dokumentation, die ausführlich den Installationsprozess begleitet. Durch das einheitliche Erscheinungsbild des Launchpad wird die Installation des WAS auf den verschiedenen Plattformen nahezu gleich ausgeführt.

IBM bietet den WAS mittlerweile für die gängisten Plattformen – beginnend mit Windows-basierten Systemen, über Linux, AIX bis hin zur iSeries, um nur einige zu nennen. Für diese Plattformen bietet das Launchpad einen einheitlichen Einstieg in die Arbeit mit WAS.

Natürlich gilt es vor einer Installation zunächst die Systemvoraussetzungen zu prüfen. Diese variieren je nach verwendeter Plattform und auch in Abhängigkeit der zu installierenden WAS-Version.

In den nächsten Kapiteln möchte ich Ihnen die Installation des WAS auf i5/OS Basis vorstellen.

Zwischen den unterschiedlichen Paketen, welche IBM im Zusammenhang mit dem WebSphere Application Server anbietet (gemeint sind hier zum Beispiel die „Base-Edition", der WAS Network Deployment) bestehen in der Regel nur geringe Abweichungen in dem Installationsprozess. Die wesentlichen Elemente der Installation sind bei den meisten WAS-Paketen identisch.

5.2.2
Seite 2

V6 WAS Installation unter i5/OS

IBM bietet neben den zuvor bereits erläuterten Express-Versionen des Anwendungs-Server auch die „große" Version in Form der verschiedenen Ausführungen des WebSphere Application Server (kurz: WAS) an.

Dabei steht die Weiterentwicklung dieser Familie weit vorn in den Entwicklungsplänen der IBM. Derzeit (Stand Mitte 2005) ist die Version 6 des WAS aktuell. Diese beinhaltet zu den Vorgängerversionen 5, 5.1 zum Teil erhebliche Verbesserungen in der Ausführungsgeschwindigkeit und bietet zum Beispiel im Einsatz mit dem WebSphere Portal Server wesentliche Funktionen für die Zentralisierung der Anwendungen.

IBM präsentiert die iSeries gerne als das Paradebeispiel für die „Systemintegration". In diesem Zusammenhang wurde unter anderem auch der WAS nativ unter dem Betriebssystem der iSeries – dem OS/400, oder genauer, dem i5/OS verfügbar gemacht.

Bei der Installation des WAS nativ auf einer iSeries gilt insbesondere, die Systemvoraussetzungen genauestens zu prüfen – denn leider eignet sich nicht jedes iSeries-System für die Installation bzw. die später zufrieden stellende Ausführung eines WAS. Hier kommt es im Wesentlichen auf das Zusammenspiel mehrerer entscheidender Komponenten wie zum Beispiel den Prozessortyp und den zugeordneten Hauptspeicher, und der schon vorhandenen Auslastung des Systems durch bereits installierte Anwendungen, an.

Derzeit (Stand WAS 6.0) werden offiziell die folgenden iSeries Prozessoren unterstützt:

- 170 ab Prozessor 2385
- 270 ab Prozessor 2252
- 720 ab Prozessor 2062
- 800 ab Prozessor 2464
- 820 ab Prozessor 2395
- 810 (alle Prozessoren)
- 825 (alle Prozessoren)
- 830 (alle Prozessoren)
- 840 (alle Prozessoren)
- 870 (alle Prozessoren)
- 890 (alle Prozessoren)
- 520 (alle Prozessoren)
- 570 (alle Prozessoren)

5.2.2.1
Seite 2

Der für die Installation erforderliche Plattenplatz kann natürlich nur für die Basis-Installation angegeben werden. Es hängt davon ab, welche Anwendungen auf dem Anwendungs-Server installiert werden. Mit der Anzahl und der Komplexität dieser Anwendungen variiert der gesamt erforderliche Plattenplatz.

Als grobe Faustregel gilt bei der Installation des WAS in der Version 6, dass auf der iSeries mindestens 1,5 GB Plattenplatz frei sein müssen.

Da im Zuge der Basis-Installation des WAS auch die Installation von einigen Beispielanwendungen angeboten wird, kann der Plattenplatz durch das Nicht-Installieren der Beispiele reduziert werden. Diese sind für das Arbeiten mit den WAS nicht erforderlich – für einen ersten Einstieg allerdings durchaus zu empfehlen!

Auch im Hinblick auf das installierte Betriebssystem, die installierten Lizenzprogramme und die erforderlichen PTFs müssen bestimmte Voraussetzungen erfüllt sein.

Der WAS 6.0 benötigt mindestens OS/400 Version V5R2M0 und die nachfolgenden Lizenzprogramme:

- 5722-SS1 Option 12 (Host Servers)
- 5722-SS1 Option 30 (Qshell)
- 5722-JV1 Option 6 (IBM Developer Kit for Java 1.4)
- 5722-AC3 (Crypto Access Provider 128-bit)

Die installierten Lizenzprogramme können wir mit Hilfe des Befehls DSPSFWRSC anzeigen.

```
_                          Software-Ressourcen anzeigen
                                                          System:   ASDEMO

 Ressour.   Option  Feature  Beschreibung
 5722SS1    7         2929    OS/400 - Bibliothek mit Beispielen
 5722SS1    8         5050    OS/400 - AFP-verträgliche Schriftarten
 5722SS1    9         5050    OS/400 - *PRV CL-Compiler-Unterstützung
 5722SS1    9         2929    OS/400 - *PRV CL-Compiler-Unterstützung
 5722SS1    12        5050    OS/400 - Host-Server
 5722SS1    12        2929    OS/400 - Host-Server
 5722SS1    13        5050    OS/400 - System Openness Includes
 5722SS1    14        5050    OS/400 - GDDM
 5722SS1    14        2929    OS/400 - GDDM
 5722SS1    18        5103    OS/400 - Media and Storage Extensions
 5722SS1    21        5050    OS/400 - Erw. Unterstützung f. Landessprachen
 5722SS1    22        5050    OS/400 - ObjectConnect
 5722SS1    22        2929    OS/400 - ObjectConnect
 5722SS1    25        5109    OS/400 - NetWare Enhanced Integration
                                                                  Weitere ...
 Eingabetaste --> Weiter

 F3=Verlassen   F11=Bibliotheken/Releases anzeigen   F12=Abbrechen
 F19=Marken anzeigen
```

DSPSFWRSC

Der WAS 6.0 wird auf der iSeries in Form eines zusätzlichen Lizenzprogramms 5733-W60 installiert.

Für die Installation ist unbedingt darauf zu achten, dass die korrekte Installations-CD für die OS/400-Version des WAS verwendet wird. Nutzen Sie für diese Installation auf keinen Fall eine Installations-CD für eine andere Betriebssystemversion!

Die für die Installation erforderlichen PTFs ändern sich relativ häufig. Deshalb verzichte ich an dieser Stelle auf eine Aufzählung der zu installierenden PTFs, sondern verweise auf die Web-Seite der IBM, welche die aktuellen PTF-Informationen für WebSphere beinhaltet:

http://www-1.ibm.com/servers/eserver/iseries/software/websphere/wsappserver/services/service.htm

Für die Installation muss das Benutzerprofil, mit dem WAS installiert wird, über ausreichende Berechtigungen verfügen (das Benutzerprofil muss mindestens die Berechtigung *ALLOBJ und *SECADM haben!). Hier empfiehlt sich die Verwendung des Benutzerprofils QSECOFR oder ein vergleichbares Profil!

5.2.2.1
Seite 4

Je nach ausgewähltem Installationsverfahren kann für die Installation des WAS 6.0 ein PC erforderlich sein – IBM stellt auch für die Installation auf der iSeries das im Zuge der Installation unter Windows erklärte „Launchpad" zur Verfügung. Wenn die Installation mit Hilfe der von IBM ausgelieferten CD ausgeführt werden soll, ist das Vorhandensein eines CD/DVD-Laufwerkes erforderlich.

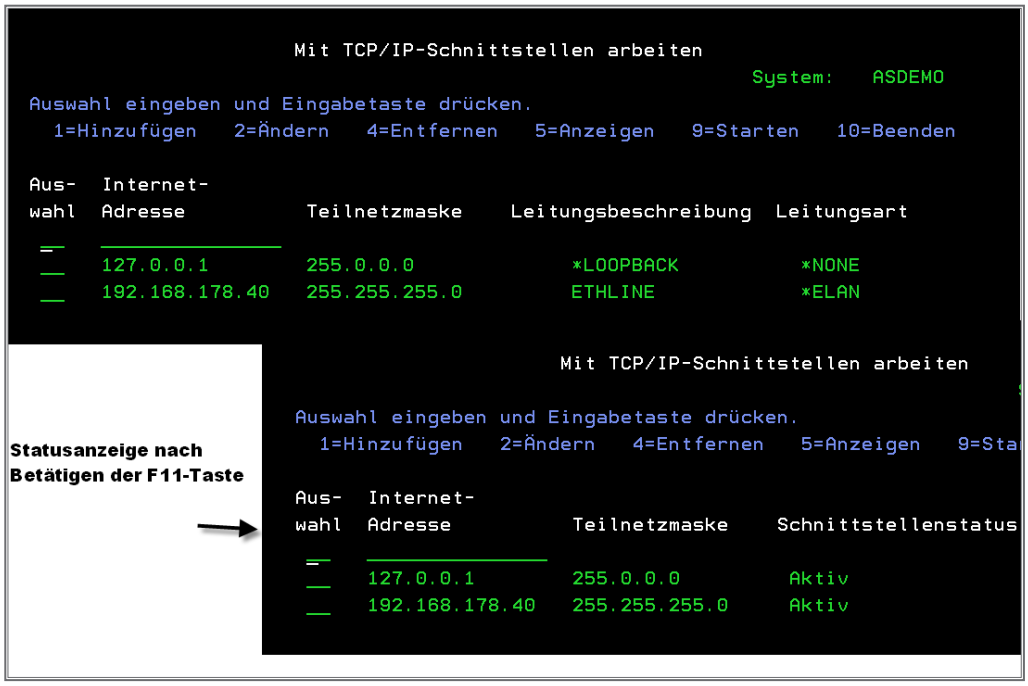
Da während der Installation einige Daten entpackt und auf dem PC zwischengespeichert werden, sollte der PC über mindestens 150 MB freien Plattenplatz verfügen. Außerdem benötigen wir für den Installationsprozess auch ein Minimum an Hauptspeicher – IBM nennt hier mindestens 256 MB.

Natürlich muss für die Installation eine Verbindung zur iSeries bestehen. Stellen Sie vor der Installation sicher, dass Sie über die korrekte IP-Adresse bzw. den Host-Namen der iSeries verfügen und das die für die Kommunikation erforderlichen Dienste (TCP/IP und Host-Server) auf der iSeries gestartet sind.

Außerdem muss TCP/IP korrekt konfiguriert sein. Dazu prüfen wir zunächst die TCP/IP-Einstellungen. Auf diese können wir aus dem Menü „TCP/IP konfigurieren" zugreifen, in das wir mit dem Befehl „GO CFGTCP" gelangen.

Die Auswahl 1 „Mit TCP/IP Schnittstellen arbeiten" beinhaltet die aktuelle TCP/IP-Leitungsdefinitionen.

In diesem Bereich muss sich unter anderem auch ein Eintrag „LOOPBACK" befinden. Der auf die IP-Adresse 127.0.0.1 verweist. Dieser Eintrag muss den Status „gestartet" aufweisen. (Den Status können Sie sich mit der F11-Taste anzeigen lassen).



TCPIP Status

Stellen Sie in diesem Bereich sicher, dass die IP-Schnittstelle, die Sie verwenden wollen, gestartet ist (Schnittstellenstatus „aktiv“).

Für die Installation empfiehlt sich das Starten aller Host-Server mit dem Befehl:

```
STRHOSTSVR SERVER(*ALL)
```

Wenn all diese Voraussetzungen gegeben sind, dann steht der Installation des WAS 6.0 auf der iSeries nichts mehr im Wege.

In unserem Beispiel installieren wir den WAS6.0 auf einer iSeries Mod. 820. Die erforderlichen Hardware- und Softwarevoraussetzungen wurden vor der Installation geprüft.

Wir werden die Installation mit dem Benutzerprofil QSECOFR durchführen, um mögliche Berechtigungsprobleme auszuschließen.

5.2.2.1
Seite 6

Für die Installation des WAS 6.0 auf der iSeries stehen verschiedene Möglichkeiten zur Auswahl:

- Installation mit Hilfe des iSeries internen CD/DVD-Laufwerkes und QShell-Funktionen
- Installation mit Hilfe des iSeries internen CD/DVD-Laufwerkes und Standard iSeries-Befehlen
- Installation mit Hilfe des automatischen Launchpad über einen PC

Ich würde in jedem Fall die Installation mit Hilfe des Launchpad empfehlen, die ich Ihnen im Rahmen der Installation des WAS auf einem Windows-basierten System beschrieben habe, denn dieses Verfahren führt uns durch den gesamten Installationsprozess.

Da sich die Verwendung des Launchpad nicht wesentlich zu der Installation des WAS auf einem Windows System unterscheidet, können Sie diese Anleitung verwenden.

Dem 5250-Fan möchte ich hier zeigen, wie die Installation des WAS 6.0 auch zunächst ohne grafische Unterstützung des Launchpad durchgeführt werden kann.

Wir wählen also für unsere Installation das iSeries interne CD-Laufwerk und das Starten über eine 5250-Emulation und QShell.

Für einige Schritte werde ich in dem nächsten Installationsverfahren allerdings doch ein wenig auf „grafische“ Hilfsmittel zurückgreifen, indem ich den iSeries Navigator für den einen oder anderen Schritt verwende.

Ausführungsschritte

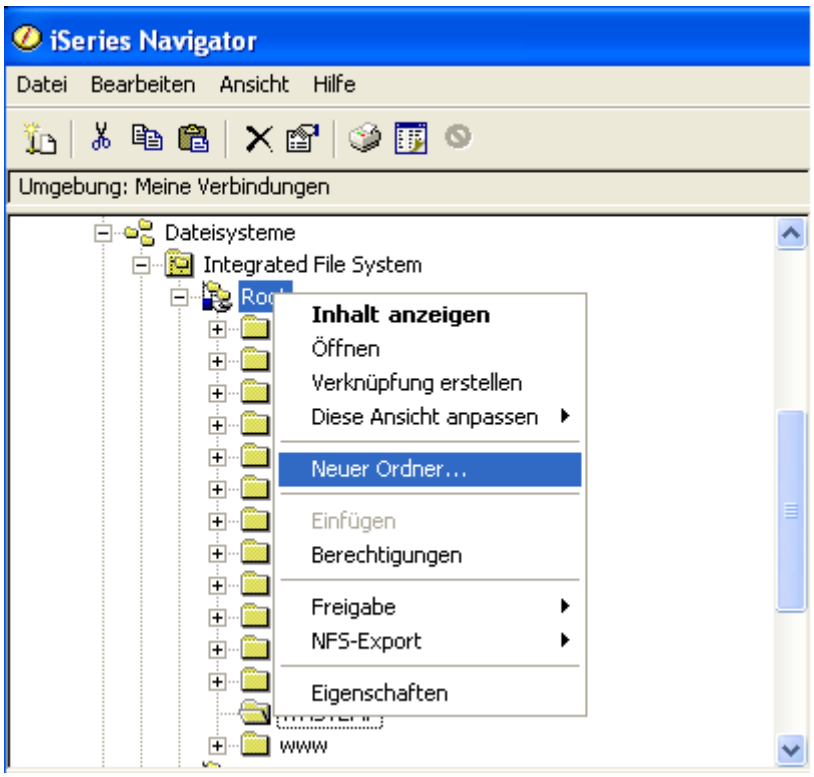
1. Legen Sie die WAS 6.0 Installations-CD in das CD-Laufwerk der iSeries ein.
2. Starten Sie die erforderlichen Host-Server – IBM empfiehlt für die Dauer des Installationsprozesses, alle Host-Server zu starten. Wir geben dazu den folgenden Befehl ein:

```
STRHOSTSVR SERVER(*ALL)
```

(Denken Sie daran, die für den Tagesbetrieb nicht erforderlichen Host-Server nach der Installation wieder zu beenden. Sollten Sie per Standard jeden Tag alle Host-Server starten, dann kann das unter Umständen ein Sicherheitsproblem darstellen!)

3. Für die Installation benötigen wir zunächst ein Verzeichnis, in das während des Installationsprozesses Informationen geschrieben werden können. Dieses Verzeichnis lege ich in unserem Beispiel in der Root der iSeries an. Bei Sicherheitsbedenken können Sie natürlich jedes anderes Verzeichnis als Speicherort auswählen.

   Für diesen Teilschritt verwende ich zunächst den iSeries Navigator. Nachdem ich dort eine Verbindung zur iSeries hergestellt habe, wähle ich den Eintrag „Dateisystem -> Integrated File System -> Root“ mit der rechten Maustaste aus und selektiere die Option „neuer Ordner“.



Verzeichnis erstellen

   Für unser Beispiel gebe ich dem Ordner den Namen „WASTEMP“.

4. Nachdem der Ordner erstellt wurde, wechsle ich in eine 5250-Emulation.

   Bevor die eigentliche Installation gestartet werden kann, müssen wir noch einige Vorarbeiten leisten. Diese Schritte sind auch bei der Verwendung des Launchpad erforderlich, lassen sich dort allerdings wesentlich einfacher – ich finde auch eleganter – ausführen.

5.2.2.1
Seite 8

5. Zunächst müssen wir die Lizenzbestimmungen akzeptieren. IBM verwendet an dieser Stelle eine etwas außergewöhnliche Methode. Auf der Installations-CD befindet sich eine Datei „Responsefile“, die wir zunächst in das zuvor erstellte Verzeichnis der iSeries übertragen müssen.

   Diese Übertragung führen wir mit dem Befehl „CPY“ aus, wie im Folgenden gezeigt wird.

   Der Parameter „TODIR“ ist dabei mit dem Verzeichnis zu füllen, das wir für die Installation des WAS definiert haben – in unserem Beispiel ist dies das Verzeichnis „WASTEMP“.

```
Auswahl oder Befehl
===> CPY OBJ('/QOPT/WEBSPHERE/WAS/RESPONSEFILE') TODIR('/WASTEMP')
```

Responsefile kopieren

6. Die soeben kopierte Datei müssen wir jetzt bearbeiten. Dazu verwenden wir innerhalb der 5250-Emulation den Befehl „WRKLNK“. Um zunächst einen Zugriff auf alle Objekte zu erhalten, geben wir als „Objekt“ einen Asterix (*) ein.

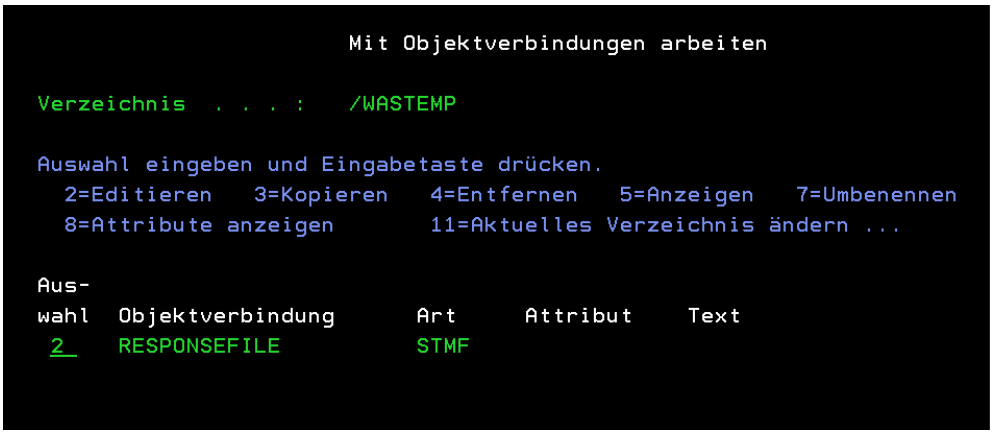
```
                    Mit Objektverbindungen arbeit. (WRKLNK)

Auswahl eingeben und Eingabetaste drücken.

Objekt . . . . . .   '*'
_____
Objektart  . . . . . .   *ALL       *ALL, *ALLDIR, *ALRTBL...
Detail . . . . . .   *PRV       *PRV, *NAME, *BASIC...
Auswahl anzeigen . . . . . .   *PRV       *PRV, *USER, *ALL
```

WRKLNK

7. In der nächsten Auflistung sollte unter anderem auch das zuvor angelegte Verzeichnis „WASTEMP“ enthalten sein. Dieses wählen wir mit der Option „5“ zur weiteren Anzeige aus.

WRKLNK Auswahl auf WASTEMP

8. In dem Verzeichnis befindet sich derzeit ein Eintrag – die Datei „Responsefile“, deren Inhalt wir anpassen müssen. Also wählen wir hier die Option „2“ für das Ändern der Datei aus.



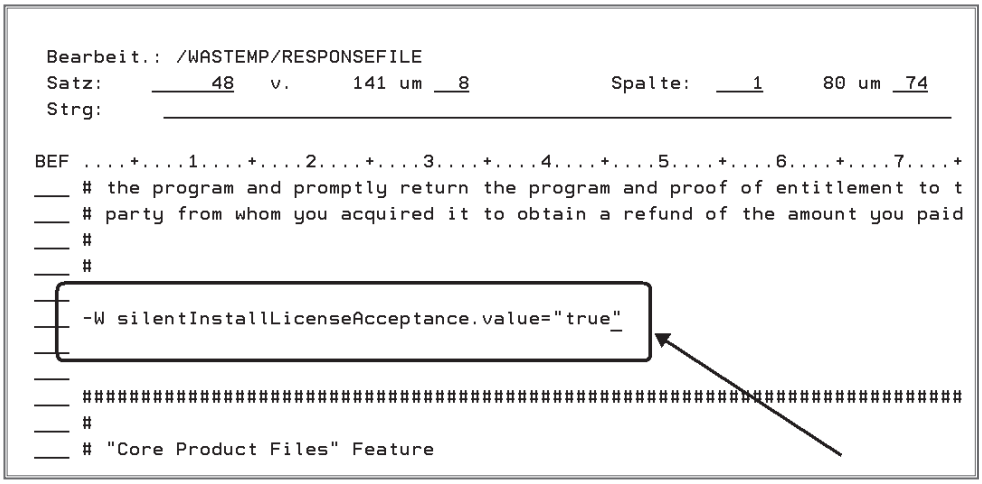
Responsefile editieren

9. Neben einigen anderen Informationen beinhaltet die Datei „Responsefile“ unter anderem auch die Bestätigung des Akzeptierens der Lizenzvereinbarungen. Diese befinden sich in dem Verzeichnis „/QOPT/WEBSPHERE/WAS/DOCS/lafiles“ und können dort eingesehen werden.

   Für die Installation des WAS ist das Akzeptieren der Lizenzbestimmungen unbedingt erforderlich. Dazu müssen wir in der Datei „Responsefile“ den Eintrag „silentInstallLicenseAcceptance.value“ von dem Auslieferungswert „false“ auf den neuen Wert „true“ ändern, was gleichbedeutend mit „Ich akzeptiere die Lizenzbestimmungen“ ist.

5.2.2.1
Seite 10

Ich blättere dazu in der Datei „Responsefile“ bis an die Stelle, wo ich die Lizenzvereinbarungen akzeptieren kann und verändere den Wert „false“ in „true“, wie in der folgenden Abbildung gezeigt.



Lizenzvereinbarung gelesen – true

Nach dem Ändern des Wertes auf „true“ verlassen wir mit der F3 -Taste den Editierbereich. Achten Sie darauf, dass die Änderungen gespeichert werden!

10. Nachdem diese vorbereitende Tätigkeit durchgeführt wurde, kann die Basis-Installation des WAS 6.0 beginnen.

    Der Installationsprozess kann mit einem QShell-Fenster ausgeführt werden. Diesen Bereich starten wir mit dem Befehl „STRQSH“, den wir in der Befehlszeile der 5250-Emulation eingeben.

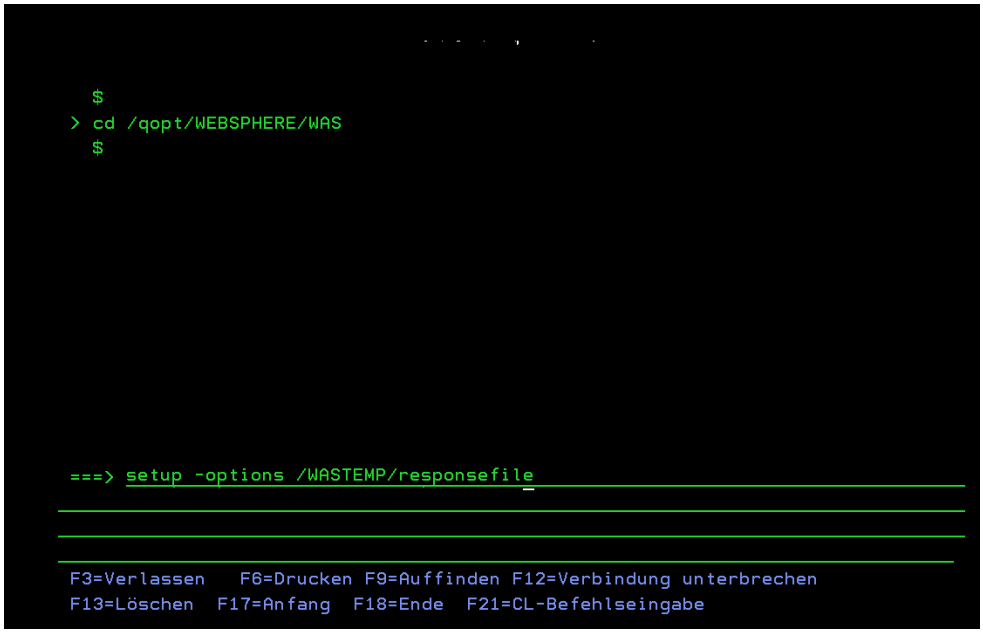
    In dem QShell-Fenster wechseln wir jetzt in das Installationsverzeichnis auf der CD. Dazu geben wir den folgenden Befehl ein:

```
cd /QOPT/WEBSPHERE/WAS
```

Nachdem wir in das Installationsverzeichnis der CD gewechselt sind, können wir die Setup-Prozedur für die WAS-Installation starten. Dazu geben wir den Startbefehl wie folgt ein:

```
SETUP -options /WASTEMP/responsefile
```

(Sollten Sie einen anderen Verzeichnisnamen verwendet haben, dann müssen Sie „WASTEMP“ durch den von Ihnen angegebenen Namen ersetzen!)

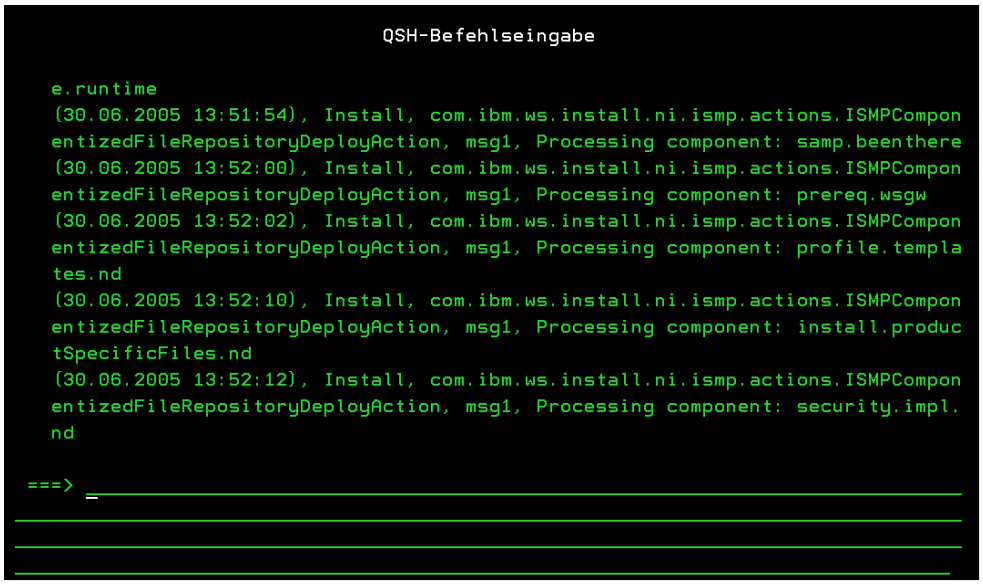


QShell Start der Installation

Jetzt heißt es: Warten!!!

In Abhängigkeit der Maschine kann dieser Schritt 90 Minuten und länger dauern! Durch den Aufruf „Setup“ wird jetzt ein WAS 6.0 Server auf der iSeries installiert. Dabei wird eine Vielzahl von Dateien entpackt und mit Hilfe von Java-Installationsroutinen auf dem System installiert.

Der jeweils aktuelle Installationsschritt wird durch eine Aktualisierung der Anzeige in dem QShell-Fenster mitgeteilt.

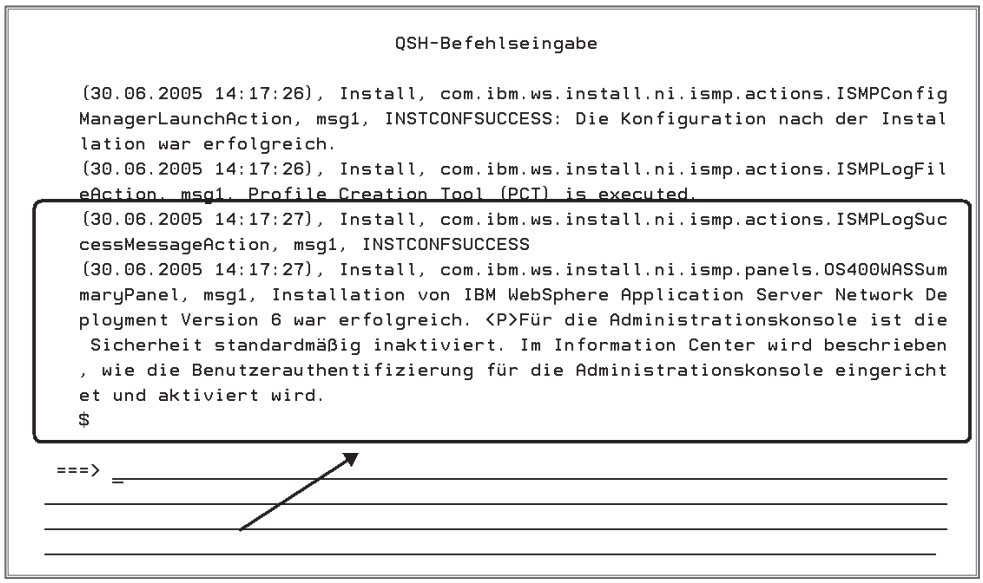


Verarbeitungsstatus

5.2.2.1
Seite 12

Während des Installationsprozesses empfiehlt es sich nicht, weitere Befehle in dem QShell-Fenster einzugeben. Achten Sie auch darauf, dass QShell-Fenster nicht etwa mit der Befehlstaste „F3“ oder „F12“ zu verlassen – denn damit würde die Installation abgebrochen werden!

Nach einer Weile sollte die Meldung erscheinen, die besagt, dass die Installation des WAS erfolgreich abgeschlossen werden konnte. Ein Beispiel für eine solche Meldung sehen Sie in der nächsten Abbildung.



Erfolgsmeldung

11. Aus Sicherheitsgründen empfiehlt es sich jetzt, alle „überflüssigen“ Host-Server zu beenden, die wir zuvor für die Installation des WAS gestartet haben. Anderenfalls wären unter Umständen „Alle Tore“ der iSeries für einen externen Zugriff geöffnet. Dazu können wir den Befehl „END-HOSTSVR“ verwenden – achten Sie aber bitte unbedingt darauf, nicht für den Normalbetrieb erforderliche Dienste „aus Versehen“ zu beenden.
12. Zum Abschluss kontrollieren wir die Auflistung der auf unserem System installierten Lizenzprogramme. Hierzu geben wir erneut den Befehl „DSPSFWRSC“ ein und blättern bis zu dem Eintrag „5733W60“ – der Lizenzprogrammnummer für den WAS 6.0.

```
                        Software-Ressourcen anzeigen
                                                          System:   ASDEMO

Ressour.   Option   Feature   Beschreibung
5722XW1    1         5101     iSeries Access Option 1
5733W60    *BASE     5050     WebSphere Application Server for OS/400 V6
5733W60    3         5103     WebSphere Application Server V6 Network Deployment
5733XT1    *BASE     5050     XML Toolkit for iSeries
5733XT1    *BASE     2929     XML Toolkit for iSeries
5733XT1    1         5050     XML Parsers version 4.0
5733XT1    2         5050     XML Parsers version 4.0 development
5733XT1    3         5050     XML Toolkit Option 3 (see documentation)
5733XT1    4         5050     XML Toolkit Option 4 (see documentation)
5733XT1    5         5050     XML Toolkit Option 5 (see documentation)
```

Lizenzprogramm 5733W60

Da wir in unserem Beispiel den WAS V6 „Network Deployment“ installiert haben, befindet sich neben der Basis-Option (5733W60 Option *BASE) auch die erweiterte Option in Form des Eintrages „5733W60 Option 3“ auf dem System.

Mit diesen Schritten ist zunächst die Basis-Installation des WAS V6 abgeschlossen.

Als Nächstes müssen unbedingt die für WAS erforderlichen PTFs installiert werden.

Beachten Sie bitte, dass in der Regel jede PTF-Lieferung über eine eigene Installations-Anleitung verfügt. Diese ist in jedem Fall zu befolgen!

Wir installieren in unserem Beispiel die aktuelle PTF mit Hilfe des Menüs „Vorläufige Programmkorrekturen“ (PTF), in das wir mit dem Befehl „GO PTF“ verzweigen.

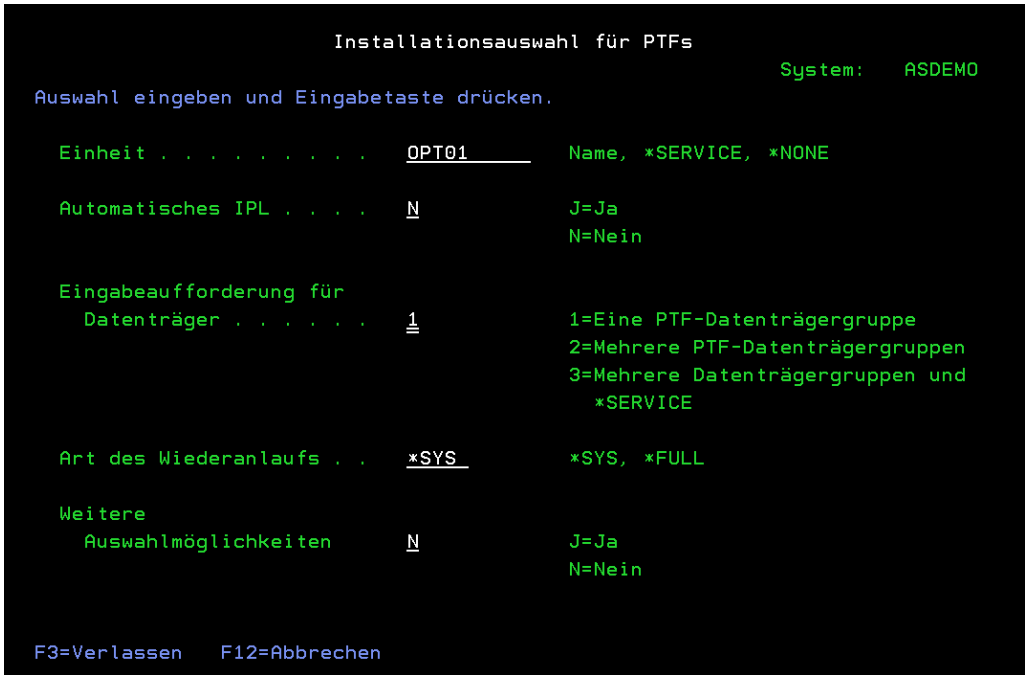
5.2.2.1
Seite 14

Nach dem Einlegen der ersten PTF-CD wählen wir die Option 8 „PTF Paket installieren“ aus.

GO PTF

Jetzt geben wir das Laufwerk an, in dem sich die PTF-CD befindet – in unserem Fall ist dies das CD-Laufwerk „OPT01“. Damit die PTF installiert werden können, muss das System neu gestartet werden. Das Einspielen der PTF kann durchaus getrennt von dem Anlegen durchgeführt werden. Dabei werden dann die PTF-Informationen auf das System kopiert und bei dem nächsten IPL aktiviert. Wir wählen dieses Verfahren aus (und denken daran, bald möglichst einen normalen IPL durchzuführen). Damit können wir die PTF im laufenden Betrieb auf das System laden, ohne etwa die Anwender in ihrer Arbeit zu stören.

Sollten Sie die Installation der PTF direkt mit einem nachfolgenden IPL abschließen wollen, dann geben Sie in den Parameter „Automatisches IPL“ ein „J“ ein – achten Sie aber darauf, dass die Maschine dann direkt nach dem Einspielen der PTF herunterfährt – das heißt, in diesem Fall sollte kein Anwender mehr auf dem System arbeiten!



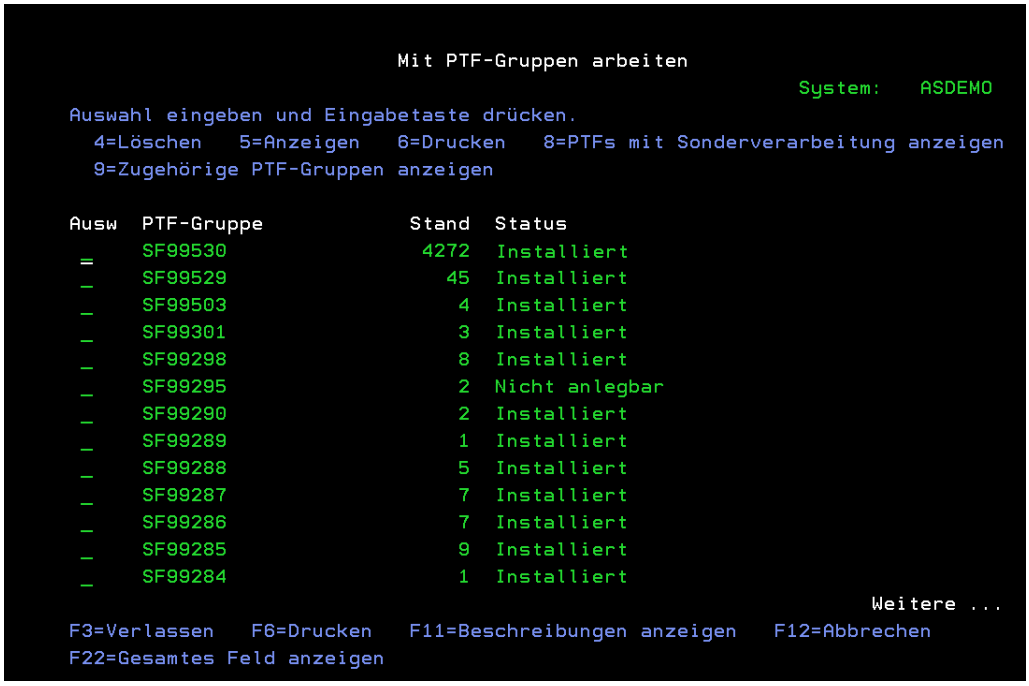
PTF installieren

Da es in der Regel eine Vielzahl zu installierender PTF gibt, kann dieser Teilabschnitt der Installation auch entsprechend lange dauern. Kalkulieren Sie also ruhig einmal eine Stunde Zeitbedarf für das Einspielen der PTFs ein.

Den aktuellen Stand der PTF – im Speziellen der Gruppen-PTF, können wir mit dem Befehl „WRKPTFGRP“ kontrollieren.

Die PTF-Gruppe für den WAS 6.0 befindet sich unter dem Eintrag „SF99301“ – diese PTF-Gruppe hat in unserem Beispiel den Stand 3.

5.2.2.1
Seite 16



WRKPTFGRP

Bevor wir jetzt die Konfiguration des WAS 6.0 fortsetzen können, müssen wir die iSeries Lizenz-Berechtigungen einstellen. Per Standard ist hier eine Benutzerlimitierung auf 0-Benutzer angegeben.

Um die mit der Lizenz erworbenen User freizuschalten, verwenden wir den Befehl WRKLICINF.

Blättern Sie in der Anzeige „Mit Lizenzberechtigungen arbeiten“ soweit, bis Sie den Eintrag „5733W60“ finden. In der Regel befinden sich in der Anzeige mehrere Einträge mit der Produktnummer 5733W60. Das hängt damit zusammen, dass automatisch mit der Installation des WAS auch der „WebSphere Application Server V6 Express“ installiert wird. In unserem Beispiel finden wir 3 Einträge, da wir für unsere Beispielinstallation den „WebSphere Application Server V6 Network Development“ installiert haben.

```
                  Mit Lizenzprogramminformationen arbeiten          ASDEMO
                                                          30.06.05  17:07:21
Systemseriennummer . . . . . . . . . . :
Prozessorgruppe  . . . . . . . . . . . :   P20

Auswahl eingeben und Eingabetaste drücken.
  1=Lizenzberechtigung hinzufügen   2=Ändern   5=Details anzeigen
  6=Details drucken   8=Mit Lizenzbenutzern arbeiten

             Lizenz-
Opt  Produkt  laufz.   Feature  Beschreibung
2_   5733W60  V6        5101    WebSphere Application Server V6 Express
__   5733W60  V6        5102    WebSphere Application Server V6 ("Basis")
__   5733W60  V6        5103    WebSphere Application Server V6 Network Deplo
```

WRKLICINF

Wir müssen für jeden dieser Einträge die Lizenzberechtigungen ändern. Dazu wählen wir die Option „2“ aus, wie in der vorhergehenden Abbildung gezeigt.

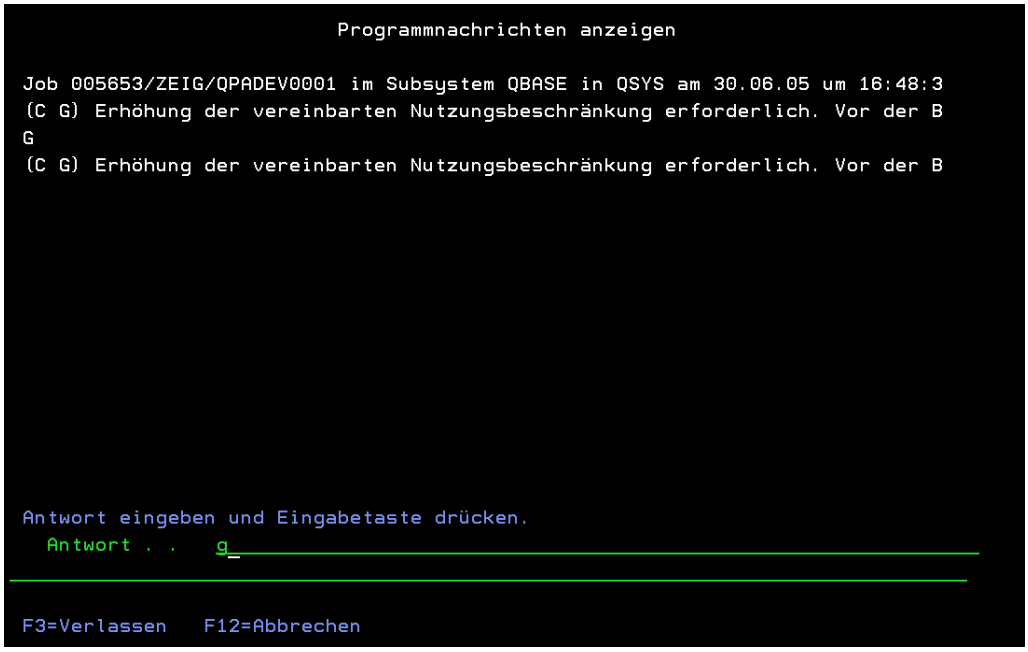
Es hängt von der von Ihnen erworbenen Lizenz ab, welche Nutzungsbeschränkung in der nächsten Anzeige eingegeben werden muss. In unserem Beispiel haben wir keine Limitierung und geben deshalb in dem Feld „Nutzungsbeschränkung“ den Wert „*Nomax“ ein.

```
                  Lizenzprogramminfo. ändern (CHGLICINF)

Auswahl eingeben und Eingabetaste drücken.

Produkt-ID . . . . . . . . . . . . > 5733W60       ID
Lizenzlaufzeit . . . . . . . . . . > V6            Vx, VxRy, VxRyMz, *ONLY
Feature  . . . . . . . . . . . . . > 5101          5001-9999
Nutzungsbeschränkung . . . . . .   *NOMAX_____     0-999999, *SAME, *NOMAX...
```

Nutzungsbeschränkung ändern

Damit wir auch wirklich sicher sind, die getätigte Änderung der Lizenzvereinbarung „bewusst“ durchgeführt zu haben, wird die anstehende Änderung nochmals mit einer Programmnachricht abgefragt – hier lassen sich Ähnlichkeiten (Wollen Sie wirklich..) zu einem anderen Betriebssystem nicht verheimlichen….

5.2.2.1
Seite 18



Programmnachrichten anzeigen

Damit sich die Änderung auch wirklich auswirken, müssen wir in dieser Anzeige die Programmnachricht mit einem „G“ beantworten. Erst dann wird die Änderung der Nutzungsbeschränkung durchgeführt.

Führen Sie diesen Schritt für alle installierten WAS-Features durch.

Danach ist die Basis-Installation des WAS abgeschlossen. Wie bereits zuvor erläutert, bedeutet der Einsatz eines Anwendungs-Server auch den Einsatz weiterer Server bzw. Schichten. Deshalb ist die Gesamtinstallation noch nicht abgeschlossen. In der Regel ist die Definition eines Web-Server für den praktischen Einsatz des WAS erforderlich. Die Installation und Konfiguration eines solchen Server werde ich Ihnen in einem weiteren Kapitel beschreiben.

V6 WAS Konfiguration unter i5/OS

Die Installation und anschließende Konfiguration eines Application-Server bezieht sich nicht nur auf den Server selbst, sondern beinhaltet noch weitere Komponenten, die für den praktischen Einsatz des Application-Server benötigt werden. Das sind unter anderem die folgenden Teilbereiche.

- Application-Server
- Administrations-Werkzeuge (z. B. Administrations-Client)
- Web-Server
- Web-Server Plug-Ins
- Java Docs

Die einzusetzenden Komponenten sind ebenso frei, wie die Kombination mehrerer Installationen auf einer physischen Maschine. So können Sie wahlweise einen einzigen WAS-Server auf einem System installieren oder bei Bedarf auch mehrere WAS-Server bzw. WAS-Instanzen auf einem physischen Rechner laufen lassen.

Im einfachsten Fall ist die Installation eines WAS als Einzelinstallation auf einem System zu sehen. In einem solchen Fall wird nur eine WAS-Instanz bzw. ein WAS-Server auf einem Rechner installiert.

Eine solche Konfiguration finden wir bei einfachen Konfigurationen bzw. auch bei Testsystemen.

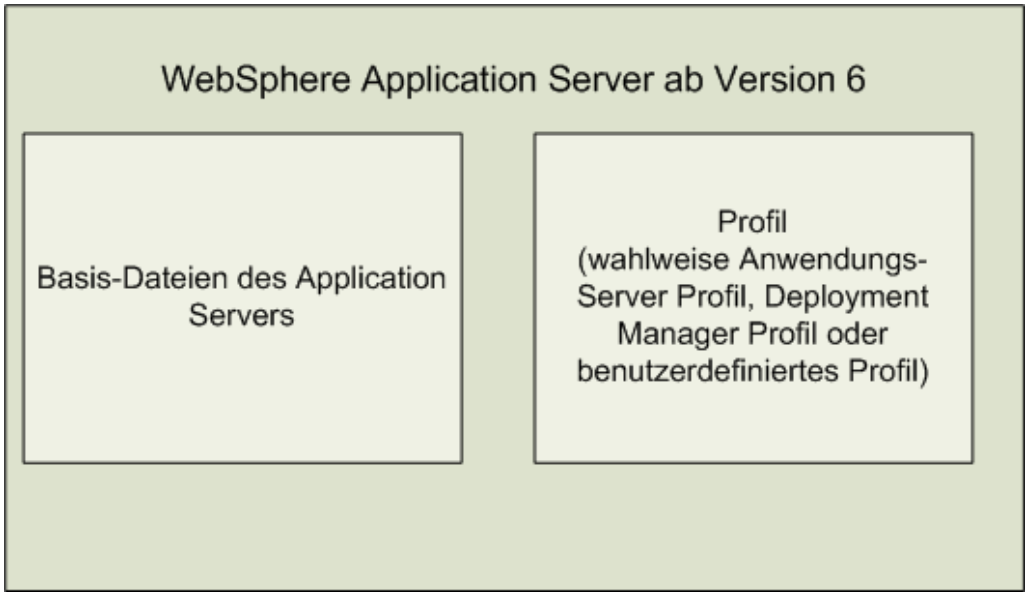
Für die Konfiguration einer solchen Umgebung genügt – in Abhängigkeit der auf dem Server auszuführenden Anwendungen – entweder die Express-Edition oder die Base-Edition. Natürlich kann auch für eine so einfache Umgebung die Network Deplyoment Edition eingesetzt werden. Das macht allerdings aus Kostengründen nur wenig Sinn. Sollten Sie jedoch die Absicht haben, nach der Installation und Konfiguration der „einfachen“ WAS-Umgebung später eine komplexere Umgebung aufbauen zu wollen, dann bietet sich natürlich für so einen Fall auch direkt die Verwendung der Network Deployment-Variante an.

Die nachfolgende Abbildung zeigt den Zusammenhang eines Profils, das neben der Basis-Installation eines WAS ein fixer Bestandteil einer komplexen WAS-Installation darstellt.

Anmerkung:

Die Basis-Installation ist in dem Kapitel 5.2.2.1 beschrieben.

5.2.2.2
Seite 2



Aufbau des WAS ab der Version 6

Mehr zu den Profilen finden Sie in dem Kapitel „5.2.9.2 Begriffsdefinitionen Application-Server".

Jeder Application-Server benötigt für die Kommunikation mit einem Client einen Web-Server. Dies ist in der Regel ein externer Web-Server, der zum Beispiel in Form des http-Server powered by Apache zur Verfügung steht. Für Testzwecke kann auf die Installation eines solchen zusätzlichen Web-Server verzichtet werden. Denn IBM liefert in den WAS integriert einen „kleinen" Web-Server mit, welcher sicherlich nicht für einen Produktiv-Einsatz geeignet ist, aber für Testszenarien durchaus in seiner Leistungsfähigkeit ausreicht. Allerdings: Häufig ist aber gerade auch das Zusammenspiel zwischen Web-Server und Application-Server in Bezug auf die Konfiguration von Bedeutung. Deshalb gilt es als Empfehlung, dass für den Test einer Anwendung für eine Produktionsfreigabe immer das Szenario gewählt wird, welches auch später in der Praxis vorgefunden wird. Das heißt in diesem Fall: der Einsatz eines externen Web-Server.

In diesem Kapitel werden wir uns mit der Konfiguration einer WAS-Umgebung beschäftigen. Dabei verwenden wir die in dem Kapitel „5.2.2. V6 WAS-Installation unter i5/OS“ installierten Komponenten des WAS in der Version 6.

Je nachdem, ob Sie die Konfiguration unmittelbar nach der Installation der Software durchführen wollen, und ob Sie für die Installation der Software des WAS das Launchpad – also die grafische Installations-Oberfläche – verwendet haben, erscheint automatisch am Ende des Installations-Vorgangs eine Frage, ob die Konfiguration des WAS sofort gestartet werden soll. In diesem Fall muss lediglich die entsprechende Auswahl getroffen werden.

Wollen Sie die Konfiguration des WAS unabhängig von der Installation der Software des WAS bzw. nach der Installation des WAS mit dem Befehlszeilenorientierten Installationsprozess durchführen, dann kann der Konfigurationsprozess natürlich auch manuell und unabhängig von dem Installationsprozess gestartet werden.

Betrachten Sie nochmals die Abbildung „Aufbau des WAS ab der Version 6“ zu Beginn dieses Kapitels. Dort finden Sie neben der Basis-Installation des WAS auch die administrative Einheit des WAS in Form eines Profils. Genau so ein Profil wird jetzt mit dem Konfigurationsprozess erstellt.

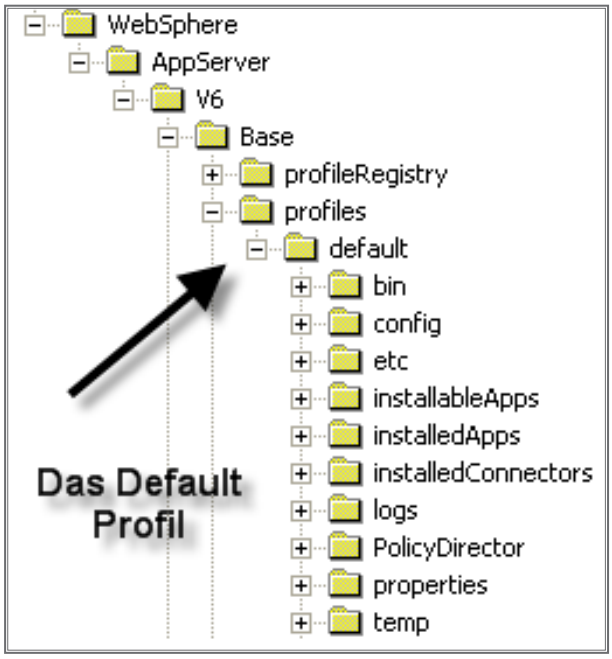
Bevor wir jedoch den Application-Server konfigurieren, müssen Sie festlegen, welche Form des Profils erstellt werden soll. Für unsere einfache Konfiguration verwenden wir das Anwendungsserver-Profil. Die Erstellung eines Deployment-Manager-Profils oder auch eines benutzerdefinierten Profils kann auf dieselbe Weise gestartet werden.

IBM bietet zwei unterschiedliche Methoden zur Profilerstellung:

1. Das Erstellen eines Profils als Hintergrund-Job. Dabei werden die erforderlichen Einstellungen mit Hilfe einer .txt-Datei beschrieben und ausgeführt.
2. Das Erstellen eines Profils mit Hilfe des grafischen Profilerstellungs-Assistenten.

Im Zuge der Installation der WAS-Software in der Version 6 wird automatisch ein Default-Profil erstellt. Ein Beispiel für solch ein Profil finden Sie in der folgenden Abbildung.

5.2.2.2
Seite 4



Das Default Profil

Diese Abbildung zeigt das Default-Profil einer Base-Edition. Ähnlich stellt sich auch die Ansicht bei der Verwendung der Network Deployment Edition dar.

Achtung

Es kann vorkommen, dass die Erstellung des Default-Profils während der Installation der WAS-Software fehlerhaft beendet wird. Deshalb sollte das Protokoll bezüglich der Erstellung des Profils überprüft werden. Dieses Protokoll trägt die Bezeichnung *wasprofile_create_default.log* und befindet sich in dem folgenden Pfad:

Base-Edition:
/QIBM/UserData/WebSphere/AppServer/V6/Base/profileRegistry/logs/wasprofile

Network Deployment Edition:
/QIBM/UserData/WebSphere/AppServer/V6/ND/profileRegistry/logs/wasprofile

In dem Protokoll sollte der Eintrag „INSTCONFSUCCESS: Erfolg: Das Profil ist jetzt vorhanden“ enthalten sein.

```
<thread>10</thread>▯  <message>Resetting the system properties to
<millis>1105277439314</millis>▯  <sequence>4860</sequence>▯  <logg
<class>com.ibm.ws.profile.cli.WSProfileCLICreateProfileInvoker</cl
<message>INSTCONFSUCCESS: Erfolg: Das Profil ist jetzt vorhanden.<
<millis>1105277439343</millis>▯  <sequence>4861</sequence>▯  <logg
<class>com.ibm.ws.profile.WSProfileCLI</class>▯  <method>invokeWSP
INSTCONFSUCCESS</message>▯</record>▯</log>▯
```

Profil wurde erfolgreich erstellt

ei der Installation der Network Deployment Edition wird neben dem Default-Profil noch ein weiteres Profil „dmgr“ erstellt. Prüfen Sie auch dessen erfolgreiche Erstellung nach dem zuvor genannten Verfahren.

Starten des WAS 6-Server

Bevor wir uns mit weiteren Konfigurations-Schritten beschäftigen, sollten wir zunächst sicherstellen, dass der WAS 6 gestartet wurde. Für die Ausführung der WebSphere-Umgebung wurde auf der i5 ein neues Subsystem „QWAS6“ generiert. In diesem Subsystem werden die aktiven Server ausgeführt. Die nachfolgende Abbildung zeigt das Subsystem und einen gestarteten Server „Server1“:

```
                        Mit aktiven Jobs arbeiten                  ASZEBIS
                                                          10.01.05  18:14:15
CPU %:    24,8     Abgelaufene Zeit:   00:00:49     Aktive Jobs:   610

Auswahl eingeben und Eingabetaste drücken.
  2=Ändern    3=Anhalten    4=Beenden    5=Arbeiten mit     6=Freigeben
  7=Nachricht anzeigen    8=Mit Spool-Dateien arbeiten   13=Unterbrechen

Ausw.  Subsystem/Job  Benutzer    Art  CPU %  Funktion        Status
 _     QWAS6          QSYS        SBS    0,0                  DEQW
 _       SERVER1      QEJBSVR     BCH   21,6  PGM-QWASSTRSVR   RUN
```

Aktiver WAS 6 Server

Ist weder das Subsystem QWAS6 noch der gewünschte Server in der Auflistung der WRKACTJOB-Einträge vorhanden, dann muss zunächst der WebSphere Application Server gestartet werden. Zuerst muss sichergestellt sein, dass das Subsystem QWAS6 in der Bibliothek QWAS6 gestartet wurde. Sie können dieses Subsystem natürlich auch mit dem Befehl STRSBS starten:

```
STRSBS SBS(QWAS6/QWAS6)
```

Anschließend muss der gewünschte Server in diesem Subsystem gestartet werden. Dazu benötigen wir entweder eine QSHell-Eingabe oder den iSeries Navigator.

5.2.2.2
Seite 6

In dem nachfolgenden Beispiel zeige ich, wie der WAS-Server mit Hilfe der Qshell Eingabe-Optionen gestartet werden kann. Dazu geben wir in der Befehlszeile der i5 den Befehl zum Starten der Qshell-Umgebung ein (STRQSH). In dem sich dann öffnenden Qshell-Befehlseingabebereich geben wir jetzt den Befehl zum Starten der Serverinstanz ein. Dieser Befehl kann in Abhängigkeit dessen, ob es sich um einen Base-Server oder einen Server in einer Network Deployment-Version handelt, durch die Angabe der jeweiligen Edition variieren. Das zeigt das nachfolgende Beispiel, in dem einmal der Server „Server1" in der Base-Edition und einmal in der Network Deployment Edition gestartet wird.

Starten des Server „Server1" in der Base-Edition:

Wechsel in das entsprechende Verzeichnis:

```
cd /QIBM/ProdData/WebSphere/AppServer/V6/base/bin
```

Starten des WAS-Server „Server1" mit dem Befehl „startServer"

```
startServer Server1
```

Starten des Server „Server1" in der Network Deployment Edition:

Wechsel in das entsprechende Verzeichnis:

```
cd /QIBM/ProdData/WebSphere/AppServer/V6/ND/bin
```

Starten des WAS-Server „Server1" mit dem Befehl „startServer"

```
startServer Server1
```

Die folgende Abbildung zeigt den Qshell-Eingabebereich und die Startanforderung für den Server „server1" in der Base-Edition.

```
> startServer server1
  CPC1221: Job 004166/QEJBSVR/SERVER1 an Jobwarteschlange QWASJOBQ in
  Bibliothek QWAS6 übergeben.
```

Starten des Server in Qshell-Befehlseingabe

Achtung

Der Name des zu startenden Server ist case-sensitiv. Achten Sie deshalb bei der Eingabe des Servernamens auf die Groß- und Kleinschreibung!

Das Starten des Server kann je nach eingesetzter Hardware und deren allgemeiner Auslastung einige Minuten dauern. Der Status des Startvorgangs kann in dem Jobprotokoll geprüft werden. Die nachfolgende Abbildung zeigt den erfolgreichen Start des Server „Server1".

```
                          Alle Nachrichten anzeigen
                                                          System:   ASZEBIS
Job . . :   SERVER1       Benutzer :   QEJBSVR       Nummer . . . :   003789

 >> CALL PGM(QWAS6/QWASSTRSVR) PARM('-profilePath' '/QIBM/UserData/WebSphere/A
    ppServer/V6/Base/profiles/default' '-server' 'server1')
    Abrechnungsdaten für 003789/QEJBSVR/SERVER1 nicht protokolliert; Ursache:
      1.
    Server wird mit Benutzerprofil QEJBSVR und JDK 1.4.2.04 gestartet.
    WebSphere-Anwendungsserver server1 ist bereit.
```

Jobprotokoll zeigt den gestarteten Server

5.2.2.2
Seite 8

Stoppen des WAS 6-Server

Neben dem Startbefehl „startServer“ gibt es noch einige weitere Befehlszeilen-Anweisungen, wie zum Beispiel den Befehl zum Stoppen des Server „stopServer“.

Bei dem Stoppen des Server sollte dieser Befehl verwendet werden. Zwar kann auch das Subsystem QWAS6 beendet werden (ENDSBS), dann besteht allerdings die Gefahr, dass nicht alle für die Beendigung des WebSphere-Bereiches erforderlichen Teilschritte korrekt abgearbeitet werden.

Da ein aktiver WebSphere Application Server aus einer Vielzahl von Teilprozessen besteht, kann es ähnlich wie bei dem Start wieder einige Minuten dauern, bis der WAS-Server mit dem Befehl „stopServer“ beendet wurde. Dieser Befehl kann ebenfalls über die Qshell-Eingabe abgesetzt werden.

Die folgende Abbildung zeigt den Stop-Befehl und die Meldung, dass der Server korrekt beendet wurde:

```
                              QSH-Befehlseingabe

> cd /QIBM/ProdData/WebSphere/AppServer/V6/base/bin
  $
> stopServer
  ADMU0116I: Toolinformationen werden in der Datei
             /QIBM/UserData/WebSphere/AppServer/V6/Base/profiles/default/logs/s
  erver1/stopServer.log
             protokolliert.
  ADMU0128I: Das Tool wird mit dem Profil default gestartet.
  ADMU3100I: Die Konfiguration des Servers wird gelesen: server1
  ADMU3201I: Es wurde eine Stoppanforderung für den Server ausgegeben. Es wird
             auf den Stoppstatus gewartet.
  ADMU4000I: Der Server server1 wurde gestoppt.
```

Stoppen des Server

Natürlich können Sie sowohl den Start- als auch den Beendigungs-Befehl automatisieren.

In Abhängigkeit der Ebene, welche Sie verwalten wollen, stehen noch weitere Start- und Stop-Befehle zur Verfügung:

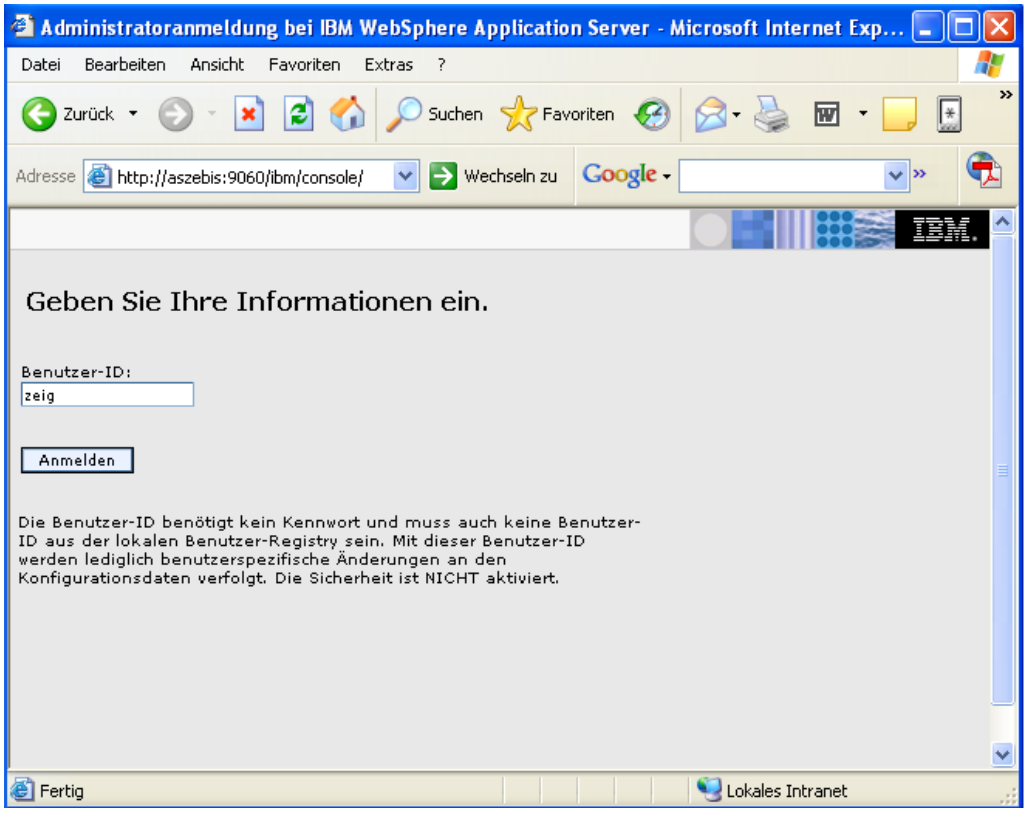
- Starten des Deployment-Managers
  startManager
- Starten des Knotens
  startNode

Nachdem der Server erfolgreich gestartet wurde, können wir den eigentlichen Konfigurationsbereich von WebSphere starten. Anders als zum Beispiel von den Express-Versionen des WebSphere in dessen Version 5 kommt bei der Version 6 des Application-Server nicht der http-Administrations-Bereich der i5 für die Verwaltung der WAS-Installation zum Einsatz, sondern ein eigener, für die Verwaltung des WAS 6 vorgesehener Administrations-Bereich.

Der Administrations-Client ist vollständig Browser-basiert und kann mit der folgenden URL aufgerufen werden:

http://*servername*:*port*/ibm/console

Ersetzen Sie den Platzhalter „servername“ durch den Namen Ihres Server. Dabei handelt es sich in der Regel um den Hostnamen der i5. Als Standard-port für die Administration wird in diesem Fall der Port 9060 verwendet. Dem zu Folge kann ein qualifizierter Aufruf wie folgt aussehen:



Start der Administrations-Konsole

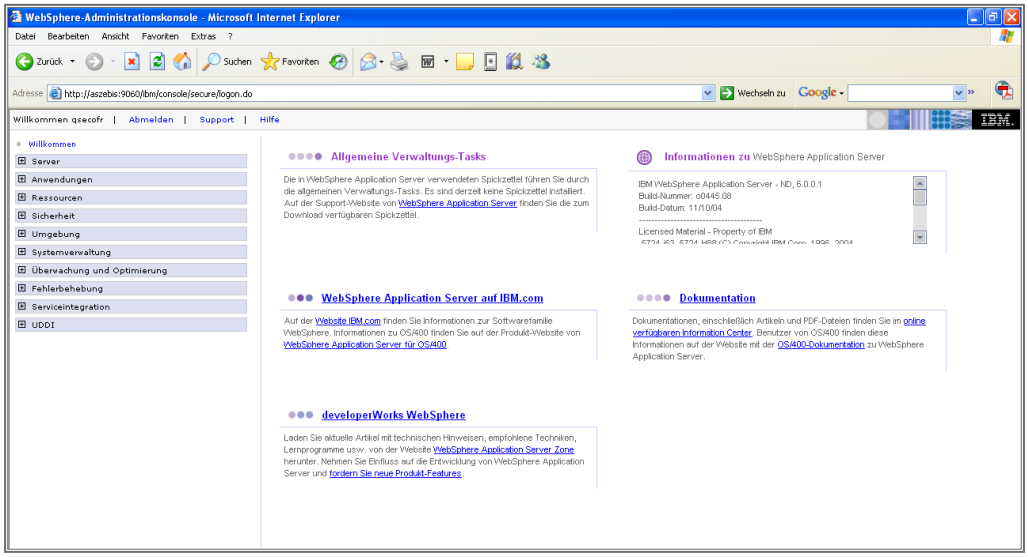
5.2.2.2
Seite 10

Wie der Hinweistext bereits aussagt, ist die hier anzugebende Benutzer-ID lediglich für Protokollierungszwecke von Bedeutung.

Anmerkung

Für die Ausführung der Administrations-Konsole müssen im Browser die Cookies aktiviert werden. Anderenfalls schlägt die Ausführung der Administrations-Konsole fehl.

Nach der Angabe eines „Pseudo-Benutzers“ erscheint die Willkommensseite der Administrations-Konsole, wie sie in der folgenden Abbildung zu sehen ist:



Die Willkommensseite der Administrations-Konsole

Einzelheiten zum Inhalt und zum Aufbau dieser Administrations-Konsole finden Sie in dem Kapitel „5.2.4.1 Die Administrations-Konsole“.

Ausgehend von dieser Einstiegsseite können wir die Verwaltung des WAS-Server durchführen. Dieser Verwaltungsbereich unterscheidet sich nicht oder nur unwesentlich in Abhängigkeit der Hardware oder des Betriebssystems, auf welcher der WAS installiert wurde.

Grundsätzlich muss in Bezug auf die Verwaltung und die Administration eines WAS-Server der allgemeine WAS-Administrationsbereich betrachtet werden. Nicht minder wichtig sind die Tätigkeiten und Einstellungen, die auf Betriebssystem-Ebene der Hardware durchgeführt werden müssen, damit die WAS-Installation optimal ausgeführt werden kann. Deshalb unterscheiden wir im Weiteren die Administration des WAS-Server auf i5 in die beiden Bereiche.

- Allgemeine WAS-Server-Verwaltung (zum Beispiel über die Administrations-Konsole)
- Verwaltung der i5-spezifischen Konfigurations-Einstellungen in Form von zum Beispiel Subsystem-Einstellungen, Systemwerten etc..

Diese beiden Teilbereiche werden wir im weiteren Verlauf in Bezug auf die i5-Integration genauer behandeln.

5.2.2.2
Seite 12

V6 WAS Update

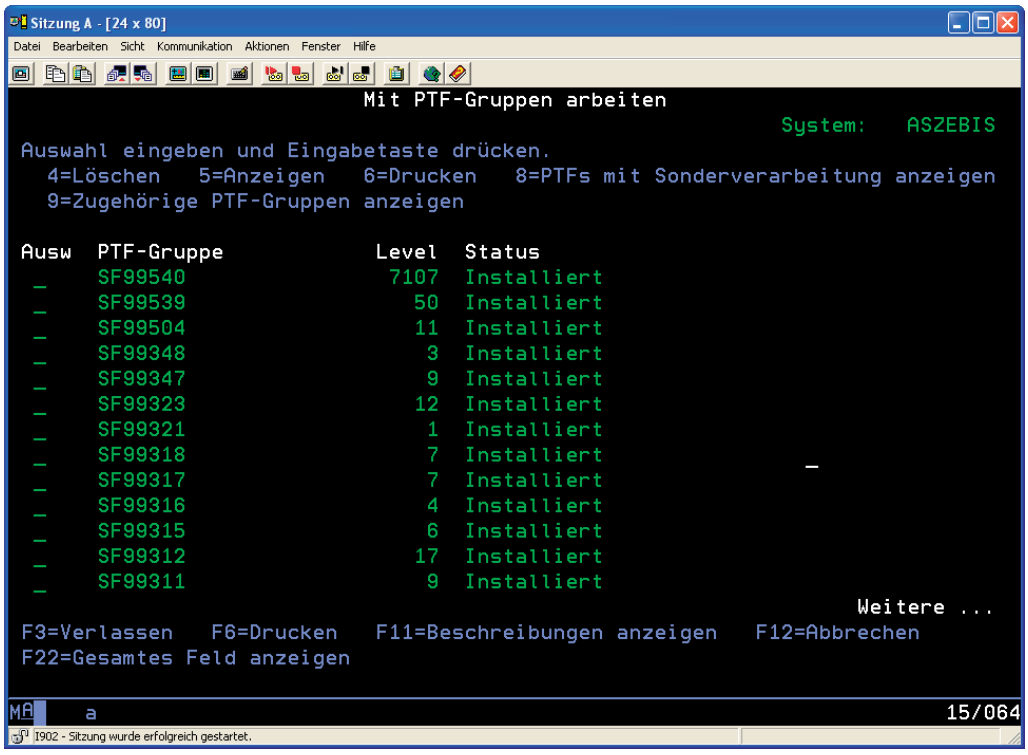
WebSphere Application Server (WAS) stehen in unterschiedlichen Versionen zur Verfügung. Diese Versionen erfahren immer wieder neue Updates, die wir im System-i-Umfeld in Form von speziellen Gruppen-PTFs installieren können. Wer den Installationsprozess solcher Gruppen-PTFs kennt, weiß, dass das Einspielen in der Regel kein Problem darstellt. So ist das auch im Bereich der Gruppen-PTFs für den/die WebSphere Application Server.

Doch wer glaubt, dass alleine das Einspielen der Gruppen-PTFs auch dazu führen würde, dass die auf dem System i konfigurierten WAS Server auf den neuesten Stand migriert werden, irrt leider.

Um die Installation der WAS auf System i vollständig durchzuführen, müssen noch zusätzliche Schritte ausgeführt werden, mit denen der Fix-Level des jeweiligen Anwendungsservers angehoben wird.

Die Basis für ein solches Anheben bildet stets der aktuelle Stand der Gruppen-PTFs native auf dem System i im Bereich von WebSphere – in Verbindung mit den DB2-Datenbank-PTFs, TCP/IP-PTFs und Java-Gruppen-PTFs.

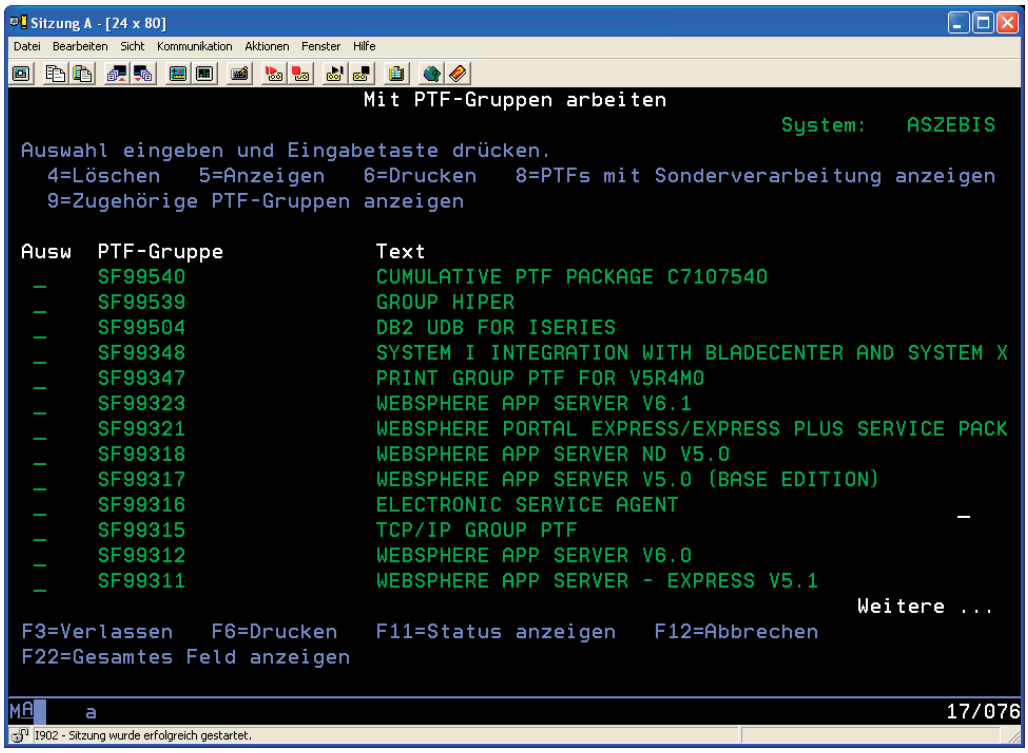
Aber schauen wir uns zunächst einmal die Gruppen-PTFs an. Diese lassen sich mit dem Befehl WRKPTFGRP anzeigen:



WRKPTFGRP

5.2.2.3
Seite 2

Mit der Taste „F11“ können Sie weitere Informationen anzeigen, wie zum Beispiel den Text zum jeweiligen Gruppen-PTF. Dieser Text gibt darüber Auskunft, zu welchem Bereich die jeweilige PTF-Gruppe gehört.



Gruppen-PTF – Textanzeige

Wie man anhand der Abbildung erkennen kann, handelt es sich beim Gruppen-PTF SF99312 beispielsweise um das Gruppen-PTF für den WebSphere Application Server in der Version 6.0, den wir hier exemplarisch behandeln.

Wir werden uns an dieser Stelle nicht weiter mit den Gruppen-PTFs oder deren Einspielung beschäftigen; wir konzentrieren uns auf den Upgrade-Prozess der WebSphere Application Server. Details zum Einspielen von PTFs finden Sie im Kapitel 2.9 „Systempflege“.

Zunächst schauen wir uns den aktuell installierten Stand an, der in Abhängigkeit der installierten Versionen des WAS variieren kann.

Informationen über die aktuell installierte Version liefert uns beispielsweise der Befehl „versionInfo“, den wir in einem Qshell-Fenster eingeben können. Die folgenden Abbildungen zeigen, wie Sie die Version ermitteln können. Dazu ist in einer 5250-Sitzung zunächst eine QSH-Befehlseingabe mit dem Befehl „STRQSH“ aufzurufen. Anschließend muss in das Installationsverzeichnis des WAS gewechselt werden, denn dessen Status soll abgerufen werden. Geben Sie dazu im QSH-Fenster folgenden Befehl ein:

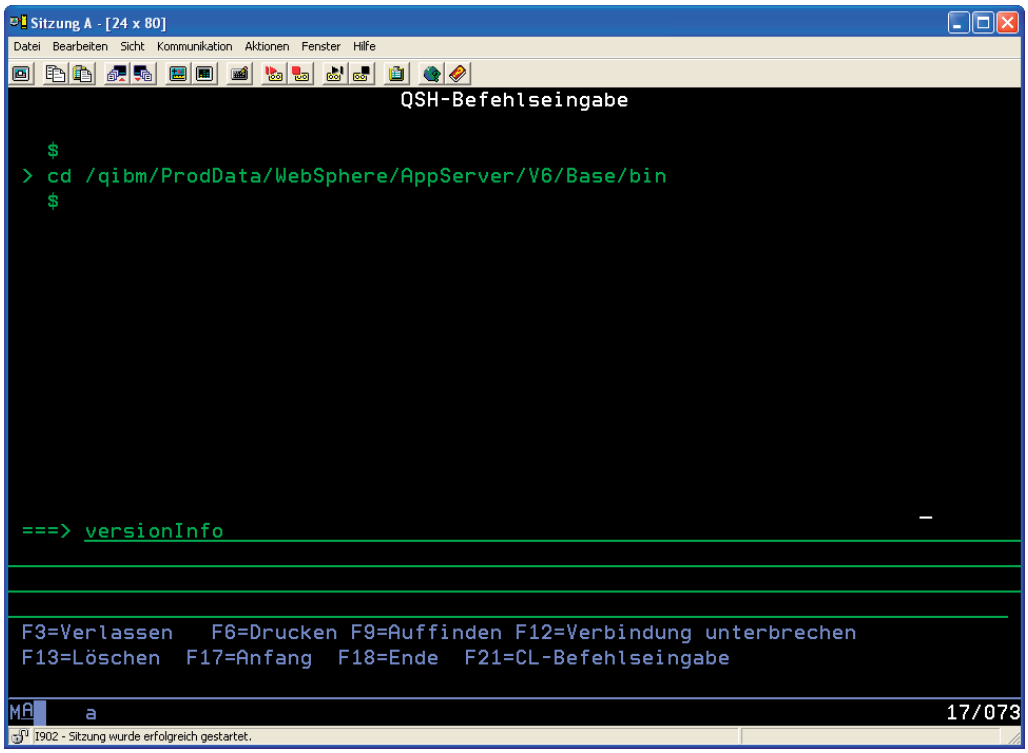
```
cd /QIBM/ProdData/WebSphere/AppServer/V6/edition/bin
```

Anmerkung

Ersetzen Sie bitte die Platzhalter „edition“ durch die bei Ihnen installierte Version. Wenn Sie beispielsweise den WebSphere Application Server Express in Version 6 nutzen, geben Sie folgenden Befehl ein:

```
cd /QIBM/ProdData/WebSphere/AppServer/V6/base/bin
```

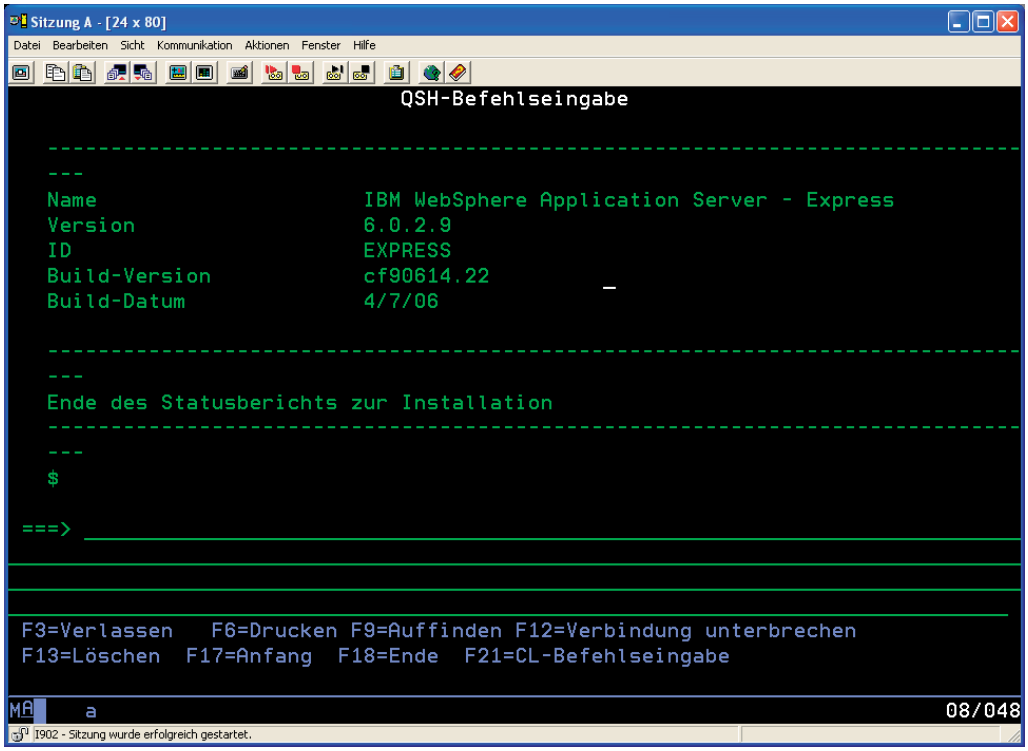
Mit dem eingegebenen Befehl sind wir in das Installationsverzeichnis des WAS gelangt. Nun geben wir den Befehl ein, mit dem wir die Version abrufen können. Dies ist der Befehl „versionInfo“. Schauen Sie sich dazu folgende Abbildung an:



Versionskontrolle

5.2.2.3
Seite 4

Das Ergebnis der Abfrage kann wie folgt aussehen:

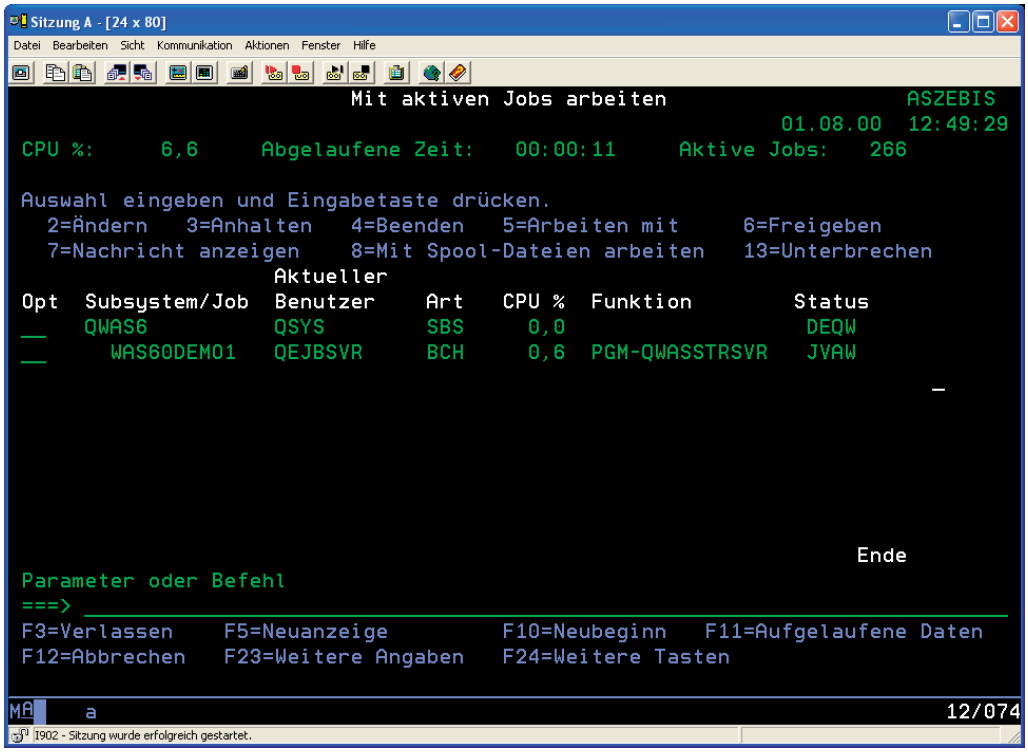


Versionsinformation

Als Ergebnis haben wir im folgenden Beispiel die Information erhalten, dass der WAS derzeit in der Version 6.0.2.9 installiert ist.

Bevor wir den WAS aktualisieren, muss sichergestellt sein, dass dazu ein Benutzerprofil verwendet wird, das über ausreichende Berechtigungen verfügt. Idealerweise verwenden wir für den Upgrade-Prozess das Benutzerprofil QSECOFR.

Als Erstes beenden wir das Subsystem, in dem der WebSphere Application Server ausgeführt wird. Im Falle der Version 6 ist dies das Subsystem QWAS6.



Aktives Subsystem QWAS6

Wenn das Subsystem QWAS6 aktiv ist, beenden Sie es mit dem Befehl:

```
ENDSBS QWAS6 *IMMED
```

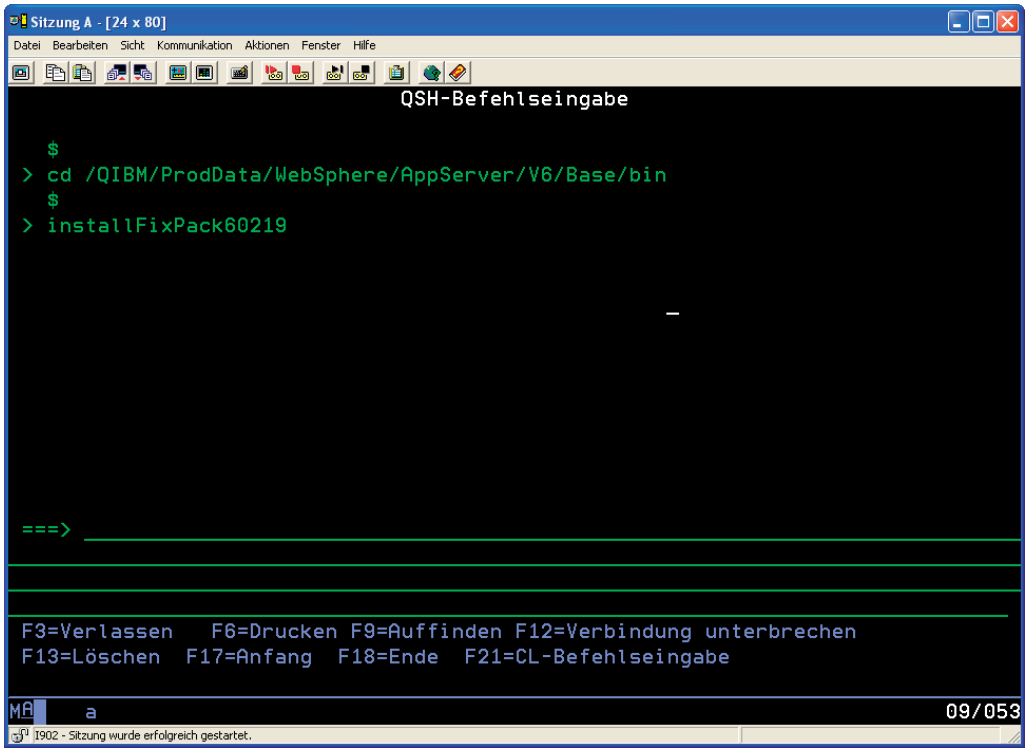
Achtung

Das Beenden sollte natürlich nur dann durchgeführt werden, wenn keine Server-Instanz ausgeführt wird, die aktiv benötigt wird!

Während des Einspielens des WAS Gruppen-PTF sind unter anderem auch einige Qshell-Skripts installiert worden, die wir für den Upgrade-Prozess benötigen. In der aktuell verwendeten Version 6.0.2.x des WAS Servers (Version 6 – Stand Herbst 2007) finden wir das folgende Skript „installFixPack60219“ im Verzeichnis:

„*QIBM/ProdData/WebSphere/AppServer/V6/Base/bin*“.

5.2.2.3
Seite 6



Interaktiver Update aus QSH

Der Update-Prozess lässt sich interaktiv aus einer QSH-Eingabe heraus starten. Man kann ihn auch mittels Batch-Befehl aufrufen, der in der Regel deutlich schneller ausgeführt wird. Den interaktiven Aufruf sehen Sie in der vorangegangenen Abbildung. Per Standard wird dieser Aufruf in das Subsystem QSYSWRK übertragen und dort ausgeführt. Alternativ kann der Update-Job auch an einen anderen Bereich übertragen werden. Dazu können wir die optionalen Parameter JOBQLIB, JOBQ und Job verwenden, die zusätzlich zum Befehl „installFix“ anzugeben sind. Schauen Sie sich dazu auch das folgende Beispiel an:

```
installFixPack60219  -jobqlib QGPL -jobq DFTJOBQ -job UPDATE
```

Alternativ zum interaktiven Aufruf lässt sich der Update-Prozess auch mit Hilfe einer Übergabe als Batch-Job starten. Dazu geben wir den folgenden Befehl ein:

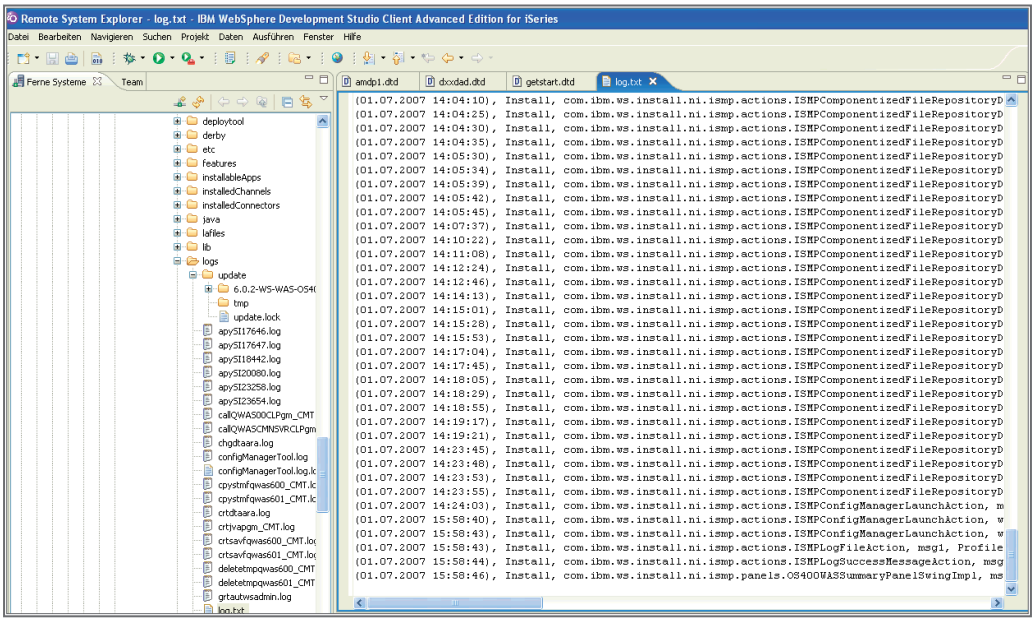
```
SBMJOB CMD(STRQSH CMD(,/QIBM/ProdData/WebSphere/AppServer/V6/Base/
updateinstaller/update‘)) JOB(UPDATE) ALWMLTTHD(*YES)
```

Unabhängig davon, ob Sie den interaktiven Weg oder die Batch-Installation gewählt haben, der Update-Prozess wird einige Zeit in Anspruch nehmen. Mit dem Befehl WRKACTJOB finden wir im Subsystem QSYSWRK die entsprechenden Update-Jobs. Allerdings geben diese nicht in jedem Fall genügend Auskunft über den Stand des Updates. Weitere Informationen finden wir im Update-Protokoll, das im Installationsverzeichnis des Anwendungs-

Servers geführt wird. Dieses Protokoll trägt die Bezeichnung „log.txt“ und befindet sich im Verzeichnis „update“.

/QIBM/ProdData/WebSphere/AppServer/V6/<edition>/logs/update/<image>

Die folgende Abbildung zeigt ein Beispiel für „log.txt“:

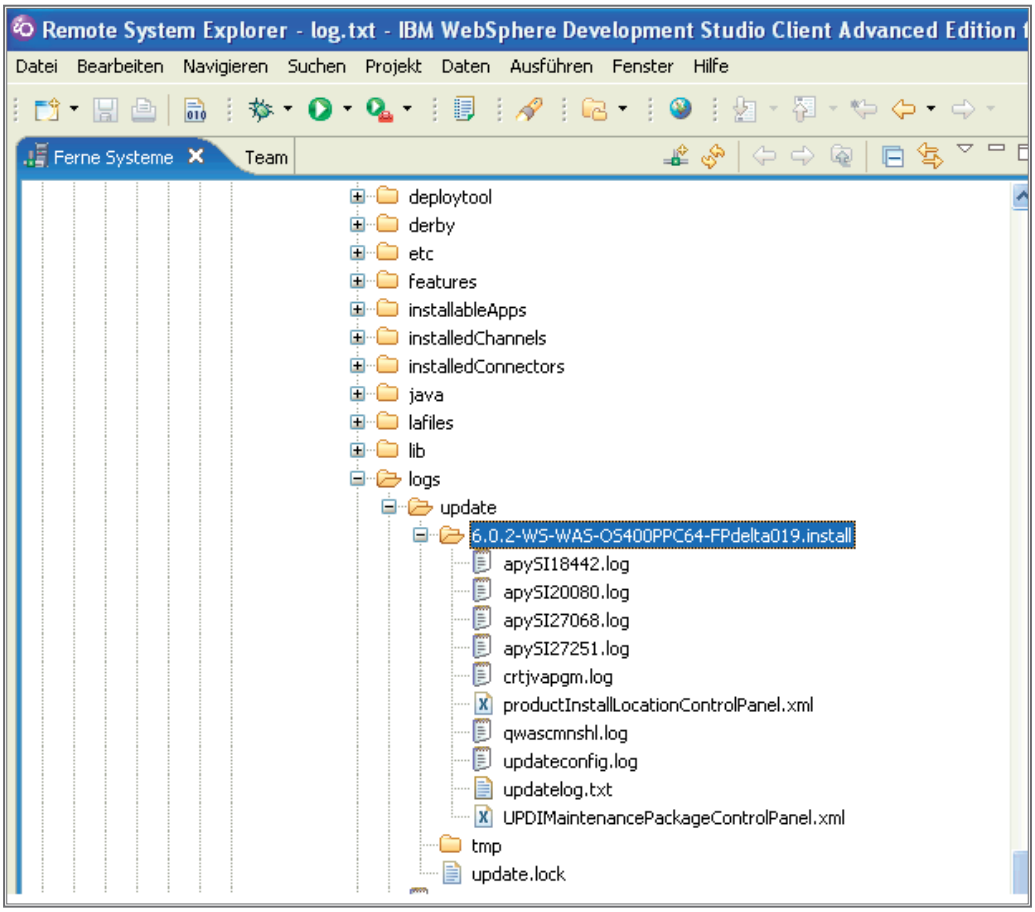


log.txt

Es kann unter Umständen vorkommen, dass im Protokoll der Update-Prozess als „fehlerhaft“ ausgewiesen wird. Das ist leider hin und wieder der Fall, wenn die eigentlichen Update-Prozesse für installierte Server-Instanzen erfolgreich durchgeführt werden konnten, einige Updates für „Standard-Instanzen“ jedoch fehlgeschlagen sind, da sie auf dem System gar nicht vorhanden sind.

Nach dem Update wird ein weiteres Unterverzeichnis innerhalb des Verzeichnisses „update“ angelegt, das Informationen zum Update-Prozess beinhaltet.

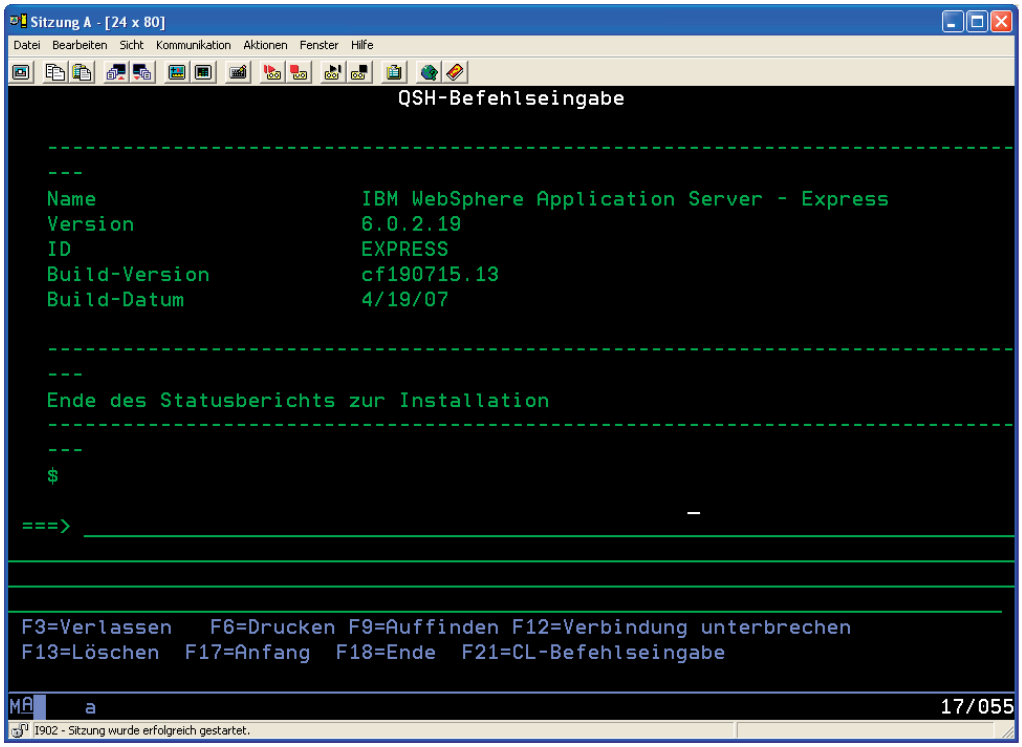
5.2.2.3
Seite 8



Update-Ergebnis

Unterhalb des Verzeichnisses „Update“ finden wir den aktuellen Stand des WAS mit aktuellem Fixpack-Stand. Wie Sie in der vorhergehenden Abbildung erkennen können, haben wir dort WAS 6.0.2 mit dem Fixpack Stand 19 installiert.

Es empfiehlt sich, nach dem Update-Prozess nochmals im Qshell-Bereich die Versionskontrolle durchzuführen, die Aufschluss über den aktuellen Status des Release-Wechsels gibt. Die folgende Abbildung zeigt den Status nach dem Wechsel auf Version 6.0.2.19.



Abschließender Status

Bevor Sie nun den WebSphere Application Server verwenden, empfiehlt es sich, den HTTP-Admin-Server erneut zu starten und erst danach den WAS Server wieder hochzufahren.

Mit diesen zusätzlichen Schritten ist der Update auf die neueste Version eines WebSphere Application Servers abgeschlossen.

5.2.2.3
Seite 10

WebSphere Application Server Enterprise Enablement für OS/400

WebSphere besteht aus einer Vielzahl von Teilbereichen und Komponenten, die für verschiedenste Anforderungen eingesetzt werden können bzw. auch eingesetzt werden müssen.

Im Zusammenhang mit der Verwendung eines IBM WebSphere Portal auf der iSeries muss zum Beispiel zusätzlich zu der Basis-Installation eines WAS-Server das so genannte „WebSphere Application Server Enterprise Enablement" installiert werden, welches für die Kommunikation zwischen dem WAS- und dem Portal-Server benötigt wird.

Anmerkung

Das WebSphere Zusatzprodukt „WebSphere Application Server Enterprise Enablement" wird nicht nur für die Anbindung des WebSphere Portal Servers benötigt, sondern ist auch eine der Voraussetzungen für zum Beispiel die Installation eines IBM Workplace.

Diese Zusatzsoftware ist in der Regel in dem Lieferumfang des WebSphere Portal Server enthalten. Dabei ist dieser Sonderzusatz meist auf einer eigenen CD vorhanden, welche die Bezeichnung „Enablement for iSeries" trägt.

Vor der Installation eines WebSphere Portal Server auf der iSeries muss diese Zusatzsoftware nach der bereits erfolgten Installation eines WAS-Server installiert werden.

Die Installation wird dabei über das CD/DVD-Laufwerk einer Arbeitsstation ausgeführt, welche über eine Netzwerkverbindung zur iSeries verfügt.

Achten Sie vor der Installation dieser Zusatzsoftware darauf, dass der WAS-Server zuvor beendet wurde. Stellen Sie vor dem Beenden des Server auch sicher, dass eventuell ausgeführte Anwendungen vor dem Beenden des WAS-Subsystems gestoppt wurden, um unvorhersehbare Situationen zu vermeiden!

Für das Beenden des WAS 5 geben wir in einer OS/400 Befehlseingabe-Zeile den folgenden Befehl ein:

```
ENDSBS SBS(QEJBAS5)
```

Tipp

Für die Installation des WebSphere Portal Server ist ein OS/400-Benutzerprofil erforderlich, das über die folgenden Mindestberechtigungen verfügen muss:

- *ALLOBJ

5.2.3
Seite 2

- *IOSYSCFG
- *SECADM (nur für das Erstellen eines neuen Datenbank Benutzerprofils)
- *JOBCTL

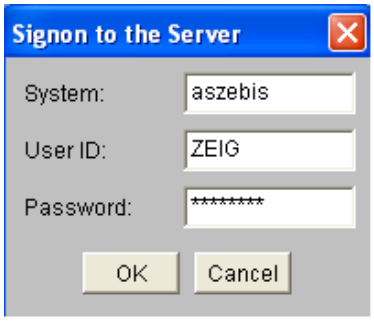
Ich empfehle an dieser Stelle, das Benutzerprofil QSECOFR oder ein vergleichbares Benutzerprofil zu verwenden.

IBM gibt zwar an, dass das Installationsprogramm für die Enablement-Software nach dem Einlegen der CD/DVD automatisch startet – allerdings ist das nicht immer der Fall.

Sollte also nach dem Einlegen der CD der Installationsassistent nicht gestartet werden, dann können Sie diesen auch manuell durch einen Klick auf die entsprechende Datei „pmeinstall“ aufrufen. Ein Beispiel dazu finden Sie in der nächsten Abbildung. Auf diesen Schritt können Sie verzichten, wenn der Installationsprozess automatisch nach dem Einlegen der CD/DVD gestartet wird.

Manueller Start

Das Java-basierte Installationsprogramm muss zunächst eine Verbindung zur iSeries herstellen. Geben Sie in dem sich öffnenden Java-Eingabebereich den IP-Namen der iSeries (oder die IP-Adresse der iSeries) an. Für die Installation benötigen Sie, wie zuvor beschrieben, ein OS/400-Benutzerprofil mit ausreichenden Berechtigungen für die Installation.



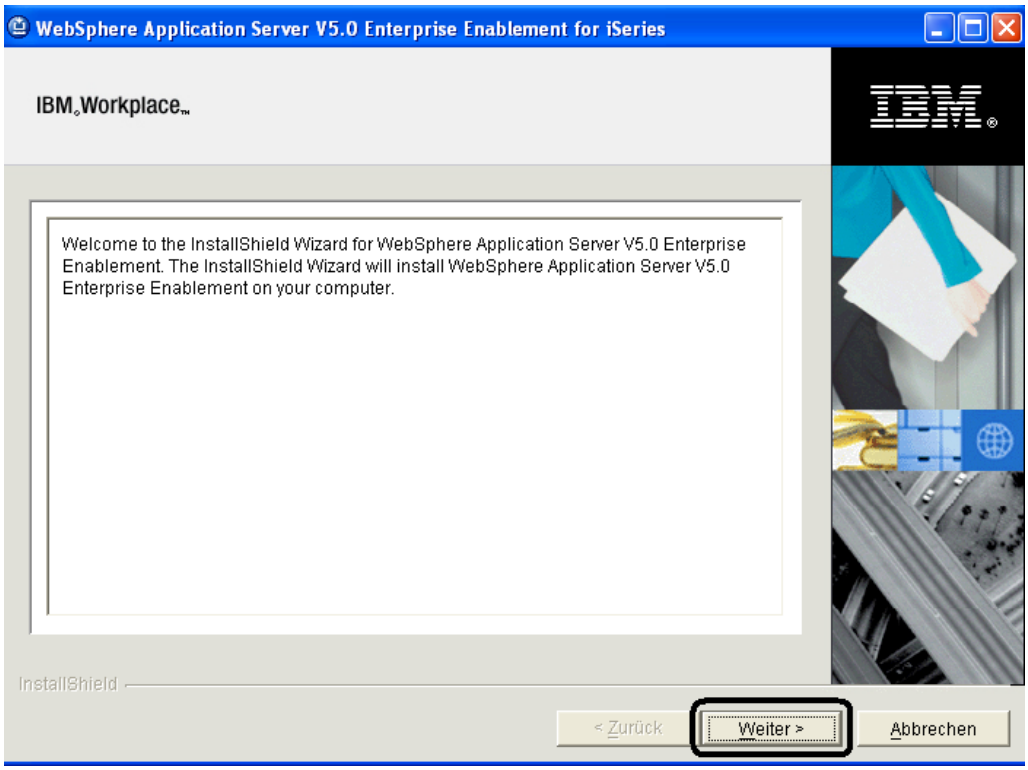
Login

Nach Eingabe des Systemnamens, des OS/400-Benutzerprofils und des dazugehörigen Kennworts klicken wir auf „OK“. Jetzt wird eine Verbindung zwischen dem PC und der iSeries hergestellt, über welche die Installation durchgeführt wird.

Achten Sie bitte darauf, dass diese Verbindung während des Installationsprozesses nicht unterbrochen wird.

Sprachauswahl

Der Installationsassistent steht uns in verschiedenen Sprachen zur Verfügung. Per Standard ist „Englisch“ vorselektiert. Zum besseren Verständnis wählen wir hier „Deutsch“ aus und klicken auf „OK“.

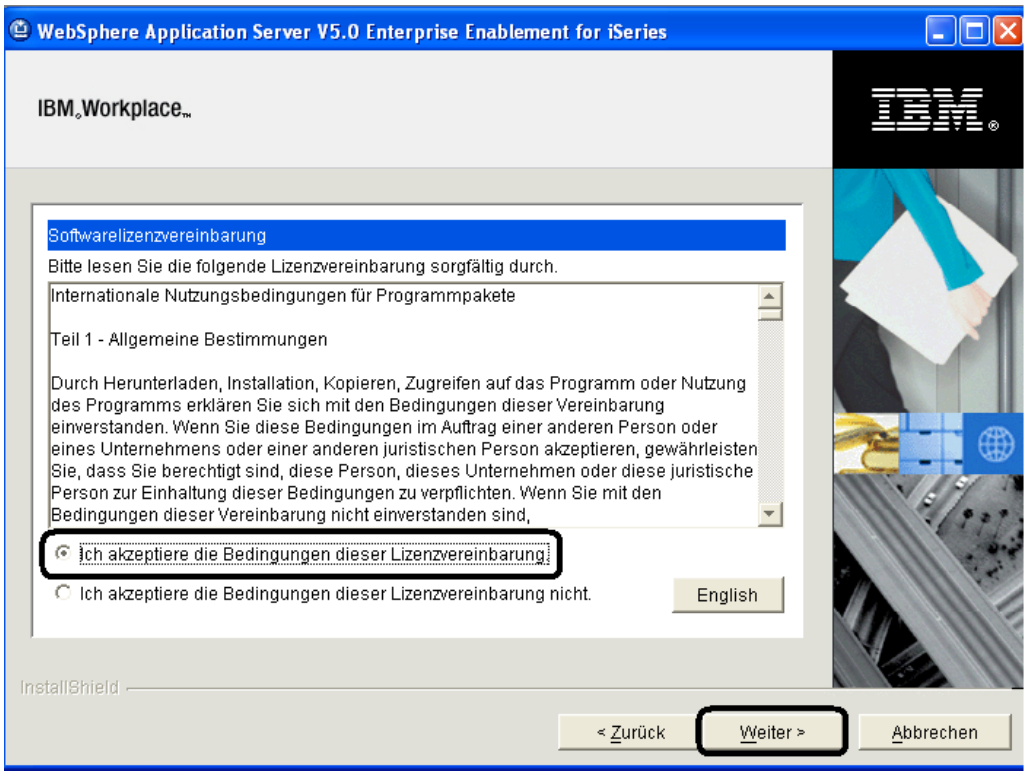


Installationsassistent

5.2.3
Seite 4

Jetzt erscheint das erste Fenster des Installationsassistenten. Dieser scheint noch ein wenig Bedenkzeit für die Übersetzung zu benötigen – denn interessanter Weise ist diese Seite trotz der Auswahl der deutschen Sprache noch in Englisch. Wir stören uns nicht weiter daran und klicken auf die Schaltfläche „Weiter".

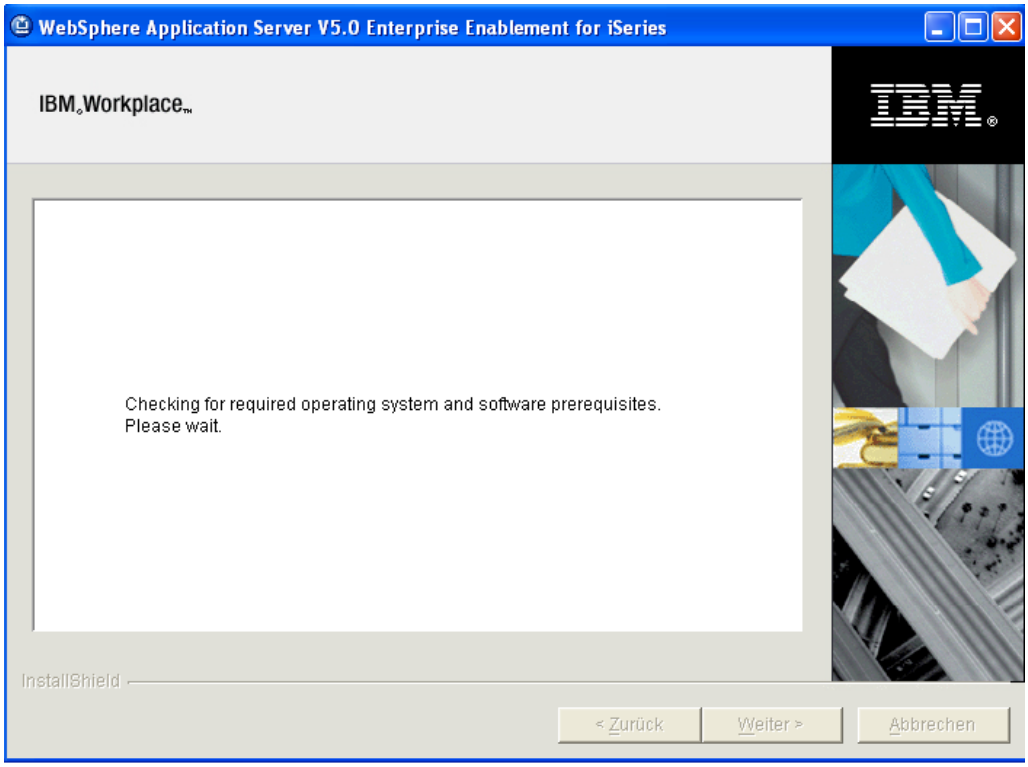
Wir kennen sie bereits – die Lizenzvereinbarungen.



Lizenzvereinbarung

Bevor wir die Installation fortsetzen können, muss die Option „Ich akzeptiere die Bedingungen dieser Lizenzvereinbarung" selektiert werden – erst dann wird die Schaltfläche „Weiter" freigegeben, die wir anschließend anklicken.

Jetzt wird zunächst eine allgemeine Prüfung der Voraussetzungen durchgeführt.

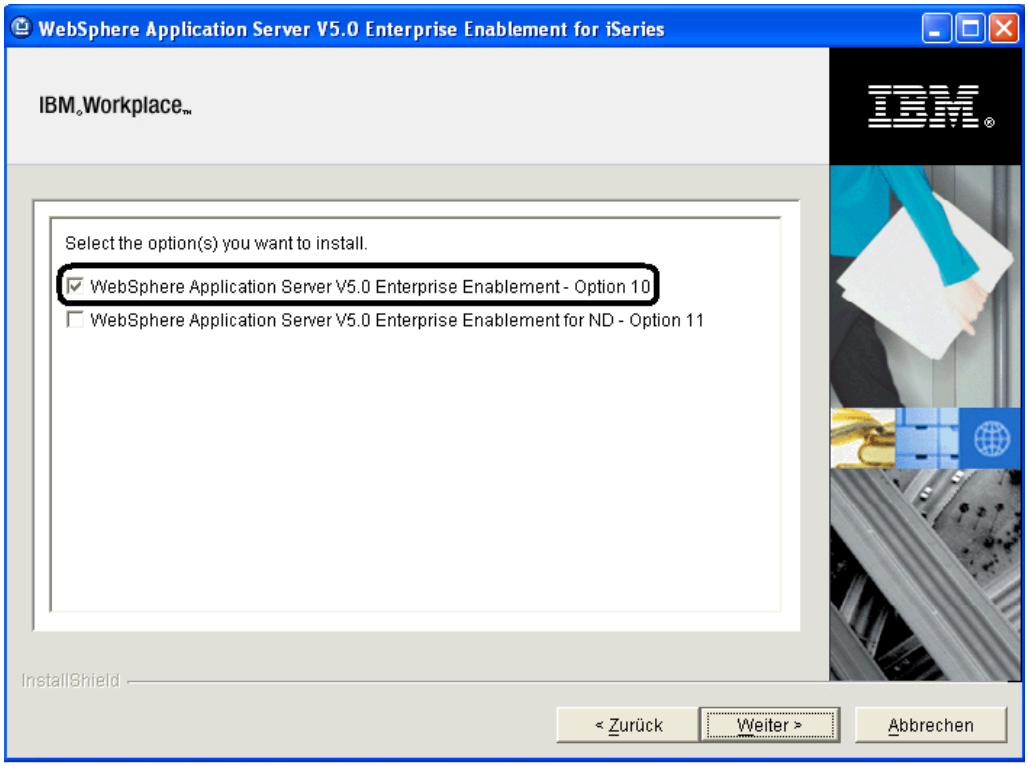


Prüfen der Voraussetzungen

Je nach Systemleistung kann diese Prüfung einige Minuten dauern. Dabei wird die iSeries auf die erforderliche Hardware und die Softwarevoraussetzungen gecheckt, welche für die erfolgreiche Installation dieses Tools erforderlich sind.

Wenn die Voraussetzungen erfüllt sind, erscheint die nächste Anzeige. Fehlen zwingend erforderliche Komponenten, dann wird eine Fehlerliste ausgegeben, in der die fehlenden Voraussetzungen aufgeführt sind.

WebSphere Enterprise Enablement V5 besteht aus zwei Teilbereichen, die nach einem Klick auf die Schaltfläche „Weiter“ zur Auswahl angeboten werden.



Zu installierende Optionen

Für die spätere Anbindung an den Portal-Server genügt die Installation der Option 10 – welche die Basis-Installation des Enterprise-Enablement darstellt. In einer komplexen, verteilten Umgebung muss zusätzlich die Option 11 installiert werden.

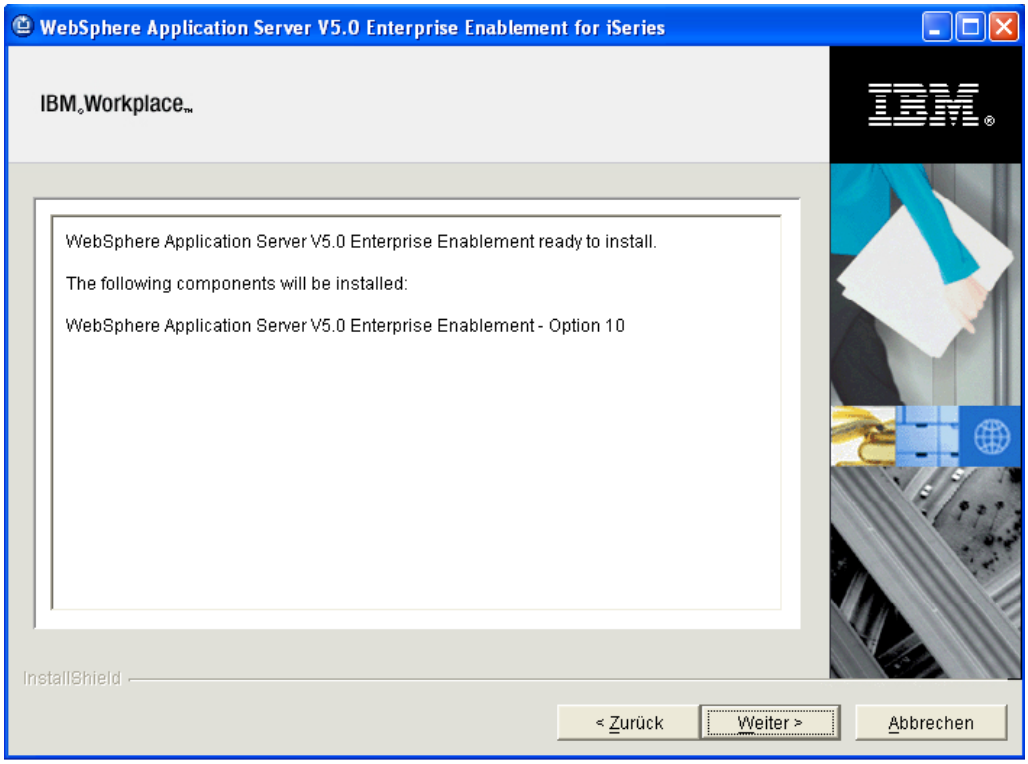
Anmerkung

Die Option 11 des „WebSphere Application Server V5.0 Enterprise Enablement for ND" wird für die Netzwerk Version des WAS benötigt (ND = Network Deployment).

Für unsere Beispielinstallation reicht die Installation der Option 10 aus.

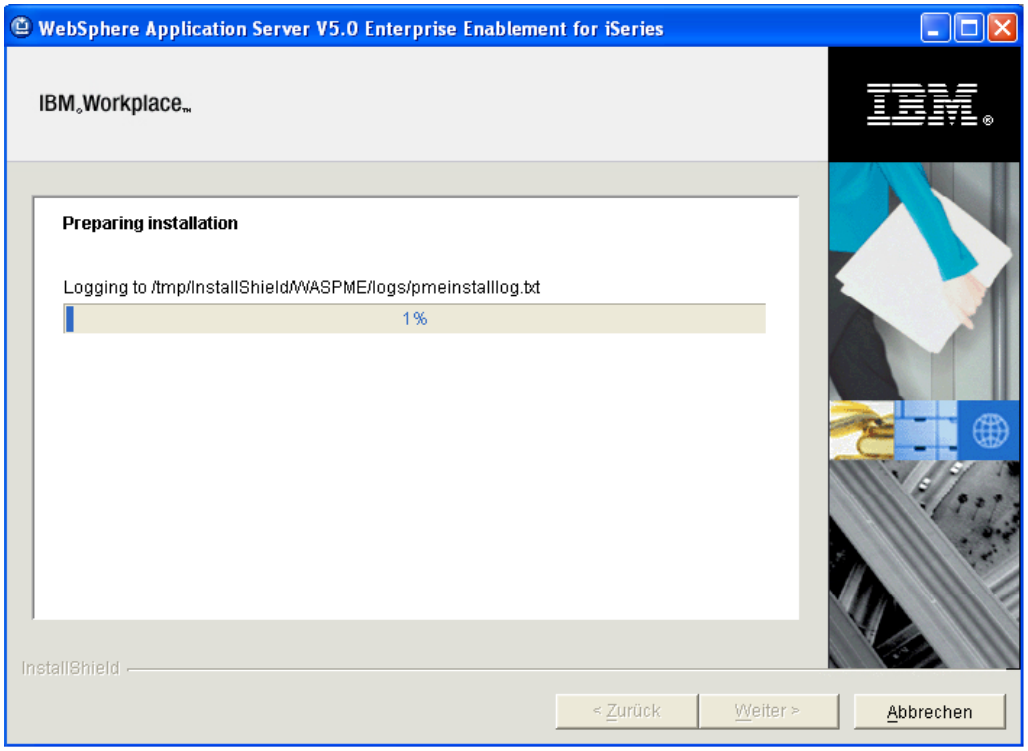
Sollten Sie die Installation beider Optionen wünschen, dann muss als Voraussetzung für die Installation der Option 11 auch die Network-Deployment-Version des WAS auf der iSeries installiert sein.

Nach der Auswahl der zu installierenden Option 10 klicken wir auf die Schaltfläche „Weiter".



Bereit zur Installation

Der Installationsassistent zeigt nochmals die von Ihnen zur Installation ausgewählte Option an. Bestätigen Sie Ihre Auswahl mit einem erneuten Klick auf die Schaltfläche „Weiter“.



Status der Installation

Leider gibt es in einigen Versionen an dieser Stelle eine kleine „Unschärfe“ während der Installation – denn die Aktualisierung der Statusanzeige funktioniert nicht immer – so lässt sich nur schwer feststellen, ob die Installation erfolgreich durchgeführt werden konnte, oder ob sie fehlgeschlagen ist.

Tipp

Sollte sich der Status der Installation nach einiger Zeit nicht ändern, dann prüfen Sie bitte, ob die Installationsdaten von dem CD/DVD-Laufwerk Ihres Rechners auf die iSeries übertragen werden. Dabei blinkt die Aktivitätsanzeige des Laufwerks.

Im Normalfall kann die Installation des Enterprise-Enablement bis zu 30 Minuten dauern (für die Installation der Option 10 – die Installation der Option 11 kann bis zu 60 Minuten in Anspruch nehmen).

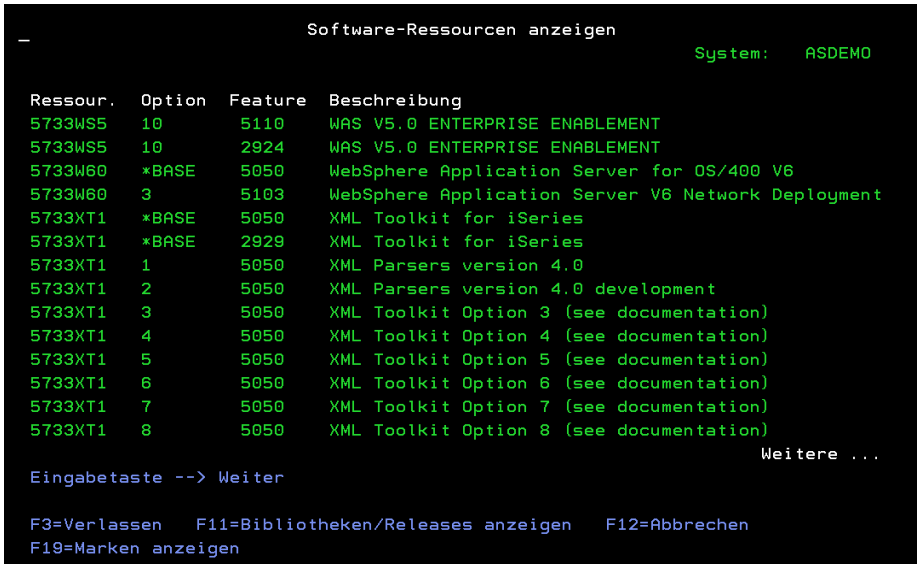
Verändert sich die Statusanzeige der Installation nicht, dann können Sie den Status der Installation auch wie folgt prüfen:

Geben Sie in einer OS/400-Befehlszeile den folgenden Befehl ein:

```
DSPSFWRSC
```

Suchen Sie in der danach dargestellten Auflistung nach den beiden Einträgen

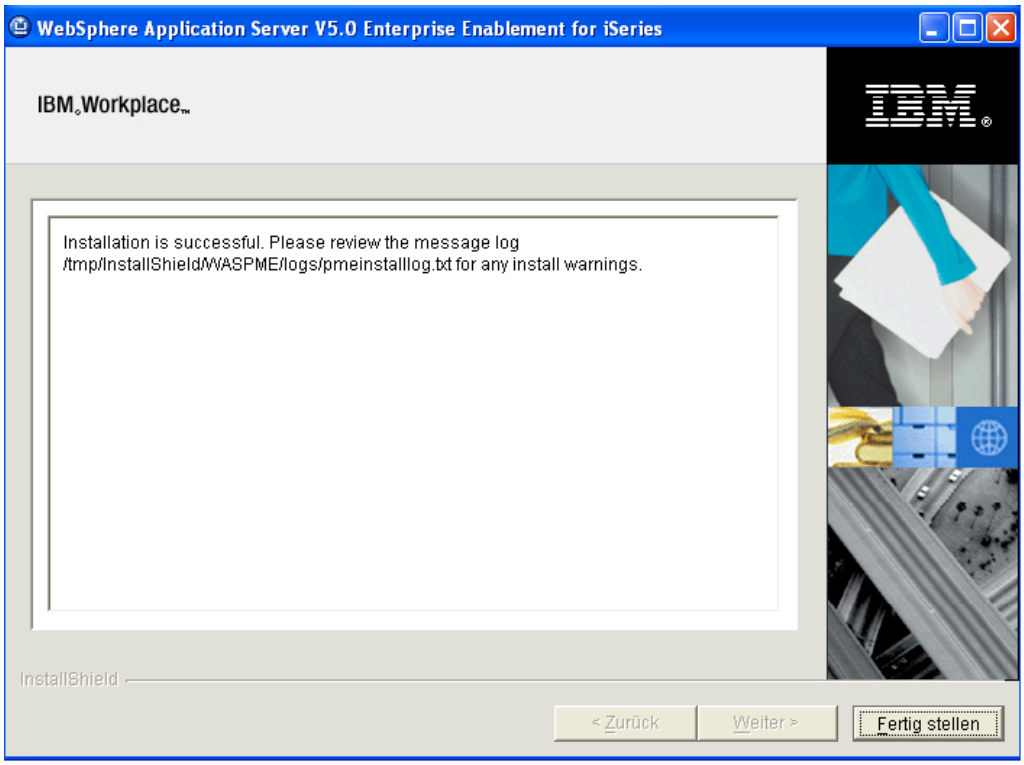
- 5733WS5 Option 10 Feature 5110 WAS V5.0 Enterprise Enablement
- 5733WS5 Option 10 Feature 2924 WAS V5.0 Enterprise Enablement



DSPSFWRSC

Sind die beiden Optionen installiert, dann war die Installation erfolgreich.

Im Normalfall erscheint – auch nach einer zwischenzeitlich nicht aktualisierten Statusanzeige – eine Erfolgsmeldung, wie sie in der nachfolgenden Abbildung enthalten ist.



Erfolgsmeldung

Klicken Sie zum Abschluss der Installation auf die Schaltfläche „Fertig stellen“. Damit wird der Installationsassistent geschlossen und Sie können mit der Installation des WebSphere Portal Server fortfahren.

5.2.3
Seite 10

Administration des WAS V6

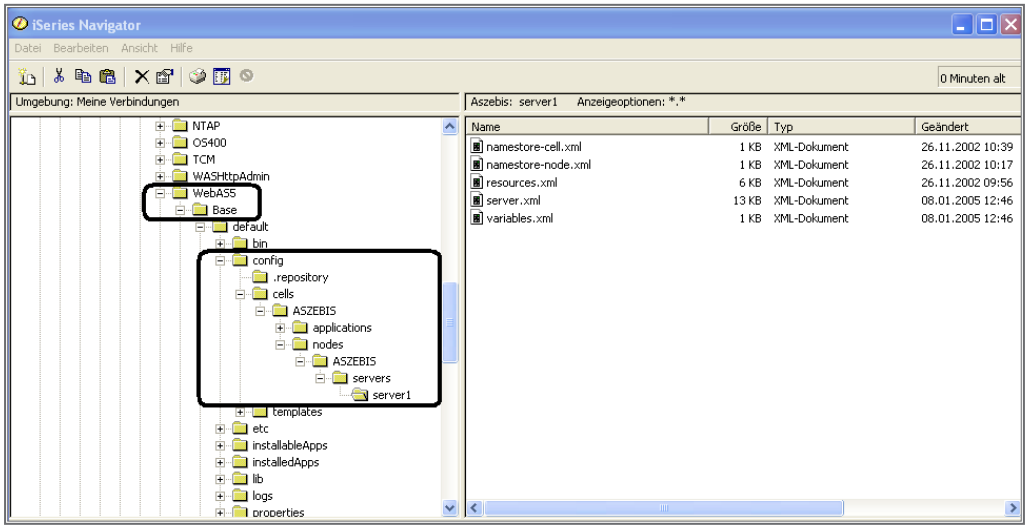
Die Administration des WAS bzw. dessen Instanzen ist eine der wichtigsten Aufgaben, die nach der Installation und Konfiguration des Server anfällt. IBM hat in den letzten Jahren die verschiedenen Teilbereiche der Verwaltung der WAS-Server stark vereinfacht. Standen uns am Anfang der Ära WebSphere vorwiegend befehlszeilenorientierte Administrations-Tools zur Verfügung, so hat sich der Trend hin zu grafischen Verwaltungs-Werkzeugen auch im WebSphere-Umfeld dominant durchsetzen können.

Die Administration einer WAS-Umgebung kann mit den folgenden Standard-Werkzeugen durchgeführt werden:

- Administrations-Konsole (bevorzugte Variante für die Administration)
- wsadmin
- Java-PIs
- Befehlszeilen-Editor

Der Einsatz der befehlszeilenorientierten Administrations-Werkzeuge hat gerade seit der Einführung der Version 5 des WAS stark an Bedeutung verloren. Aber auch heute noch können wir die Konfigurations-Einstellungen der aktuellen Server-Editionen mit Hilfe von Texteditoren verändern. Derzeit (Stand Version 6) sind noch nicht alle Konfigurations-Elemente für die Verarbeitung mit den grafischen Werkzeugen ausgelegt. Einige Komponenten müssen manuell verarbeitet werden. Dazu gehört unter anderem auch die nachfolgende Datei, die mit einem Editor bearbeitet werden muss:

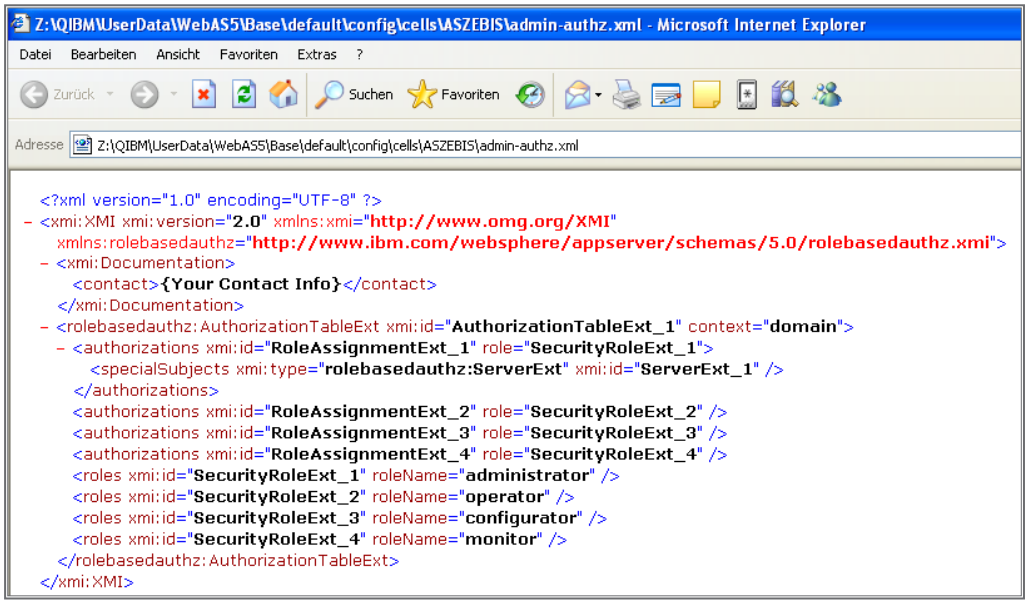
- admin-authz.xml in dem Verzeichnis config/cells/zellenname



admin-authz

5.2.4
Seite 2

Diese Datei – es handelt sich um eine XML-Datei – kann mit einem beliebigen XML-Editor verwaltet werden. Die folgende Abbildung zeigt den Inhalt der Datei admin-authz.xml in einem Standard-Editor.



Inhalt der admin-authz

Ähnlich wie der Zugriff und auch die Verwaltung dieser Administrations-Teildatei kann die Mehrheit der WAS-Konfigurations-Dateien auf diese Weise verwaltet werden. Wie Sie anhand der Abbildung oben erkennen können, ist der Inhalt der einzelnen Konfigurations-Dateien sehr spezifisch. Anpassungen erfordern ein gewisses Know-how und sind häufig mit Abhängigkeiten verbunden, die nicht auf den ersten Blick ersichtlich sind. Deshalb empfiehlt sich der Einsatz dieser „manuellen“ Werkzeuge nur für wirklich erfahrene Administratoren, die tatsächlich wissen, welche Auswirkungen die einzelnen Änderungen haben und welche Abhängigkeiten in den verschiedenen Teilbereichen zu beachten sind.

Grafische Administrations-Werkzeuge

Bedingt durch die integrierte Syntaxprüfung der grafischen Tools sind diese gerade auch für weniger erfahrene WAS-Administratoren wesentlich einfacher und auch effektiver im Einsatz, als die Administration mit Hilfe von einfachen Editoren. Deshalb gilt als Empfehlung die grafische Administrations-Oberfläche zu verwenden, und die Editor-basierte Verwaltungsmöglichkeit nur in den Fällen einzusetzen, für die diese unbedingt erforderlich ist.

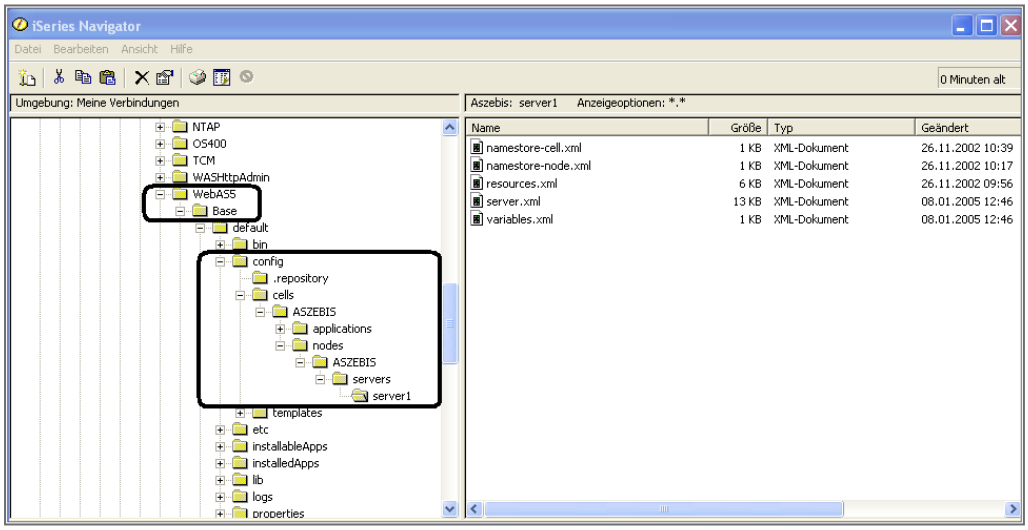
Unabhängig davon, welche Form der Administration Ihrer WAS-Umgebung Sie auch wählen – allen ist gemeinsam, dass die Konfigurations-Einstellungen immer in Konfigurations-Dateien gespeichert werden. Die Konfigurations-Angaben sind in so genannten Konfigurations-Dokumenten abgelegt. Meist handelt es sich dabei um XML-basierte Dateien (wie sie exemplarisch in der Abbildung „Inhalt der admin-authz“ zu sehen sind), mit denen unter anderem auch die verschiedenen Hierarchien der WAS-Installation abgebildet werden.

Der Aufbau der Konfigurations-Dateien ermöglicht unter anderem auch die Abbildung von komplexeren WAS-Umgebungen in Form der Network Deployment Installation oder auch der Cluster-Implementierung. Die hierarchische Struktur der Konfigurations-Daten ermöglicht so zum Beispiel auf der Knoten-Ebene die Replizierung der Einstellungen. Der hierarchische Aufbau der Konfigurations-Dokumente variiert zwischen den verschiedenen Versionen. Gerade mit der Einführung der Version 6 des WAS wurden, bedingt durch die Profile, welche als zusätzliche administrative Einheit eingeführt wurden, auch im Bereich der Administration einige Veränderungen eingeführt. Diese wirken sich natürlich auch auf die Konfigurations-Dokumente aus.

Schauen wir uns einmal den Aufbau der Konfigurations-Dokumente an.

An oberster Stelle der Konfigurations-Einstellungen befindet sich das Verzeichnis „cells“. In diesem finden wir für jede konfigurierte Zelle einen Untereintrag, welcher der Bezeichnung der Zelle entspricht.

Die folgende Abbildung zeigt den Aufbau dieser Zellen-Informationen für eine Base-Edition des WAS in der Version 5.



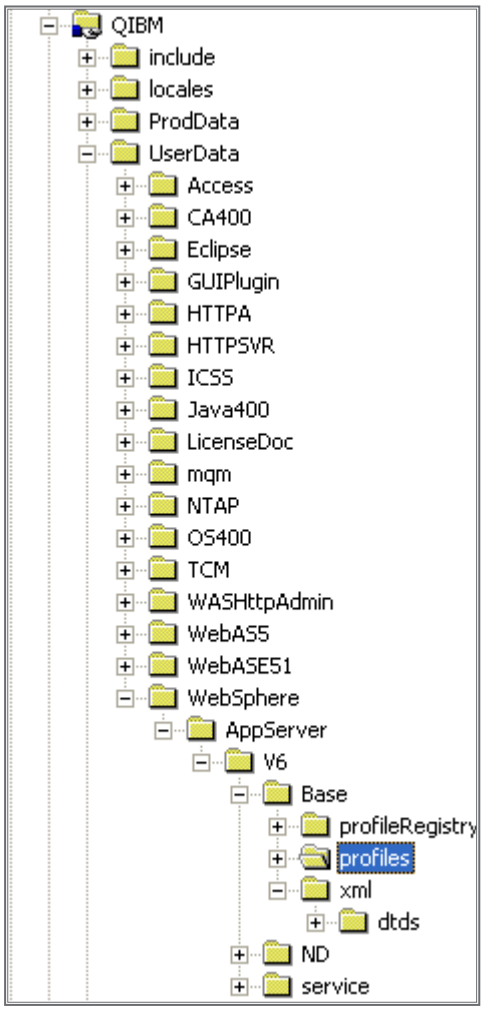
Struktur der Konfigurations-Dokumente

5.2.4
Seite 4

Wie man anhand der Abbildung sehen kann, befindet sich in dem Unterverzeichnis „config“ unter anderem auch der Eintrag „cells“. In diesem Verzeichnis befinden sich die einzelnen Zellen – in unserem Beispiel ist lediglich eine Zelle „ASZEBIS“ vorhanden. Natürlich können Sie auch mehrere organisatorische Einheiten in Form von verschiedenen Zellen definieren.

Jede Zelle besteht aus den ausführbaren Anwendungen (Verzeichnis „applications“) und den einzelnen Knoten (Nodes).

Je nach verwendeter Version (WAS in der Version 5 oder in der Version 6) kann der Aufbau der Hierarchie der Konfigurations-Dateien variieren. Die nachfolgende Abbildung zeigt den grundsätzlichen Aufbau der Konfigurations-Elemente einer WAS 6 Installation.



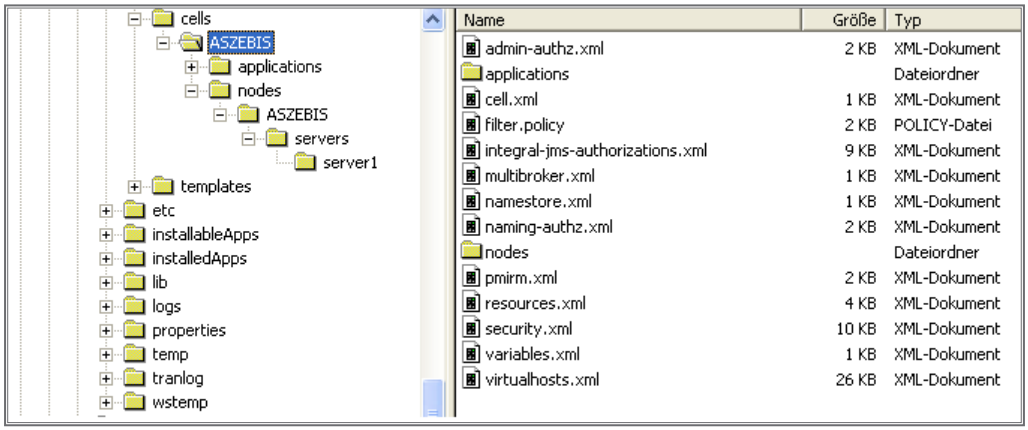
WAS 6 Konfigurations-Struktur

Die Installation der Version 6 wird in dem Verzeichnis „WebSphere/AppServer/V6“ realisiert. Unterhalb dieses Verzeichnisses befinden sich dann, je nachdem ob die Base-Edition (Base) oder die Network Deployment Edition (ND) installiert wurde, eines oder mehrere Unterverzeichnisse. In der folgenden Abbildung sehen Sie die Konfigurations-Struktur eines WAS in der Version 6. Auf diesem System sind beide Editionen (Base und ND) installiert. Die jeweiligen Konfigurations-Einstellungen befinden sich dann in dem Verzeichnis „profiles“.

Bezogen auf die Konfigurations-Elemente eines WAS 5-Server finden wir die drei Haupt-Ebenen:

cells

Beinhaltet die Konfigurations-Einstellungen auf Zellenebene. Dieses Verzeichnis besteht aus verschiedenen XML-Konfigurations-Dateien, die neben den sicherheitsrelevanten Informationen (Datei „security.xml“) unter anderem auch die Definition der virtuellen Hosts (virtualhosts.xml) enthält.



Inhalt des Verzeichnisses cells

nodes

Dieses Verzeichnis bildet die Knotendefinitionen der WAS-Konfiguration ab. Für jeden definierten Knoten existiert ein eigenes Unterverzeichnis, über das die Konfigurations-Einstellungen vorgenommen werden können. In dem Unterverzeichnis „nodes“ können dieselben Konfigurations-Elemente vorhanden sein, wie sie auch auf Ebene der Zelle existieren. So kann zum Beispiel die Datei „ressources.xml“ sowohl auf Zellen-Ebene, aber auch auf Knoten-Ebene vorhanden sein. In einem solchen Fall überschreiben die Angaben auf Knoten-Ebene die Definitionen der Zellen-Angaben.

5.2.4
Seite 6

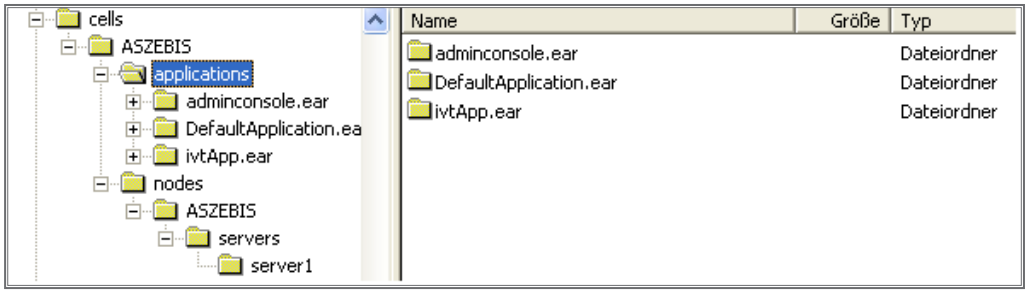
servers

Dieses Verzeichnis beinhaltet die einzelnen Server, welche in dieser Hierarchie erstellt wurden.

Diese zuvor genannten Teilbereiche sind charakteristisch für die drei Konfigurations-Ebenen innerhalb einer WAS 5-Umgebung. Zusätzlich existiert für die auf dem Server installierten Anwendungen ein weiteres Verzeichnis:

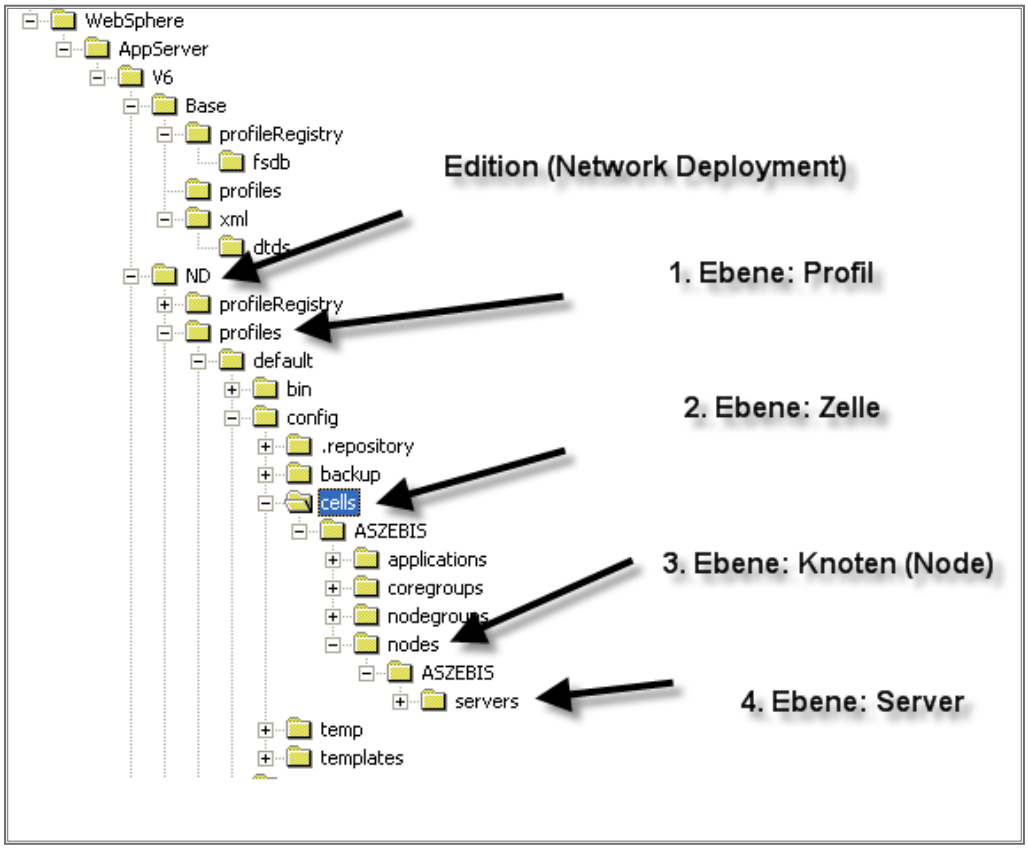
applications

In diesem Verzeichnis sind alle auf diesem WAS-Server installierten Anwendungen enthalten. Unter anderem befinden sich hier auch die Standard-Anwendungen in Form der Administrations-Konsole. Ausführbare Anwendungen sind mit der Endung „.ear“ in diesem Verzeichnis gespeichert. Zusätzlich befinden sich die installierten Anwendungen auch in dem Verzeichnis „installed Apps“, das als Unterverzeichnis der WAS-Konfiguration existiert. Jede Anwendung wird in ihrem eigenen Unterverzeichnis gespeichert. So lassen sich auch komplexere Anwendungen strukturiert abbilden.



Installierte Anwendungen

Bei einer WAS 6-Umgebung finden wir dieselben Komponenten wieder. Allerdings ist diesen eine weitere Stufe in Form eines Profils vorgeschaltet, wie die folgende Abbildung zeigt:



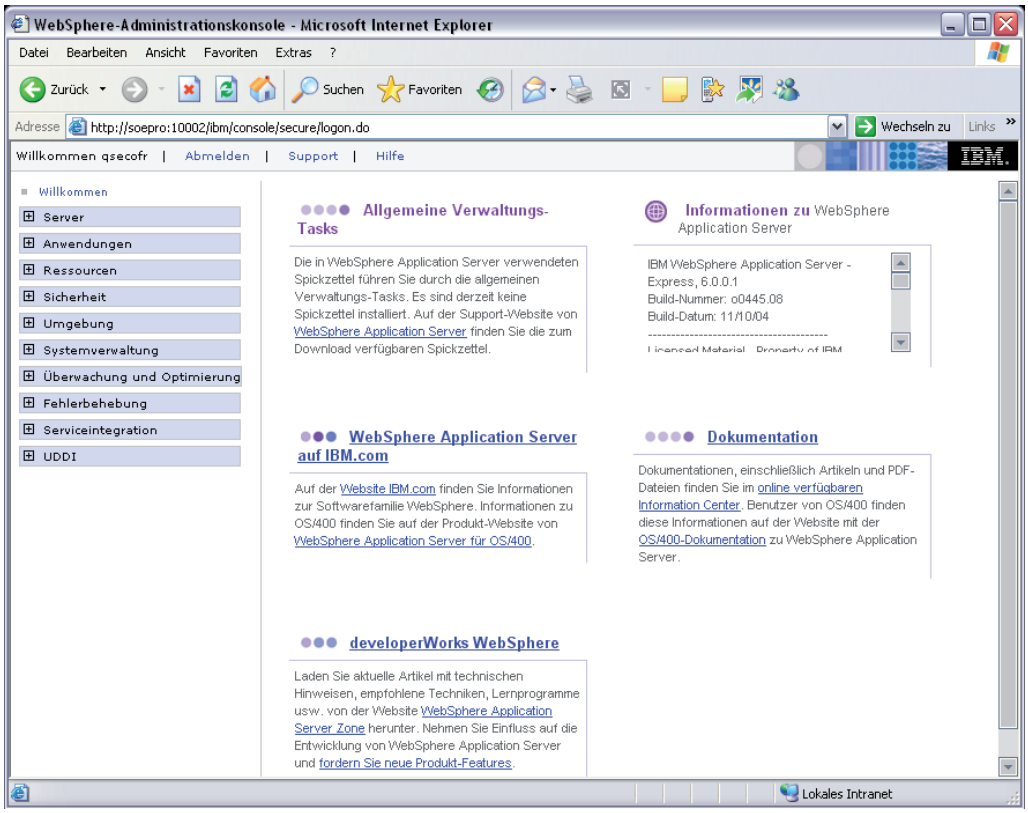
Konfigurations-Ebenen eines WAS 6-Server

Je nachdem, welche Änderungen an der Konfiguration bzw. welche administrativen Aufgaben ausgeführt werden sollen, können oder müssen diese auf unterschiedlichen Konfigurationsebenen durchgeführt werden. Dabei ist unter anderem auch zu beachten, dass Konfigurations-Einstellungen „doppelt“ vorhanden sein können, nämlich auf zwei unterschiedlichen hierarchischen Ebenen. Dabei ist wichtig zu wissen, welche der Ebenen für die jeweiligen Anforderungen bei der Ausführung des WAS verwendet wird. So können zum Beispiel Konfigurations-Einstellungen auf Profil-Ebene als allgemein gültige Einstellungen für alle Bereiche (also auch alle Zellen und Knoten) einer WAS-Installation gemeinsam verwendet werden. Abweichungen an diesen Konfigurations-Einstellungen können auf den tieferen Ebenen realisiert werden, indem – abweichend zu den allgemeinen Vorgaben – die individuellen Konfigurations-Angaben für zum Beispiel einen Knoten auf Knotenebene definiert werden. Dabei „überschreiben“ diese Angaben auf Knotenebene dann die Einstellungen der allgemeinen Ebene – nämlich in diesem Beispiel des Profils.

5.2.4
Seite 8

Die Administrationskonsole des WAS

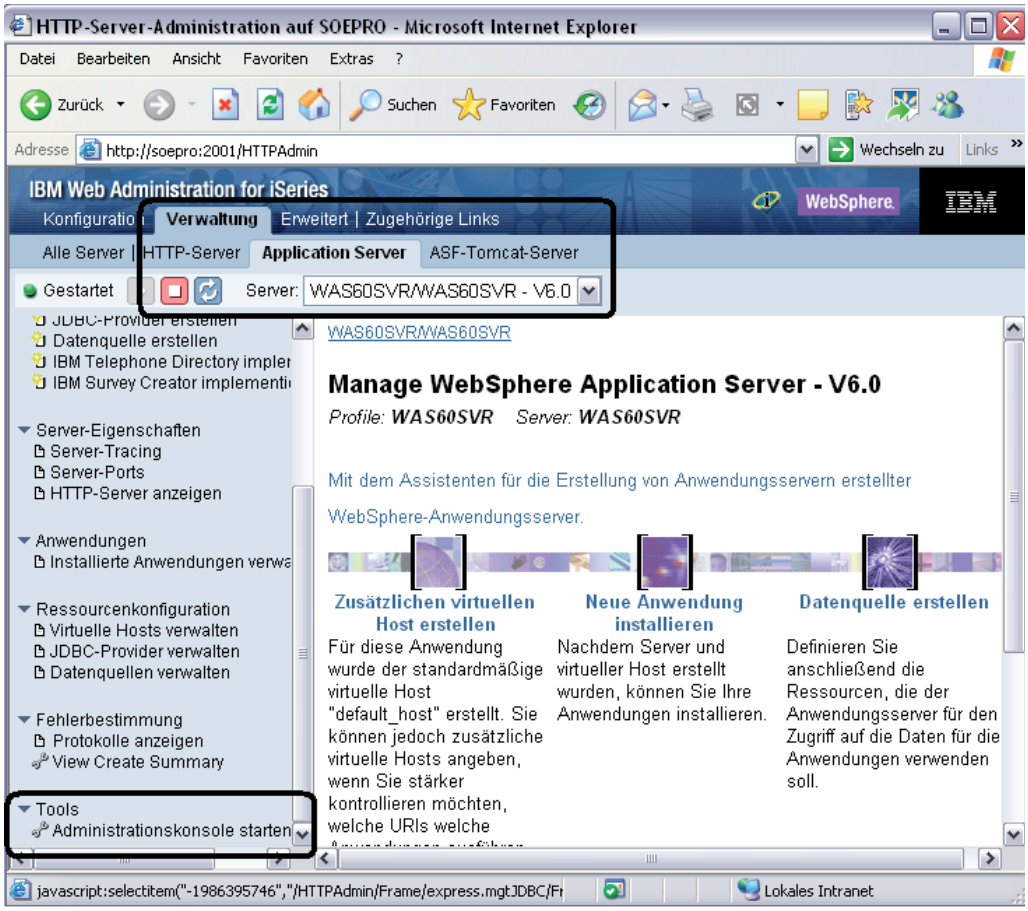
Seit der Version 6 des WebSphere Application Servers (kurz: WAS) hat IBM die Verwaltung dieses Servers in eine eigene Administrationskonsole übertragen. Diese Konsole dient der Verwaltung des oder der WAS-Server und bietet auf der Willkommensseite per Standard auch Zugang zu den wichtigsten Informationsquellen rund um das Thema WAS.



Der Startbildschirm der Administrationskonsole

Diese Konsole lässt sich entweder über den allgemeinen http-Server-Verwaltungsbereich öffnen oder durch die Direkteingabe der URL für die Startseite der Adminstrationskonsole. In der folgenden Abbildung sehen Sie den Start über den http-Verwaltungsbereich. Mit einem Klick auf den Eintrag „Administrationskonsole starten“ im linken Navigationsbereich wird die Adminstrationskonsole gestartet.

5.2.4.1
Seite 2

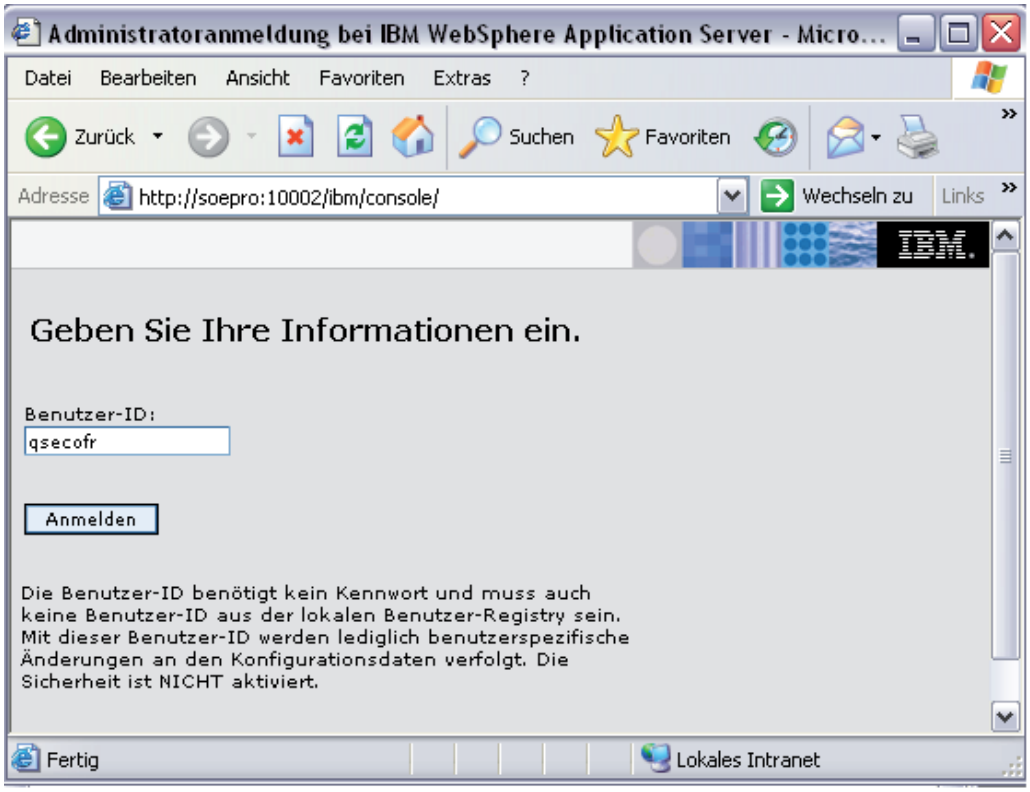


Starten der Adminstrationskonsole aus dem http-Verwaltungsbereich

Alternativ dazu können Sie auch direkt die URL für die Adminstrationskonsole eingeben. In Abhängigkeit des von Ihnen bei der Konfiguration des Servers festgelegten Ports müssen Sie diesen in dem folgenden Beispiel ersetzen, um den Verwaltungsbereich Ihres WAS-Servers zu starten. Bei einer Standardinstallation wird der Port 9060 als Verwaltungsport für den WAS verwendet.

http://servername:port/ibm/console/

Nach der Starteingabe müssen wir uns mit einem Benutzer im Verwaltungsbereich anmelden:



Anmeldung

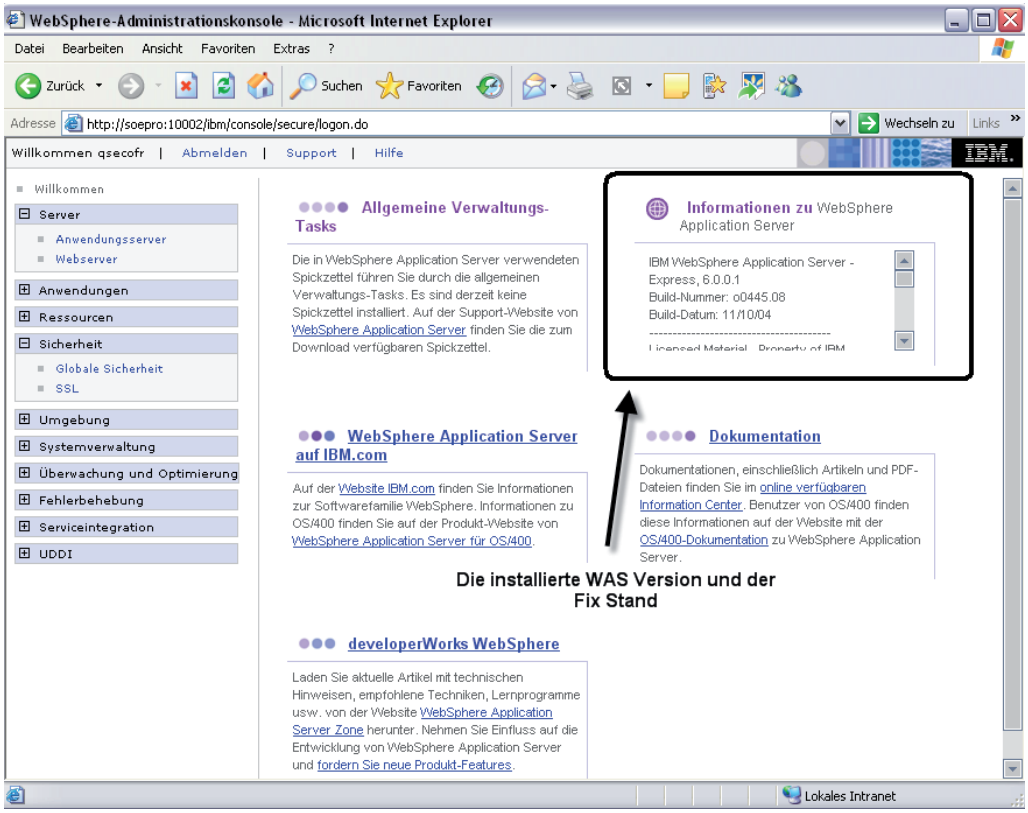
Für die Anmeldung können wir irgendeinen Namen eingeben, denn es erfolgt keine Prüfung, ob es sich bei dem angegebenen Wert tatsächlich um eine Benutzer-ID auf dem System i5 handelt. Dieser eingegebene Wert dient lediglich der Protokollierung von Änderungen an den Konfigurationseinstellungen des WAS. Erst wenn Sie die in dem WAS enthaltenen Sicherheitseinstellungen definiert haben, ist der Zugang zu diesem allgemeinen Verwaltungsbereich nur noch mit einem gültigen Benutzer und dem dazugehörigen Kennwort möglich.

Tipp

Sie sollten in jedem Fall die Sicherheitseinstellungen der WAS-Administrationskonsole so ändern, dass eine Anmeldung an dem Server ohne eine entsprechende Berechtigung nicht möglich ist. Denken Sie daran: Dieser Bereich ist die Steuerzentrale des WAS. Ein Missbrauch kann, bewusst oder unbewusst zu einem Systemausfall führen! Die erforderlichen Einstellungen können Sie in dem Bereich „Globale Sicherheit“ vornehmen.

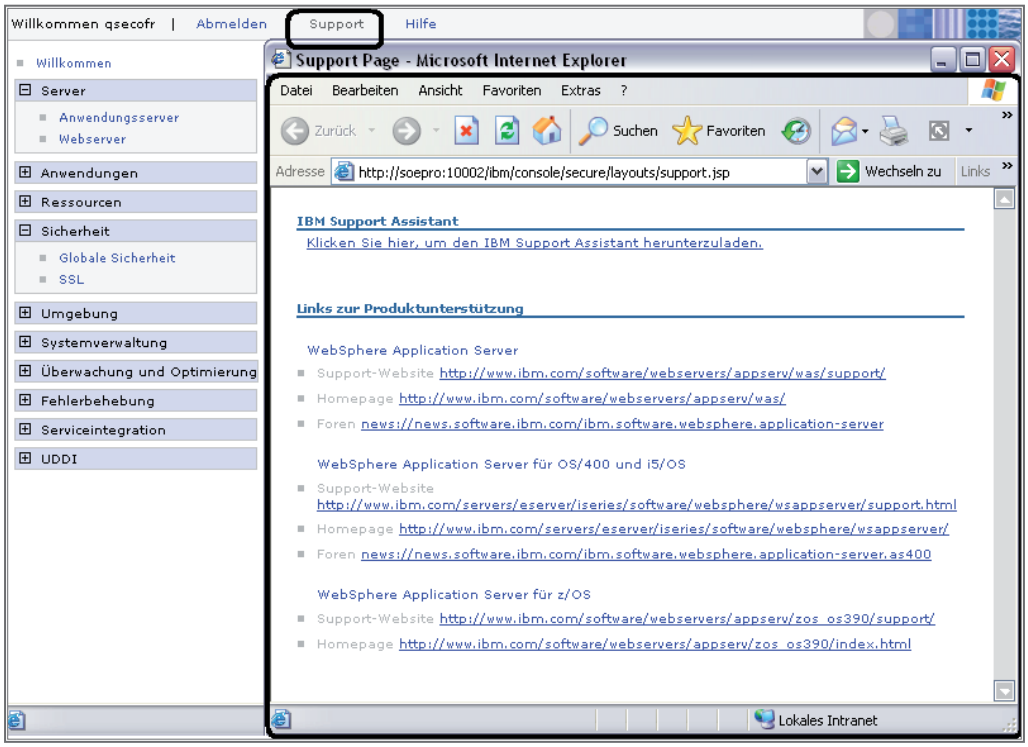
5.2.4.1
Seite 4

Mit der Willkommensseite erhalten wir einen direkten Zugriff auf weiterführende Informationen im Internet. Außerdem finden wir hier auch die aktuell installierte Version des WAS, die neben der Versionsnummer auch den installierten Fixstand beinhaltet. Dieser ist zum Beispiel bei Supportanfragen wichtig.



Die Willkommensseite

Im oberen Bereich der Seite können wir auf zusätzliche Hilfefunktionen zugreifen. Außerdem können wir mit einem Klick auf „Support“ direkt Wartungsanfragen stellen. Mit Hilfe eines zusätzlichen Tools in Form des IBM Support Assistant lassen sich Supportanfragen automatisch an den Servicegeber übertragen. Zusätzlich befindet sich in diesem Bereich auch eine Fülle an hilfreichen Links, über die wir zusätzliche Informationen zu dem Betrieb des WAS finden.



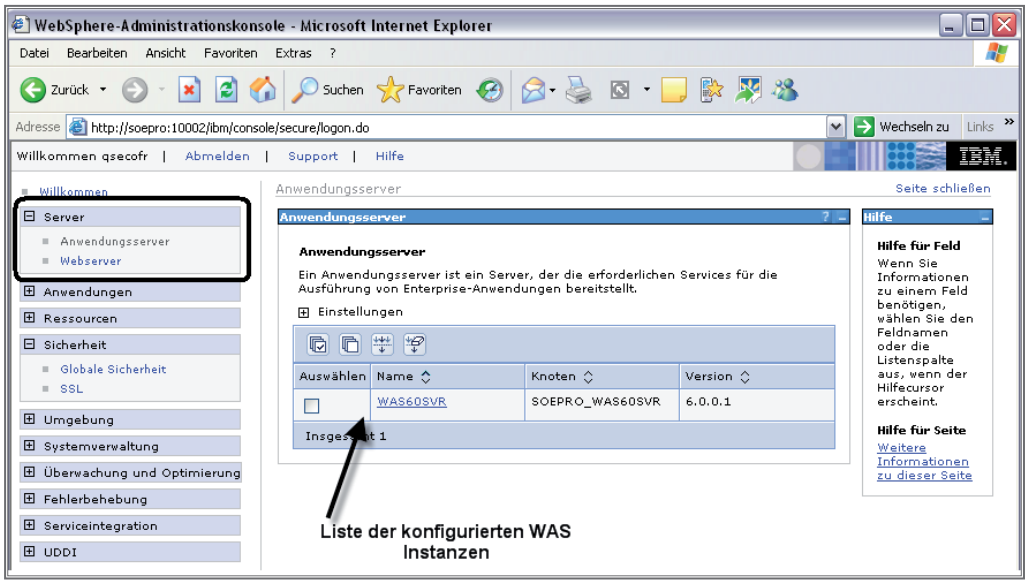
Der Supportbereich

In dem linken Bereich der Administrationskonsole befindet sich der so genannte Navigationsbereich, über den wir Zugang zu den verschiedenen Teilbereichen der Verwaltung des WAS-Servers erhalten.

Wir wollen nicht jeden einzelnen dieser Bereiche in diesem einleitenden Teil behandeln. Mit dem Abschnitt „Server“ erhalten wir Zugriff auf die verschiedenen WAS und Webserver, die auf dem Rechner installiert sind. Diese beiden Teilbereiche lassen sich getrennt voneinander verwalten. In Abhängigkeit der auf dem Rechner installierten bzw. konfigurierten WAS-Server kann die Übersicht mehrere Einträge beinhalten, wobei jeder einzelne einen auf dem Rechner konfigurierten Server darstellt. Es macht in komplexen Umgebungen durchaus auch Sinn, mehrere WebSphere Application Server bzw. mehrere Instanzen dieser Server zu konfigurieren. Damit lassen sich zum Beispiel die unterschiedlichen Anwendungen separat auf eigenen Servern betreiben. Dies hat durchaus Vorteile – denken wir allein an die Anforderungen im Hinblick auf Hochverfügbarkeit von Webshops, die auf Basis eines solchen Servers ausgeführt werden können. Ein Webshop erfordert zudem spezielle Einstellungen unter Berücksichtigung der Sicherheit. Es versteht sich von selbst, dass ein solcher Webshop anders zu behandeln ist, wie beispielsweise ein Management Verwaltungssystem, das ebenfalls auf Basis eines Anwendungsservers in einer weiteren WAS-Instanz ausgeführt wird. Die Liste der möglichen Gründe für

5.2.4.1
Seite 6

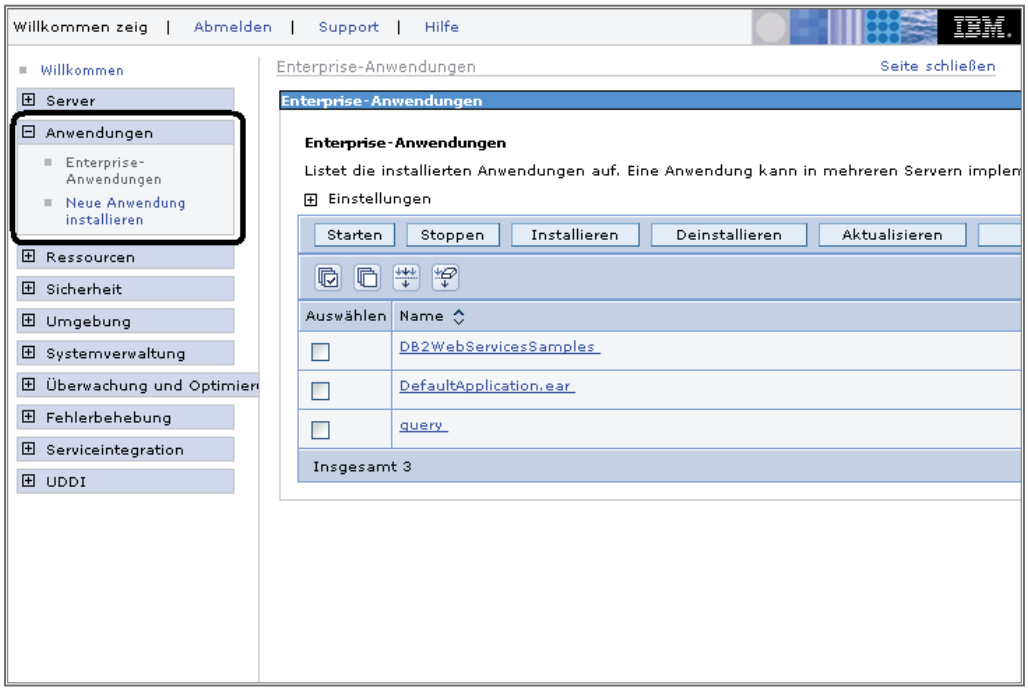
das Betreiben verschiedener WAS-Instanzen ist schier endlos. Begnügen wir uns also mit der Tatsache, dass in dem Bereich „Anwendungsserver“ verschiedene Server aufgelistet sein können, deren Administration jeweils separat durchgeführt werden kann.



Zu verwaltende Anwendungsserver

In dieser Auflistung können wir den zu verwaltenden Server auswählen und diese im Detail verwalten. Neben der Verwaltung der Anwendungsserver können wir durch einen zweiten Eintrag in dem Navigationsbereich auch die konfigurierten Webserver verwalten, welche zusammen mit den Anwendungsservern eingesetzt werden.

Das wichtigste bei einem Anwendungsserver sind die Anwendungen, die auf dem Server ausgeführt werden sollen. Bevor eine solche Anwendung auf dem Server aktiviert werden kann, ist diese zunächst dort zu aktivieren bzw. zu installieren. Bei diesem Prozess werden die erforderlichen Komponenten der Anwendung in den WAS eingebunden und stehen dann, sobald die Anwendung gestartet wurde, für die Anwender bereit. Die auf dem Server installierten Anwendungen können in dem nächsten Teilbereich „Anwendungen“ verwaltet werden. Dieser Bereich dient neben der Verwaltung der bereits installierten Anwendungen auch der Installation neuer Anwendungen:



Anwendungsverwaltung

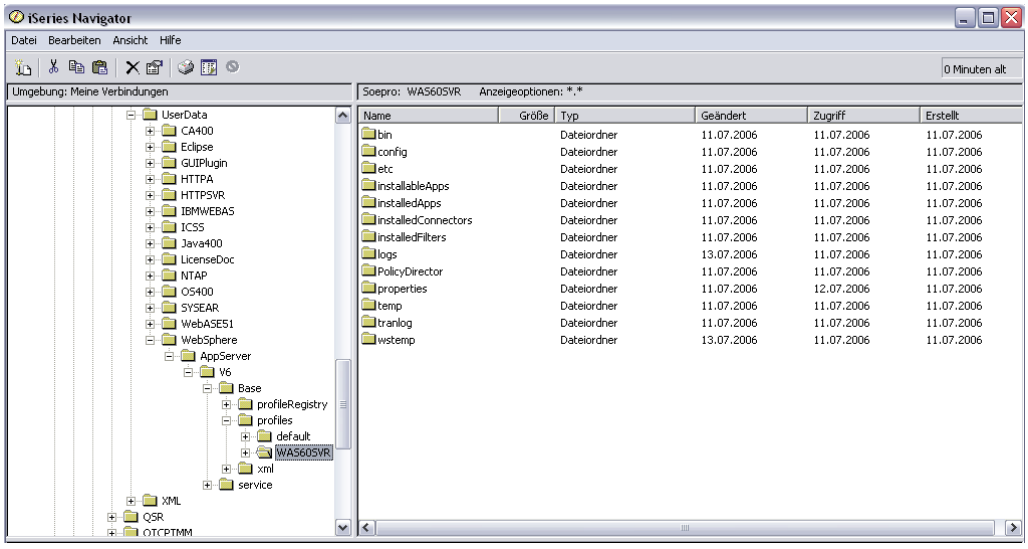
Während der Installation des Servers bietet IBM optional auch die Installation einiger Beispielanwendungen an: Wenn Sie diese Option selektiert haben, dann erscheinen einige Anwendungen, die in dem Bereich „Neue Anwendungen installieren“ für eine Installation und eine spätere Nutzung ausgewählt werden können.

Neben der Installation steht in diesem Bereich auch das Aktualisieren oder beispielsweise das Deinstallieren von Anwendungen zur Auswahl bereit. Wir werden uns mit den Details zu der Anwendungsverwaltung in einem eigenen Kapitel eingehend beschäftigen.

Alle Verwaltungsarbeiten, die mit der Administrationskonsole durchgeführt werden, finden sich in den jeweiligen Konfigurationsdateien des WAS wieder. Alle Konfigurationseinstellungen werden dabei in Form von XML-Dateien in dem jeweiligen Stammverzeichnis des WAS gespeichert. Für jedes WAS-Profil werden diese Einstellungen in dem folgenden Verzeichnispfad abgelegt:

UserData/WebSphere/AppServer/V6/Base/profiles

5.2.4.1
Seite 8



Die Konfigurationsdateien des WAS 6

Diese Pfadangabe unterscheidet sich von den Konfigurationseinstellungen auf anderen Plattformen und bezieht sich nur auf die Installation des WAS auf dem System i5. Geübte Administratoren können die gewünschten Einstellungen auch durch eine native Eingabe der Änderungen in den Konfigurationsdateien durchführen. Allerdings fehlen bei solchen Änderungen dann etwaige Abhängigkeitsprüfungen, die sich unter Umständen dann negativ auf die Ausführung des WAS auswirken könnten. Deshalb empfiehlt es sich, für die Änderungen an den WAS-Konfigurationseinstellungen ein Tool wie die Administrationskonsole zu verwenden.

Anwendungsverwaltung

Das wichtigste bei dem Betrieb eines Anwendungsservers sind die Anwendungen, die auf dem Server ausgeführt werden. In der Regel sind dies „moderne" Java- oder auch J2EE-Anwendungen, die auf einem WAS-Server ausgeführt werden. Aber nicht nur solche Anwendungen lassen sich auf Basis eines WAS-Servers betreiben. Ein solcher Server dient auch als Grundlage für weitere Produkte aus dem Hause IBM, wie zum Beispiel das Ausführen eines Portal-Servers oder auch die Bereitstellung von Diensten und Funktionen im Workplace-Umfeld.

Doch was sind das eigentlich für Anwendungen, die auf dem WAS ausgeführt werden können? Neben den zuvor bereits genannten Anwendungsbereichen ist es für uns als i5-Betreiber sicher auch interessant, dass wir, basierend auf den alt gewachsenen RPG- und Cobol-Anwendungen moderne Oberflächen oder Front Ends schreiben können, mit deren Hilfe eine Web-basierte Oberfläche eingesetzt werden kann. Mit Hilfe von verschiedenen Werkzeugen, die IBM zum Beispiel mit dem WebSphere Development Studio Client (kurz: WDSC) bereitstellt, lassen sich die erforderlichen Schnittstellen generieren, welche es uns erlauben, die bisherigen 5250-Anwendungen mit nur wenigen Anpassungen in Verbindung mit „modernen" Oberflächen einzusetzen. Natürlich stehen uns in diesem Bereich der Anwendungen auch alle anderen Anwendungsmodelle zum Einsatz zur Verfügung, so dass wir natürlich auch die modernsten Java- und J2EE-Anwendungen auf dem WAS installieren und ausführen können. Auch die neuerdings nativen auf der i5 verfügbaren PHP-Anwendungen lassen sich ideal auf dem WAS ausführen.

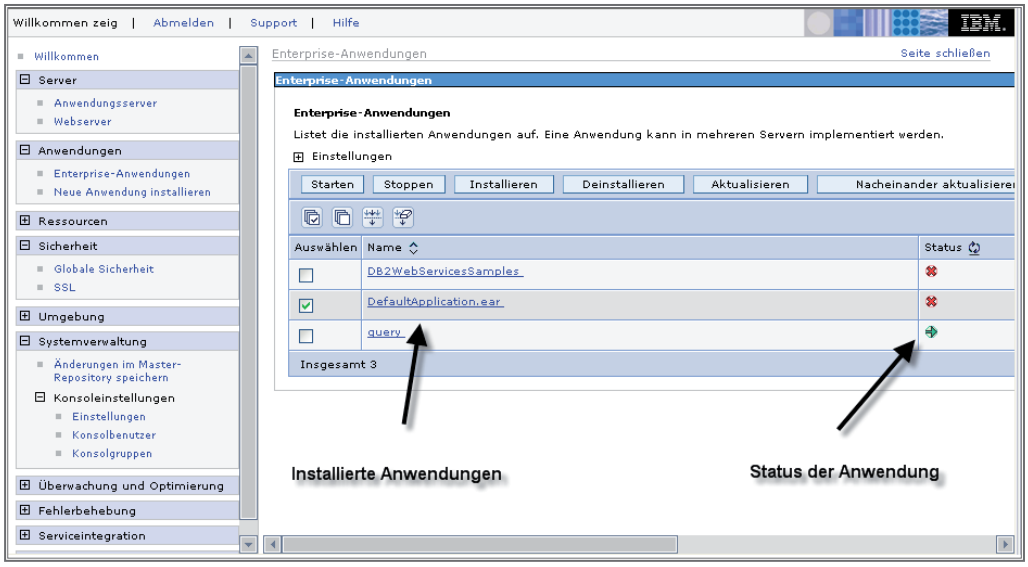
Natürlich sind bei der Aufzählung von potentiell auf dem WAS ausführbaren Anwendungen auch WebServices oder SOA-Projekte nicht zu vergessen. Diese lassen sich auf Basis von WebSphere realisieren.

An dieser Stelle wollen wir uns mit der Bereitstellung bzw. der Installation von Anwendungen auf dem Server beschäftigen. Grundsätzlich muss eine Anwendung auf dem Server installiert (deployed) werden, damit sie auch tatsächlich genutzt werden kann. Dabei handelt es sich im Wesentlichen um einen recht einfachen Prozess.

Anwendungen lassen sich mit Hilfe der Administrationskonsole verwalten. Neben der Installation von neuen Anwendungen lassen sich in dem Anwendungsbereich der Konsole auch bereits installierte Anwendungen verwalten. Diese können gestartet oder gestoppt, aber auch aktualisiert oder deinstalliert werden.

5.2.4.2
Seite 2

Per Standard liefert IBM einige Beispielanwendungen aus, welche auf dem Server bei der Installation optional installiert werden können. Wählen Sie für die Verwaltung dieser Anwendungen im Navigationsbereich den Eintrag „Anwendungen / Enterprise-Anwendungen“. Wenn diese Beispielanwendungen installiert wurden, dann erscheinen diese in dem Verwaltungsbereich, wie es die folgende Abbildung zeigt.



Anwendungen verwalten

In dem rechten Anzeigenbereich finden wir ein Symbol, das Auskunft über den aktuellen Status der jeweiligen Anwendung gibt. Ein grüner Pfeil bedeutet, dass diese Anwendung aktiv ist und für einen Einsatz bereitsteht. Ein rotes Kreuz bedeutet, dass diese Anwendung zwar auf dem Server installiert, aber derzeit deaktiviert ist.

Um eine solche Anwendung zu aktivieren, wählen wir den gewünschten Eintrag mit einem Klick auf die Spalte „Auswählen“ aus und starten die Anwendung anschließend mit einem Klick auf „Starten“.

Starten einer Anwendung

Das Starten der Anwendung kann je nach Anwendung variieren.

Die einzelnen Anwendungen lassen sich auch in Bezug auf deren Einstellungen auf dem WAS-Server verwalten. Wenn die Einstellungen einer Anwendung angepasst werden müssen, dann genügt ein Doppelklick auf die jeweilige Anwendung.

5.2.4.2

Seite 4

Installation neuer Anwendungen

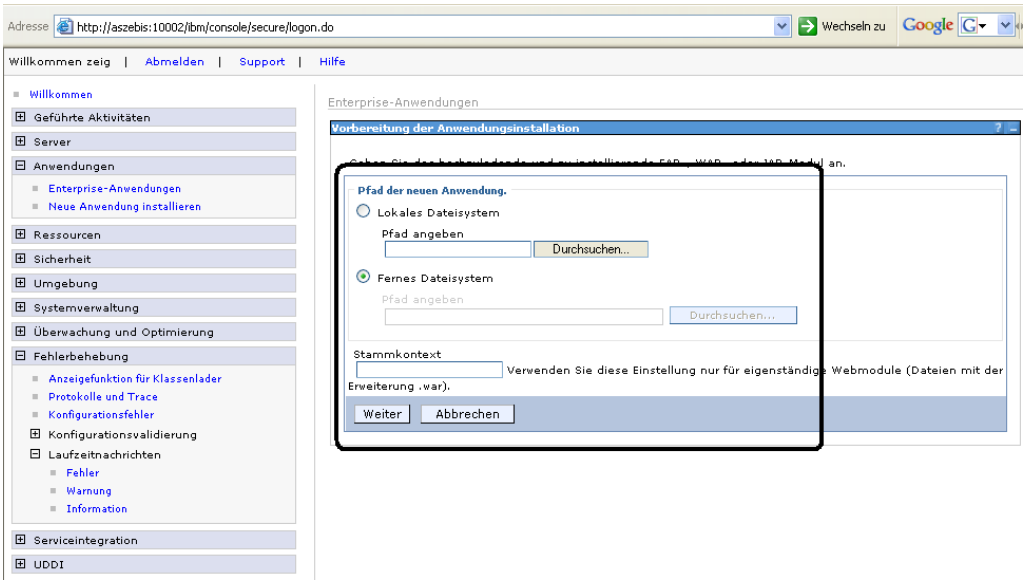
Neue Anwendungen lassen sich auch über die Administrationskonsole auf dem WAS installieren. Dazu steht im linken Navigationsbereich der Konsole der Abschnitt „Anwendungen / neue Anwendung installieren“ zur Verfügung:

Auswahl im Navigationsbereich

Bei der Auswahl der zu installierenden Anwendung gilt, dass es sich dabei wahlweise um ein Enterprise Archive (.ear Datei) oder aber um ein einzelnes Webmodul (.war Datei) handeln kann. Bei letzterer ist allerdings zu beachten, dass ein Stammkontext für diese .war Datei erforderlich ist.

Wir können wahlweise auf installierbare Anwendungen auf dem lokalen System zurückgreifen, oder aber auch eine bereitgestellte Anwendung von einem fernen System installieren.

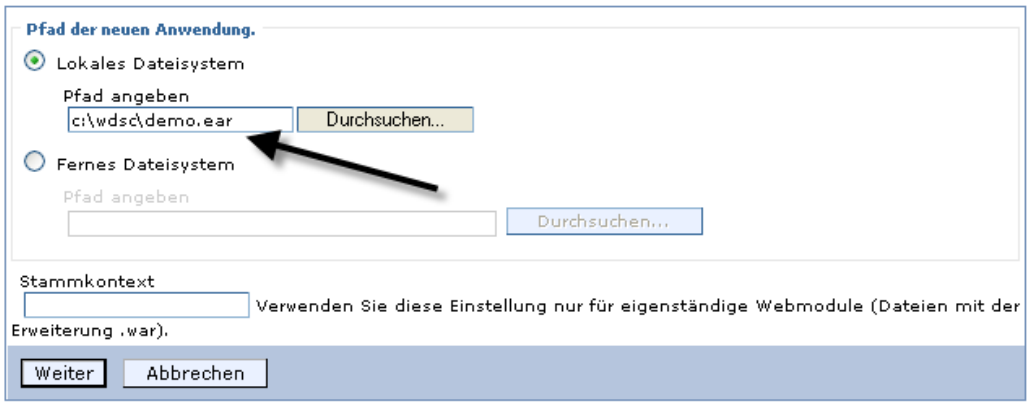
5.2.4.2.1
Seite 2



Installationspfad

Die Bezeichnungen sind an dieser Stelle ein wenig irreführend – denn „lokales Dateisystem“ bedeutet an dieser Stelle in der Tat der PC, auf welchem die Administrationskonsole ausgeführt wird und nicht etwa der Rechner, welcher zur Ausführung des WAS zum Einsatz kommt!

Sollte die neu zu installierende Anwendung also von einem PC oder einem Netzlaufwerk abgelegt worden sein, welches von dem PC aus verwendet werden kann, dann können wir mit einem Klick auf die Schaltfläche „Durchsuchen“ den Pfad und den Namen der zu installierenden Anwendung schnell ermitteln, oder diesen, falls er bekannt ist, direkt in dem dafür vorgesehenen Feld eingeben.

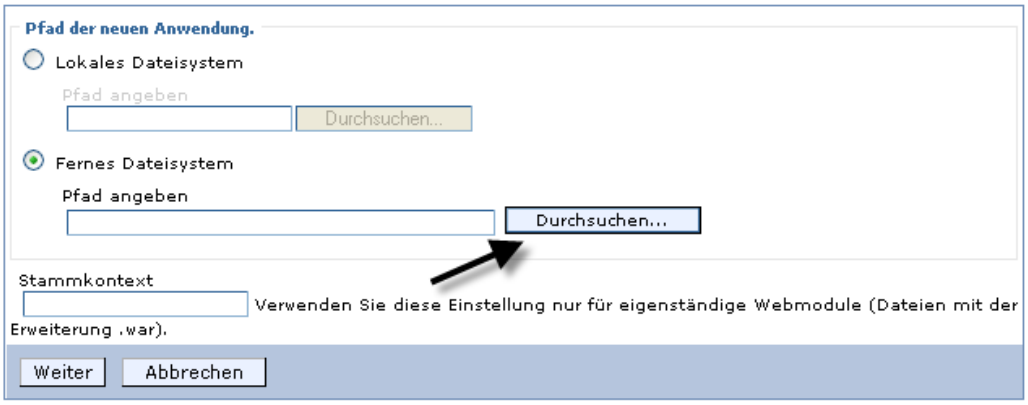


Verwendung des lokalen Dateisystems

Alternativ dazu können wir auch Anwendungen installieren, welche wir zuvor in das dafür vorgesehene Verzeichnis des WAS gestellt haben.

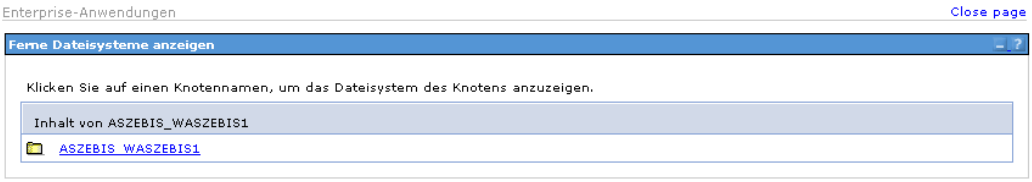
In unserem Beispiel haben wir die neu zu installierende Anwendung in Form einer EAR-Datei in das dafür vorgesehene Verzeichnis „installableApps“ der WAS-Installation gestellt.

Um diese Anwendung zu ermitteln, klicken wir auf die Schaltfläche „Durchsuchen“ neben dem Eintrag „Fernes Dateisystem“:



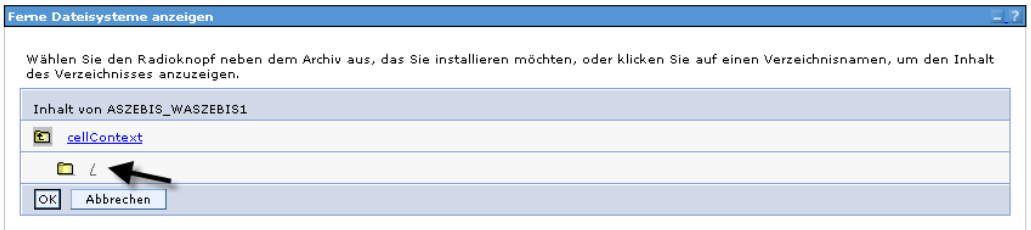
Verwendung des fernen Dateisystems

Nun bietet uns der Verwaltungsbereich eine Liste der verfügbaren fernen Systeme an. In unserem Fall wird der Knotenname des WAS angezeigt, den wir für die weitere Verarbeitung auswählen.



005 Fernes Dateisystem auswählen

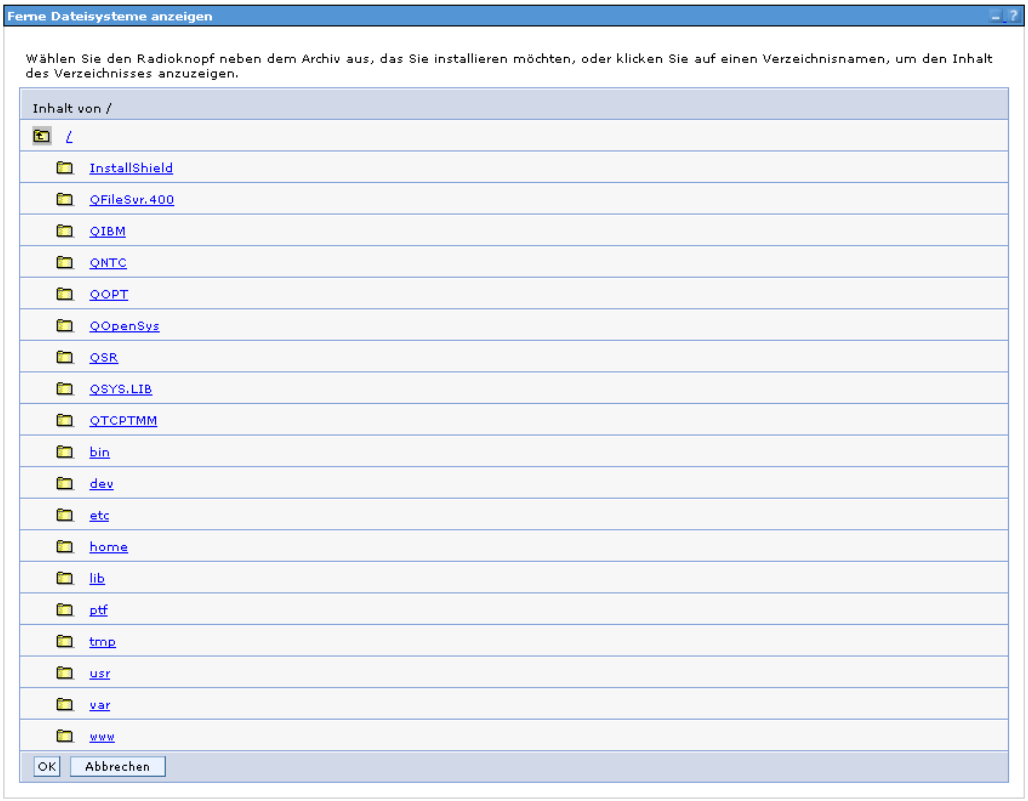
Die nächste Anzeige ist ein wenig gewöhnungsbedürftig (zumindest für i5-Anwender). Um eine Übersicht aller Verzeichnisse zu erhalten, die auf der i5 (genauer in dem IFS) verfügbar sind, klicken wir auf den unteren Eintrag:



Cell-Struktur

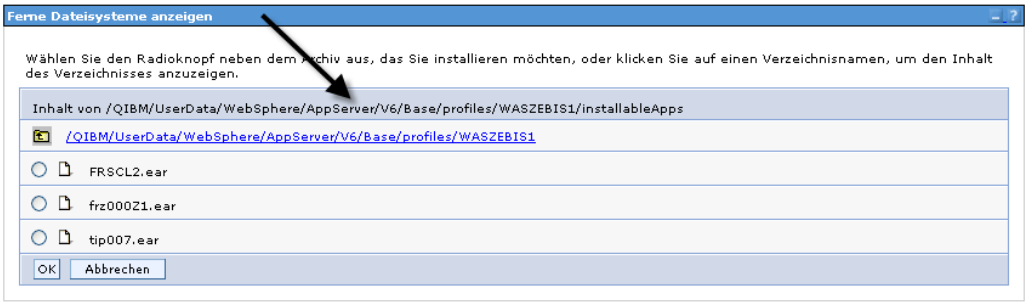
Damit gelangen wir in eine Auflistung der Ressourcen, welche auf der i5 zur Verfügung stehen.

5.2.4.2.1
Seite 4



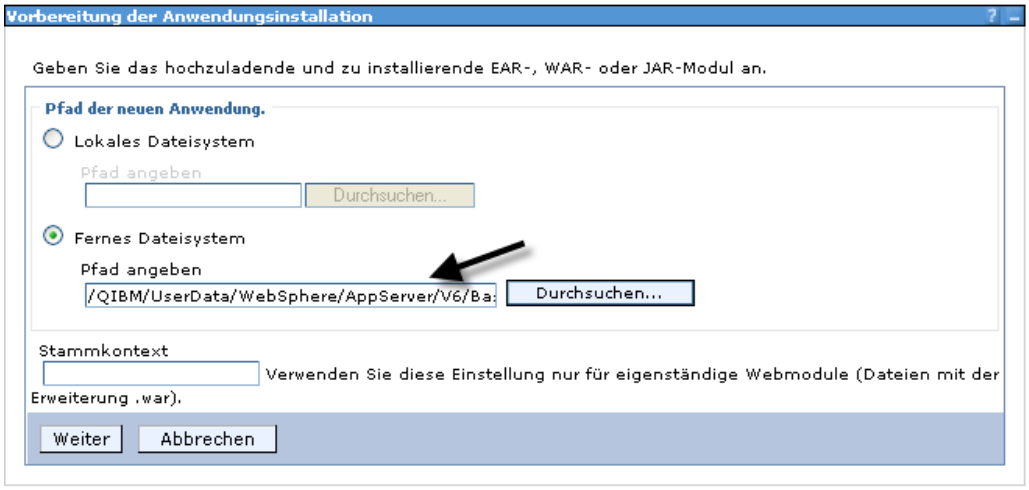
IFS-Inhalt

Ich möchte an dieser Stelle nicht jede weitere Ebene darstellen, sondern wechsle in das Unterverzeichnis qibm/UserData/WebSphere/AppServer/V6/Base/profiles/servername/installableApps (der Platzhalter „servername“ ist durch den Namen des Servers zu ersetzen, wie es das folgende Beispiel zeigt:



Installierbare Anwendungen

In unserem Beispiel können wir aus drei verschiedenen installierbaren Anwendungen auswählen. Wir klicken auf den Eintrag „tip007.ear“ und bestätigen diese Auswahl mit einem Klick auf die Schaltfläche „OK“. Damit wird unsere Auswahl in das Einstiegsfenster übernommen.



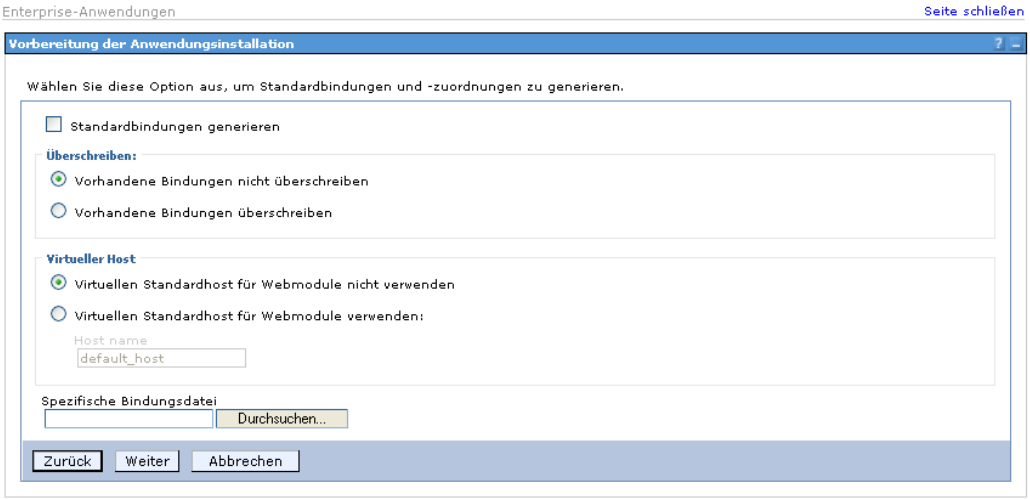
Übernommene Auswahl

Wie man unschwer erahnen kann: Eine Direkteingabe des Pfads mit der zu installierenden Anwendung ist natürlich auch möglich!

Der Bereich „Stammkontext“ ist nur für die Installation von .war Dateien anwendbar.

Um die neu zu installierende Anwendung auf dem Server verfügbar zu machen, klicken wir auf die Schaltfläche „Weiter“.

In der nächsten Anzeige können wir die Angaben zum virtuellen Host festlegen.



Weitere Angaben der zu installierenden Anwendung

Wenn wir die Option „Standardbindungen generieren“ selektieren, dann haben wir an dieser Stelle die Möglichkeit, eventuell noch fehlende Bindings und Mappings zu vervollständigen.

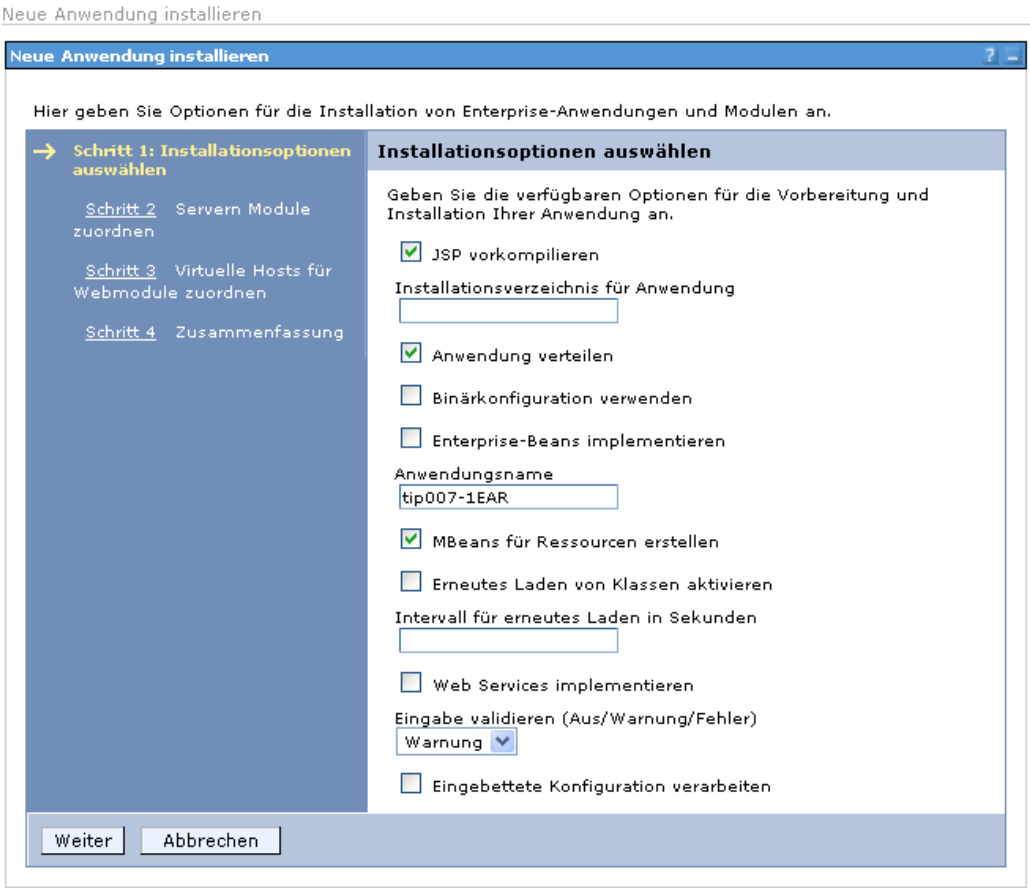
5.2.4.2.1
Seite 6

Anmerkung

Unter einer „Bindung“ verstehen wir die Umsetzung von lokalen auf einen globalen Namensbereich, während „Mapping“ die Zuordnung von einzelnen Modulen zu den Serverbereichen darstellt.

Wir verzichten an dieser Stelle auf die Möglichkeit der Generierung von Standardbindungen und setzen die Installation mit einem Klick auf „Weiter“ fort.

Als nächstes legen wir weitere Optionen für die Installation der Anwendung fest. Einige dieser Angaben haben wir bereits bei der Installation einer Anwendung über den Web-Administrator kennen gelernt.



Installationsoptionen auswählen

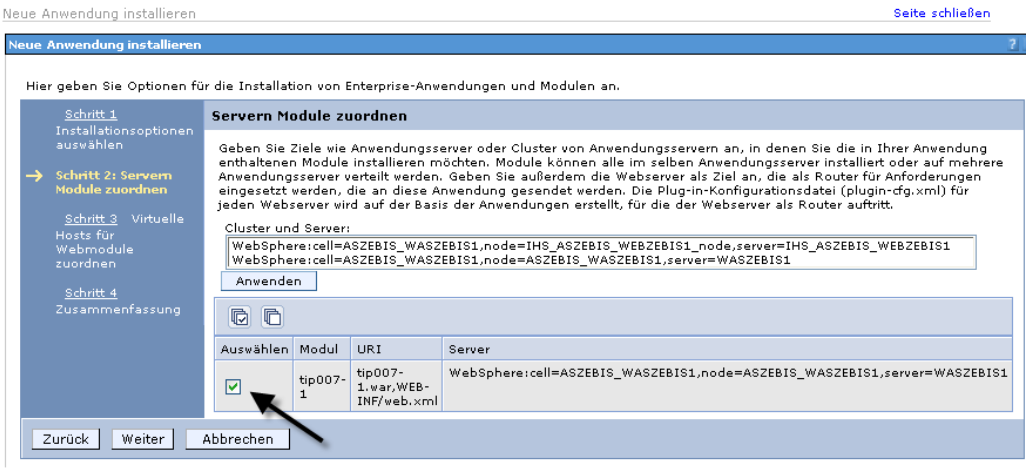
- JSP vorkompilieren
  Mit dieser Angabe legen wir fest, ob die mit dieser Anwendung genutzten JSPs bei der Installation übersetzt werden sollen, oder erst dann, wenn diese erstmals in einer Anwendung aufgerufen werden. Diese Einstellung kann sich beträchtlich auf die Geschwindigkeit auswirken!

- Installationsverzeichnis für Anwendung
  Hier können wir ein Installationsverzeichnis angeben, das von den Standardvorgaben abweicht.
- Anwendung verteilen
  Mit dieser Angabe steuern wir, ob die Anwendung konfiguriert wird oder ob eine Verteilung der Binaries auf die verschiedenen Knoten erfolgen soll.
- Binärkonfiguration verwenden
  Angabe, ob die Erweiterungen und Bindings aus der Datei deployment.xml verwendet werden sollen, oder ob diese Angaben aus der .ear Datei genutzt werden.
- Enterprise Beans implementieren
  Mit dieser Option legen wir fest, dass der Deployment Code der Enterprise Java Beans generiert werden soll.
- Anwendungsname
  Definiert den Namen der zu installierenden und später auszuführenden Anwendung.
- MBeans für Ressourcen erstellen
  Mit dieser Option legen wir fest, ob für die Teilbereiche der Anwendung in Form von z.B. JSPs oder Servlets MBeans (Managed Beans) angelegt werden sollen.
- Erneutes Laden von Klassen aktivieren
  Mit dieser Option legen wir fest, ob die Anwendungen auf Änderungen überprüft werden sollen. Wenn diese Option gewählt wurde, dann muss in dem Folgefeld „Intervall für erneutes Laden in Sekunden“ der Zeitwert entsprechend angegeben werden.
- Web-Services implementieren
  Hiermit definieren wir, ob für die Installation das Web Services Deploy Tool verwendet werden soll.
- Eingabe validieren
  Diese Option legt fest, ob die Referenzen der Ressourcen geprüft werden sollen. Wird eine fehlerhafte Referenz gefunden, dann wird die Installation der Anwendung unterbrochen, wenn der Wert in diesem Feld auf „Fehler“ gesetzt wurde. Bei der Auswahl „Warnung“ wird die Installation bei einem Fehler fortgesetzt – es erfolgt allerdings eine Protokollierung dieses Hinweises, so dass entsprechende Korrekturen möglich sind.
- Eingebettete Konfiguration vorbereiten
  Definiert, ob zusätzliche Informationen der EAR verarbeitet werden sollen.

5.2.4.2.1
Seite 8

Wenn alle erforderlichen Angaben getätigt worden sind, setzen wir die Installation der Anwendung mit einem Klick auf die Schaltfläche „Weiter“ fort. Für unser Beispiel übernehmen wir alle Voreinstellungen, wie sie von dem System vorgeschlagen wurden.

Damit gelangen wir in eine Auflistung der zu installierenden Anwendungen. Dort wählen wir „unsere“ Anwendung mit einem Klick in das Auswahlkästchen aus.

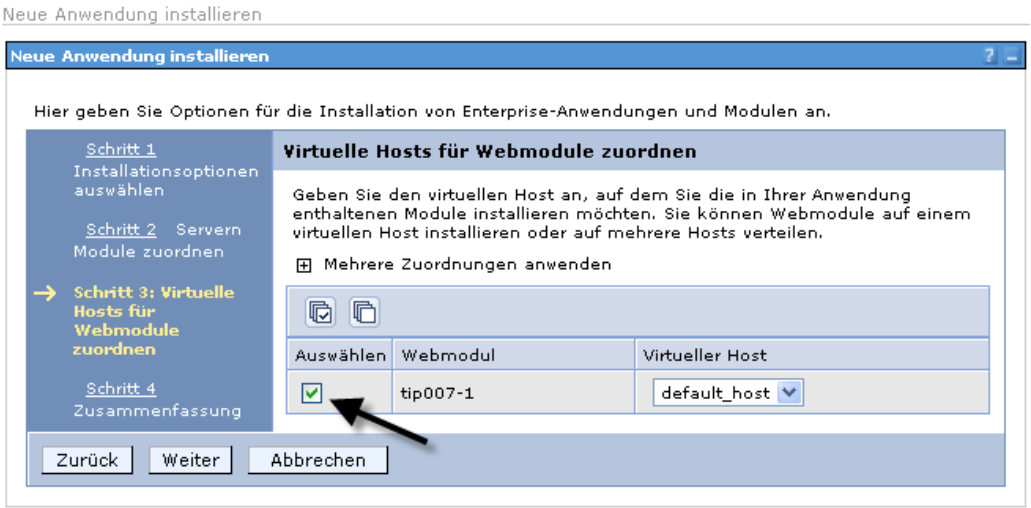


Server-Module zuordnen

In dieser Anzeige definieren wir die Aufteilung der Module, die bei Bedarf auch auf verschiedene Instanzen oder Cluster verteilt werden können.

Auch in dieser Anzeige klicken wir erneut auf „Weiter“.

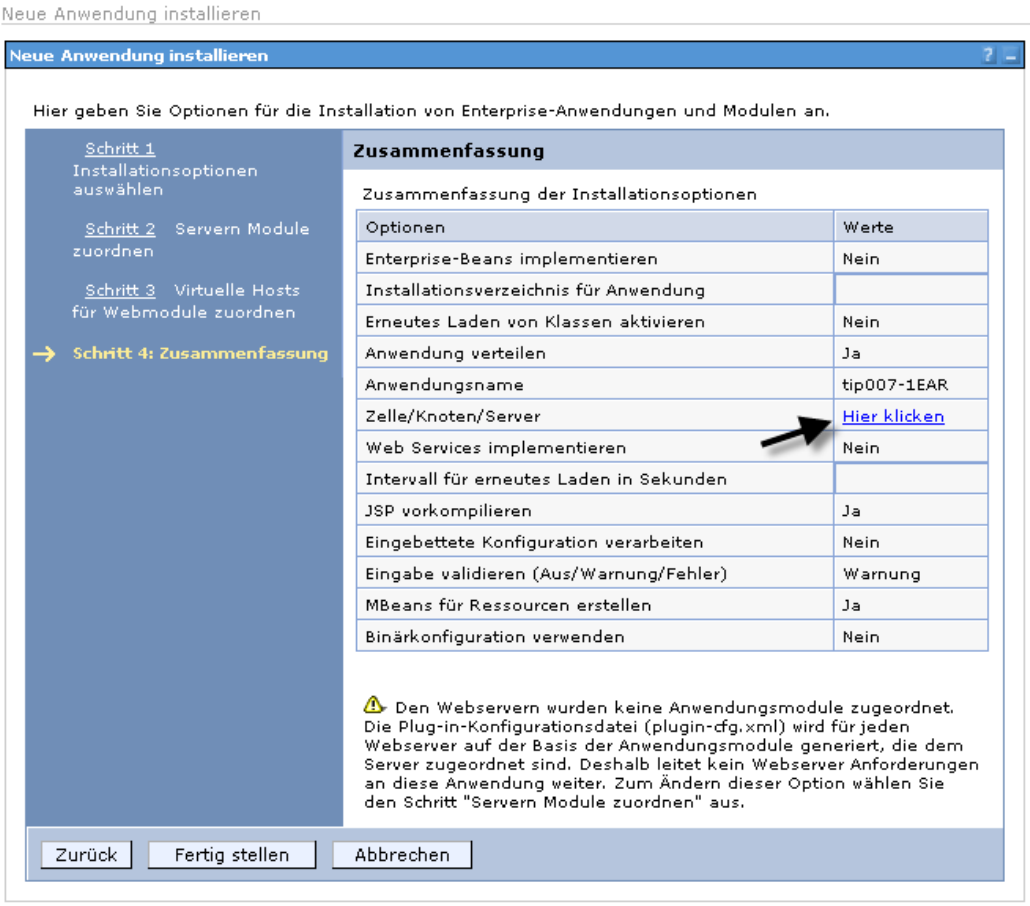
Als nächstes ordnen wir die neue Anwendung einem virtuellen Host zu. Auch hier ist es möglich, die Anwendung auf unterschiedliche virtuelle Hosts zu verteilen.



Zuordnung zu virtuellen Hosts

Denken Sie auch in dieser Anzeige an die Selektion „Auswählen“ und setzen Sie die Installation mit einem Klick auf „Weiter“ fort.

Damit haben wir die für die Installation erforderlichen Schritte abgeschlossen und können nun die Anwendung auf dem WAS installieren. Die Administrationskonsole bietet nochmals eine Zusammenfassung unserer Angaben. Sollten noch Veränderungen erforderlich sein, dann können wir diese durchführen, indem wir auf „zurück“ klicken und damit Schritt für Schritt in der Liste der Definitionsfenster zurück springen.



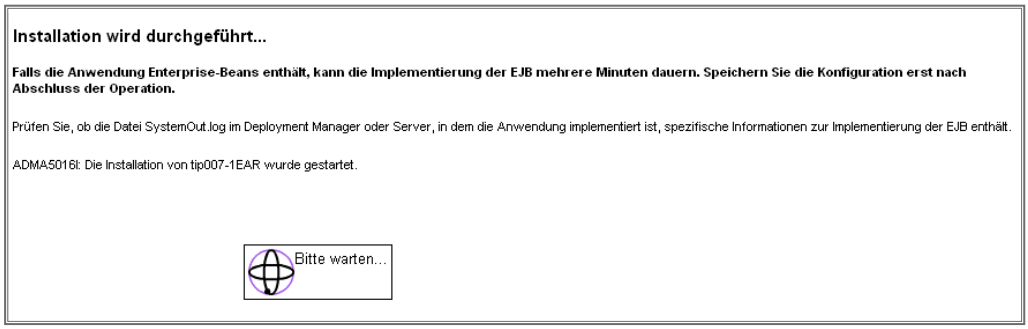
Zusammenfassung

Prüfen wir doch einmal unsere Zusammenfassung. Beachten Sie unbedingt den Hinweisbereich im unteren Abschnitt der Anzeige. Dort befindet sich in unserem Fall ein Fehlerhinweis. In der Auflistung der von uns getätigten Definitionen befindet sich außerdem ein Eintrag „Hier klicken“.

Nachdem mögliche Fehler oder Warnungen bearbeitet worden sind, können wir die Auswahlen für die Installation abschließen und unsere Anwendung mit einem Klick auf „Fertig stellen“ auf dem WAS installieren.

Im Laufe der Installation erhalten wir verschiedene Statusmeldungen, wie sie auch in der folgenden Abbildung zu sehen ist:

5.2.4.2.1
Seite 10



Fortschrittsmeldung

Es kann je nach Maschine und Auslastung einige Minuten dauern. Als Ergebnis erhalten wir dann eine Anzeige, ähnlich der folgenden Abbildung:

Abschlussmeldung

Damit haben wir eine neue Anwendung auf dem WAS mit Hilfe der Administrationskonsole angelegt.

Problembehandlung

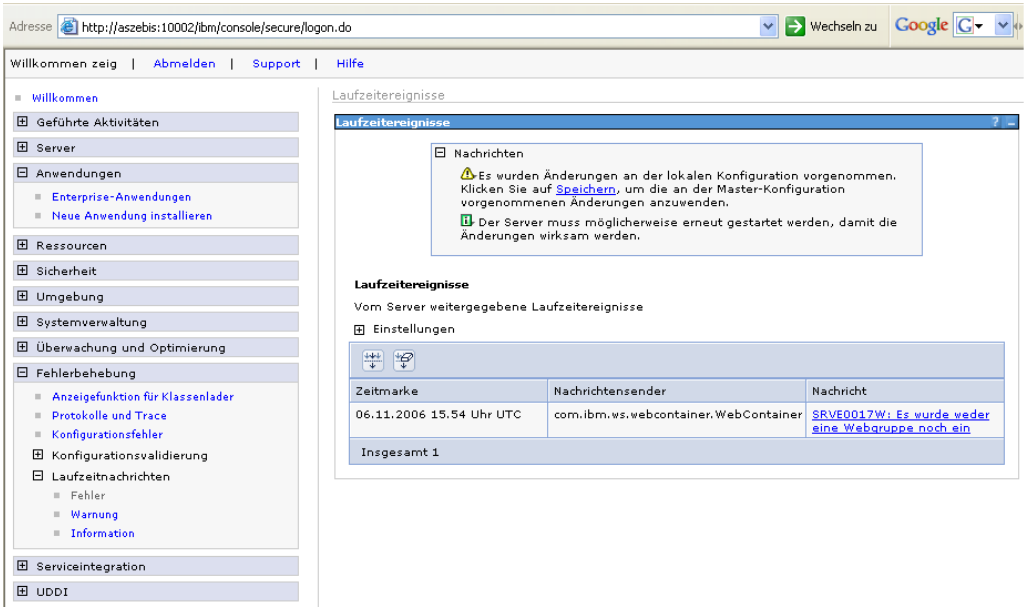
Der Betrieb einer derart komplexen Anwendungsumgebung, wie sie auch WebSphere darstellt, verläuft leider in der Praxis nicht immer ohne die eine oder andere Komplikation. Solche Komplikationen können kleinere Unregelmäßigkeiten sein, welche es zwar auf Dauer zu beseitigen gilt, deren Existenz jedoch nicht die Lauffähigkeit des WAS beeinflusst. Es können aber leider auch schwerwiegende Probleme sein, welche sich massiv auf die Verfügbarkeit oder die Stabilität des WAS auswirken.

Aus diesen Gründen ist es unbedingt erforderlich, dass man sich geeigneter Werkzeuge und Anwendungen bedient, mit deren Hilfe eine Überwachung der Installation auf der einen Seite und eine Fehleranalyse und Fehlerbeseitigung auf der anderen Seite möglich ist.

Bei der Ausführung von WebSphere auf dem System i5 können die Probleme und Unregelmäßigkeiten, welche im Zusammenhang mit WebSphere auftreten, zum Einen auf der Plattform i5 und deren Betriebssystem i5/OS liegen, zum Anderen können es aber auch Probleme sein, welche WebSphere-intern auftreten können.

Durch die Integration von WebSphere auf der i5 stehen uns verschiedene Einstiegspunkte zur Verfügung, mit deren Hilfe wir Problembehandlungen durchführen können.

Da wären einmal die i5-Bordmittel in Form z.B. der Standardprotokolle auf Betriebssystemseite zu nennen, außerdem können wir den Webadministrator für die Problemanalyse der WAS-Installation verwenden, oder die WAS-Administrationskonsole einsetzen, welche seit der Version 6 des WAS zur Verfügung steht.



Fehlerbehebung in der Administrationskonsole

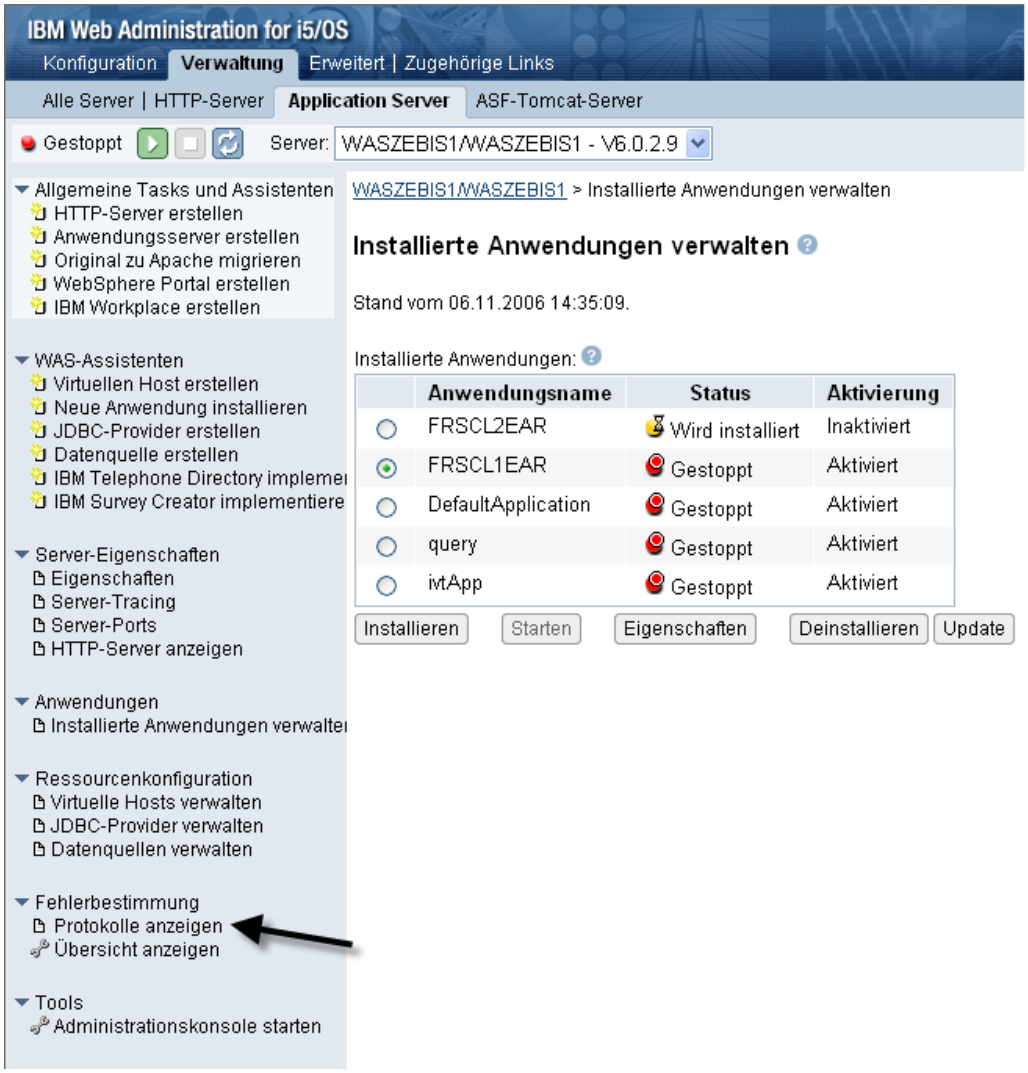
5.2.8
Seite 2

WAS-Protokoll

Für die Administration einer Umgebung wie WebSphere ist es unumgänglich, eine zuverlässige Quelle zu haben, in der wir neben Informationen über wichtige Ereignisse auch Fehlereinträge finden, die sich während der Laufzeit des WebSphere Anwendungsservers ereignet haben. Diese Quelle bildet im WebSphere-Umfeld verschiedene Protokolle, die standardmäßig auf dem Server geführt werden. Diese Protokolle liefern wichtige Informationen über die WebSphere-Installation und die Anwendungen, die auf dem WAS ausgeführt werden.

Die Grundkonfiguration der Protokolle wird während der Erstellung einer WebSphere-Instanz definiert.

Die Protokolle lassen sich über den Eintrag „Protokolle anzeigen“ im linken Navigationsbereich aufrufen.



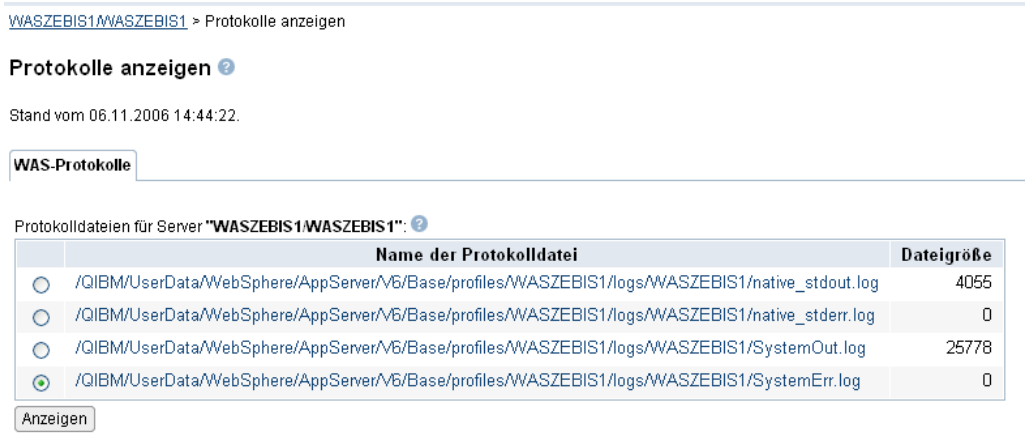
Auswahl-Protokolle anzeigen

5.2.8.1
Seite 2

Wie Sie unschwer erkennen können, befindet sich der Eintrag „Protokolle anzeigen“ in dem Bereich „Fehlerbestimmung“, in dem unter anderem auch die Auswahl „Übersicht anzeigen“ enthalten ist.

Protokolle lassen sich aber nicht nur in erster Linie zur Fehlerbestimmung einsetzen, sondern sind auch für die Analyse der WebSphere-Installation ein wichtiges Hilfsmittel.

Nach der Auswahl „Protokolle anzeigen“ erhalten wir zunächst eine Übersicht aller für diesen Server bzw. diese Instanz verfügbaren Protokolle, wie Sie in der folgenden Abbildung zu sehen sind:



Übersicht der Protokolle

In dieser Übersicht finden wir nicht nur den Namen des Protokolls, sondern auch den vollständigen Pfad, unter dem das Protokoll gespeichert ist. Darüber hinaus erkennen wir auch auf einen Blick die Dateigröße.

Tipp

Sollte der WebSphere Application Server einmal nicht mehr gestartet werden können, dann lassen sich die Protokolle auch direkt mit Hilfe eines Editors öffnen, indem in dem Verzeichnis (das allerdings bekannt sein muss!) das jeweilige Protokoll geöffnet wird.

Je nach Dauer der Aufzeichnung kann ein solches Serverprotokoll schon einen beachtlichen Umfang annehmen. Natürlich richtet sich die Anzahl der Einträge, die in einer gewissen Zeit in einem Protokoll auflaufen, nach der Anzahl der Aktionen und protokollierten Prozesse. Wir können aber auch in den Protokolleigenschaften definieren, dass bei dem Erreichen einer bestimmten Größe das Protokoll automatisch umbenannt und ein neues Protokoll erzeugt wird. Dieser Prozess läuft automatisch für die Standardprotokolle des WAS.

Wie Sie anhand der vorhergehenden Abbildung sehen können, verfügt unsere WebSphere-Instanz über derzeit vier verschiedene Protokolle, deren Bedeutung ich Ihnen nachfolgend zeigen möchte:

Protokoll SystemOut

Bei diesem Protokoll handelt es sich um die Aufzeichnung von JVM-relevanten Aktionen. Dieses Protokoll wird für jeden WebSphere Application Server Prozess erstellt. Dies kann der Applikationsserver selbst sein, ein Teilbereich innerhalb eines Clusters, ein Node Agent oder ein Deployment Manager.

Dieses Protokoll wird neben dem Protokoll SystemErr dazu verwendet, Fehler innerhalb der JVM aufzuzeichnen. Damit lassen sich nicht nur automatisch JVM-Fehler protokollieren, sondern innerhalb der Anwendungen können wir dieses Protokoll auch zur manuellen Ausgabe von Protokolleinträgen nutzen, indem wir mit einer der Methoden print(); println() oder printStackTrace() Einträge in das Protokoll SystemOut ausgeben.

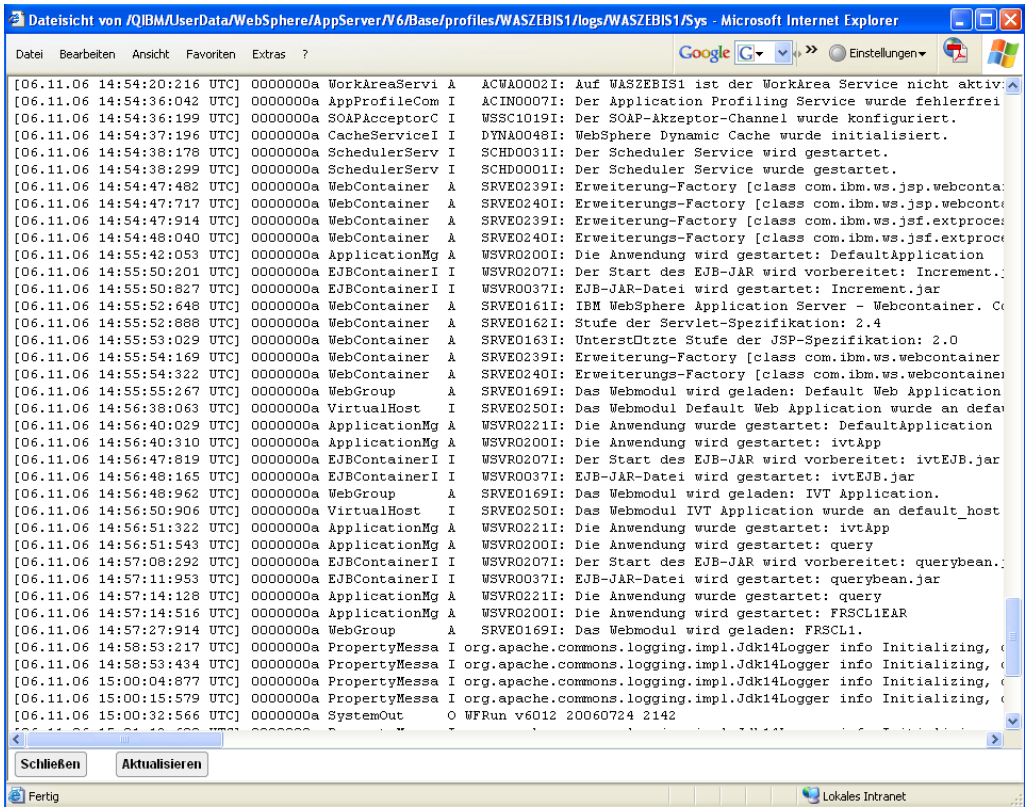
SystemOut dient im Wesentlichen der Überwachung der Aktivitäten und der Verfügbarkeit des WAS.

Tipp

Sollte es im Rahmen der Installation oder Konfiguration des WAS Probleme geben, dann werden diese in dem Protokoll SystemOur.log aufgezeichnet. Dieses können Sie dann mit einem Texteditor öffnen und mit Hilfe der Einträge eine Fehlerbehebung starten.

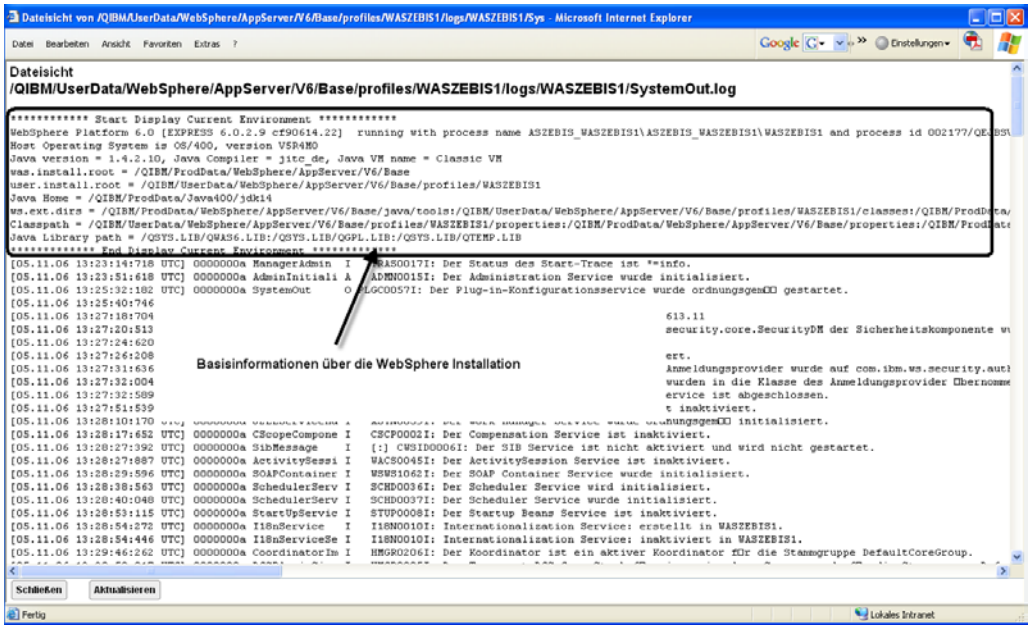
In dem nachfolgenden Beispiel habe ich einmal das Protokoll SystemOut geöffnet. Wie man leicht erkennen kann, werden in diesem Protokoll alle möglichen Aktionen aufgezeichnet. Neben der Aktivierung von Standardprozessen innerhalb der WAS-Installation sehen wir dort auch die gestarteten Anwendungen. Diese Informationen sind für Analysen von Bedeutung.

5.2.8.1
Seite 4



SystemOut-Protokoll

Während sich die aktuellen Einträge am Ende des Protokolls befinden, enthält das Protokoll am Anfang wichtige Informationen über die WAS-Installation. Damit lassen sich zum Beispiel auch wichtige Informationen auslesen, die bei der Anforderung von Softwaresupport durch IBM benötigt werden. Ein Beispiel für solche Informationen finden Sie in der folgenden Abbildung:



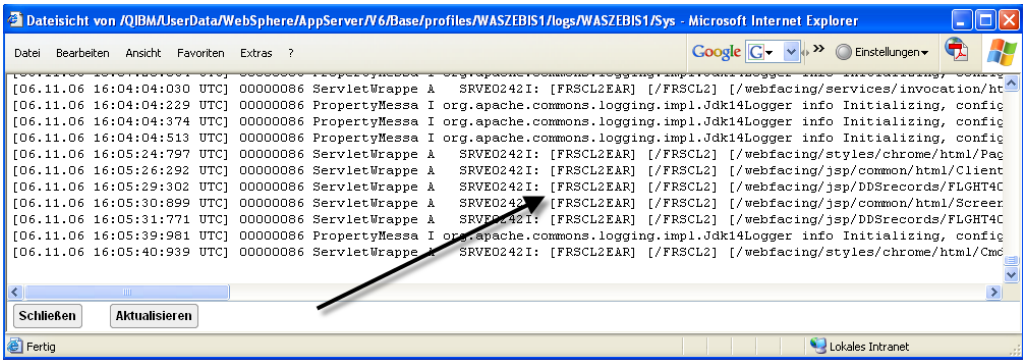
Basisinformationen zum WAS

Neben der installierten WebSphere Version (in unserem Beispiel verwenden wir die Version 6.0.2.9 des WebSphere Application Server Express) finden wir dort auch den Prozessnamen sowie die aktuellen Informationen der Plattform, unter der WAS installiert wurde.

In diesem Zusammenhang hier noch einige wichtige Informationen, welche wir dem Protokoll entnehmen können:

- Java Version und weitere Details wie Classpath Angabe
- Installationsverzeichnis der WAS-Installation (in dem Beispiel ist das / QIBM/ProdData/WebSphere/AppServer/V6/Base)
- Installationsverzeichnis der WebSphere Instanz (in dem Beispiel /QIBM/ UserData/WebSphere/AppServer/V6/Base/profiles/WASZEBIS1)

Neben diesen Basisinformationen werden beispielsweise auch alle aktiven Anwendungen bzw. bestimmte Aktionen und Statusinformationen dieser Anwendungen in diesem Protokoll aufgezeichnet, wie wir es an dem Beispiel einer Anwendung „FRSCL2EAR“ sehen können. In dem Protokoll finden wir dazu Informationen über den Start der Anwendung, die Initialisierung und auch mögliche Fehler, die im Zusammenhang mit der Anwendung aufgetreten sind.



Informationen über Anwendungen

5.2.8.1
Seite 6

Protokoll SystemErr

In diesem Standardprotokoll werden schwerwiegende Systemfehler aufgezeichnet. Die Einträge erfolgen in Form von Trace-Informationen zu den jeweiligen Stacks, in denen ein Fehler aufgetreten ist. Im Idealfall erhalten wir die folgende Anzeige des Protokolls:

Ein leeres Protokoll SystemErr

Protokoll native_stdout

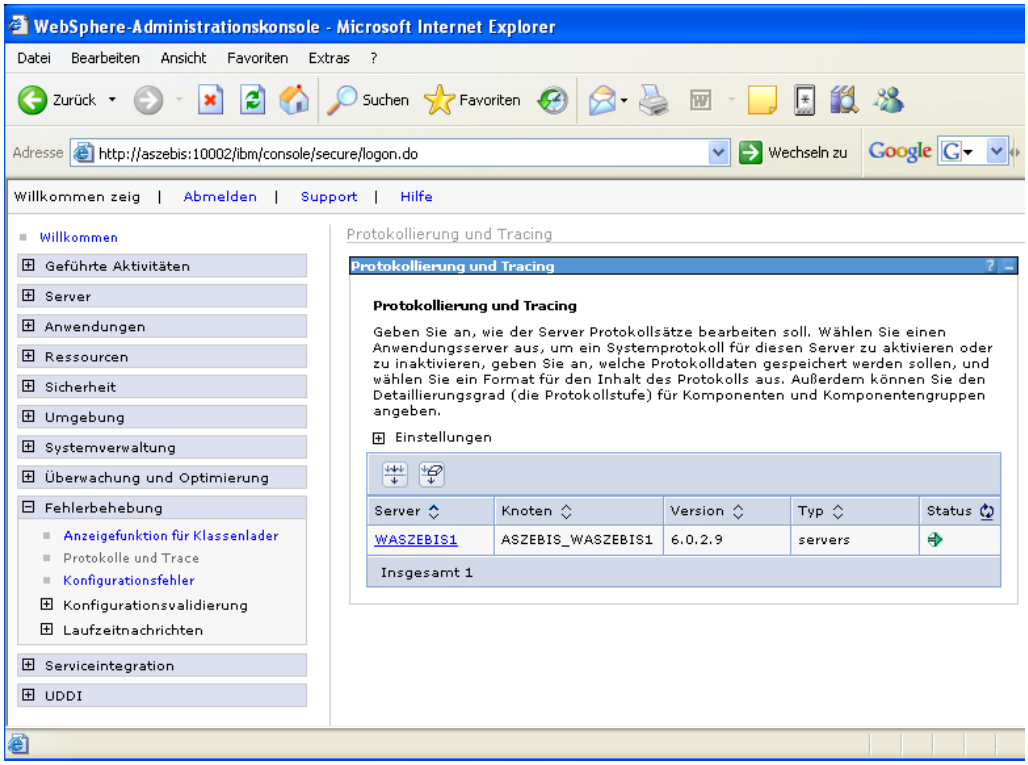
Im Gegensatz zu den beiden zuvor genannten Protokollen werden in dem native_stdout.log keine JVM abhängigen Fehler aufgezeichnet. Vielmehr dient dieses Protokoll der Aufnahme von Unregelmäßigkeiten, welche im Zusammenhang mit der nativen Installation und Ausführung des WAS im Zusammenhang stehen. In diesem Protokoll werden zum Beispiel .dll Informationen weg geschrieben.

Protokoll native_stderr

Neben dem Log native_stdout dient auch das Protokoll native_stderr der Aufzeichnung von nativen Prozessen oder Unregelmäßigkeiten, die beim Ausführen von WAS in einem bestimmten Betriebssystem (Windows, Unix, i5/OS) auftreten.

Wahlweise zu den zuvor gezeigten Zugriffen auf die verschiedenen Protokolle können wir diese auch über die Administationskonsole öffnen.

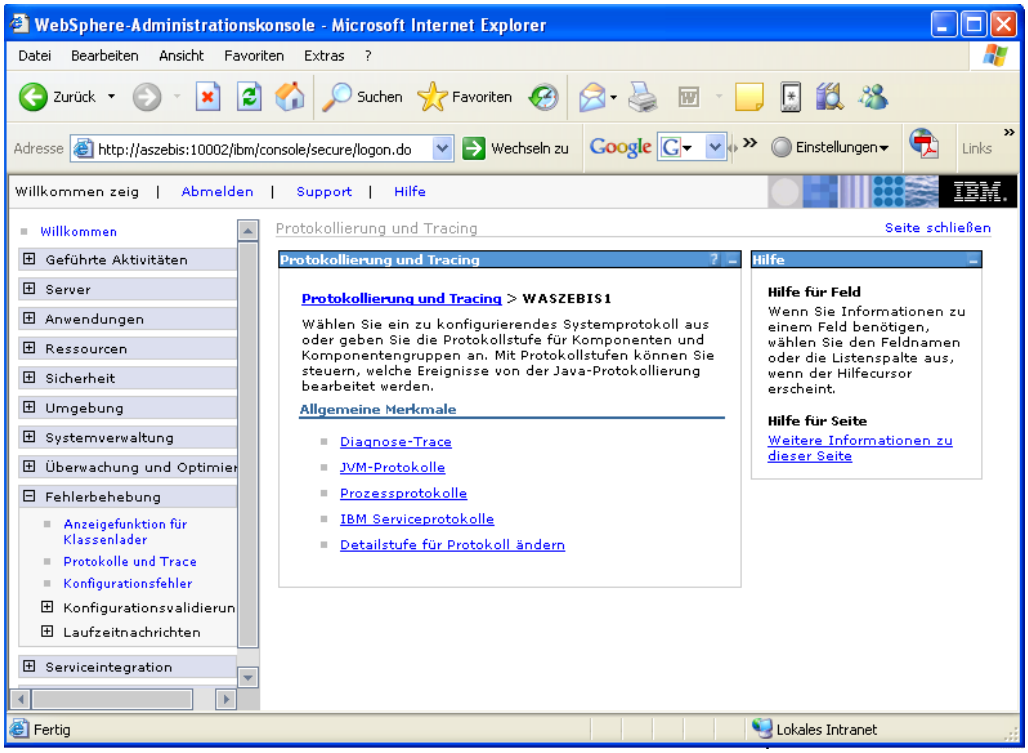
Dazu steht uns nach dem Starten der Administrationskonsole im linken Navigationsbereich der Abschnitt „Fehlerbehebung“ zur Verfügung.



Fehlerbehebung in der Administrationskonsole

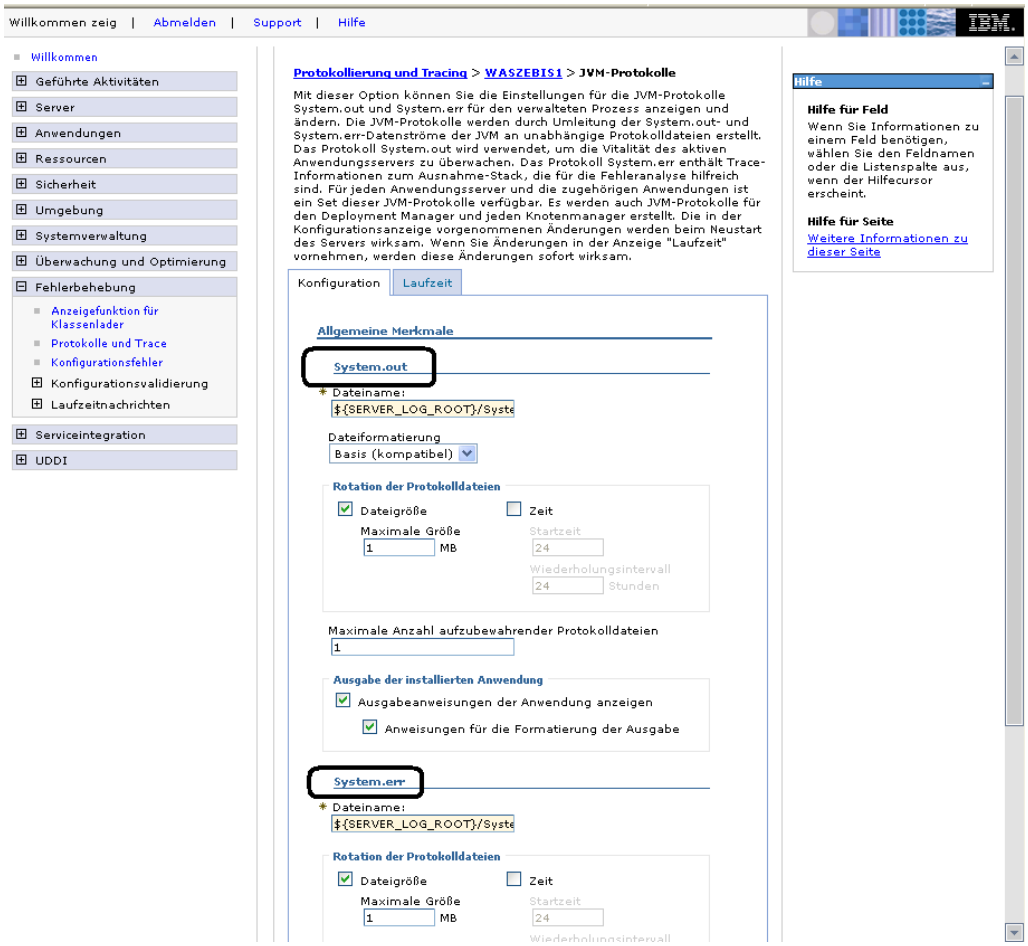
Dieser Bereich erlaubt uns neben der Ansicht der Protokollinhalte auch das Anpassen der Einstellungen für die Protokolle. Für das Arbeiten mit den Protokollen wählen wir nun in dem Hauptbereich der Anzeige den Server aus, dessen Protokolle wir einsehen bzw. verwalten wollen.

5.2.8.1
Seite 8



Protokolldetails WAS-Server

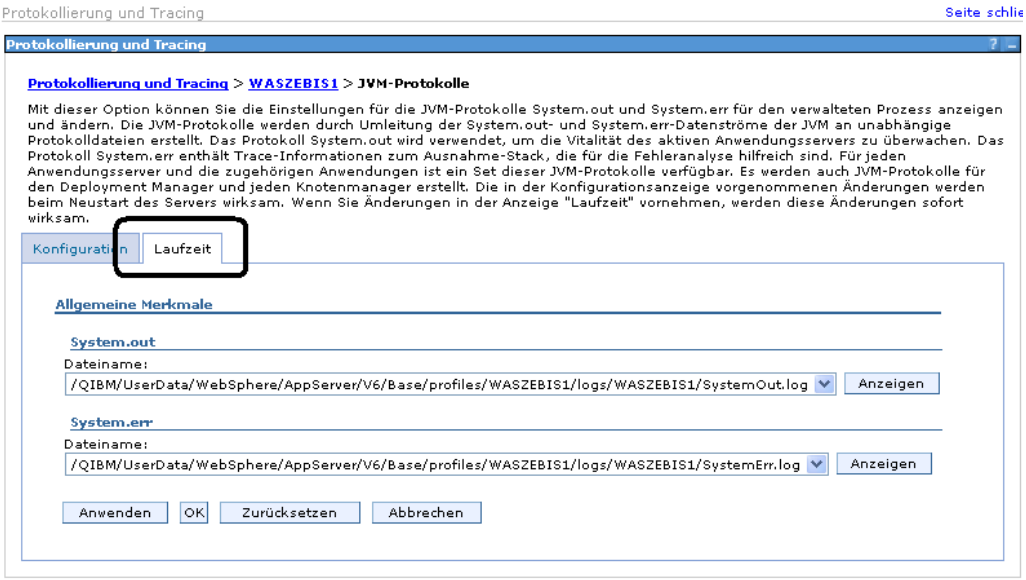
In dieser Übersicht finden wir die Anordnung der Protokolle leicht angepasst. In dem nachfolgenden Beispiel habe ich den Punkt „JVM-Protokolle“ selektiert. Damit gelangen wir in die Verwaltungsanzeige der Protokolle, welche JVM betreffende Informationen aufzeichnen. Zuvor habe ich Ihnen diese beiden Protokolle „System.Out“ und „System.Err“ bereits kurz vorgestellt.



JVM-Protokolle

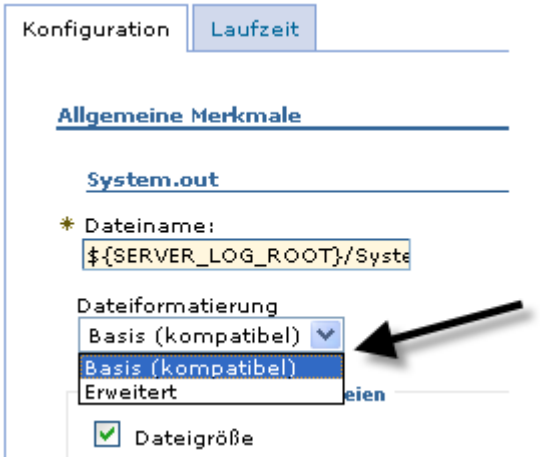
Dieser Anzeigenbereich ist in zwei Abschnitte unterteilt. In dem Tabellenbereich „Laufzeit“ finden wir die genauen Pfadangaben zu den beiden JVM-Protokollen:

5.2.8.1
Seite 10



Laufzeitangaben

Doch kehren wir nochmals zu dem Tabellenbereich „Konfiguration“ zurück. Dort lassen sich einige Grundeinstellungen in Bezug auf die Protokollverwendung definieren. Unter anderem können wir zwischen verschiedenen Formatierungen wählen:



Formatierung

- Basis
  - Bei dieser Einstellung handelt es sich um den Standardwert. Dabei werden nur Basisinformationen protokolliert. Der Aufbau der Nachrichten ist dabei wie folgt:

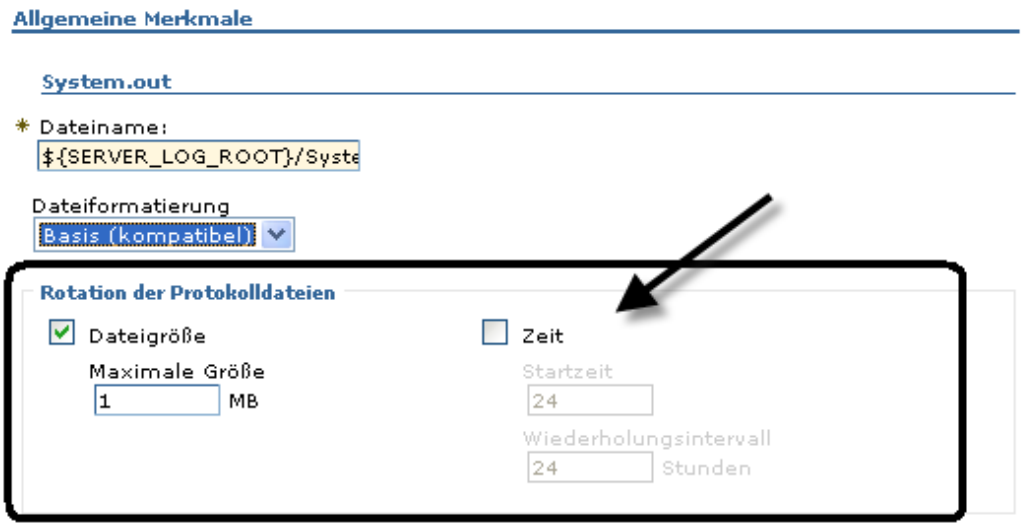
    <timestamp><threadID><shortName><eventType>[class] [method]<message>

- Erweitert
  - Bei dieser Selektion erhalten wir neben den Basisinformationen auch Informationen zu einem Ereignis. Der Aufbau sieht dabei so aus:

    <timestamp><threadID><eventType><UOW><source=longName>[class][method]<Organization><Product><Component><message>

Wie zuvor bereits erwähnt, kann die Größe eines Protokolls nach verschiedenen Kriterien sehr schnell anwachsen. Damit ein solches Protokoll nicht zu unübersichtlich wird, lassen sich automatische Mechanismen einsetzen, mit deren Hilfe wir automatisch ein bestehendes Protokoll abhängen und durch ein neues ersetzen können. Dabei werden die abgehängten Informationen weggesichert und WAS erstellt automatisch ein neues Protokoll.

Als Schwellenwerte für eine solche Selbstorganisation kann wahlweise eine Größe definiert werden, bei deren Überschreiten automatisch ein neues Protokoll erzeugt wird, oder wir können diesen Automatismus mittels einer Zeit steuern. Beide Möglichkeiten lassen sich in dem Bereich „Rotation der Protokolldateien“ verwalten, wie es die nachfolgende Abbildung zeigt.



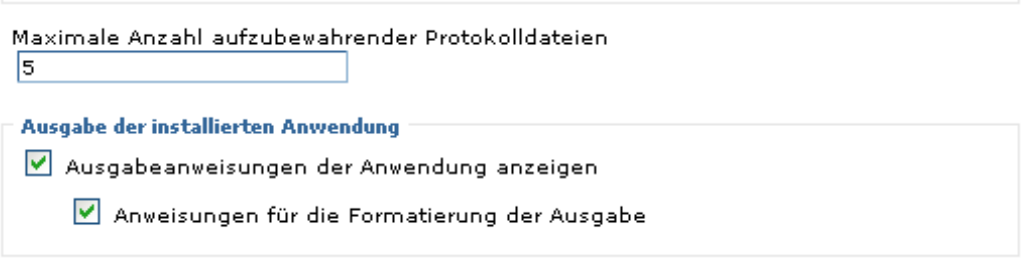
Rotation der Protokolldateien

Es lässt sich zeitgleich nur einer der beiden Faktoren „Dateigröße“ oder „Zeit“ verwenden.

Während wir in dem Feld „Dateigröße“ einen MB als Wert eintragen, bei dessen Überschreiten ein neues Protokoll angelegt wird, definieren wir mit der Startzeit den Zeitpunkt, zu dem der Zyklus für die Protokollverwaltung gestartet werden soll. In Verbindung mit der Angabe „Wiederholungsintervall“ lässt sich damit eine zeitbasierte Verwaltung der Protokolle erreichen.

5.2.8.1
Seite 12

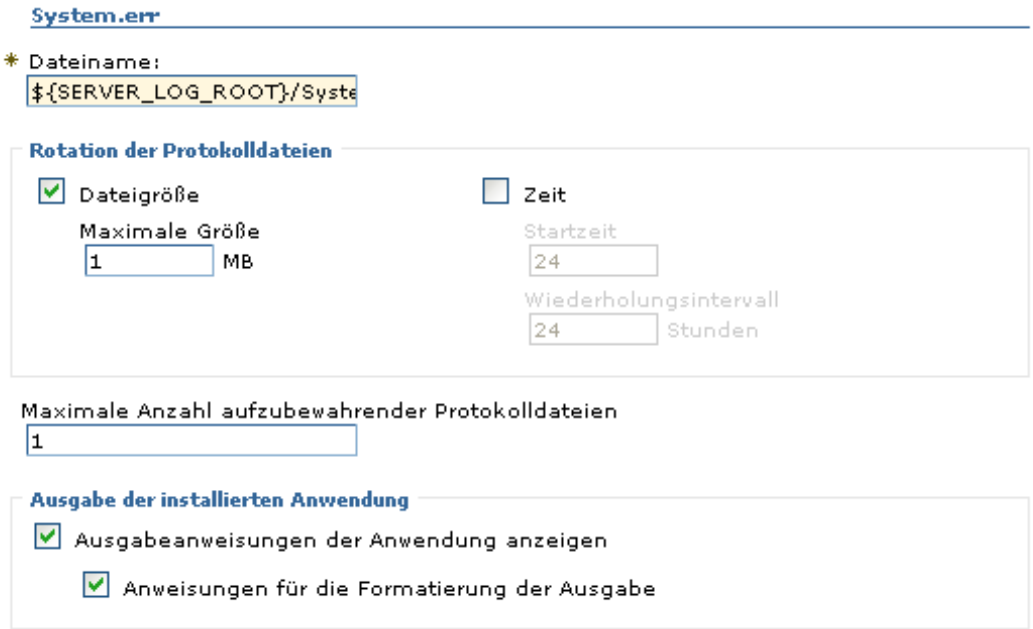
Wenn der Zyklus eines Protokollwechsels zu klein gewählt wird, dann laufen wir Gefahr, dass wichtige Protokollinformationen verloren gehen. Aber auch dafür ist eine Vorkehrung getroffen. Mit der Angabe der maximalen Anzahl der aufzubewahrenden Protokolldateien bestimmen wir, wie viele Versionen des Protokolls gespeichert bleiben sollen.



Anzahl der aufzubewahrenden Protokolldateien

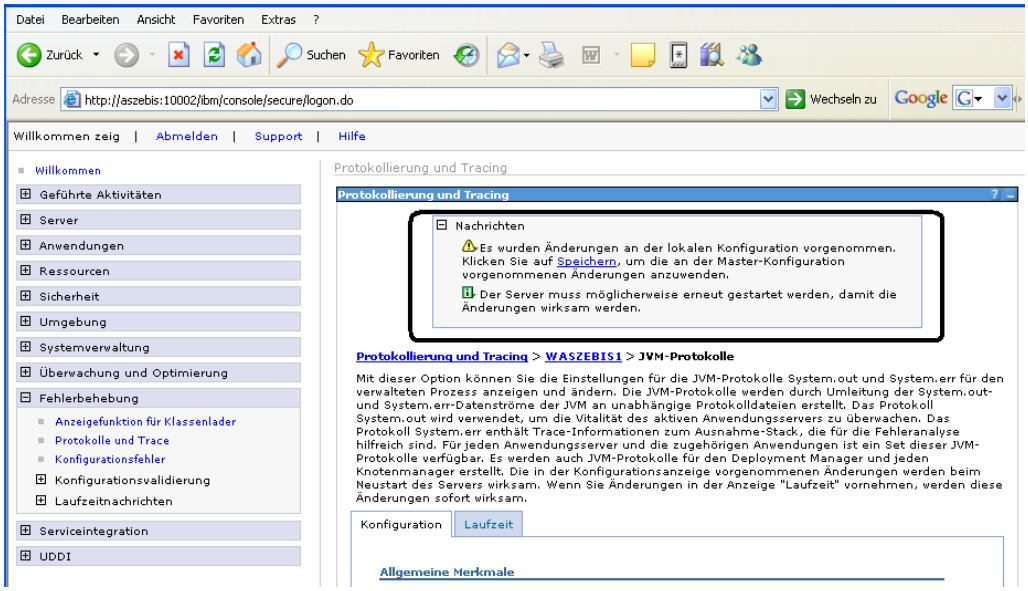
Abschließend können wir mit dem Bereich „Ausgabe der installierten Anwendungen“ beeinflussen, ob Informationen, welche über Methoden wie z.B. print() oder println() in das Protokoll geschrieben wurden, automatisch mit angezeigt werden sollen.

Diese Einstellungen beziehen sich zwar in der Erläuterung auf das Protokoll SystemOut, im unteren Bereich der Anzeige finden wir die gleichen Informationen nochmals für das Protokoll SystemErr.



Einstellungen für SystemErr

Sollten Sie Änderungen an den Einstellungen durchgeführt haben, dann achten Sie bitte auch auf den oberen Bereich des Verwaltungsabschnitts – denn dort könnte ein Hinweis erscheinen, welcher mitteilt, dass Änderungen an den Einstellungen durchgeführt wurden, welche einen Neustart des Servers bedingen.



Neustart ist erforderlich

5.2.8.1
Seite 14

Begriffsdefinitionen Application-Server

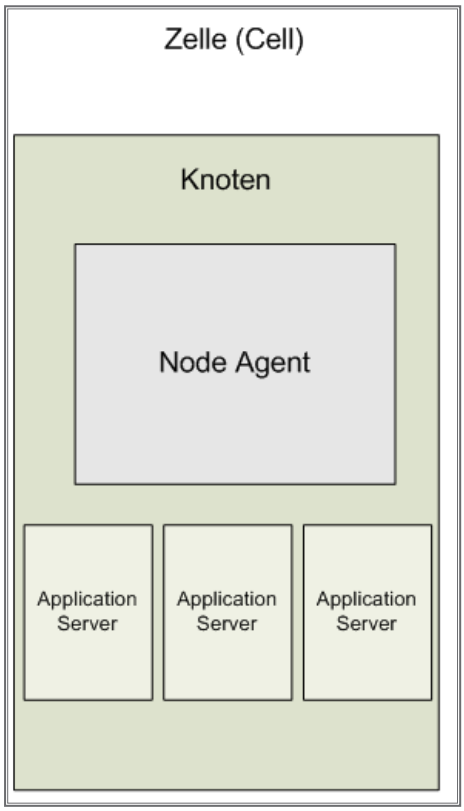
Die Technologie des IBM WebSphere Application Server basiert auf verschiedenen Komponenten, deren Definition ich Ihnen an dieser Stelle erläutern möchte.

Bei der Vorbereitung und Planung einer WAS-Installation sind einige Punkte zu beachten, die sich zum Teil später im produktiven Einsatz des Systems wesentlich auswirken können.

So gilt es zunächst festzulegen, ob die Anforderungen mit einem einzelnen WAS realisiert werden können. Oder ob für komplexe Ausführungsschritte bzw. aus Verfügbarkeits Aspekten mehrere WAS parallel installiert werden sollen, welche die Aufgaben gemeinsam übernehmen. Aus Performance-Gründen und in Bezug auf Sicherheits- und Verfügbarkeits-Überlegungen ist es wichtig, wie der WAS-Server bzw. dessen Instanzen aufgebaut sind. Diese einzelnen hierarchischen Teilbereiche – zum Beispiel Zellen und Knoten – spielen eine wesentliche Rolle.

Da es immer wieder Verwirrungen bei der Vielzahl der „neuen" Begriffe gibt, möchte ich an dieser Stelle einige der wesentlichen Bestandteile bzw. Begriffe erläutern, die beim Einsatz eines Application-Server von Bedeutung sind. In diesem Zusammenhang ist zunächst der Aufbau eines WAS im Hinblick auf dessen Komponenten von Bedeutung.

Schauen wir uns dazu die folgende Abbildung an:



Struktur eines WAS Express

5.2.9
Seite 2

In dieser Abbildung sehen Sie die einzelnen Teilbereiche eines WAS in der Express-Version oder auch eines WAS in der Base-Edition.

Diese Abbildung beinhaltet einige Begriffe, die ich Ihnen nun erläutern möchte, denn sie spielen bei der Konzeption und der Administration einer WAS-Umgebung eine wesentliche Rolle.

Knoten

Unter einem Knoten (oder auch als „Node“ bezeichnet) verstehen wir im Zusammenhang mit einem Application-Server eine Gruppierung von Diensten und Prozessen. Ein Knoten stellt zudem eine organisatorische Einheit dar und basiert auf einheitlichen Konfigurations-Einstellungen. Jeder einzelne Server, der mit einer Single-Installation erstellt wurde, verfügt über einen solchen Knoten.

Bei der Installation eines neuen WAS-Server oder auch nur einer Instanz werden die Konfigurations-Einstellungen auf Basis eines solchen Knotens abgelegt. Mit der Einführung der Profile in der Version 6 kann diese Definition der Knoten auch um die Ebene der Profile erweitert werden. So bezeichnen wir jedes Profil ebenfalls als „Knoten“.

Knotengruppe

Die Knotengruppe stellt eine weitere organisatorische Einheit im Zusammenhang mit WAS dar. Dabei werden gleichartige Knoten zu einer Gruppe zusammengefasst. Auf diese Weise lassen sich zum Beispiel WAS-Server mit gleichartigen Funktionen (zum Beispiel für die Verarbeitung der ERP-Anforderung) oder WAS-Installationen auf gleichartigen Betriebssystemen zusammenfassen. Letzteres macht deshalb Sinn, weil in einigen Fällen – betrachten wir hier insbesondere die Installation des WAS auf i5 – doch kleine Abweichungen zu anderen Plattformen auffallen. Außerdem erleichtert es die Administrationstätigkeit immens, wenn die Spezifika der einzelnen Plattformen gleichartig zu behandeln und zu administrieren sind.

Zelle (Cell)

Das Gruppieren der WAS-Installationen und WAS-Komponenten geht noch weiter. Für administrative Aufgaben stehen uns die Zellen zur Verfügung. Diese stellen Gruppierungen von verschiedenen Knoten dar. Bei der Verwendung eines WAS-Express bzw. auch der Base-Edition eines WAS-Server besteht eine Zelle immer genau aus einem Knoten. Im Gegensatz dazu kann beim Einsatz einer Network Deployment Edition oder auch eines WAS-Cluster eine Zelle aus einer Vielzahl von Knoten bestehen. Bei der Zusammenfassung von Knoten zu Zellen können neben den Konfigurations-Einstellungen auch die Anwendungen in ein und demselben Repository gespeichert werden, von wo aus diese zentral administrierbar und nutzbar sind.

Bei der Definition von solchen Zellen kann der Administrator entscheiden, nach welchen Kriterien die Zellen zusammengesetzt werden sollen – also auch, aus welchen einzelnen Knoten eine Zelle besteht. Wie alle anderen Definitionseinstellungen, so werden auch die Zellen-Definitionen in einem XML-Dokument gespeichert. Wie man anhand der freien Definition durch einen Administrator schon schließen kann, bilden Zellen im Wesentlichen administrative Einheiten, deren Inhalt frei von dem Administrator nach dessen Anforderungen definiert werden können.

Node-Agent

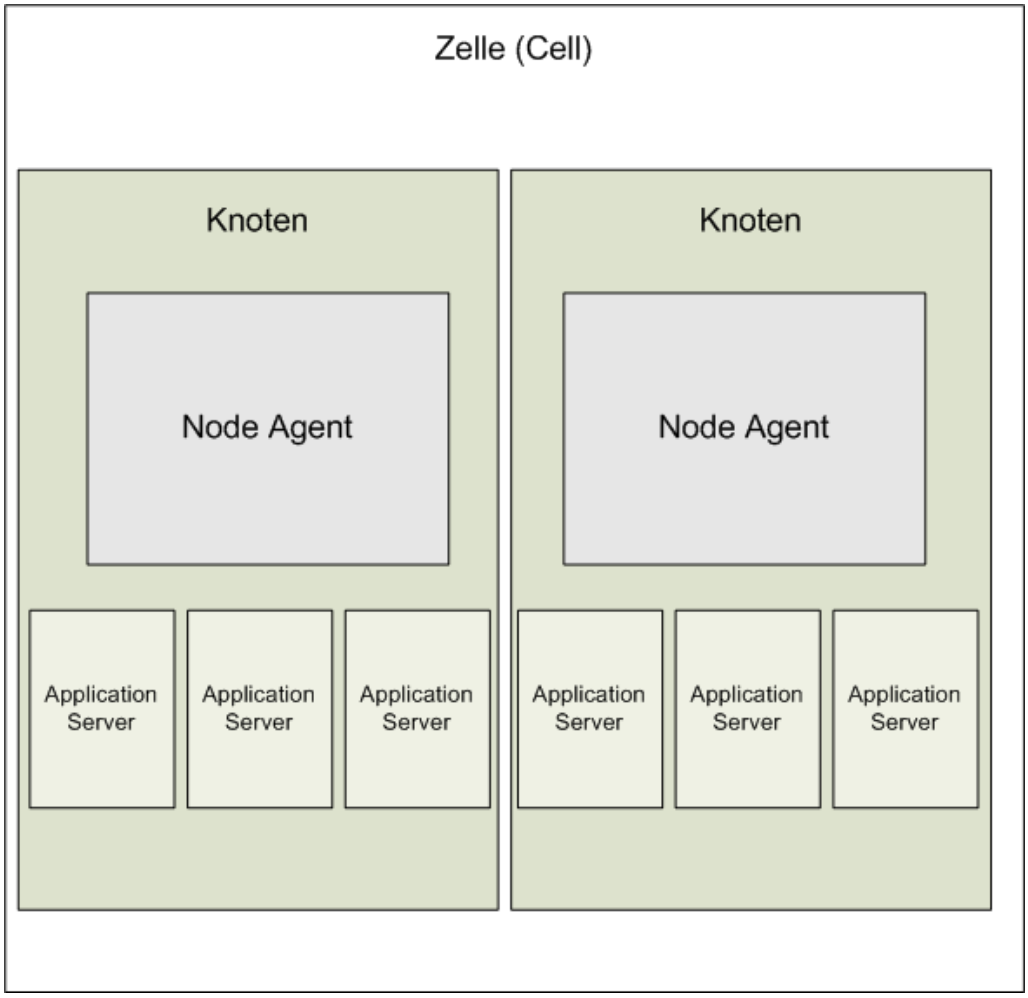
Node-Agenten stellen ebenfalls administrative Einheiten dar. Sie spielen bei dem Einsatz von WAS-ND eine wesentliche Rolle. Node-Agenten werden dazu verwendet, um eine zentrale Verwaltung der Konfigurations-Einstellungen und eine gemeinsame Prozesssteuerung verschiedener WAS-Server zu ermöglichen. Node-Agenten können zum Beispiel zur Überwachung der verschiedenen WAS-Server eingesetzt werden. Neben der Verfügbarkeits-Überwachung sind Synchronisations-Aufgaben eine der Schwerpunkte im Einsatz von Node-Agenten.

Der Einsatz solcher Node-Agenten ist nicht zwingend erforderlich. Allerdings wird bei der Definition eines Knotens, der zu einer Zelle hinzugefügt wird, automatisch ein solcher Node-Agent generiert. Grundsätzlich unterscheiden wir zwei verschiedene Knotentypen:

- Managed Nodes
  Dabei handelt es sich um Knoten, deren Verwaltung über Node-Agenten durchführt wird.
- Unmanaged Nodes
  Knoten, denen kein Node-Agent zugeordnet wurde.

Wie zuvor erwähnt, kann die Komplexität dieser zuvor genannten Teilbereiche je nach Installation der WebSphere-Umgebung mehr oder weniger ausgeprägt sein. Bei dem Einsatz eines WebSphere in der Base-Edition bzw. in der Express-Version finden wir die Zuordnung von nur einem Knoten zu einer Zelle. Anders verhält sich dies beispielsweise bei dem Einsatz der Network Deployment Edition. In einem solchen Fall können einer Zelle mehrere Knoten zugeordnet sein, wie die folgende Abbildung zeigt.

5.2.9
Seite 4



WAS-Struktur komplex

Repository

Der Einsatz von mehreren WAS-Servern wirkt sich natürlich auch auf deren Administration aus. Je mehr Application-Server installiert werden, desto komplexer kann deren Administration auch werden. Damit diese nicht zu kompliziert und unübersichtlich wird, stellt IBM einige Optionen zur Verfügung, welche die gemeinsame Administration mehrerer WAS-Server ermöglichen. Bei dem Einsatz von Knoten und Zellen ist die zentrale Verwaltung der Konfigurations-Einstellungen und auch der auf den Servern auszuführenden Anwendungen über ein so genanntes Repository möglich. Dabei handelt es sich um verschiedene zentrale XML-Dateien, in denen die Konfigurations-Einstellungen von mehreren Servern gespeichert sind. Über Werkzeuge in den Administrations-Oberflächen des WAS können diese Konfigurations- und auch Anwendungs-Einstellungen zentral verwaltet werden.

Deployment Manager

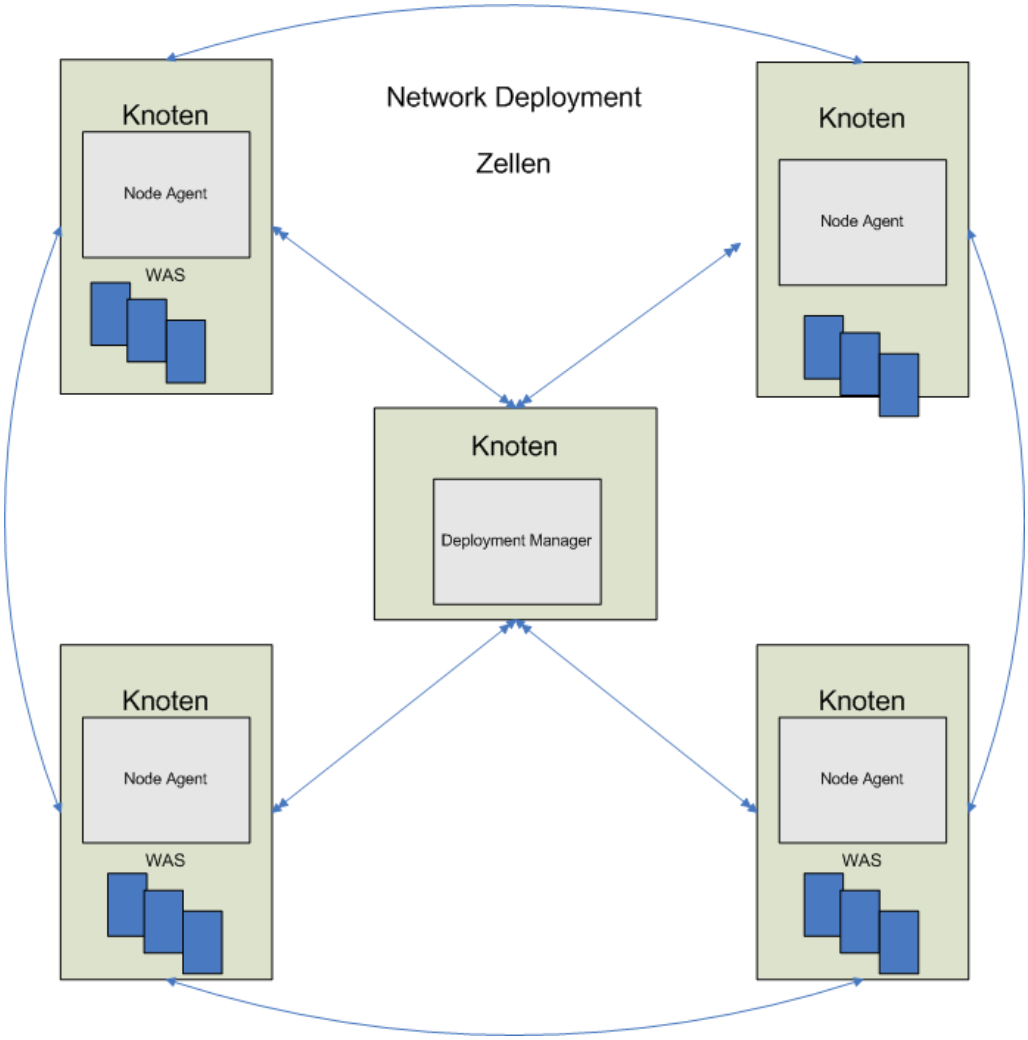
Für die Verwaltung mehrerer Server, welche auf dem Einsatz eines Repository basiert, kann der Deployment Manager eingesetzt werden. Dabei handelt es sich um eine Instanz eines WAS-Server, die ausschließlich für administrative Aufgaben eingesetzt wird.

Network Deployment

Einfache Installationen basieren meist auf dem Einsatz von einem einzelnen WAS oder wenigen WAS-Servern. Je komplexer die Installationen jedoch werden, umso mehr nimmt auch die Anforderung an die Leistungsfähigkeit des oder der Server und deren Ausfallsicherheit zu. Wie auch in anderen Bereichen der IT, so stehen uns auch bei dem Einsatz von WAS-Servern so genannte Cluster zur Verfügung. Die Basis für ein solches WAS-Cluster stellt eine Erweiterung der Standard-Edition des WAS dar – das so genannte „Network Deployment“ – eine Ausbaustufe des WAS-Server, mit der neben dem Einzelbetrieb auch die Option einer verteilten Installation realisiert werden kann. Bei einer solchen Installation agieren mehrere, voneinander getrennt installierte WAS-Server als Einheit.

Bei dem Einsatz des Network Deployment verfügt jeder einzelne Knoten über einen Node-Agenten, über den die gemeinsame Verwaltung ermöglicht wird.

Die folgende Abbildung zeigt schematisch den Aufbau des Network Deployment.



Zellen-Network-Deployment

Der Vorteil liegt im Wesentlichen im Einsatz von mehreren JVMs, die unabhängig voneinander agieren können. Auf diese Weise lassen sich die Auslastungen der Server elegant steuern und auch die Verfügbarkeit der Anwendungen ist durch die quasi Redundanz der Systeme sichergestellt.

Bei dem Einsatz der Network Deployment-Variante werden verschiedene WAS-Server-Installationen zentral verwaltet. Dabei muss ein zentraler Administrations-Server eingerichtet werden. Auf diesem Server werden die Administrations-Werkzeuge installiert und auch ausgeführt. Die Überwachung der zu diesem Network gehörenden Server, also der Zelle, übernimmt dabei der Deployment Program Manager (dmgr). Damit die Überwachung der einzelnen Knoten zentral durchgeführt werden kann, werden auf Knotenebene die Node-Agenten benötigt, welche direkt mit dem „dmgr“ kommunizieren.

Cluster

Je komplexer die Anwendungen, welche auf einem WAS-Server ausgeführt werden sollen, umso mehr kann das Verteilen der anfallenden Last auf mehrere Server Sinn machen. Mit dem Einsatz der Network Deployment Edition des WAS besteht auch die Option, mehrere Server zu einer interagierenden Einheit zusammenzufassen. Dadurch können über komplexe interne Strukturen die Lasten, welche von einer oder von mehreren Anwendungen auf einer WAS-Installation entstehen, auf verschiedene WAS-Server verteilt werden. Die logische Zusammenfassung mehrerer, miteinander zusammen arbeitender Server wird also als „Cluster" bezeichnet.

Die Bildung eines Cluster wird nach folgendem Kriterium differenziert:

- Vertikales Cluster
  In diesem Fall werden die einzelnen Server-Instanzen über einen einzelnen Knoten abgebildet.
- Horizontales Cluster
  Sobald die Server über mehrere Knoten abgebildet werden, spricht man von einem horizontalen Cluster.

Profil

Die Profile stehen ab der Version 6 des WAS zur Verfügung. Waren zum Beispiel in der Version 5 des WAS der Knoten bzw. die Zelle als administrative Einheiten von Bedeutung, so wurden ab der Version 6 die Profile als Administrations-Hülle eingesetzt. Bei einem Profil handelt es sich um spezifische Einstellungen bezogen auf die Konfigurationen des WAS und der Anwendungen, welche auf dem Server installiert sind. Für jeden WAS-Server ab der Version 6 muss mindestens ein solches Profil – also eine Laufzeitumgebung zur Administration – erstellt werden.

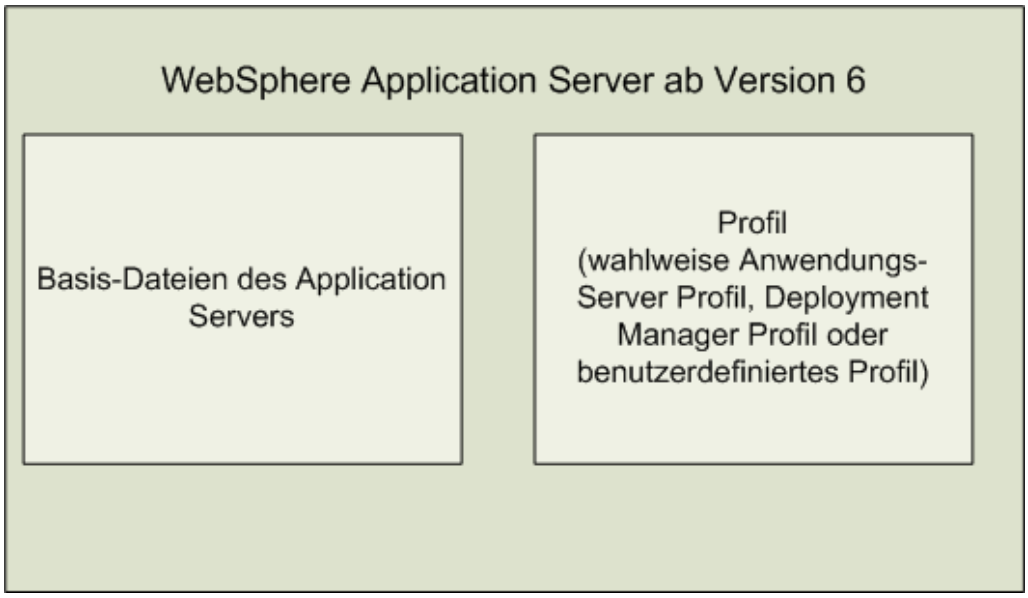
Je nachdem, welche Edition des WAS eingesetzt wird, kommen unterschiedliche Profile zum Einsatz:

- Anwendungsserver-Profil
  Bei der Verwendung eines Anwendungsserver-Profils wird ein Server konfiguriert, welcher als Standard-Server betrachtet wird. Ein solcher Anwendungsserver kann auch in eine komplexe Server-Umgebung eingebunden werden.
- Deployment Manager-Profil
  Bei dieser Form des Profils werden mehrere Server innerhalb des Profils abgebildet. Dieses Profil stellt quasi die Schnittstelle für die zentrale Verwaltung mehrerer WAS-Server dar. Dabei können sowohl verschiedene Server, welche auf einer physischen Maschine installiert sind, in einem Deployment Manager-Profil zusammengefasst werden, als auch Server, die auf verschiedenen physischen Maschinen installiert sind.

- Benutzerdefiniertes Profil
  Unter einem benutzerdefinierten Profil verstehen wir einen zunächst leeren Knoten, der erst später mit Leben gefüllt wird. Ein solches Profil bzw. ein solcher Knoten kann individuell auf die jeweiligen Bedürfnisse angepasst werden. Mit Hilfe eines solchen Profils kann unter anderem auch eine Cluster-Definition realisiert werden.

Die Bedeutung der Profile wird bereits bei der Installation eines WAS ab der Version 6 deutlich. Die Installationsschritte werden seit der Version 6 in die zwei folgenden Teilbereiche gesplittet:

- Installation der Basis des WAS-Server
- Erstellung eines Profils
  Hier kann wahlweise ein Anwendungsserver-Profil, Deployment-Server-Profil oder benutzerdefiniertes Profil erstellt werden

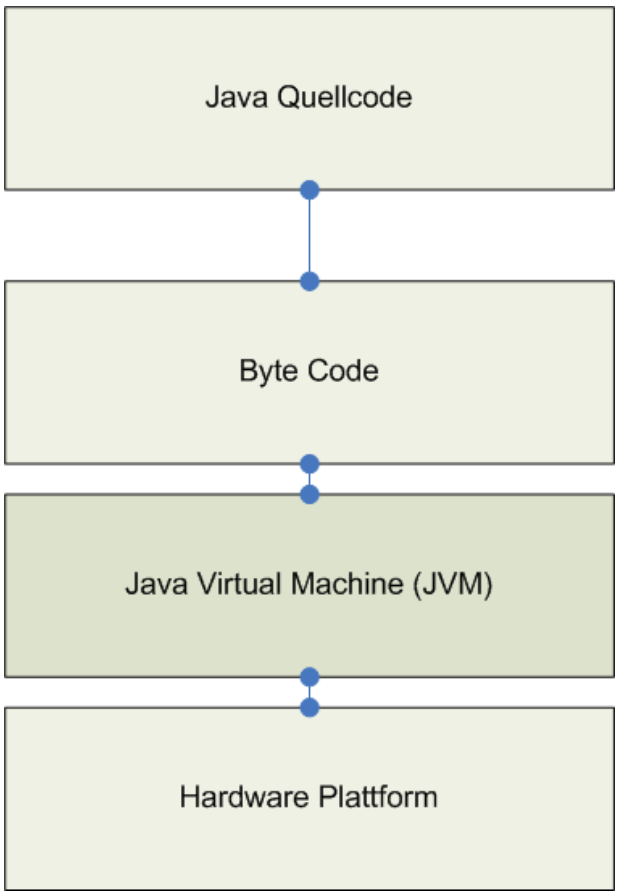


Aufbau des WAS ab der Version 6

JVM

JVM, oder ausgeschrieben Java Virtual Machine, ist eine der Schlüsselfunktionen im Java-Umfeld. Mit Hilfe der JVM erhält Java erst die „Plattform-Unabhängigkeit". Grundsätzlich benötigt jedes Java-Programm eine Laufzeitumgebung, unter welcher das Programm oder die Anwendung ausgeführt werden kann. Diese stellt die JVM dar. Häufig finden wir in der Literatur für solch eine Laufzeitumgebung eine weitere Bezeichnung: „Interpreter".

Die nachfolgende Skizze zeigt die einzelnen Komponenten in einer JVM-Hierarchie:



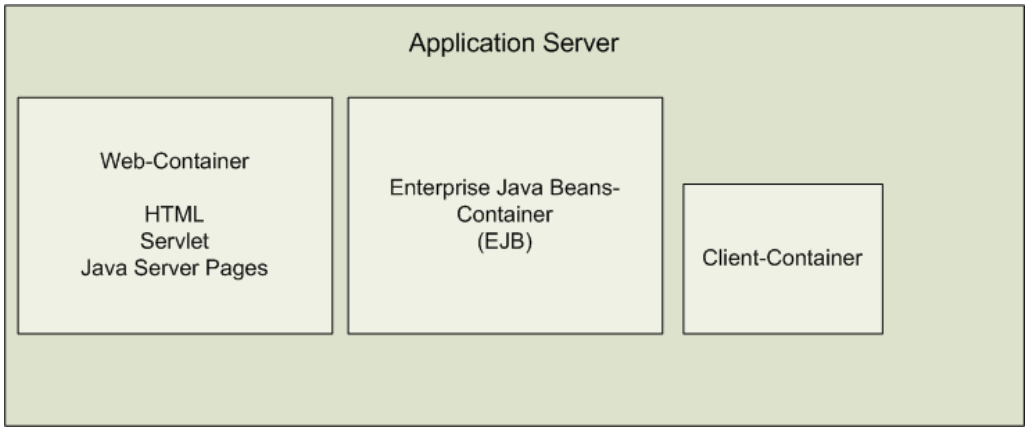
JVM-Komponenten

Container

Der Begriff „Container“ ist nicht nur in der WebSphere-Umgebung von Bedeutung, sondern bezieht sich als allgemein gültige Definition auch auf Begriffe aus den J2EE und die .NET-Welten. Als Container wird in diesem Zusammenhang die Basis-Laufzeitumgebung für J2EE-Anwendungen bezeichnet.

5.2.9
Seite 10

Grundsätzlich werden bei der Ausführung mit einem Application-Server drei verschiedene Container-Bereiche unterschieden, wie sie in der folgenden Abbildung zu sehen sind:



Container

Die beiden Container-Ebenen

- Web-Container und
- EJB-Container

stellen die Hauptkomponenten eines Application-Server dar.

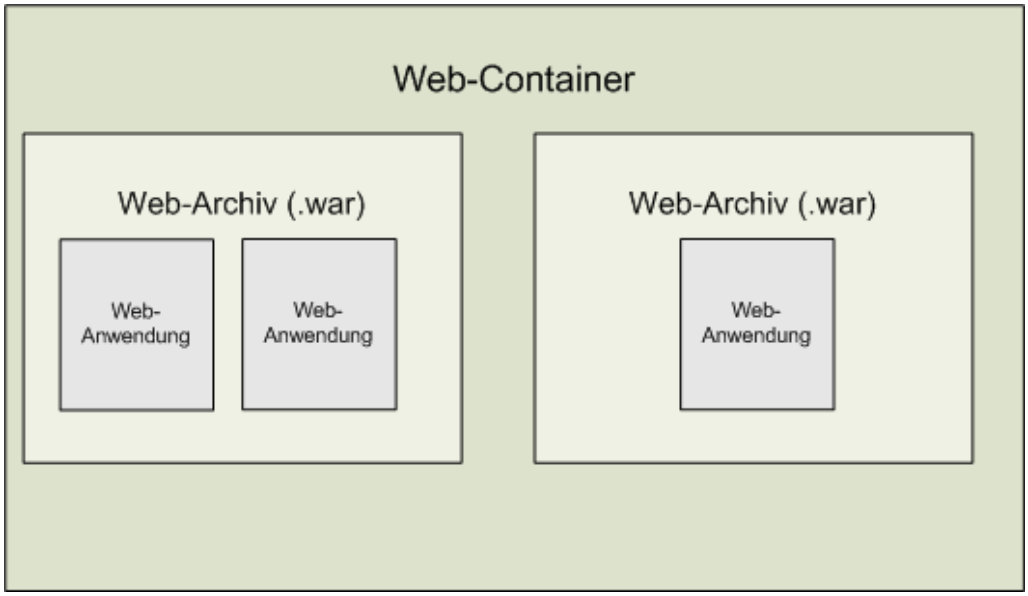
Web-Container

Der Web-Container übernimmt die Kommunikation und die Verarbeitung der Anforderungen im HTTP-Umfeld. Damit werden die Transaktionen, welche von einem Client aus gestartet werden, verarbeitet. Auf der anderen Seite werden die Ergebnisse, welche an den Browser übertragen werden müssen, über diesen Web-Container verteilt. Damit bildet der Web-Container die Laufzeitumgebung für alle Web-basierten Transaktionen.

Web-Container verarbeiten neben den klassischen HTTP-Anforderungen unter anderem auch Servlets und Java Server Pages (JSPs).

Da in einem WAS-Umfeld die Web-Aktionen in der Regel über einen externen Web-Server ausgeführt werden, kann über den Web-Container auch die Definition bezogen auf die Anbindung eines externen Web-Server durchgeführt werden.

Wesentlicher Bestandteil eines Web-Containers sind die Web-Archive. Dabei handelt es sich um einzelne oder auch mehrere, voneinander abhängige Web-Anwendungen. Diese Web-Anwendungen werden in Web-Archiven verwaltet. Diese Web-Archive erkennen wir an den Endungen (.war).



Web-Container

EJB-Container

Die zweite Säule der Container bildet neben dem Web-Container der EJB-Container. Wie der Name es bereits vermuten lässt, bildet dieser Teilbereich die Ausführungsebene für die Enterprise Java Beans (kurz: EJBs). Jede EJB benötigt für deren Ausführung eine Laufzeitumgebung. Diese ist im Falle des WAS nicht direkt auf dem Server selbst implementiert, sondern wird über den EJB-Container realisiert.

Die Notwendigkeit des Einsatzes eines solchen EJB-Containers ist eng mit dem Einsatz von J2EE verbunden. J2EE bedeutet die Server-seitige Ausführung von Anwendungen. Damit ist – im Gegensatz zu den J2SE (Java 2 Standard Edition) auf Client-Ausführung ausgelegten Anwendungen – die Steuerung und die Ausführungsebene auf den Server, in unserem Fall also auf den Application-Server, verlagert worden.

Alle auf dem EJB-Container ausgeführten Dateien bzw. Anwendungen sind an deren Endung „.jar“ zu erkennen.

5.2.9
Seite 12

Client-Container

Die dritte Container-Variante stellt neben den beiden auf dem Server erforderlichen Container einen Zusatz auf der Client-Seite dar – der Client-Container. Mit diesem Zusatz-Container ist es möglich, auf dem Client J2EE-Anwendungen in einer EJB-Umgebung auszuführen.

Virtueller Host

Dieser Begriff ist uns auch zum Beispiel aus dem Domino-Umfeld und dem Einsatz von HTTP-Servern bekannt. Mit dem Einsatz der virtuellen Hosts können wir auf einem physischen Rechner mehrere spezifische Server simulieren. Das hat den Vorteil, dass aus Sicht der Application-Server auf einem Rechner mehrere Application-Server parallel installiert werden können. Diese Application-Server sind ab einer Ebene logisch voneinander getrennt und erlauben so zum Beispiel die Aufteilung von Anwendungen auf unterschiedliche IP-Adressen. Damit lassen sich unterschiedliche Anwendungen und Anforderungen leicht auf einem System abbilden.

WebSphere-Sicherheit

Sicherheit spielt in einem Bereich, der unterschiedlichen Benutzergruppen zur Verfügung gestellt wird, eine essentielle Rolle. Die Sicherheitseinstellungen lassen sich dabei nicht auf einen einzelnen Teilbereich beschränken, sie müssen allgemein betrachtet werden. Anwendungsumgebungen, die durch WebSphere gebildet werden, bestehen aus einer Vielzahl unterschiedlicher Ebenen, die teilweise separat gegen einen unberechtigten Zugriff abgesichert werden müssen.

Sicherheit in einer WebSphere-Umgebung bedeutet demnach zum Beispiel Sicherheitsüberlegungen auf Ebene von:

- Netzwerk
- Web-Server
- Anwendungs-Server
- Administration
- Anwendung
- Datenbank
- Verschlüsselung
- Middleware
- LDAP
- Betriebssystem
- Single Signon
- Web-Sicherheit
- EJB-Sicherheit

Diese Beispielliste lässt sich beliebig erweitern.

Neben den Ebenen der Sicherheitsbetrachtung differenziert man Konfigurationsbereiche auf dem WebSphere Application-Server, wenn man die Möglichkeiten der Einstellungen der sicherheitsrelevanten Optionen prüft, beispielsweise in Verwaltungssicherung und Anwendungssicherheit.

Verwaltungssicherheit

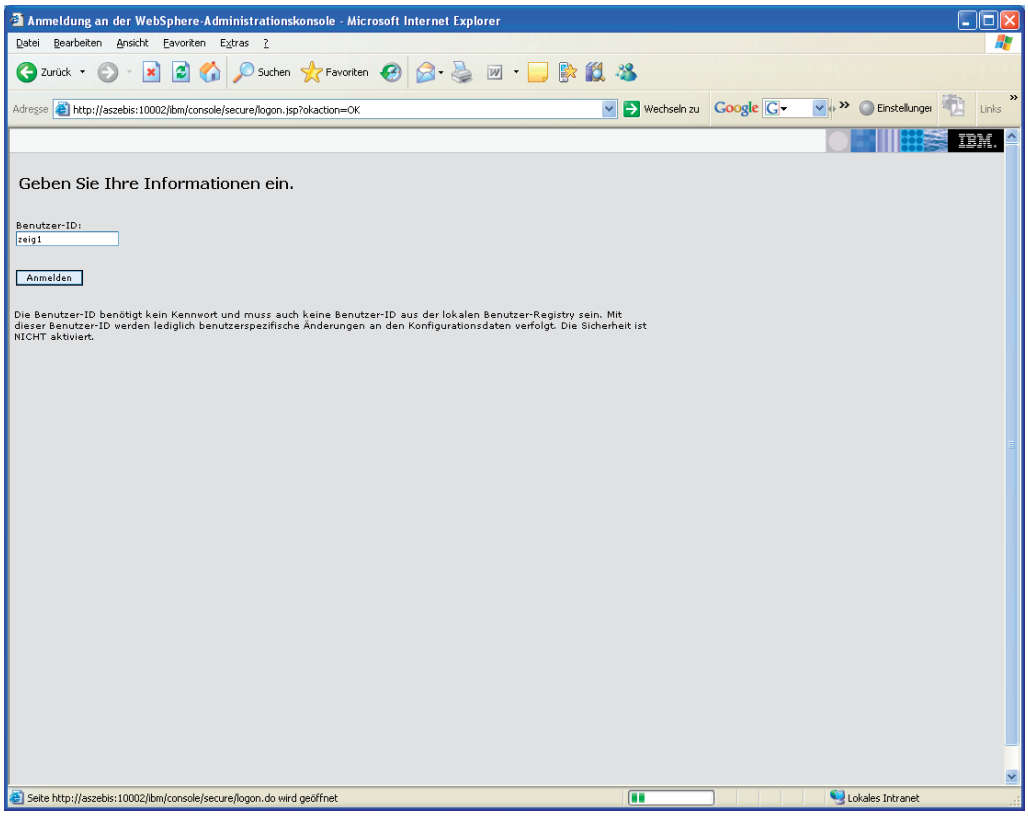
Unter Verwaltungssicherheit verstehen wir grundsätzliche Sicherheitseinstellungen auf einem WebSphere Application-Server in Bezug auf Administration. Wenn die Verwaltungssicherheit aktiviert ist, lassen sich auch die damit verbundenen Sicherheitseinstellungen vornehmen. Mit dieser Ebene der Sicherheitskonfiguration lassen sich nachfolgende Bereiche konfigurieren:

5.2.10
Seite 2

- Sicherheitseinstellungen für Administrationskonsole
- Authentifizierungsverfahren allgemein
- Authentifizierung der HTTP-Verbindungen
- SSL-Verschlüsselung
- Authentifizierung von IIOP-Clients

In diesem Kapitel möchte ich Ihnen die wesentlichen Funktionen aufzeigen, die in Bezug auf die Sicherheit mit WebSphere Application-Server eine Rolle spielen.

Wenn Sie den WebSphere Application-Server installiert haben und den Administrationsbereich starten, dann werden Sie in jedem Fall dazu aufgefordert, einen Benutzer anzugeben.

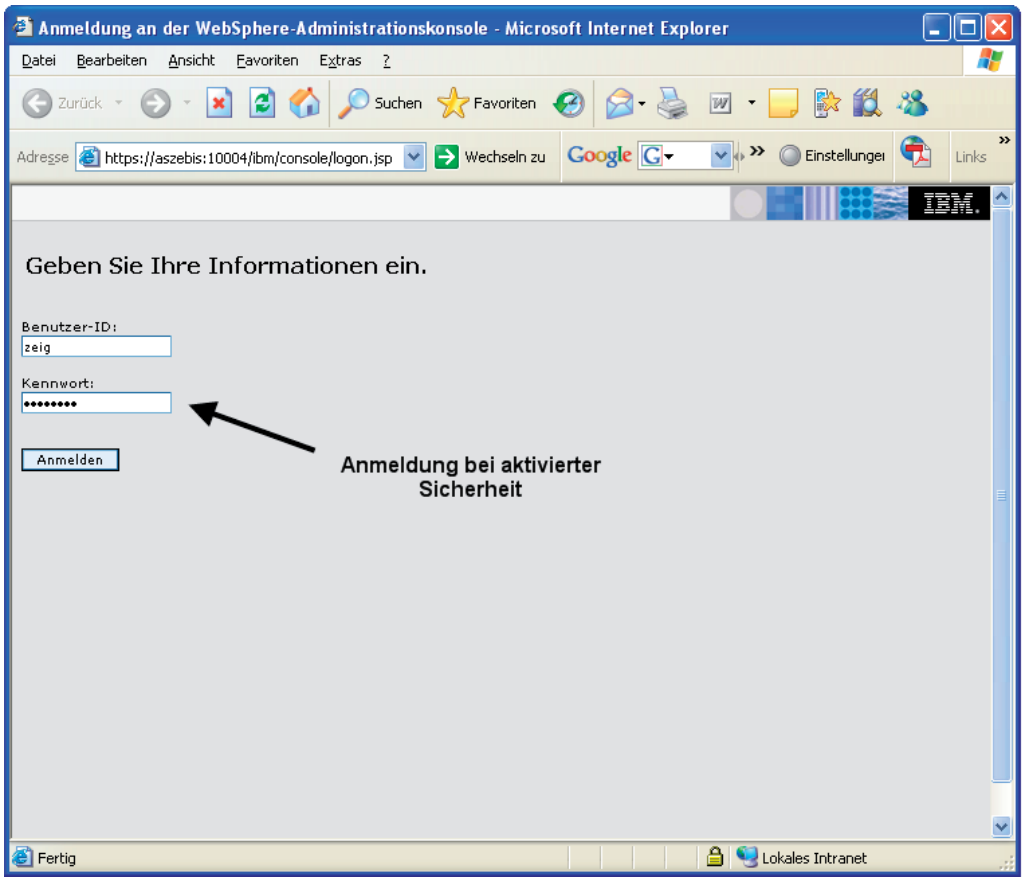


Benutzer-ID angeben

Beachten Sie den Hinweistext in diesem Fenster. Wenn Sie nicht die Sicherheitseinstellungen für den WebSphere Application-Server angepasst haben, können Sie als „Benutzer“ einen beliebigen Eintrag vornehmen. Dieser wird nicht überprüft – weder auf Existenz noch auf Berechtigung; er wird lediglich für Protokollierungen verwendet.

Lassen Sie sich also nicht in jedem Fall von dieser Anmeldeanzeige irreführen – sie symbolisiert keine Sicherheitseinstellungen. Erst wenn Sie die globale

Sicherheit für den Server aktiviert haben, wird eine Prüfung auf den eingegebenen Benutzer und dessen Berechtigung zur Verwaltung des Servers durchgeführt. Wie das funktioniert, erfahren Sie im weiteren Verlauf dieses Kapitels. Ein Beispiel für eine Anmeldung mit aktivierter Sicherheit finden Sie in der folgenden Abbildung:



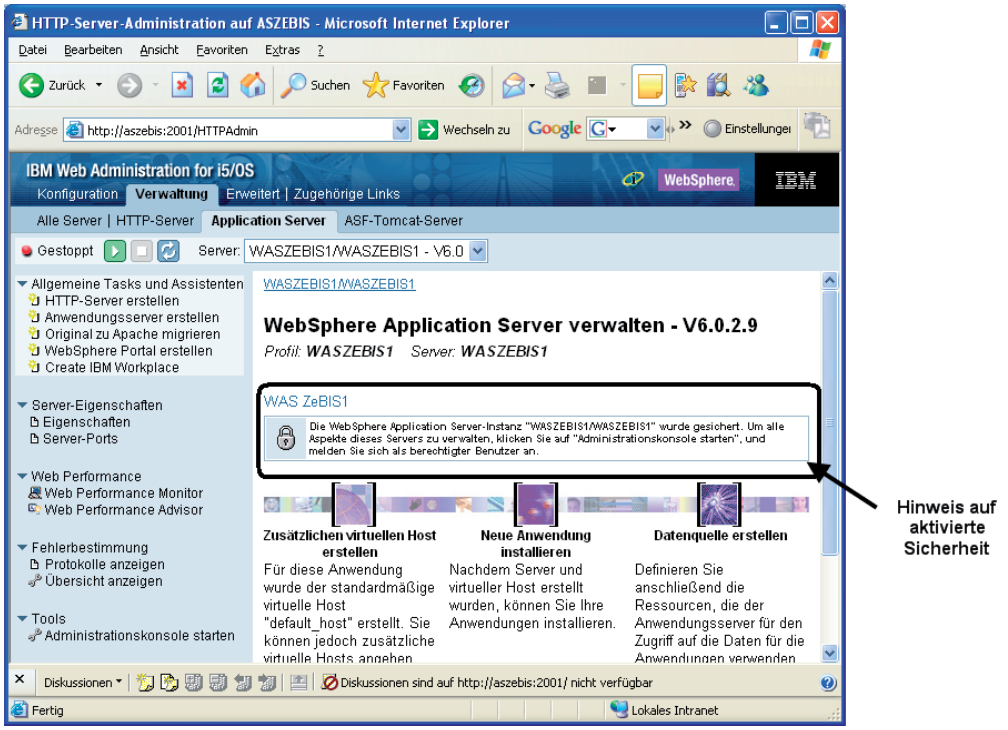
Anmeldung bei aktivierter Sicherheit

Es empfiehlt sich, dass mit einer fehlenden Sicherheitskonfiguration wirklich jeder, der Zugang zum Server erhält, diesen administrieren kann.

Erst wenn die Sicherheit auf dem WebSphere Application-Server aktiviert worden ist, wird mit dem Anmeldefenster eine Prüfung durchgeführt. Diese Prüfung zeigt, ob der Benutzer (und das bei aktiver Sicherheit zusätzlich anzugebende Kennwort) korrekt ist und ob er zuvor als Administrator für diesen Bereich festgelegt wurde.

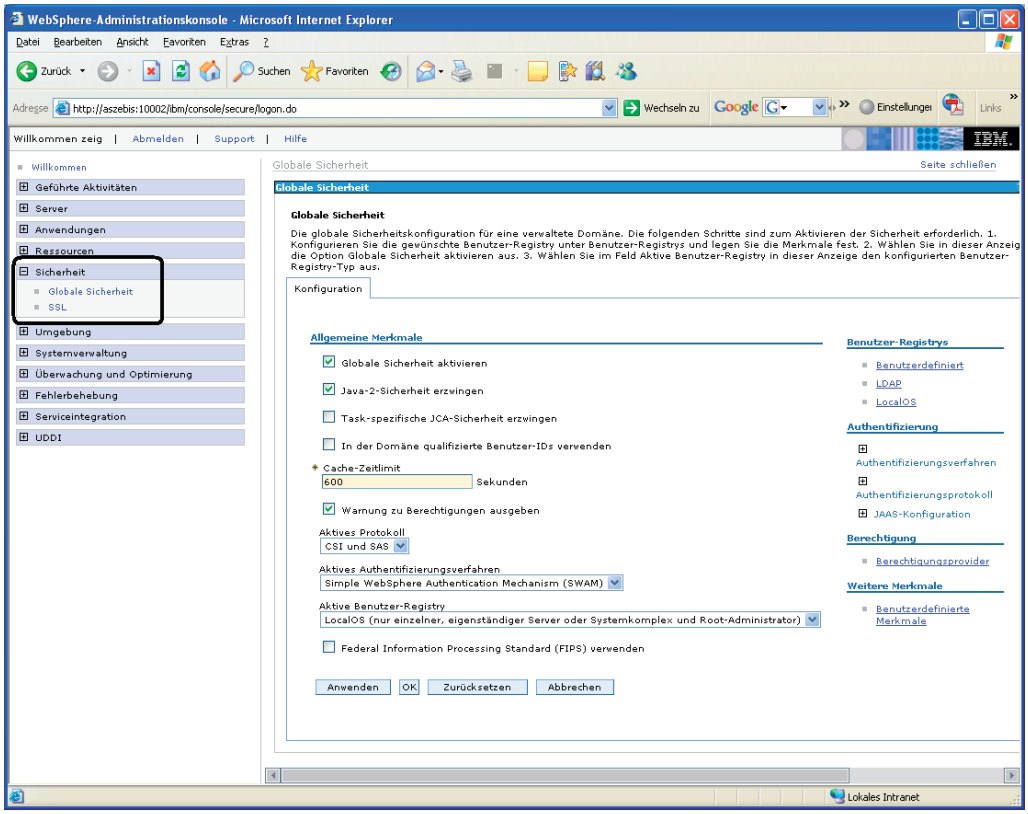
Die folgende Abbildung weist bereits auf Ebene der allgemeinen Server-Verwaltung hin, dass dieser WebSphere Application-Server mit Sicherheitseinstellungen abgesichert worden ist:

5.2.10
Seite 4



Gesicherter Server

Die folgende Abbildung zeigt den allgemeinen Verwaltungsbereich der Sicherheitseinstellungen in der Administrationskonsole eines WebSphere Application-Servers, Version 6.



Sicherheit im WebSphere-Administrator

Grundsätzlich gilt es, in Bezug auf Sicherheit in einer WebSphere-Umgebung folgende Sicherheitsbereiche zu bedenken und nach Bedarf entsprechend zu konfigurieren:

Authentifizierungsprotokoll

Dieses Protokoll wird bei aktivierter Sicherheit auf einem WebSphere Application-Server benötigt, um RMI über IIOP durchzuführen.

RMI:
RMI ist die Abkürzung für „Remote Method Invocation“ und stellt eine recht einfache und vor allem schnelle Möglichkeit dar, in einer Java-Umgebung eine Kommunikationsstruktur aufzubauen. Diese Form der Technologie ist in modernen Anwendungen häufig anzutreffen.

IIOP
Bei IIOP handelt es sich um die Abkürzung für „Internet Inter-ORB Protocol“ – ein Protokoll, das für Methodenaufrufe von Objekten auf fernen Rechner verwendet wird.

5.2.10
Seite 6

Authentifizierungsverfahren

Innerhalb eines WebSphere Application-Servers wird LTPA (Lightweight Third Party Authentification) als Standard-Authentifizierungsverfahren verwendet. LTPA bildet die Grundlage für die Implementierung von Single-Sign-on-Lösungen in einer WebSphere-Umgebung.

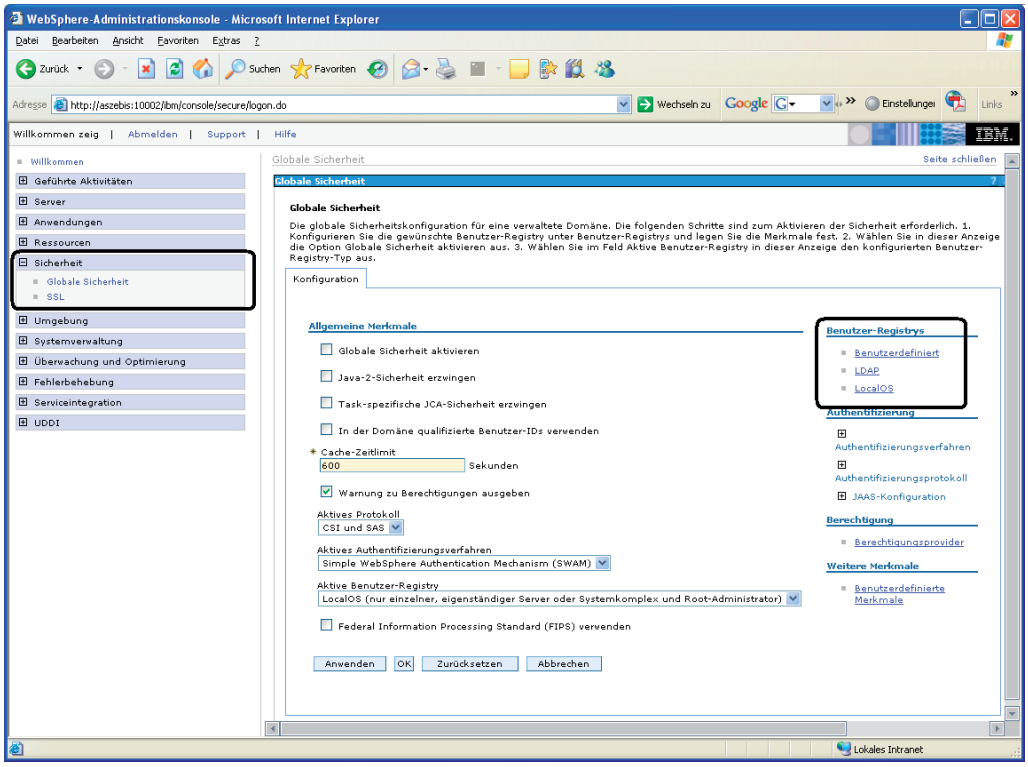
Benutzer Repository

Für die Benutzerregistrierung kann in einer WebSphere-Installation auf unterschiedliche Standards zugegriffen werden:

- Benutzerverwaltung des lokalen Betriebssystems
- LDAP-Registrierung
- Benutzerdefinierte Benutzerverwaltung und Registrierung

Administrative Sicherheit

Der Begriff der „Administrativen Sicherheit“ wurde von IBM im Laufe der Bereitstellung des WebSphere Application-Servers Version 6.x eingeführt. Bis zum Erscheinen der Version 6.1 wurde dieser Bereich als „Globale Sicherheit“ bezeichnet. Welche der beiden Bezeichnungen treffender ist, dürfte Ansichtssache sein. Der neuere Begriff „Administrative Sicherheit“ bezeichnet jedoch genauer, um welchen Bereich der Sicherheitseinstellungen es geht: um den Bereich der Administration des WebSphere Application-Servers und dessen Schutz gegen unberechtigte Verwendung. Ist dieser Bereich der Sicherheit erst einmal konfiguriert, können wir uns auch mit den übrigen sicherheitsrelevanten Themen und Teilbereichen innerhalb von WebSphere beschäftigen.



Benutzer-Registrys

Der Bereich der Sicherheitseinstellungen wird zunächst im linken Navigationsabschnitt in die beiden Gruppen „Globale Sicherheit“ und „SSL“ unterteilt. An dieser Stelle beschäftigen wir uns zunächst mit der „Globalen Sicherheit“ beziehungsweise mit der „Administrativen Sicherheit“.

Mit einem Klick auf den Eintrag „Globale Sicherheit“ gelangen wir in die vorhergehende Abbildung. Wie man erkennen kann, ist die Globale Sicherheit noch nicht aktiviert. Dies ist der Auslieferungszustand eines WebSphere Application-Servers.

5.2.10.1
Seite 2

Während im mittleren Bereich die allgemeinen Sicherheitsdefinitionen vorgenommen werden können, sollten wir uns zunächst mit dem rechten Bereich, genauer mit dem Abschnitt „Benutzer-Registrys“ beschäftigen.

Für die Anmeldung an einem WebSphere Application-Server sowie für die Absicherung der darauf laufenden Anwendungen bietet IBM unterschiedliche Optionen, um gültige Benutzer zu identifizieren. Diese Optionen sind im Abschnitt „Benutzer-Registrys“ in Form der folgenden Hauptgruppen zu finden:

- Benutzerdefiniert
- LDAP
- Local/OS

Für die Anmeldung am Anwendungs-Server ist festzulegen, wo die WebSphere-Benutzer gepflegt und wie sie geprüft werden sollen. Das Festlegen der korrekten Registry wirkt sich auf nahezu alle Bereiche der WebSphere-Installation aus. Wählen Sie deshalb sorgfältig die korrekte Basis für die Sicherheitskonfiguration.

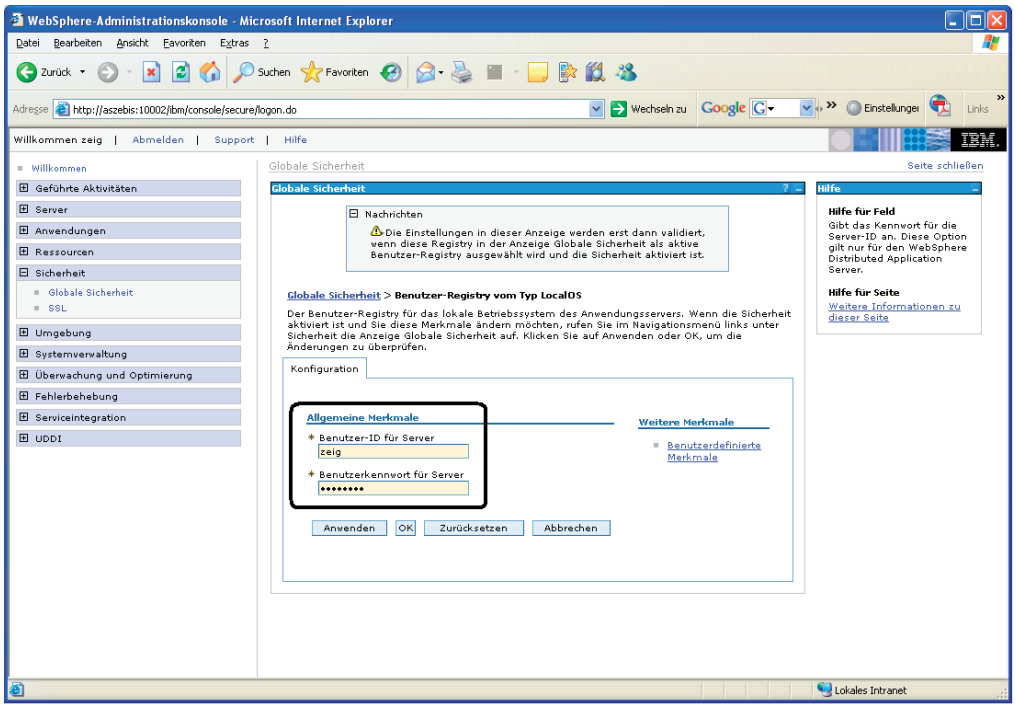
Wenn Sie keine Vorgaben machen, dann nutzt der Sicherheitsprozess per Standard die Prüfung gegen das lokale Betriebssystem – in unserem Falle gegen die im i5/OS gepflegten Benutzer. Achten Sie darauf, dass der später anzugebende Benutzer über ausreichende Berechtigung für die Ausführung der Aufgaben besitzt.

Während die unterschiedlichen Registry-Optionen auch verschiedene Verfahren der Benutzerverwaltung mit sich bringen, verfügen alle drei Optionen über eine gemeinsame Benutzer-ID für den Server. Dabei handelt es sich um eine besondere ID, die für die allgemeine Verwaltung im WebSphere-Umfeld verwendet wird. Diese ID ist eine Art „Sonderadministrator“ und sollte aus diesem Grund auch nur einem ausgewählten Personenkreis zugänglich gemacht werden.

Wenn Sie erstmals die Sicherheit auf Ihrem Server aktivieren möchten, dann ist die Reihenfolge der Schritte, die ich Ihnen nachfolgend exemplarisch aufzeigen werde, in gewisser Weise beliebig. Es bleibt Ihnen überlassen, ob Sie zunächst die Benutzer-Registry festlegen (wie in diesem Beispiel) oder ob Sie als Erstes die „Globale Sicherheit“ aktivieren.

Um die globale oder administrative Sicherheit auf dem Server zu aktivieren, führen wir nun folgende Schritte aus. Klicken Sie dazu zunächst im rechten Abschnitt auf den Bereich „Benutzer-Registrys“ und wählen Sie dort das gewünschte Registrierverfahren aus. Für unser Beispiel nutze ich das lokale Betriebssystem für die Authentifizierung, da dessen Konfiguration am einfachsten durchzuführen ist.

Mit der Auswahl „LocalOS“ gelangen wir anschließend in folgende Anzeige:



Benutzer-ID für den Server

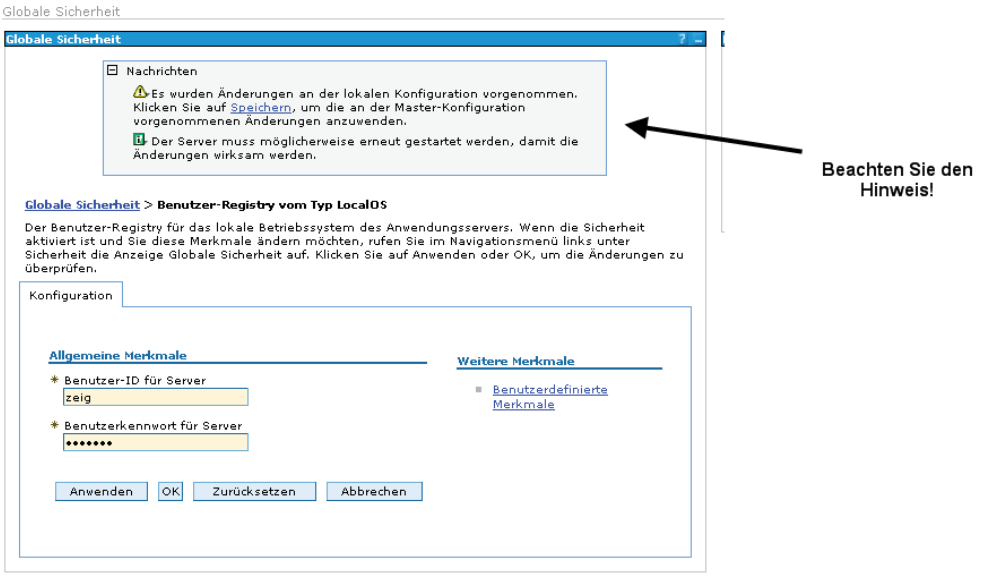
Als Erstes müssen wir die Benutzer-ID angeben, die als Hauptadministrator für den Server verwendet werden soll. Mit dieser ID wird der Server später ausgeführt. Hier muss es sich um einen in der ausgewählten Registry gültigen Eintrag handeln – beispielsweise um einen im i5/OS-definierten Benutzer. Das dazugehörige Kennwort wird später bei der Anmeldung zur WebSphere-Administration ebenso geprüft wie der eingegebene Benutzer.

Diese Vorgaben werden in einem speziellen Bereich innerhalb der WebSphere-Konfiguration gespeichert.

Bestätigen Sie Ihre Eingaben mit einem Klick auf die Schaltfläche „Anwenden“.

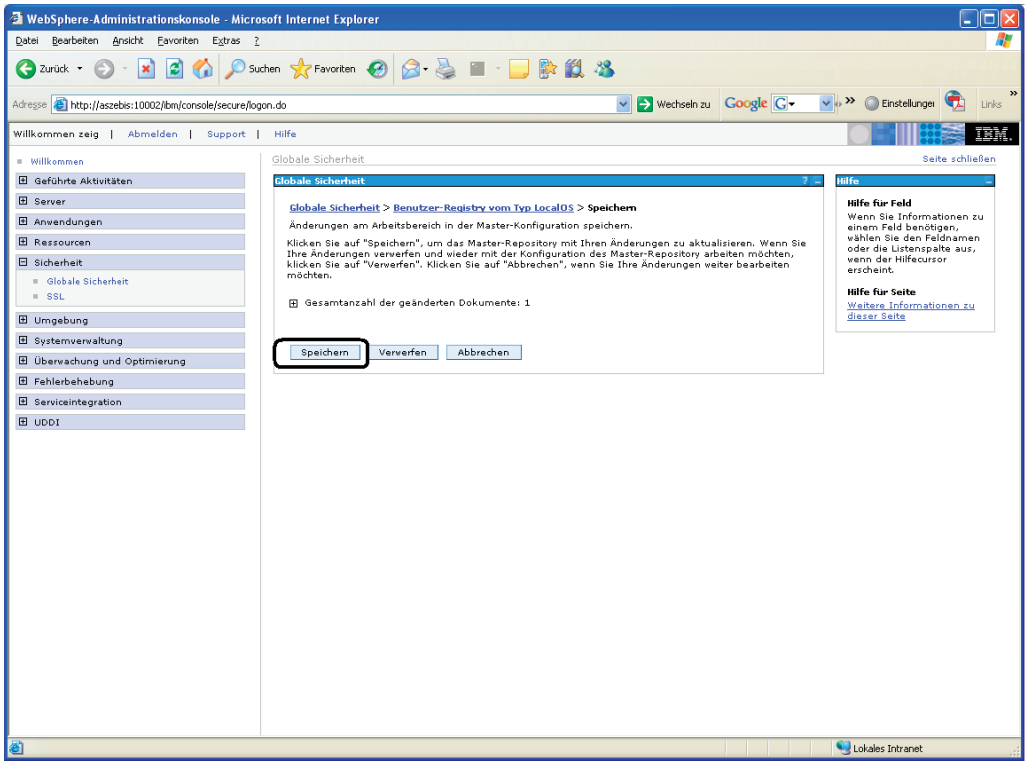
Nach diesen Änderungen müssen eventuell Anpassungen in den unterschiedlichen Ebenen der WebSphere-Konfiguration durchgeführt werden. Diese Änderungen werden in Form einer Hinweisnachricht ausgegeben.

5.2.10.1
Seite 4



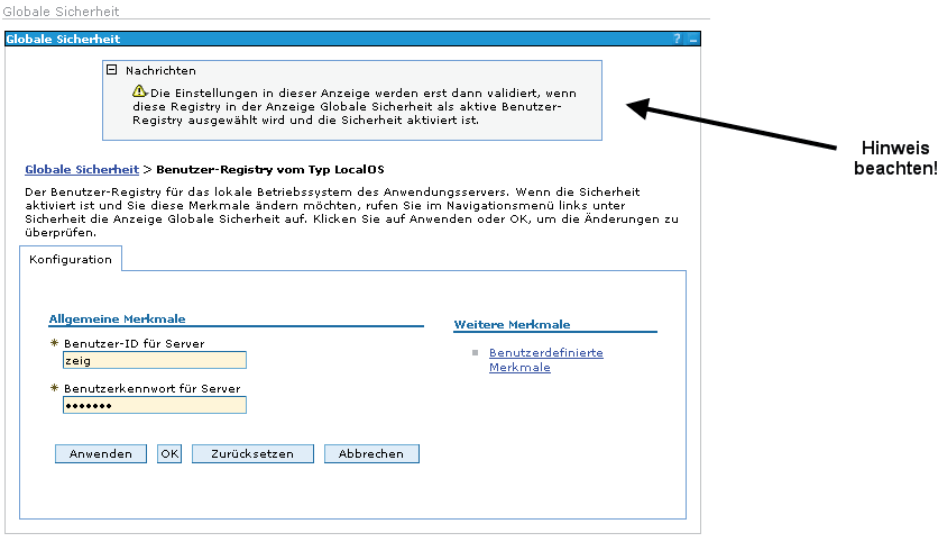
Hinweis beachten

Speichern Sie die anstehenden Änderungen mit einem Klick auf den markierten Textabschnitt „Speichern“. Bestätigen Sie die darauf erscheinende Anzeige, indem Sie auf die Schaltfläche „Speichern“ klicken – siehe folgende Abbildung.



Konfigurationseinstellungen speichern

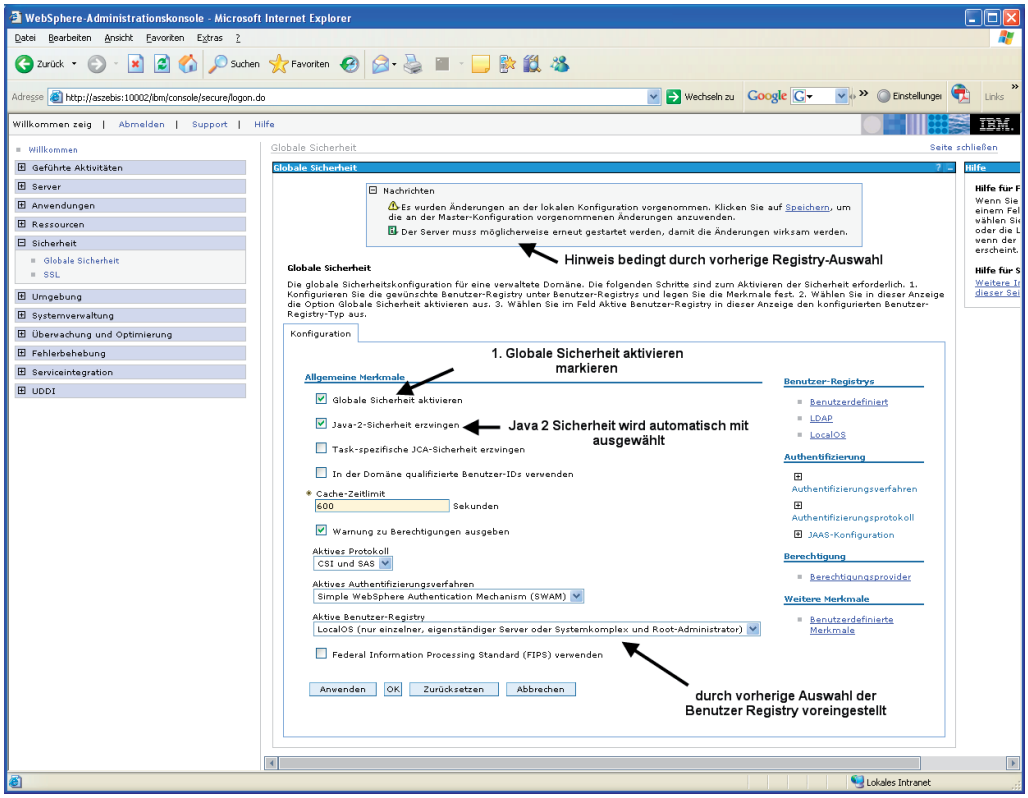
Mit diesen Angaben haben wir die Registry, die zu verwenden ist, definiert. Damit ist die Sicherheit auf dem WAS jedoch noch nicht aktiviert. Eine Nachricht weist auf diese Tatsache hin: Die globale beziehungsweise administrative Sicherheit muss auf dem WebSphere-Server aktiviert werden.



Hinweis beachten

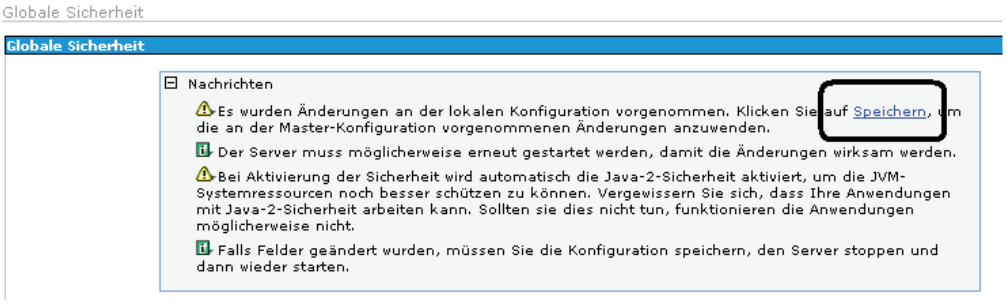
Das Aktivieren der administrativen Sicherheit wird mit einem Klick auf das Auswahlkennzeichen links neben dem Eintrag „Globale Sicherheit aktivieren“ durchgeführt. Wenn Sie diese Option selektieren, wird automatisch auch die zweite Selektion „Java-2-Sicherheit erzwingen“ aktiviert. Diese Option ist erforderlich, um auch die Java-Ebene abzusichern. Allerdings sollten Sie dann bei weiterem Betrieb des WAS darauf achten, dass die Anwendungen auch auf diese aktivierte Sicherheit entwickelt worden sind und ordnungsgemäß ausgeführt werden können.

5.2.10.1
Seite 6



Globale Sicherheit aktivieren

Mit Aktivierung der administrativen Sicherheit haben wir wieder Änderungen an der WebSphere-Konfiguration durchgeführt, die ein „Speichern“ erforderlich machen.

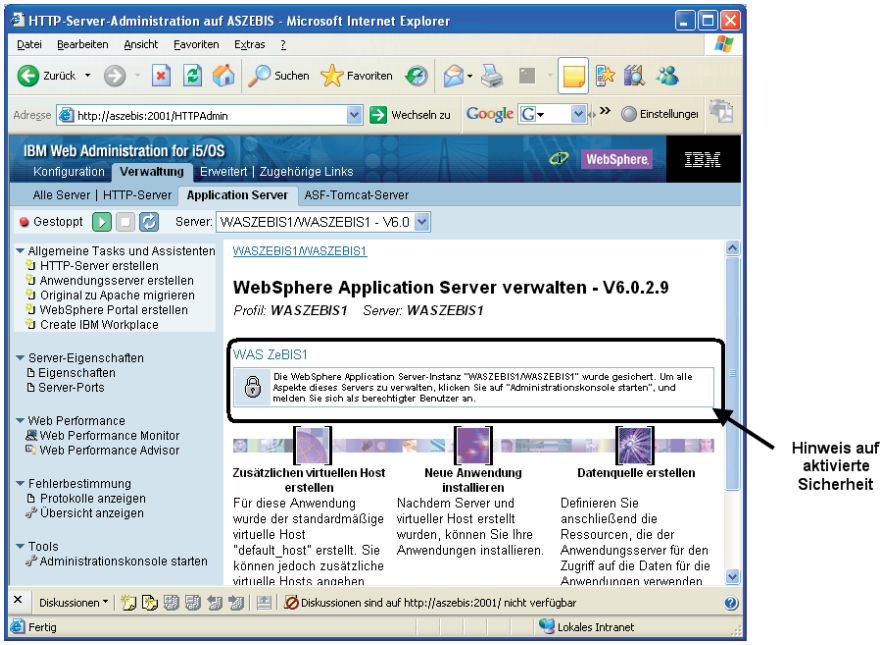


Änderungen speichern

Klicken Sie auch in dieser Nachricht wieder auf den Bereich „Speichern“ und bestätigen Sie dadurch die Aktivierung der Änderungen.

Speichern

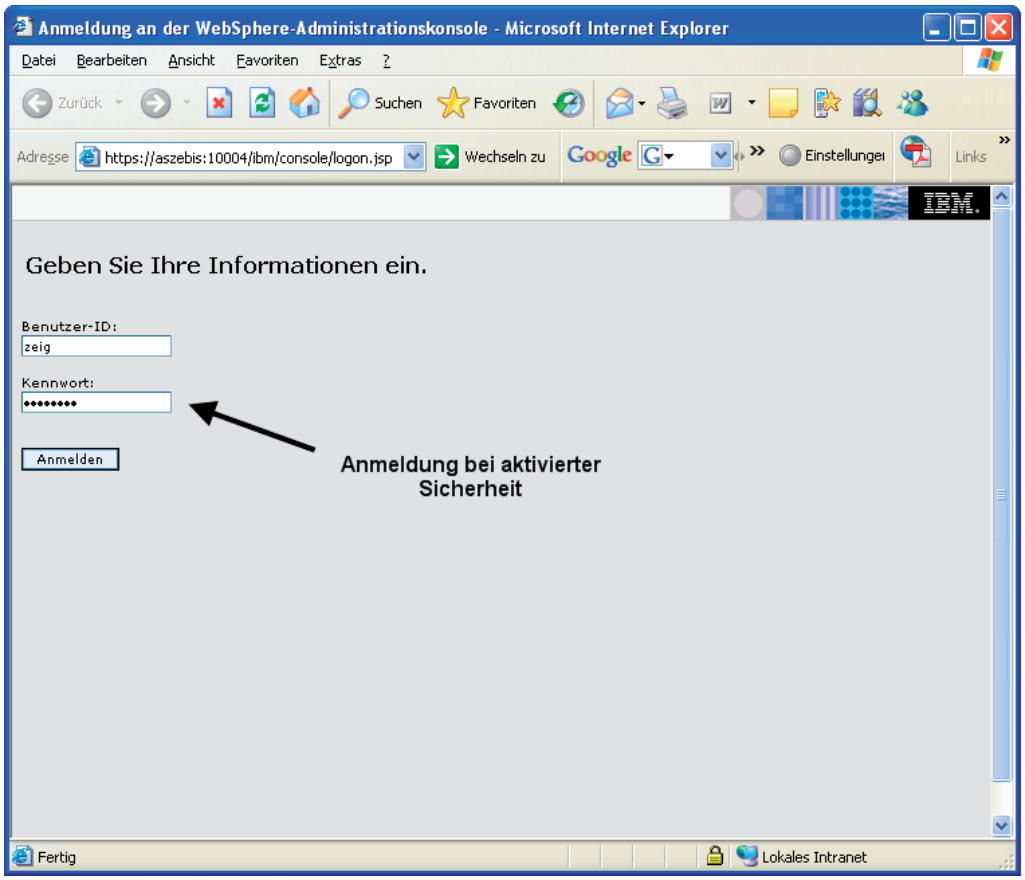
Nach erneutem Start des WebSphere Application-Servers ist nun die administrative Sicherheit aktiviert. Diese Einstellung lässt sich bereits anhand der Server-Übersicht feststellen. Dort erhalten wir einen Hinweis auf die Sicherheitsaktivierung:



Gesicherter Server

Wenn Sie nun die Administrationskonsole für diesen Server starten, werden Sie zur Eingabe der Benutzer-ID für den WAS sowie dem dazugehörigen Kennwort aufgefordert. Nun erfolgt eine Prüfung, ob Sie die korrekte Benutzer-ID angegeben haben und damit den Verwaltungsbereich öffnen dürfen.

5.2.10.1
Seite 8



Anmeldung bei aktivierter Sicherheit

Sicherung und Wiederherstellung WAS 6.0

Die Sicherung einer WebSphere-Umgebung ist natürlich – wie in anderen IT-Bereichen auch – ein wichtiges Thema.

Durch die Implementierung von WebSphere auf dem System i sind die verschiedenen Bereiche der WebSphere-Implementierung bei einem Sicherungsszenario entsprechend zu berücksichtigen.

Die Standard-Befehle (SAVxxx) des System i für die Sicherungsprozesse (z. B. im Menü GO SAVE) lassen sich auch für die WebSphere-Umgebung anwenden. Die im Folgenden gezeigten Sicherungsbefehle führen alle zu einer Sicherung im WebSphere-Umfeld. Beachten Sie bitte, dass die einzelnen Komponenten einer WAS-Installation auf unterschiedlichste Ebenen und Bereiche des System i verteilt sein können (Bibliotheken, Verzeichnisse, Berechtigungen etc). Die Gesamtheit dieser einzelnen Fragmente bietet erst im Wiederherstellungsfall die Erfolgsgarantie. Achten Sie deshalb sorgfältig darauf, dass auch tatsächlich alle Bestandteile der WAS-Umgebung gesichert werden.

Während Systemeinstellungen in größeren zeitlichen Abständen oder beispielsweise nach Konfigurationsänderungen oder dem Einspielen von Fixes gesichert werden sollten, ist die Datensicherung der eigentlichen Benutzerdaten täglich empfehlenswert.

Während die verteilten Komponenten der WAS-Integration auf dem System i die Planung der Sicherungsprozesse der WAS-Umgebung etwas komplexer erscheinen lassen als dies bei anderen System i-Bereichen der Fall ist, kommt dem Wiederherstellungsprozess einer WAS-Umgebung dieselbe Bedeutung zu. Allerdings nur, wenn eine Wiederherstellung auf demselben System durchgeführt werden soll.

Wird der Wiederherstellungsprozess beispielsweise auf einem anderen System durchgeführt, dann müssen die unterschiedlichen Einstellungen angepasst werden. Datenbanknamen, TCP/IP-Definitionen und Berechtigungsstrukturen sind eng mit dem System i verbunden und bedürfen in einem Wiederherstellungsszenario auf einem anderen System besonderer Beachtung. Die Zuordnung und Konfiguration von WebServer-Instanzen muss ebenfalls bei der Datensicherung und Wiederherstellung berücksichtigt werden. Doch dazu später mehr.

14. Ergänzung 3/2008

Betrachten wir zunächst einmal die Implementierung von WebSphere auf dem System i.

Die Implementierung der reinen WAS-Komponenten wird über folgende Bereiche abgebildet:

- Bibliotheken
  – (z. B. QWAS6x)
- Verzeichnisse im IFS
  – (z. B. in QIBM/ProdData)
- Besondere Benutzerprofile für die WAS-Integration

Bibliotheken

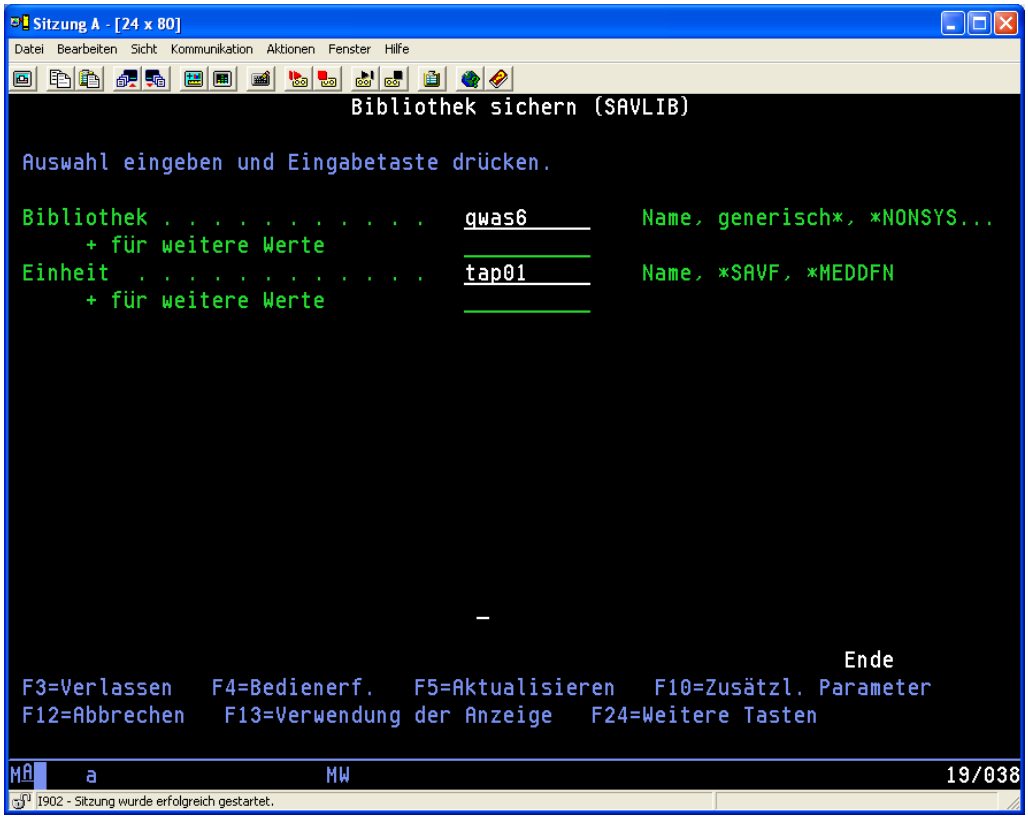
Die Implementierung von WebSphere native auf dem System i wird unter anderem über das klassische Bibliothekskonstrukt realisiert. Die verwendeten Bibliotheken können je nach installierter WAS-Version differenzierte Namen tragen.

Bei dem Einsatz der Version 6.0 des WAS finden wir auf dem System die Bibliothek QWAS60. Diese dient als Basis für alle WAS-Instanzen, welche auf dem System i konfiguriert werden.

Bibliothek QWAS6

Die Bibliothek QWAS6 beinhaltet unter anderem die folgenden Bestandteile:

- Produktbeschreibung
- Details zu den installierten Optionen
- Subsystembeschreibung QWAS6 für die Ausführung des WAS
- Jobbeschreibung QWASJOBD
- Ausgabewarteschlange QWASOUTQ
- Nachrichtendaten QWAS6MSGF
- Klassendefinition QWAS6



SAVLIB QWAS6

Die automatischen Sicherungsfunktionen in dem Menü „GO SAVE“ sichern ebenfalls den Inhalt dieser Bibliotheken mit. Dabei sind natürlich die Besonderheiten in Bezug auf die Sicherung von System- und Benutzerdaten zu beachten.

Anmerkung

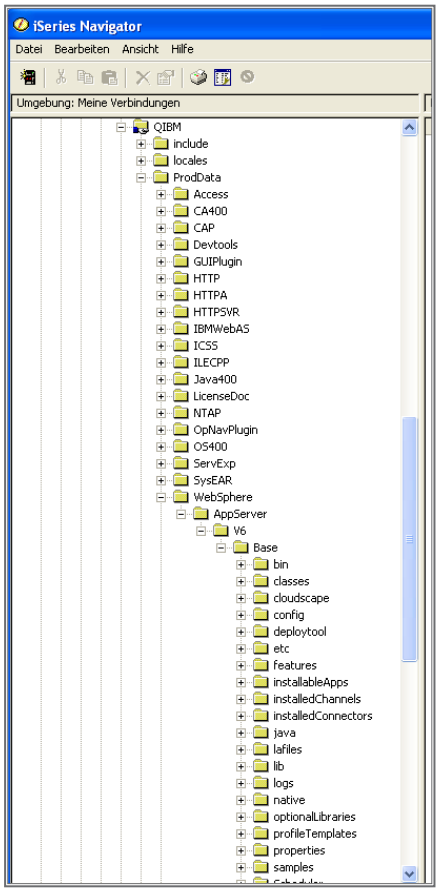
Achten Sie darauf, dass bei einer Gesamtsicherung unter Umständen die Inhalte von Ausgabewarteschlangen oder anderen Warteschlangen nicht mitgesichert werden!

5.2.11
Seite 4

Verzeichnisse

Neben den zuvor genannten Bibliotheken bilden die Verzeichnisse innerhalb des IFS einen wesentlichen Bestandteil der WebSphere-Installation auf dem System i. Bis zur Version 6.0 kann der WAS auf einem System i nur in von IBM vorgegebenen Verzeichnissen installiert werden. Dabei werden die Komponenten des WAS in den eigentlichen Systemteil und den Benutzerbereich differenziert. Die beiden Hauptkomponenten der WAS-Installation in dem IFS stellen sich deshalb wie folgt dar:

/QIBM/ProdData/WebSphere/AppServer/V6
Dieses Verzeichnis beinhaltet die Systemkonfiguration und Basisdaten. Je nach installierter Version können innerhalb des Verzeichnisses V6 unterschiedliche Einträge erscheinen, welche sich aus den installierten WAS-Versionen (Base, Network Deployment etc.) ergeben.



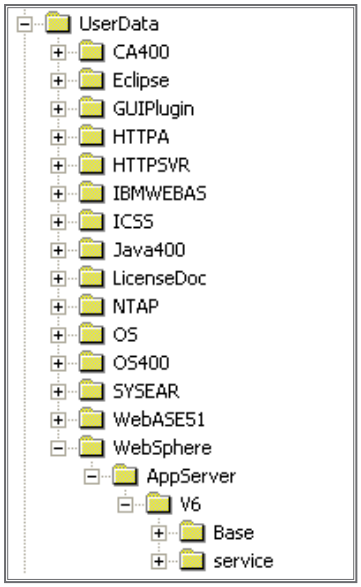
Verzeichnisstruktur WAS 6.0

Anmerkung

Wie bei anderen systemeigenen Bereichen dürfen auch in dem Verzeichnis ProdData KEINE Veränderungen durchgeführt werden. Dieses Verzeichnis dient als Systemverzeichnis. Notwendige Einstellungen müssen in dem Benutzerbereich – also dem Verzeichnisbereich UserData vorgenommen werden!

/QIBM/UserData/WebSphere/AppServer/V6

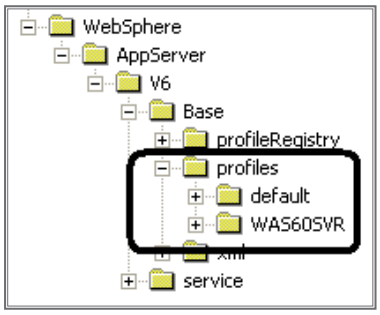
In diesem Verzeichnis finden wir die Benutzerdaten. Auch hier ist, wie bereits in dem ProdData-Bereich gezeigt, die Serverinstallation über ein Unterverzeichnis innerhalb von V6 abgebildet. Die Sicherung dieses UserData-Inhalts stellt eine Benutzerdatensicherung aus Sicht des WAS dar – allerdings ohne beispielsweise Datenbankinhalte.



Verzeichnisstruktur UserData

Der Inhalt des Verzeichnisses UserData dient der Speicherung verschiedener Serverinstallationen. Wenn Sie auf Ihrem System unterschiedliche Profile verwenden, dann wird jedes Profil in einem separaten Verzeichnis gespeichert. Damit kann man bei Bedarf auch einzelne Serverprofile sichern, um diese bei Bedarf wiederherstellen zu können.

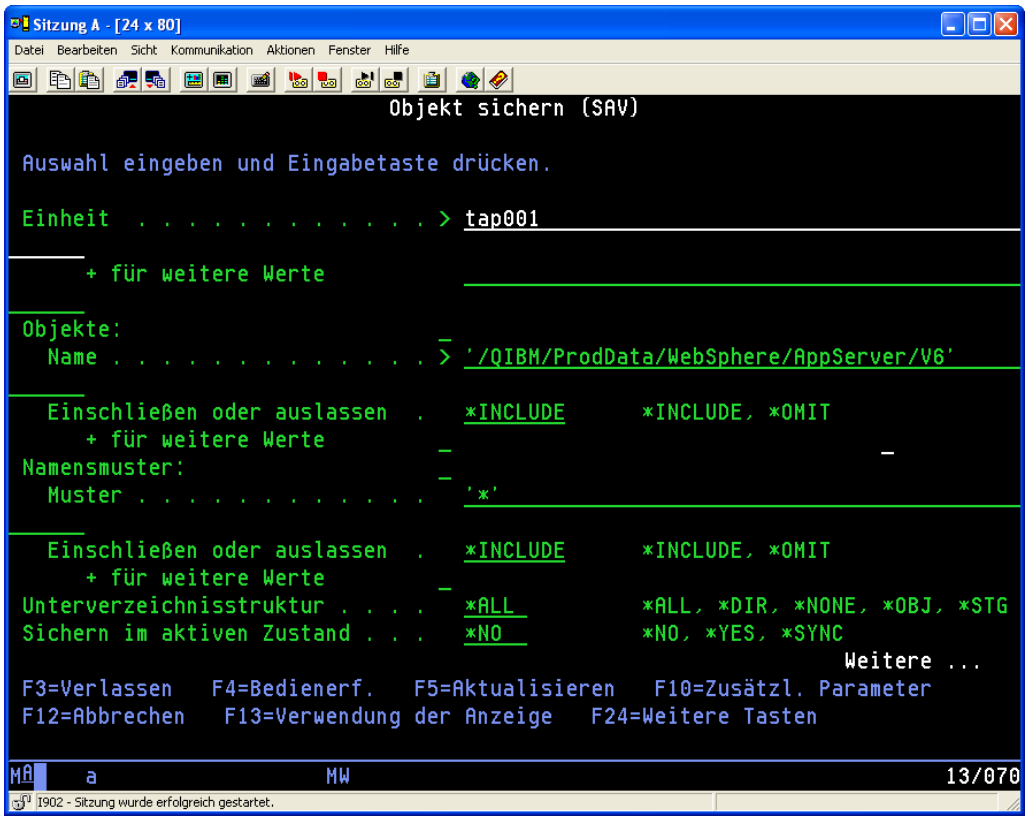
Die folgende Abbildung zeigt ein Beispiel für solche Profile. Neben dem im Standard installierten Profil „default“ sehen Sie auch ein individuelles Profil WAS60SVR.



Profile

Im Regelfall bietet es sich an, die WAS-Umgebung komplett zu sichern. Damit schließt man etwaige Deltas beim Wiederherstellen aus. Sollte es jedoch Gründe für eine Reduzierung der Datensicherung geben, dann wäre die Beschränkung auf einzelne Profile innerhalb des UserData-Verzeichnisses denkbar.

Die Sicherung der Verzeichnisse kann beispielsweise mit dem Befehl „SAV“ durchgeführt werden.



SAV

Anmerkung

Der Befehl SAV kann auch für die Sicherung einzelner Bereiche einer WAS-Installation angewendet werden. Wenn Sie beispielsweise eine einzelne Anwendung eines WAS sichern wollen, dann ermitteln Sie in dem Installationsverzeichnis den Namen der installierten Anwendung und erweitern den Parameter „Objekte“ in dem Befehl SAV um die zusätzlichen Pfadangaben.

Benutzerprofil

Häufig werden sie bei der Datensicherung vergessen bzw. bei separaten Sicherungsprozessen nicht bedacht – die Benutzerprofile. Es ist die Arbeitsweise der Sicherheitsmechanismen des System i, dass jedes Objekt einen Eigner hat. Darüber hinaus sind Benutzerprofile maßgeblich für die Ausführung von Jobs auf dem System – zumindest, was deren Berechtigungssteuerung betrifft. Die Ausführung des WebSphere Application Servers auf dem System i basiert auf bestimmten Berechtigungen und Eigentumsverhältnissen. Sind diese nicht gegeben, kann es bei dem Betrieb des WAS zu Problemen kommen oder

der Server lässt sich im Extremfall wegen fehlerhafter Berechtigungsprüfung nicht starten. Stellen Sie deshalb bei einem WAS-Sicherungskonzept sicher, dass neben den Bibliotheken und Verzeichnissen auch die Benutzerprofile des WAS gesichert werden. Dabei sind die folgenden Benutzerprofile wichtig:

- QEJB
- QEJBSVR

Anmerkung

Bei der Installation eines WAS kann auch ein von dem Standard abweichendes Benutzerprofil für die Ausführung des WAS vorgegeben werden. Stellen Sie in einem solchen Fall sicher, dass auch dieses Profil mitgesichert wird!

Für die Sicherung eines Benutzerprofils verwenden wir den Befehl SAVSECDTA:

SAVSECDTA

Ab der Version 6.0.2 kann eine WebSphere Produktinstallation auch mit dem Befehl „Lizenzprogramm sichern“ SAVLICPGM gesichert werden.

SAVLICPGM

Dieser Sicherungsbefehl sichert das Lizenzprogramm des WAS mit den installierten PTFs. Im Wiederherstellungsfall kann die WAS-Basis mit dem Befehl RSTLICPGM mit den gesicherten PTFs zurückgespeichert werden. Benutzerdaten werden bei diesem Sicherungsbefehl jedoch nicht berücksichtigt.

Anmerkung

Der Befehl SAVLICPGM muss für alle Optionen der WAS-Installation durchgeführt werden. Haben Sie auf Ihrem System beispielsweise die Express Variante installiert, dann muss neben der in der vorherigen Abbildung gezeigten Option *BASE auch die Option 1 mit einem separaten SAVLICPGM-Befehl gesichert werden!

Datenbank

Mit diesen Sicherungsroutinen ist die eigentliche WAS-Umgebung des System i gesichert. Allerdings macht eine WAS-Umgebung nur in Verbindung mit der Datenbank Sinn. Die Sicherungsprozesse der Datenbank sind nicht Bestandteil dieses Kapitels. Ausgehend davon, dass die Datenbank des WAS auf der DB2/UDB des System i basiert, lassen sich die Inhalte mit den zuvor gezeigten Befehlen (z. B. SAVLIB, SAVOBJ) sichern.

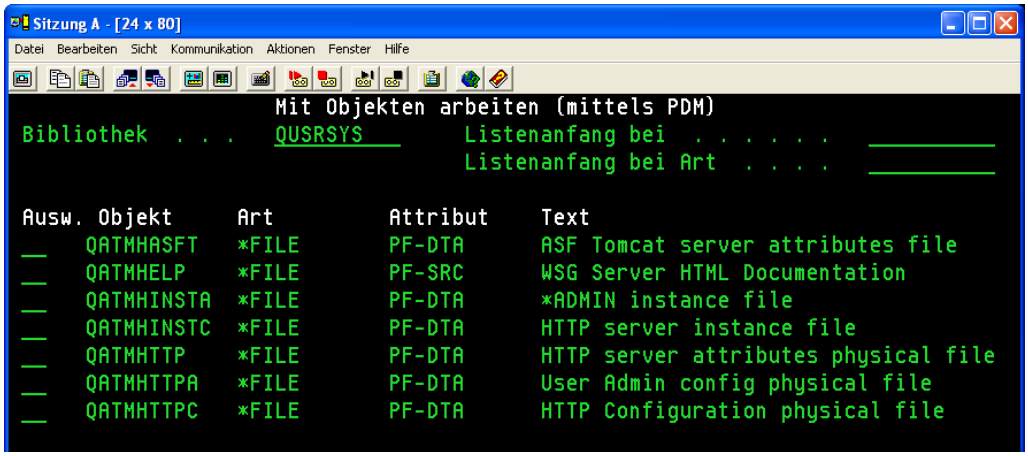
WebServer

Neben der Datenbank bildet auch der WebServer einen wichtigen Bestandteil der WAS-Installation. Die Sicherung des http-Servers mit den Konfigurationseinstellungen ist deshalb im Gesamtkonzept der WAS-Datensicherung zu berücksichtigen!

Generell gelten die zuvor genannten Sicherungsbefehle auch für die Datensicherung des WebServers. Beispielsweise lässt sich der Systemteil mit Hilfe des Befehl SAVLICPGM sichern. Ein Beispiel finden Sie in der folgenden Abbildung:

SAVLICPGM 5722DG1

Zusätzlich wird bei dem Betrieb des WebServers die Bibliothek QUSRSYS einbezogen. Diese speichert verschiedene Konfigurationseinstellungen. Deshalb ist es auch erforderlich, die Bibliothek QUSRSYS mit in das Datensicherungskonzept des WAS zu integrieren. In der folgenden Abbildung sehen Sie einige Objekte, welche für den Betrieb des WebServers auf dem System i wichtig sind.



QUSRSYS http-Objekte

Alternativ zur Sicherung der gesamten QUSRSYS können auch einzelne Objekte gesichert werden. Der zugehörige Befehl SAVOBJ kann beispielsweise wie folgt angewendet werden:

```
SAVOBJ OBJ(QATMHINSTC) LIB(QUSRSYS) DEV(TAP01) OBJTYPE(*FILE)
```

Mit der in dem vorherigen Beispiel gesicherten Datei QATMHINSTC werden die http-Server-Instanzen gespeichert. Diese können im Bedarfsfall mit dem Befehl RSTOBJ auf das System zurückgespeichert werden.

Anmerkung

Wie auch bei anderen Anwendungsbereichen, so bieten die Sicherungswerkzeuge des System i unter anderem auch die Option, den WAS im laufenden Betrieb (Save while active – kurz: SWA) zu sichern. Allerdings muss bei einem solche SWA-Sicherungsverfahren sichergestellt werden, dass die einzelnen Anwendungen für dieses Sicherungsverfahren geeignet sind. Offene Zugriffspfade, Datenintegrität und Objektsperren sind häufig ein KO-Kriterium für das Sichern im aktiven Zustand. Deshalb gilt es im Zweifelsfall, die gesamte WAS-Umgebung vor der Datensicherung zu beenden, um einen exklusiven Zugriff auf die Inhalte des WAS zu gewährleisten.

5.2.11
Seite 10

Portal-Server

In den vorhergehenden Kapiteln habe ich Ihnen die Installation und die Konfiguration der Anwendungsserver näher gebracht. Solche Anwendungsserver können, zum Beispiel in Form des IBM WebSphere Application Server und dessen verschiedenen Ausprägungen als Basis für die Implementierung von „modernen" J2EE Anwendungen genutzt werden.

In den letzten Jahren hat sich der Trend im Bereich der Anwendungen und deren Präsentation für die Anwender verändert. Das Internet hat es uns vorgemacht: Es bietet den Internetusern optimierte und strukturierte Informationen, die nach Möglichkeit auf einer oder wenigen Webseiten Platz finden. Diese optimierten Webseiten stellen einen benutzerfreundlichen Einstiegspunkt für den Zugriff auf bestimmte Informationen dar. Seien es Webseiten wie die von MSN, von Banken, ebay und nicht zuletzt auch die der IBM. All diesen Webseiten ist eines gemein: Sie versuchen, eine Schlüsselstelle für die Anforderungen und Informationen der Benutzer zu bilden.

Dabei werden die für den Benutzer wichtigen Informationen zunächst zentral auf einer Seite zusammengefasst bzw. sind leicht von dieser aus erreichbar. Nehmen wir als einfaches Beispiel die Webseite der IBM (www.iseries.ibm.com), welche diese für iSeries-Kunden und Interessenten zur Verfügung stellt:

Beispiel für ein Portal

Wie Sie anhand dieser Webseite erkennen können, sind die einzelnen Teilbereiche in kleinen Fensterabschnitten innerhalb der Webseite eingebettet. Diese einzelnen Fensterbereiche (auch Portlets genannt), enthalten jeweils für sich abgekapselt von den übrigen Inhalten der Webseite spezielle Informationen. So bietet IBM zum Beispiel einen Navigationsbereich, über den wir in die verschiedenen Teilbereiche verzweigen können, zusätzliche Bereiche, in denen Business Partner und auch Kunden weitere Informationen bieten und vieles mehr.

Wir wollen jetzt keine Analyse der IBM-Webseite durchführen, sondern deren Aufbau als ein Beispiel für die so genannten Portale verwenden.

Ein weiteres Beispiel für ein Portal ist die folgende Webseite, die sich „zufällig“ auf die WebSphere Produkte der IBM bezieht:

WebSphere Webseite als Portal

Auch hier sehen Sie sehr schön die Aufteilung der Webseite in einzelne Fensterabschnitte oder Portlets.

Portale stellen also, wie der Name bereits aussagt, eine Art „Tür“ als zentralen Zugang auf die für den Benutzer wichtigen Informationen dar.

Ein Portal selbst besteht dabei aus einer beliebigen Anzahl von Fensterbereichen, von denen jeder für sich abgekapselt von den übrigen Bereichen, besondere Informationen enthält. Diese Informationen können in ein Portal mit Hilfe so genannter Portlets (oder Fensterabschnitte) eingebunden werden.

Natürlich lassen sich mit Hilfe eines Portals auch nur beschränkt Informationen auf einer einzigen Präsentationsseite darstellen. Es versteht sich von selbst, dass über ein Portal auch noch weitere Informationen zur Verfügung gestellt werden können, die über entsprechende Verweise in Form von Links eingebunden werden können. Aber wir haben es hier mit Portalen – also „Türen“ zu tun, die uns den Zugang zu weiteren Informationen eröffnen.

Berechtigungsprüfungen kontrollieren außerdem, ob ein Benutzer, der sich zum Beispiel für den Zugriff auf bestimmte Informationen zunächst authentifizieren muss, berechtigt ist, auf diese Informationen zuzugreifen. In diesem Zusammenhang spielen dann auch Techniken wie Single Sign On (SSO) etc. eine wichtige Rolle.

Die Zielgruppen bzw. Benutzergruppen können dabei aus den unterschiedlichsten Bereichen kommen. Nehmen wir als Beispiel Kunden, Interessenten, Lieferanten, Mitarbeiter etc. In diesem Zusammenhang spricht man auch von unterschiedlichen Portalen:

- Informations-Portale
  Webseiten, die als Einstiegsseiten für weitere Informationen dienen (z.B. MSN)
- Portale mit interaktiven Anwendungen
  Integration von 5250-Anwendungen, CRM, Messaging, Mailing, etc.
- B2E Mitarbeiter-Portale
  Für die Information von Mitarbeitern. Zum Beispiel „Schwarze Brett-Funktionen", Urlaubsplanung, Informationssysteme, Urlaubslisten, Anwesenheitszeitverwaltung etc.
- B2B Geschäftspartner-Portale
  Lieferantenanbindung etc.
- B2C Kunden-Portale
  Für Kundenanbindung, Webshops, Lieferverfolgung, Auskunftssysteme etc.
- E-Business-Portale
  Marktplätze, Handelsplattformen (z.B. ebay)

Aber wie bereits gesagt, diente das Internet lediglich als eine Art Ideengeber in der Form, dass der zentrale, auf die Benutzergruppe optimierte Zugriff schnell eine Anforderung aufkeimen ließ, die auch für unternehmensinterne Zwecke durchaus sinnvoll erschien.

Heute hat diese Technik ihren Siegeszug sowohl im Internet, als auch in den Intranets der Unternehmen angetreten. Als flexibler und dennoch für den Benutzer optimierter Arbeitsbereich werden heute Portale dazu genutzt Informationen, die aus unterschiedlichsten Anwendungen stammen können, zentral aufbereitet und für den Benutzer optimiert dargestellt zu präsentieren.

Der wesentliche Vorteil dieser Technik liegt darin, dass für die Ausführung der Portale durch den Endbenutzer keine spezielle Client-Software erforderlich ist, die auf dessen Arbeitsplatz installiert werden muss. Die Portalfunktionen werden nämlich innerhalb eines Browser (Microsoft Internet Explorer, Gozilla, Firefox etc.) ausgeführt. Doch nur ein Vorteil wäre sicher zu wenig, um sich mit dieser Form der Server zu beschäftigen.

Optimierte Zugänge, zentrale Informations-Darstellung, SSO-Unterstützungen, integrierte Suchfunktionen, einfach einzurichtende Kommunikation zwischen heterogenen Systemen, die in die Portlets integriert werden und Web-Services sind nur einige weitere „Schlagworte", die in diesem Zusammenhang zu nennen sind.

Per Definition können wir also zusammenfassen:
Bei einem Portal handelt es sich um Web-basierte Anwendungen, in die verschiedene Inhalte, Dienste und Funktionen optimiert für die jeweiligen Benutzer oder Benutzergruppen integriert sind. Die Basis dieser Technik bzw. Darstellung wird als Portal bezeichnet. Die einzelnen Informationsbereiche oder Fenster werden „Portlets“ genannt.

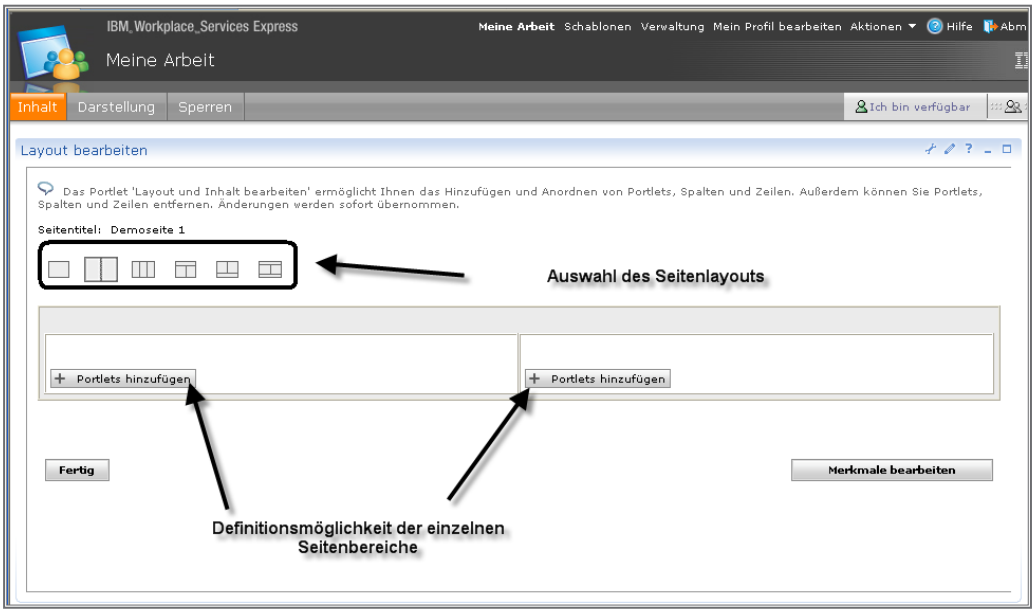
Interessant wird diese Portal-Technologie natürlich auch für iSeries-Benutzer. Haben Sie nicht auch manchmal den Wunsch, Ihren Benutzern zentral einen Einstieg zu allen Informationen zu bieten – einschließlich der Möglichkeit der Integration von 5250-Anwendungen?

Solche Wünsche können zum Beispiel mit Portalen realisiert werden. Dabei kann ein Portal zum Beispiel die folgenden Informationen bzw. Portlets enthalten:

- Eingangspostfach der Mails
- Kalender
- 5250-Emulation (iSeries Access)
- CRM-Portlets
- MS Office Integration
- etc.

Der Aufbau eines Portals kann dabei auf bestimmten Grundmustern oder Vorgaben basieren. IBM bietet zum Beispiel vorgefertigte Portalseiten an, die in einzelne Fensterabschnitte unterteilt sind. Diese Fensterabschnitte – oder Portlets – können dann mit den gewünschten Informationen versehen werden. Sowohl das Layout der gesamten Portalseite, als auch der einzelne Portlet lässt sich dann individuell einstellen und konfigurieren.

Die nachfolgende Abbildung enthält ein Gerüst von einer Portalseite, wie sie zum Beispiel im IBM Workplace enthalten ist:



Portal-Details

Aber wir wollen an dieser Stelle nicht zu tief gehen. Details zu dem Aufbau von Portalen und Portlets finden Sie in den nachfolgenden Unterkapiteln.

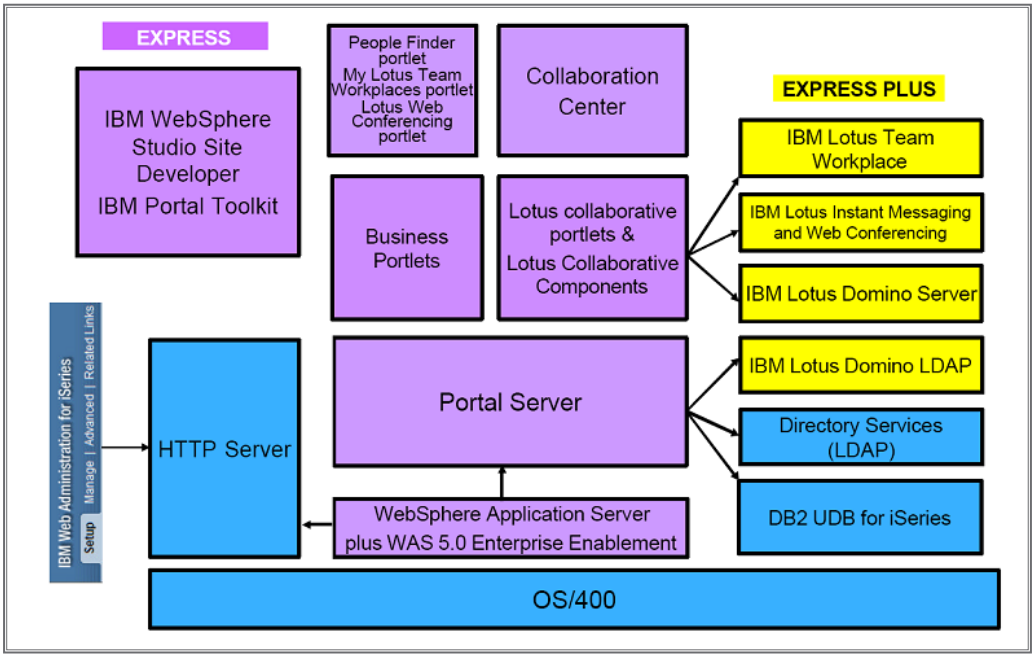
In diesem Unterkapitel werden wir uns jetzt mit den Details zu solchen Portalen in Form der Portal-Server beschäftigen. Dabei liegt der Fokus klar auf den von IBM in diesem Bereich angebotenen Produkten. IBM ist „nur“ ein Unternehmen, das neben vielen anderen Global-Playern der IT auf Portal-Technologien setzt und diese derzeit sehr stark weiter entwickelt. Wir folgen also keinem Ruf der IBM, wenn wir auf deren Webseiten und vielen anderen Informationsquellen mehr und mehr mit Portalen, Portal-Servern, Portlets & Co. konfrontiert werden, sondern stellen uns vielmehr einer modernen Technik, die wir in Zukunft wohl nur noch schwer umgehen können.

Neben den Produkten, die IBM in diesem Bereich in Form des „WebSphere Portal Server“ (in verschiedenen Formen) anbietet, finden wir auf dem Markt unter anderem auch noch die folgenden Portal-Server:

- mySAP Enterprise Portal
- BEA WebLogic Portal
- Oracle Application Server Portal
- SUN ONE Portal Server
- Apache Jetspeed
- Und noch viele mehr!

Die Integration eines Portal-Server in eine Systemumgebung bedarf einiger Planungen, wie und wo die einzelnen Komponenten installiert werden müssen. Denn ein Portal-Server benötigt verschiedene weitere Dienste oder

Anwendungen, wie zum Beispiel einen Web-Server, einen Anwendungsserver und noch einige Komponenten mehr, wenn es darum geht, zusätzliche Dienste wie Single-Signon, LDAP etc. zu realisieren. Schauen Sie sich dazu auch die nächste Abbildung an, in der eine Integration eines WebSphere Portal Server auf der iSeries gezeigt wird. Unabhängig von dem Betriebssystem OS/400, das in diesem Beispiel verwendet wird, sind die einzelnen Komponenten auch auf anderen Betriebssystemen erforderlich.



Integration von WebSphere Portal

In der folgenden Abbildung wird der Portal-Server auf Basis eines WebSphere Application Server installiert. Dieser kommuniziert mit einem Web-Server, der als Frontend für die Benutzereingaben verwendet wird.

5.3
Seite 8

WebSphere Portal Server

IBM bietet, wie bereits erwähnt, in ihrem Portfolio zusätzlich zu den bereits genannten Anwendungs- oder Application-Servern auch Portal-Server an: die „WebSphere Portal Server“. Wie der Name bereits zeigt, werden die Portal-Server unter dem Dach der WebSphere Familie geführt – einer der Gründe ist die direkte Abhängigkeit der WebSphere Portal Server zu einem WebSphere Application Server.

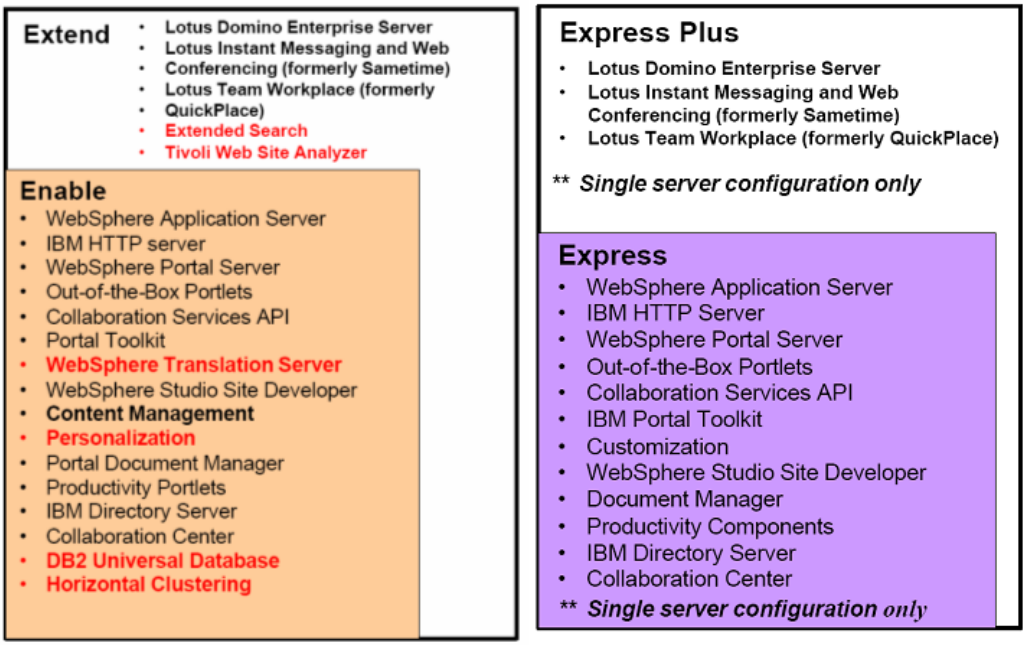
Wie wir es bereits von den WebSphere Application Servern kennen, stehen uns auch im Bereich der Portal-Server verschiedene Versionen oder Pakete zur Verfügung. Diese unterscheiden sich neben der unterstützten Plattform auch in dem jeweiligen Leistungsumfang und durch Besonderheiten im Bereich der Integration von Anwendungen (z.B. 5250-Portltes) oder durch die Spezifika, welche mit einer Installation auf einer iSeries verbunden sind.

So bietet IBM zum Beispiel die folgenden Versionen des WebSphere Portal an (Stand Herbst 2005):

- WebSphere Portal for Multiplatforms
- WebSphere Portal Express for Multiplatforms
  Windows, Linux, z-Series, iSeries….
- WebSphere Portal Enable for zSeries
- WebSphere Commerce Portal
- …

Dabei unterscheiden sich die unterschiedlichen Versionen zum Teil erheblich. Schauen Sie sich dazu auch die folgende Abbildung an, in der ich Ihnen die Unterschiede zwischen den verschiedenen Versionen im Bereich „WebSphere Portal Express“ aufzeige.

5.3.1
Seite 2



WebSphere Portal Express

WebSphere Portal Server ist eine J2EE Anwendung, für die wir als Basis einen WebSphere Application Server benötigen. Dieser WebSphere Application Server dient lediglich als Basis für die Installation des Portal-Server und übernimmt die Services und die Kommunikation mit weiteren Diensten oder auch den Anwendern.

Der WebSphere Portal Server selbst übernimmt in diesem Zusammenspiel die erforderlichen Dienste in Hinblick auf die Integration der Anwendungen und der Darstellung der Informationen in den einzelnen Bereichen eines Portals.

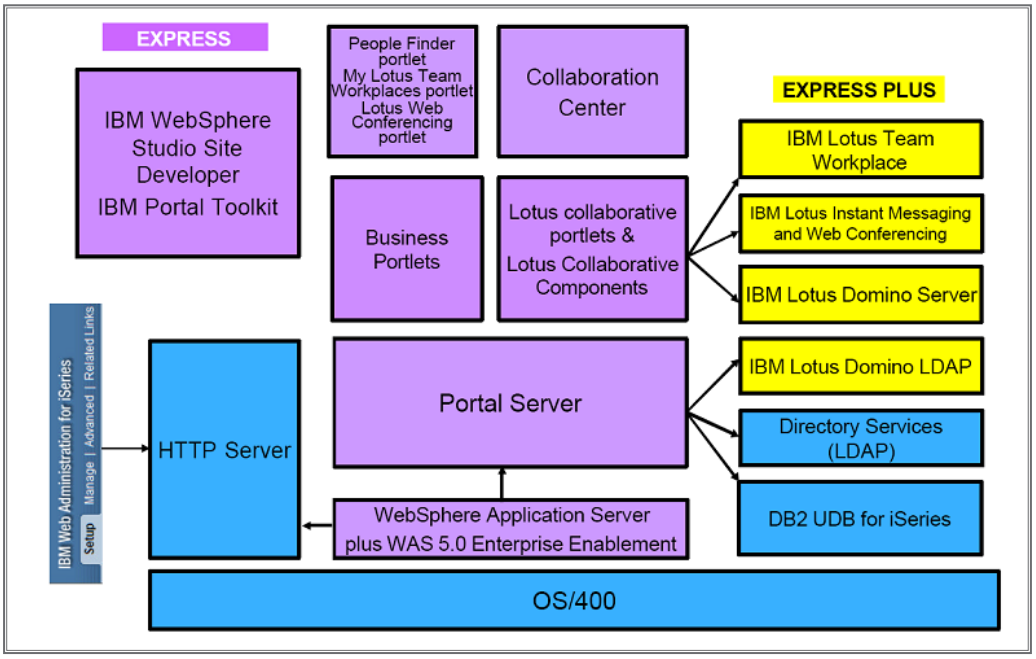
IBM liefert in diesem Zusammenhang derzeit zum Beispiel die folgenden Komponenten in der Version 5.0 aus:

- WebSphere Portal
- WebSphere Application Server
- IBM Directory Server
- DB2 Enterprise Edition
- WebSphere Studio Site Developer
- WebSphere Portal Content Publishing
- Collaborative Services
- Portal Toolkit
- WebSphere Application Server

Wenn Sie eine so genannte Extended Edition einsetzen, dann erhalten Sie außerdem noch weitere Komponenten, wie zum Beispiel:

- Collaboration Center
- Domino Enterprise Server
- Tivoli Access Manager
- Etc.

Die folgende Abbildung skizziert die einzelnen Komponenten und im Speziellen auch die Integration bzw. die Installation auf Basis des OS/400 der iSeries.



Integration von WebSphere Portal

So verwirrend diese Abbildung auf den ersten Blick auch ist: lassen Sie sich dadurch nicht abschrecken! Wie Sie später noch sehen werden, stellt die Installation und Konfiguration kein so großes Problem dar, wie diese Anzeige vermuten lässt. IBM hat im Bereich der benutzerfreundlichen Installation und Konfiguration in der letzten Zeit enorm gute Arbeit geleistet. Damit sollten auch in den einzelnen Bereichen ungeübte bzw. unerfahrene Administratoren trotzdem in der Lage sein, die einzelnen Komponenten installieren zu können. Dies werden Sie in den nachfolgenden Kapiteln sehen, in denen ich mich mit der Installation und der Konfiguration des WebSphere Portal beschäftigen werde.

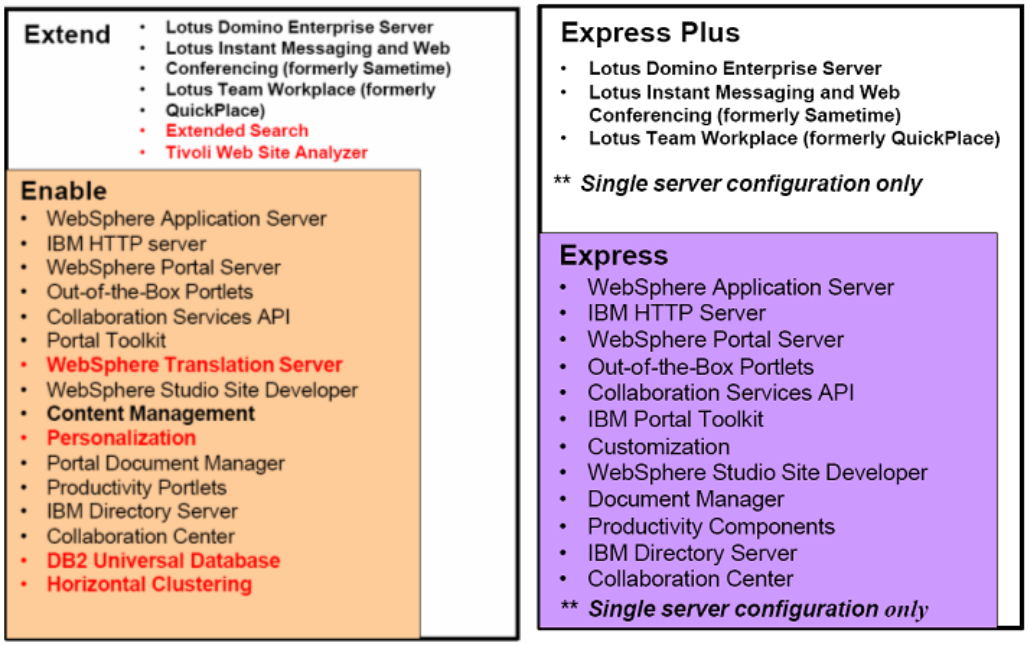
5.3.1 Seite 4

WebSphere Portal Server Express Installation auf iSeries

WebSphere Portal steht auch in verschiedenen Varianten für die Installation auf Basis des Betriebssystems der iSeries – dem OS/400 – zur Verfügung.

Unter anderem können wir hier auch die Express-Version einsetzen, deren Installation ich Ihnen in der Version 5.0.2.3 in diesem Kapitel zeigen möchte. Die Version bezieht sich auf den Release-Stand des Portal-Servers – nicht etwa auf Kapitelnummerierungen!!!

Welche Funktionen bzw. Elemente Bestandteil des IBM WebSphere Portal Express sind, sehen Sie in der folgenden Abbildung:



WebSphere Portal Express

Installationsvoraussetzungen

Bevor wir mit der Installation des IBM WebSphere Portal Express beginnen, müssen wir die Voraussetzungen prüfen, die an die Hardware und die Software gestellt werden.

Ich hatte vor gar nicht allzu langer Zeit ein interessantes Gespräch mit einem Professor der Wirtschaftsinformatik, dem ich von den vielfältigen Möglichkeiten der iSeries berichtete. Dabei habe ich dann auch die Möglichkeiten der WebSphere Integration – im Speziellen auch des WebSphere Portals – erwähnt. Wie Sie gleich sehen werden, sind die Voraussetzungen, die an eine iSeries in Bezug auf die Hardware gestellt werden, damit der WebSphere Portal Server wunschgemäß – also auch performant – ausgeführt wird, enorm hoch. Diese Ansicht hatte ich bis dato. Als ich mich ein wenig verwundert über mindestens 2 GB Hauptspeicher für die Ausführung einer WebSphere Portal Instanz

zeigte (die in der Tat als Mindestvoraussetzung zu sehen ist!), wunderte sich dieser Professor:

„Wir reden hier von einer J2EE-Anwendung, da sind solche Anforderungen doch vollkommen normal. Ihre genannten Voraussetzungen bewegen sich da sicher im unteren Level dessen, was wir heute als Standard für J2EE Anwendungen einsetzen." – war seine profane Aussage.

Was ich damit sagen möchte: Lassen Sie die Mindestanforderungen erst einmal auf sich wirken, bevor Sie dieses Kapitel überspringen und den Entschluss fassen, dass WebSphere Portal als Plattform Ihrer vielleicht zukünftigen Lösungen aufgrund der Anforderungen an das System nicht in Frage kommt. Vielleicht sind wir nur ein wenig verwöhnt von den so genügsamen Anwendungen, die derzeit auf der iSeries realisiert sind.

Ich denke, wir müssen uns über kurz oder lang an solche Mindestanforderungen gewöhnen:

Hardwarevoraussetzungen:

- Mind. Mod. 810 oder höher
- Mind. 2 GB Hauptspeicher (für die WebSphere Portal Instanz!)
- Mind. 2 GB Plattenplatz für die Basisinstallation
  Zzgl. der für Ihre Portalinhalte erforderliche Plattenplatz
- Mind. 750 CPW für die Ausführung der WebSphere Portal Instanz

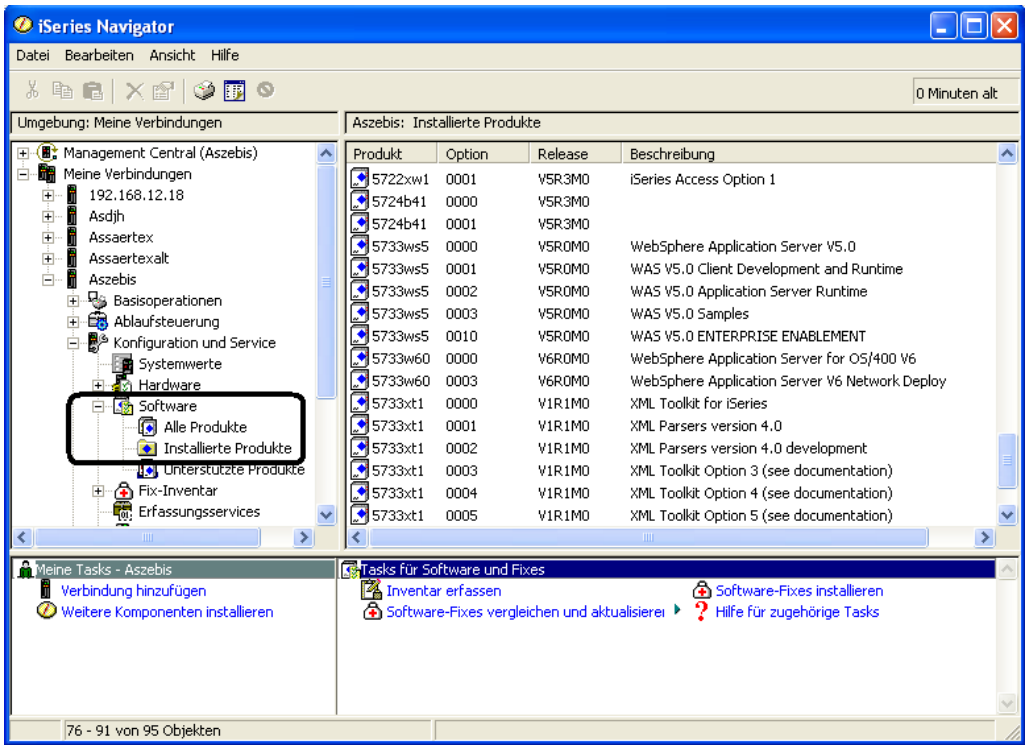
Beachten Sie bitte, dass die Anforderungen an den Hauptspeicher pro Web Sphere Portal Instanz gelten. Sollten Sie also zum Beispiel eine Test-Instanz und eine Produktiv-Instanz installieren, dann benötigen Sie entsprechend 4 GB Hauptspeicher!

Softwarevoraussetzungen:

Die Anforderungen an die Software variieren, wie die an die Hardware auch, je nach zu installierender Version des WebSphere Portals. Deshalb möchte ich auch darauf hinweisen, dass die aktuellen Anforderungsinformationen für jedes neue Release auf den Webseiten der IBM zu finden sind. Deshalb sollen die hier jetzt genannten Voraussetzungen als grobe Richtlinie gelten und stellen keine übertragbare Auflistung für zukünftige Release-Versionen dar!

Lizenzprogramm	Option	Beschreibung
5722-SS1		OS/400 ab Version V5R2
5722-SS1	12	Host Server
5722-SS1	30	QSHELL
5722-SS1	33	PASE
5722-JV1	*BASE	Developer Kit für Java
5722-JV1	5	Java Developer Kit 1.3
5722-JV1	6	Java Developer Kit 1.4
5722-TC1	*BASE	TCPIP
5722-DG1	*BASE	http-Server
5722-WS5	*BASE	WebSphere Application Server V5.0
5722-WS5	1	WebSphere Application Server V5.0 Client development and runtime
5722-WS5	2	WebSphere Application Server V5.0 Application server runtime
5722-WS5	5	WebSphere Application Server V5.0 Network Deployment (ND, optional)
5799-PTL		iSeries Tools for Developer

Die Voraussetzungen auf der iSeries können wir zum Beispiel mit dem iSeries Navigator prüfen:



Softwarevoraussetzungen

Zusätzlich zu den installierten Lizenzprogrammen müssen auch die erforderlichen PTF – gerade auch für den WebSphere Application Server und den Web-Server auf der iSeries installiert worden sein. Prüfen Sie mittels der Informationen auf den Webseiten der IBM, welcher PTF-Stand für die von Ihnen zu installierende Version als Mindestvoraussetzung benötigt wird. Die entsprechenden Informationen finden Sie unter anderem auf den folgenden Webseiten:

http://www-306.ibm.com/software/genservers/portalexpress/support/

oder

http://www-03.ibm.com/servers/eserver/iseries/software/websphere/portal/portalee.html

Anmerkung

In der Regel liefert IBM zwar bereits zusammen mit dem WebSphere Portal Server auch die erforderlichen PTFs aus, ich empfehle aber in jedem Fall zu prüfen, ob der mitgelieferte PTF-Stand auch wirklich dem erforderlichen Stand entspricht!

Achten Sie darauf, dass vor der Installation von WebSphere Portal Express die zusätzlich erforderlichen Varianten wie zum Beispiel der WebSphere Application Server installiert sind. Details zu dessen Installation finden Sie in dem Kapitel 5.2 ff. Dasselbe gilt auch für die Anbindung eines Web-Server.

Die Installation von „WebSphere Portal Express for OS/400 on iSeries“ erfolgt mittels der von IBM ausgelieferten CDs. In dem Lieferumfang sind in der Regel neben dem WebSphere Portal Server auch die übrigen Komponenten, wie der WebSphere Application Server, WebSphere Application Server Enterprise Enablement für OS400 und alle vorhandenen PTFs enthalten.

Zusätzlich muss vor der Installation der für den WebSphere Portal Server genutzte WebSphere Application Server gestartet sein. Sie können das entsprechende Subsystem wie folgt starten:

```
STRSBS SBSD(QEJBAS5/QEJBAS5)
```

Details zu dem Starten des WebSphere Application Server finden Sie auch in dem Kapitel 5.2 ff.

Achten Sie darauf, dass sowohl der Start des WAS als auch die Installation des WebSphere Portal Server mit einem Benutzerprofil ausgeführt werden, das mindestens über die Berechtigungen *ALLOBJ, *JOBCTL und *IOSYSCFG verfügt. Ich empfehle für die Installation die Verwendung des Benutzerprofils QSECOFR oder eines vergleichbaren Benutzerprofils.

Die Installation des WebSphere Portal Server Express wird in dem Verzeichnissystem der iSeries vorgenommen. Dabei wird folgendes Verzeichnis verwendet:

/QIBM/ProdData/PortalServer5/

Dieses Verzeichnis dient als Installationsziel für unseren Portal Server und wird während der Installation automatisch angelegt.

Der Lieferumfang eines WebSphere Portal Server für die iSeries kann sich durchaus sehen lassen. Denn neben der Software für den eigentlichen WebSphere Portal Server Express beinhaltet die Lieferung auch einen WebSphere Application Server, erforderliche PTFs und je nachdem, welches Paket Sie bestellt haben, auch noch einige weitere Anwendungen in Form von zum Beispiel Collaboration Software und die Tools, die Sie für die Anwendungsentwicklung mit Portal-Servern benötigen. Da dieser Lieferumfang in der Vergangenheit wohl in dem einen oder anderen Fall zu Unsicherheiten und Problemen während des Installationsprozesses geführt hat, ist der Teil der CD-Lieferung, mit welchem wir die Installation starten, mit dem Aufkleber „Start here“ versehen.

Für die Installation benötigen wir einen PC mit einem CD- oder DVD-Laufwerk und einer Netzwerkverbindung zu der iSeries, auf welcher der WebSphere Portal Server installiert werden soll. Auf dem PC sollte für die spätere Konfiguration auch ein Standard Web-Browser vorhanden sein. Details zu den Anforderungen an den PC finden Sie auch auf den Webseiten der IBM.

Grundsätzlich können wir zwischen zwei unterschiedlichen Installationsverfahren wählen:

- Die grafische Installation
  Hier wird die Installation mit Hilfe eines Browser-basierten Installationsassistenten durchgeführt (empfohlene Methode!)
- Die Installation mit Hilfe Befehlszeilen-basierter Befehle
  Diese Installation kann mit einer 5250-Sitzung bzw. einer QSHELL-Sitzung durchgeführt werden

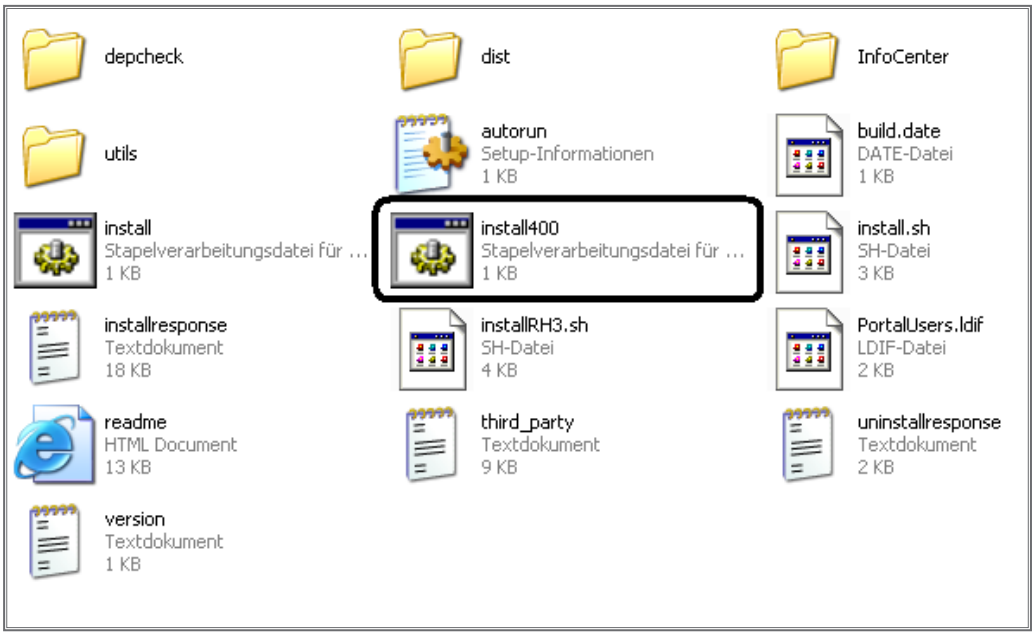
Ich empfehle die Verwendung des graphischen Installationsassistenten, der weitestgehend automatisch durch die Installation führt. Außerdem bietet dieser Installationsassistent zusätzliche Hilfe-Hinweise, mit denen auch Newcomer in der Lage sein sollten, die Installation erfolgreich durchzuführen.

Der graphische Installationsassistent wird zusammen mit der Installations-CD des WebSphere Portals ausgeliefert.

Natürlich bedarf es vor der Installation eines WebSphere Portals auch einer entsprechenden Analyse, wie das Portal-Umfeld aussehen soll, wie und wo die einzelnen, für WebSphere Portal erforderlichen Komponenten installiert

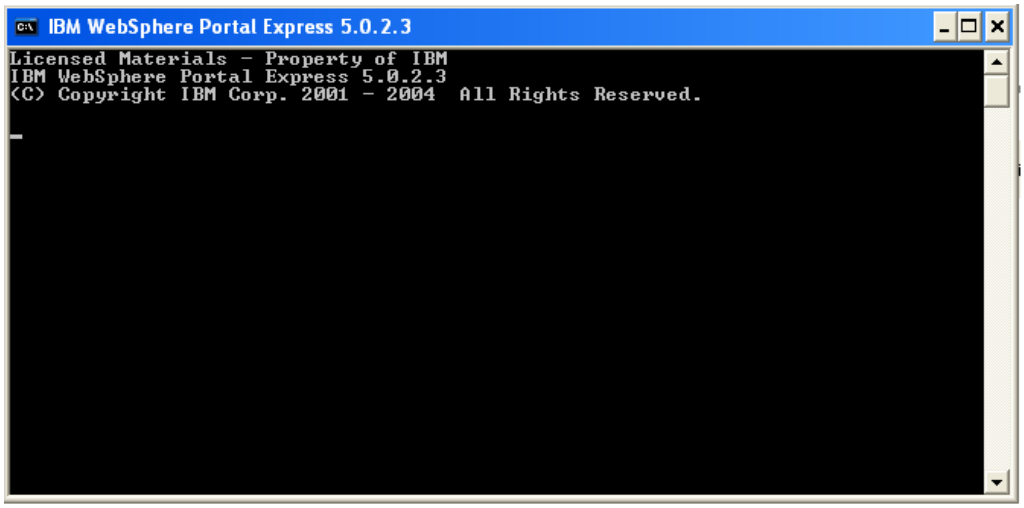
sind (diese müssen nicht zwangsläufig alle auf einer iSeries installiert sein!) und noch vieles mehr. Details dazu finden Sie in einem weiteren Kapitel, in dem wir uns mit der Planung einer WebSphere Portal-Umgebung und deren möglichen Szenarien einer Realisierung beschäftigen.

Nach dem Einlegen der 1. Installations-CD des WebSphere Portals startet der Installationsassistent, der uns durch die Installation führt, in der Regel automatisch. Sollte dieser wider Erwarten nicht automatisch anlaufen, dann kann er mit einem Klick auf den entsprechenden Eintrag „install400“ gestartet werden:



Inhalt der Installations-CD WebSphere Portal for iSeries

Nach dem Start wird ein zusätzliches DOS-Fenster geöffnet. Dieses dient nur der Hintergrundverarbeitung und als Start-Fenster für den Installationsassistenten und braucht nicht weiter beachtet zu werden.



Startvorbereitung

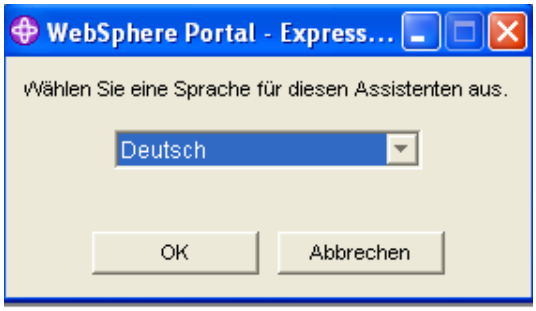
Der Installationsprozess versucht zunächst, eine Verbindung zur iSeries herzustellen, auf welcher der WebSphere Portal Server installiert werden soll. Dazu erscheint ein Login-Fenster, indem wir neben dem Hostnamen der iSeries auch ein gültiges iSeries Benutzerprofil nebst Kennwort angeben müssen.

Login an der iSeries

Anstelle des iSeries Hostnamens in dem Feld „System“ können wir in dieser Anmeldeanzeige auch die IP-Adresse der iSeries angeben. Verwenden Sie als USER-ID ein Benutzerprofil, das über ausreichende Berechtigungen verfügt. Idealerweise verwenden wir für die Installation das Benutzerprofil QSECOFR oder ein vergleichbares Benutzerprofil. Nach der Eingabe des Kennwortes für das OS/400-Benutzerprofil und einem Klick auf die Schaltfläche „OK“ wird jetzt versucht, eine Verbindung zur iSeries herzustellen.

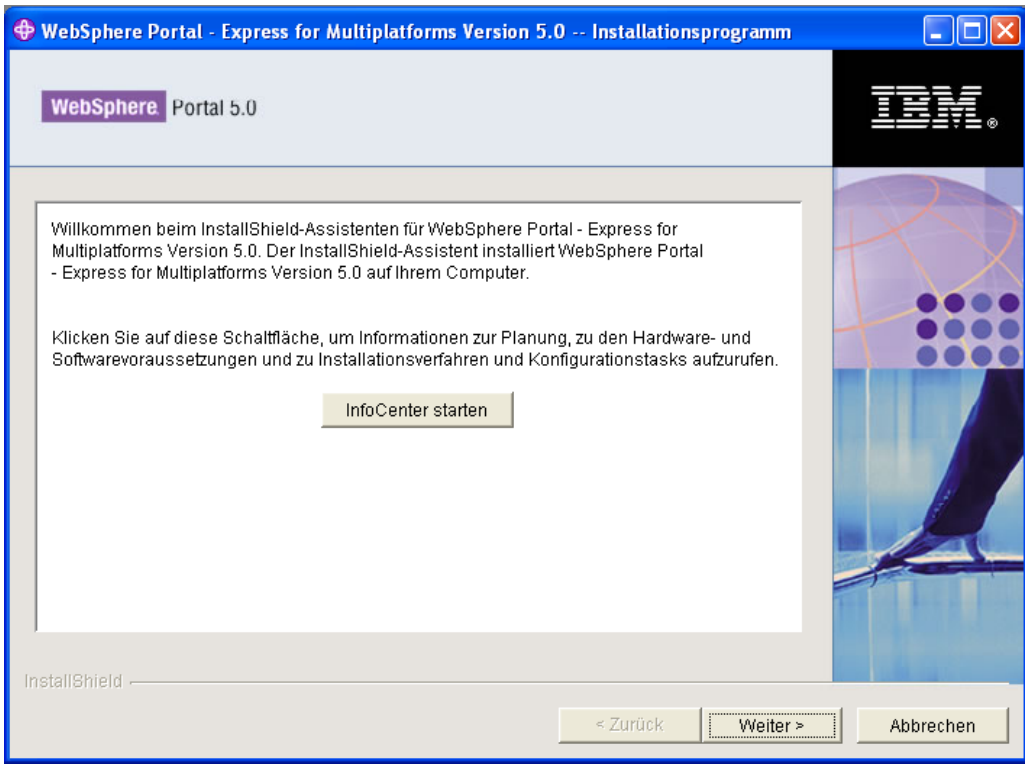
Schlägt diese Verbindungsherstellung fehl, dann erscheint eine entsprechende Fehlermeldung. In einem solchen Fall müssen Sie erst die Verbindungsangaben korrigieren, bevor mit der Installation fortgefahren werden kann.

Der Installationsassistent steht uns in verschiedenen Sprachen zur Verfügung. Wir können die Standardeinstellung „Englisch“ auch mit der deutschen Version ersetzen:



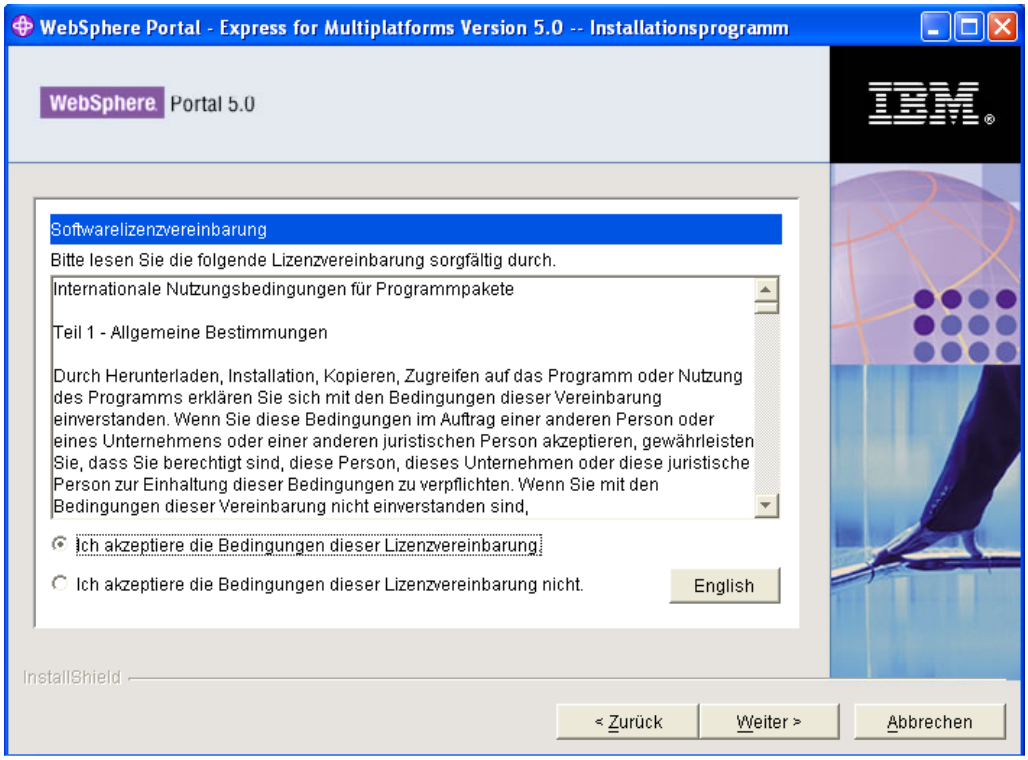
Auswahl der Sprache für den Installationsassistenten

Nach der Auswahl der Sprache für den Installationsassistenten (die nichts mit der zu installierenden Sprachversion des WebSphere Portal Server zu tun hat!), erhalten wir eine weitere Informationsanzeige. Hier haben wir die Möglichkeit des Zugriffs auf weitere Informationen, indem wir auf die Schaltfläche „InfoCenter starten“ klicken, wie in der nächsten Abbildung gezeigt. Diese optionale Auswahl können wir auch überspringen und den Installationsprozess jetzt mit einem Klick auf die Schaltfläche „Weiter“ starten.



Und noch weitere Informationen

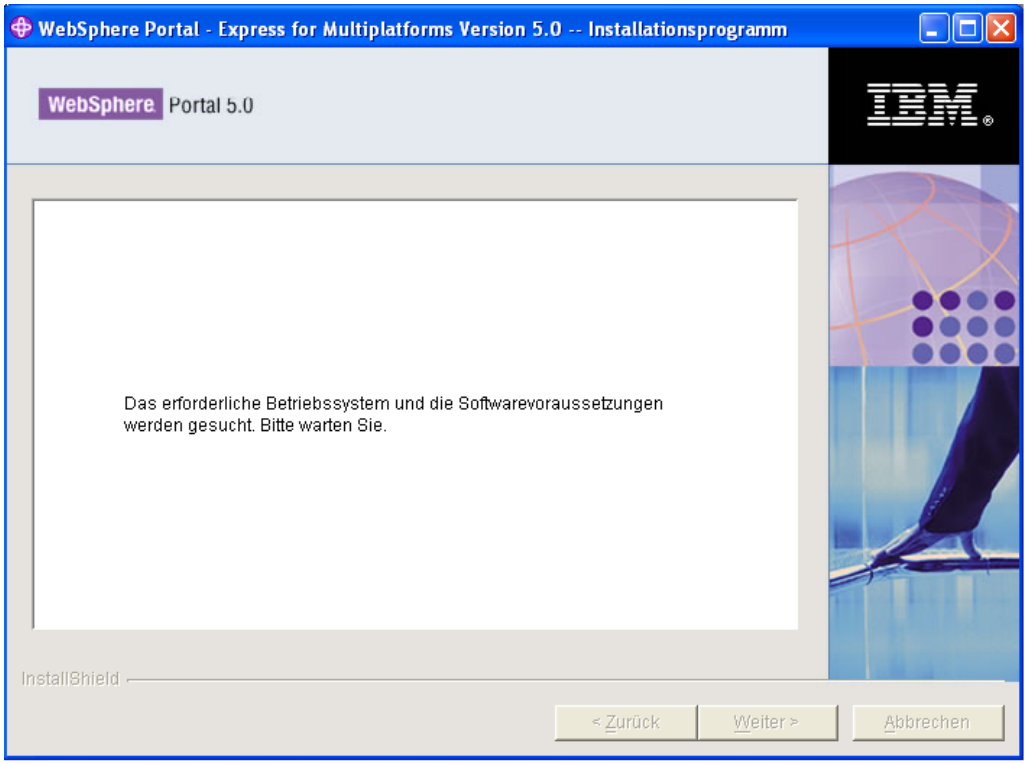
Wie sicher von anderen Anwendungen bekannt, müssen die Lizenzbestimmungen erst akzeptiert werden, bevor die Installation fortgesetzt werden kann. Wählen Sie deshalb die „Akzeptieren“-Option und klicken Sie anschließend wieder auf die Schaltfläche „Weiter“.



Lizenzbestimmungen

Wir haben uns zwar im Vorfeld bereits die Systemvoraussetzungen angeschaut – zusätzlich werden diese in bestimmten Bereichen jetzt vom Installationsprozess nochmals geprüft. Insbesondere geht es bei dieser Prüfung unter anderem um den Speicherplatz und die vorhandenen bzw. benötigten Lizenzprogramme auf der iSeries.

5.3.2
Seite 10



Prüfen der Systemvoraussetzungen

Nachdem die Systemprüfung abgeschlossen ist, erscheint entweder eine Auflistung aller fehlenden Komponenten, die dann vor dem Fortsetzen der Installation zunächst installiert werden müssen, oder – im Idealfall – eine Meldung, dass die Prüfung erfolgreich abgeschlossen wurde.

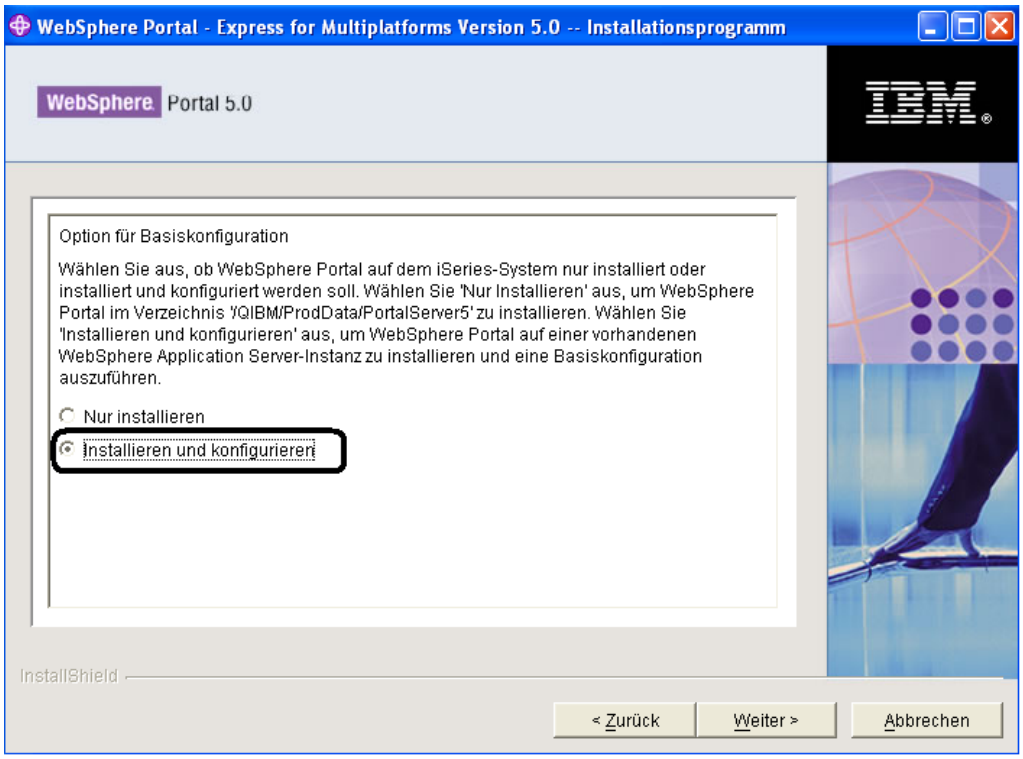
Der Installationsassistent kann jetzt auf zwei unterschiedliche Arten eingesetzt werden:

- Zur Installation der Software des WebSphere Portal Server Express
  Mit dieser Option wird lediglich der Installations-Code des WebSphere Portal Server Express in dem Verzeichnis /QIBM/ProdData/PortalServer5“ durchgeführt. Eine Konfiguration, welche für eine Nutzung des Portal Server erforderlich ist, erfolgt damit nicht!
- Zur Installation und Konfiguration eines WebSphere Portal Server Express
  Mit dieser (empfohlenen) Option wird nicht nur der Installationscode für den Portal Server auf der iSeries angelegt, sondern anschließend direkt die erforderlichen Konfigurationsschritte zum Erstellen eines WebSphere Portals (oder einer entsprechenden Instanz) gestartet.

Ich empfehle an dieser Stelle die Option „Installieren und Konfigurieren“. Klicken Sie nach deren Auswahl auf die Schaltfläche „Weiter“.

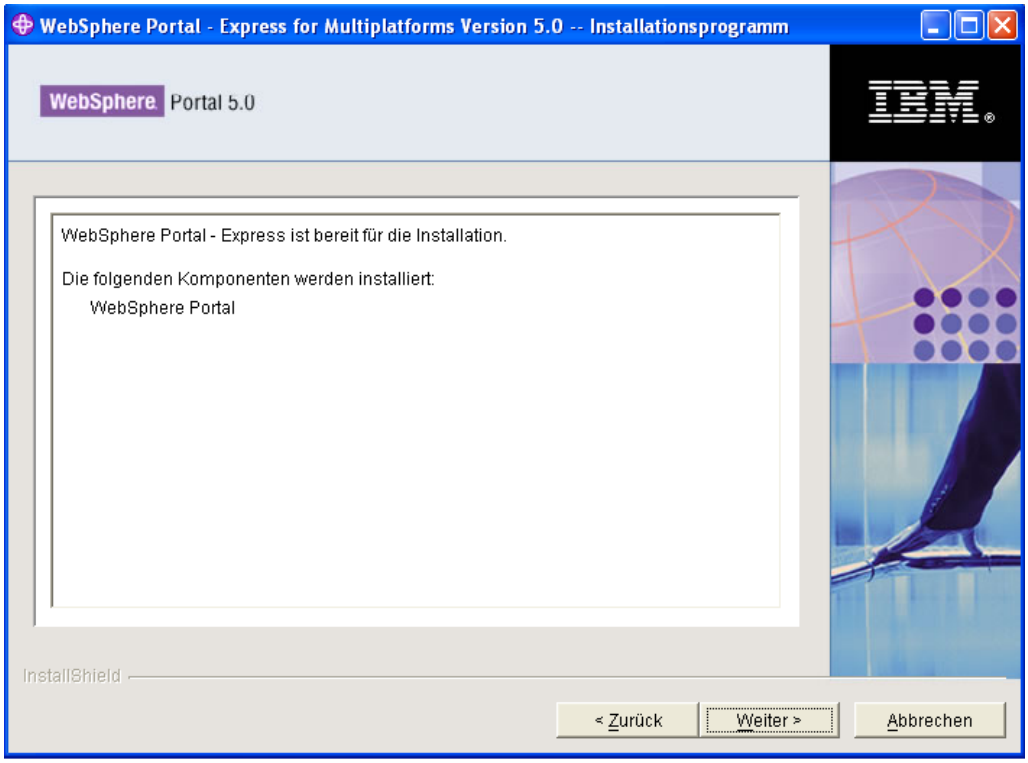
Anmerkung

Diese Empfehlung ist keine Bedingung. Sollten Sie zum Beispiel nur die Software installieren wollen, dann kann natürlich jederzeit später die Konfiguration eines WebSphere Portal Server erfolgen.



Installieren und Konfigurieren

Damit sind alle für die Installation erforderlichen Angaben getätigt worden. Bisher ist noch kein wirklicher Installationsschritt auf der iSeries ausgeführt worden. Die eigentliche Installation starten wir, indem wir in der nächsten Abbildung auf die Schaltfläche „Weiter“ klicken.



Installationsbestätigung

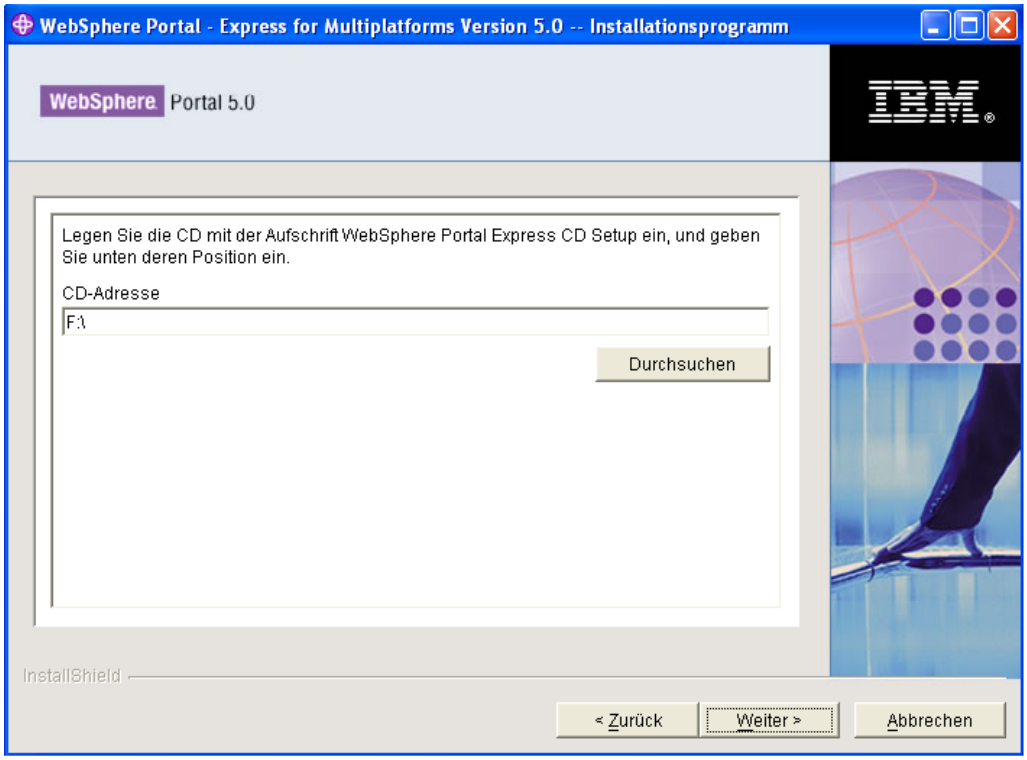
Sollte sich die CD noch nicht in dem Laufwerk befinden, dann legen Sie jetzt die CD mit der Aufschrift „WebSphere Portal Server Express for OS/400 on iSeries – Setup“ in das CD/DVD-Laufwerk ein.

Anmerkung

Es kann vorkommen, dass diese Meldung auch dann erscheint, wenn sich die CD bereits in dem Laufwerk befindet. Lassen Sie sich dadurch nicht irritieren.

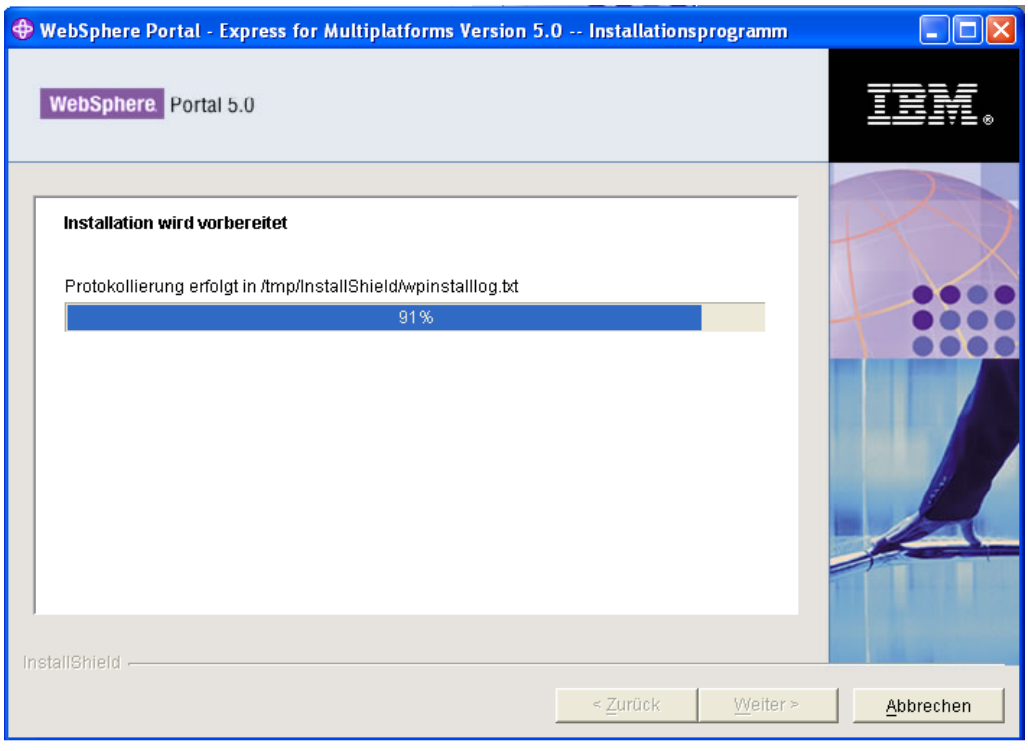
Wenn Sie sicher sind, dass sich die korrekte CD in dem Laufwerk befindet, klicken Sie auf die Schaltfläche „Weiter“.

Alternativ zur Installation über das CD-Laufwerk können Sie den Installationscode natürlich auch vorher auf ein Plattenlaufwerk kopieren. In dieser Anzeige müssen Sie auf jeden Fall das korrekte Laufwerk (und eventuell den Pfad) angeben, von dem aus die Installation durchgeführt werden soll.



CD einlegen

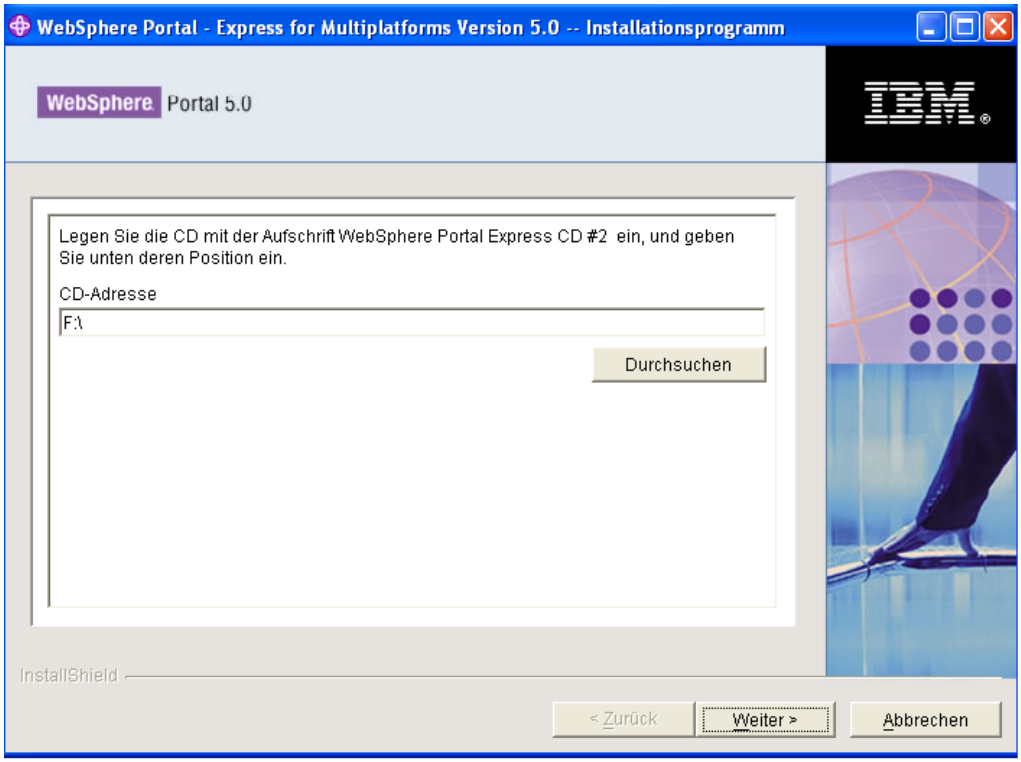
Jetzt wird die Installation des WebSphere Portal Server durchgeführt. Der Fortschritt des Kopierens des Installationscodes auf die iSeries wird dabei mit einer Statusanzeige dargestellt.



Installationsvorbereitung

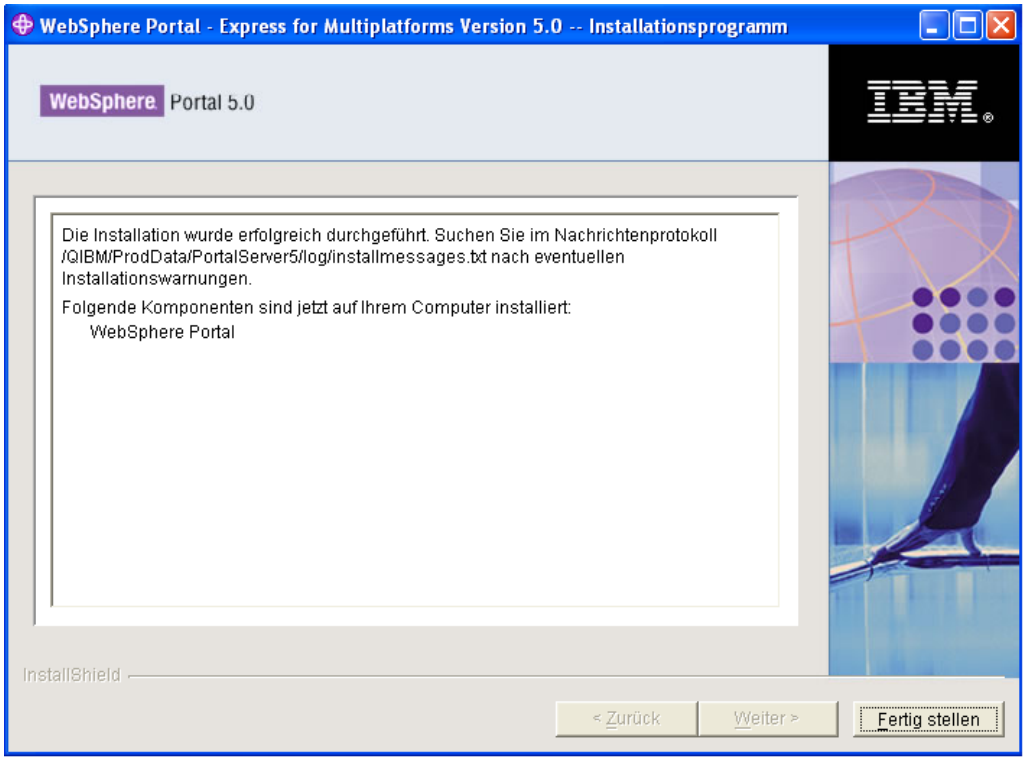
5.3.2
Seite 14

In der Regel passt der Installationscode des WebSphere Portal nicht auf eine CD. Deshalb kann unter Umständen eine Nachricht ausgegeben werden, wenn die Folge-CD eingelegt werden muss. Eine solche Meldung sehen Sie in der nächsten Abbildung:



Nächste CD einlegen

Nach dem Einlegen der korrekten Folge-CD klicken wir auf die Schaltfläche „Weiter“ und setzen damit den Installationsprozess fort.



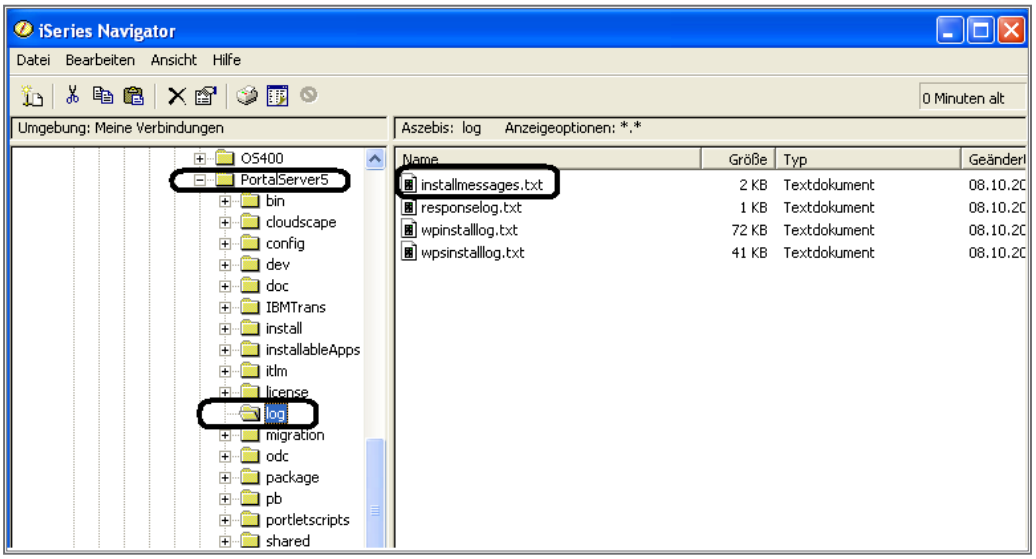
Abschluss der Installation

Nach dem vollständigen Übertragen und Installieren des Installationscodes auf die iSeries erfolgt eine entsprechende Nachricht. Hier klicken wir jetzt auf die Schaltfläche „Fertig stellen“, um den Installationsprozess zu beenden.

Ich empfehle in jedem Fall die erfolgreiche Installation nochmals zu kontrollieren. Während des Installationsprozesses wurde eine Log-Datei auf der iSeries angelegt, in der wir den Status der Installation und eventuell aufgetretene Probleme finden. Diese Protokoll-Datei befindet sich in dem Verzeichnis „QIBM/ProdData/PortalServer/log“ und trägt den Namen „installmessages.txt“.

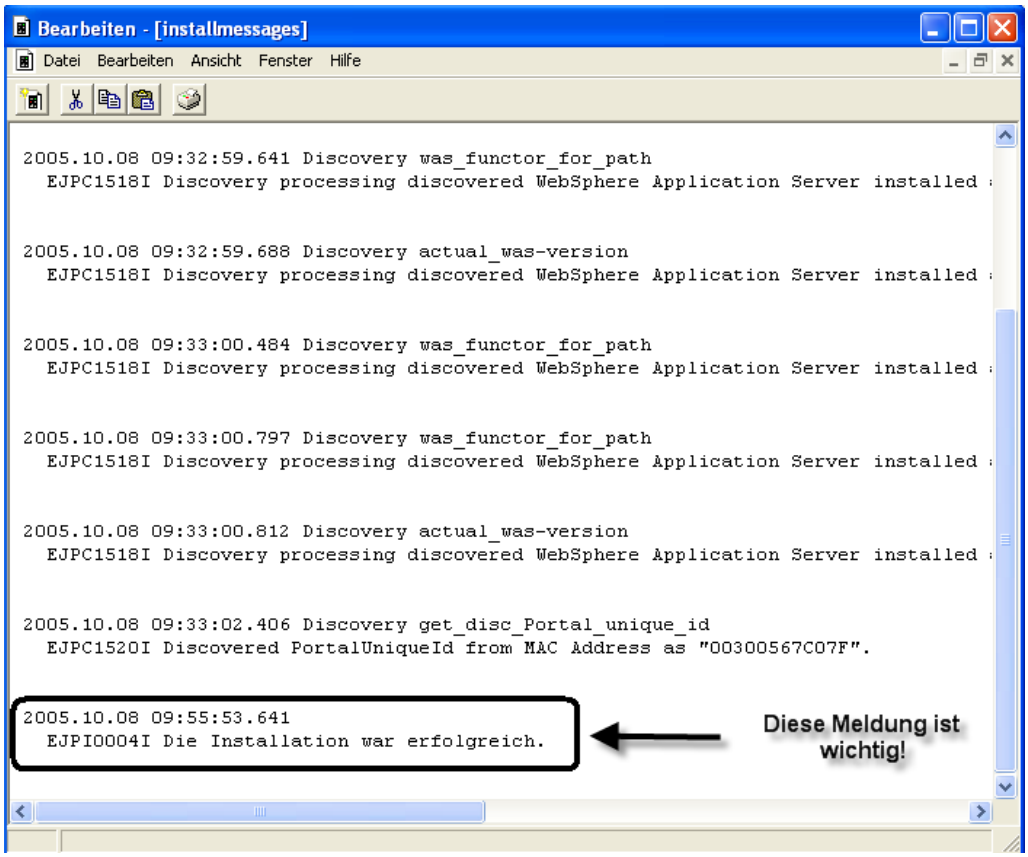
In der folgenden Abbildung sehen Sie diese Protokolldatei, dargestellt mit dem iSeries Navigator.

5.3.2
Seite 16



Speicherlokation des Installationsprotokolls

Das Installationsprotokoll können wir öffnen, indem wir es mit dem Cursor markieren und mit der rechten Maustaste auswählen. Jetzt können wir die Option „Bearbeiten“ auswählen, womit die .txt-Datei in einem Editorfenster geöffnet wird. Ein Beispiel finden Sie in der nachfolgenden Abbildung.



Inhalt des Installationsprotokolls

In dem Installationsprotokoll sind die wesentlichen Schritte der Installation protokolliert. Wichtig ist hier der letzte Eintrag, der auf eine erfolgreiche Installation hinweisen muss.

Damit ist die eigentliche Installation des WebSphere Portal Server Express auf der iSeries beendet.

Allerdings muss der Server erst noch konfiguriert werden, bevor er später tatsächlich produktiv eingesetzt werden kann.

Je nachdem, welche Option in dem Installationsassistenten ausgewählt wurde (nur „Installation“ oder „Installation und Konfiguration“) wird der Konfigurationsprozess jetzt automatisch gestartet, oder ein solcher Prozess muss anschließend manuell ausgelöst werden.

5.3.2
Seite 18

Allgemeine Serverdienste

Neben den vorherigen Kapiteln, in denen wir die unterschiedlichen Server und ihre Funktionen dezidiert behandelt haben, möchte ich Ihnen mit diesem Kapitel einen Einblick in die auf dem System i vorhandenen „sonstigen" Serverdienste und Funktionen geben.

Dabei werden wir uns mit den unterschiedlichen TCP/IP-Serverdiensten beschäftigen, welche in der Regel Bestandteil der Basisauslieferung der Maschine sind.

10. Ergänzung 3/2007

5.4
Seite 2

Verzeichnisserver

Jeder kennt die Problematik verteilter Informationen und Daten innerhalb eines Netzwerks, und dass zum Teil erheblicher Aufwand damit verbunden ist, diese Informationen wieder aufzufinden. Die Fülle an Informationen, welche verteilt auf den unterschiedlichsten Systemen abgelegt sind, nimmt beinahe täglich zu. Da ist es schon fast eine Herausforderung, die Zugriffe auf die gewünschten Daten in einer akzeptablen Zeit realisieren zu können. Da herkömmliche Techniken für solche Zugriffe nicht optimal genug arbeiten, wurde vor einiger Zeit ein neuer Typ des Zugriffs auf Informationen entwickelt, den wir heute als LDAP oder Verzeichnisdienst kennen.

LDAP-Funktionen sind dabei auf den Lesezugriff von Informationen optimiert und weniger dazu geeignet, Daten zu verarbeiten, welche ständigen Änderungen unterlegen sind. Die Lesezugriffe werden dabei mit einem Minimum an Ressourcen ausgeführt, was zur Folge hat, dass diese Zugriffe – vergleicht man sie mit den Lesezugriffen auf eine herkömmliche Datenbankdatei – ein Mehrfaches an Geschwindigkeit aufweisen. Das ist auch für die Bereiche, welche sich der LDAP-Funktionen bedienen, dringend erforderlich.

Doch schauen wir uns zunächst einmal die Grundlagen ein wenig genauer an:

Man kann die Funktionen von LDAP bzw. einem Verzeichnisdienst mit einer Art Telefonbuch oder mit den „Gelben Seiten“ vergleichen, einem einfachen Such- und Finde-Mechanismus für bestimmte Informationen. Durch die Techniken, die bei den Verzeichnisdiensten angewendet werden, ist ein optimaler und schneller Zugriff auf die gewünschten Informationen möglich.

Wenn man über den Einsatz von Verzeichnisdiensten nachdenkt, dann ist damit häufig die Suche nach Informationen zu Personen verbunden. Gerade in größeren Unternehmen haben die Adressbücher einen derartigen Umfang angenommen, dass eine schnelle Suche nach Mitarbeitern oder anderen Kontaktadressen gar nicht mehr so einfach ist. Dabei können uns die Verzeichnisdienste entlasten und variable Möglichkeiten der Suche bereitstellen. Der Einsatz der Verzeichnisdienste ist natürlich nicht auf den Einsatz in Verbindung mit Personensuchen oder Adressen beschränkt, sondern lässt sich nach Belieben verwenden. Wenn ein Anwender beispielsweise nach Informationen sucht, welcher Drucker über bestimmte Eigenschaften verfügt und diese Eigenschaften in einem Verzeichnis hinterlegt sind, dann kann der Verzeichnisdienst den Mitarbeiter darüber informieren, auf welchem Drucker dieser die Ausdrucke optimal erzeugen lassen kann.

Man könnte – obwohl dieser Vergleich nicht wirklich zutreffend ist – ein Verzeichnis auch mit einer Datenbank vergleichen. Eines der Merkmale eines Verzeichnisses ist die Tatsache, dass es in der Hauptsache Lesezugriffe sind, welche ausgeführt werden. Dabei können zeitgleich Tausende von Zugriffen

auf das Verzeichnis ausgeführt werden. Während eine große Anzahl von Lesezugriffen auf ein solches Verzeichnis kein Problem darstellt (denn dafür ist ein Verzeichnis schließlich gedacht), sind die Möglichkeiten des Veränderns der Inhalte eines Verzeichnisses auf ein Minimum beschränkt. Ein Verzeichnis ist nicht wirklich dazu geeignet, größere Datenmanipulationen an dem Inhalt der Verzeichnisdaten durchzuführen.

Doch wozu das alles?

In den letzten Jahren haben sich verstärkt Produkte oder Standards entwickelt, welche auf die Besonderheiten der Daten, welche mit LDAP bereitgestellt werden können, aufbauen. Dabei wurde mit dem Anwachsen der Datenmengen auch eine Grenze überschritten, mit der ein herkömmlicher Zugriff auf die gewünschten Informationen nicht mehr in einer angemessenen Zeit realisiert werden konnte.

Gerade auch im Zusammenhang mit dem Einsatz von WebSphere und Tivoli Produkten ist die Verwendung von LDAP fast unumgänglich. Aber nicht nur dort in den „komplexen" Anwendungen ist LDAP Voraussetzung für die Realisierung von Anforderungen, sondern es findet auch mehr und mehr Einsatz zusammen mit Anwendungen, welche in vielen Unternehmen zum Einsatz kommen.

Nachfolgend einige Beispiele für den Einsatz von LDAP auch in Verbindung mit dem System i:

- Single Signon (SSO) Funktionen
  - Inkl. der Enterprise Identity Mapping (EIM) Funktionen
- Allgemeine Benutzerverwaltung
  - Auch plattformübergreifend
- Elektronisches Telefon- und Adressbuch
- Ressourcenverwaltung
  - Hardwareressourcen, Raumverwaltung, Fuhrparkverwaltung etc.
- HelpDesk-Anwendungen
- etc.

Bevor wir uns mit den Details der Integration der Serverdienste auf der i5 beschäftigen, wollen wir uns zunächst einen Überblick darüber verschaffen, was sich eigentlich hinter einem Verzeichnisdienst und LDAP verbirgt.

LDAP steht als Abkürzung für „Lightweight Directory Access Protocol" (oder auf Deutsch: einfaches Protokoll für den Zugriff auf Verzeichnisse) und stellt – wie der Begriff schon zum Ausdruck bringt – einen optimierten Zugriff auf Verzeichnisse dar.

LDAP ist ein definierter Standard für die Kommunikation mit verschiedenen Datenbanken in einem Netzwerk. Manchmal kann man es durchaus auch als „Übersicht über das Chaos“ bezeichnen …

Mit LDAP besteht u. a. die Möglichkeit, Adressbücher auch nach anderen Kriterien und Stichwörtern zu durchsuchen, als nach dem Namen des Eintrages und so schnell das zu finden, wonach gesucht wird.

In UNIX-Systemen sind solche Mechanismen bereits seit einiger Zeit bekannt. Das Netzwerk-Informations-System (NIS) katalogisiert die Informationen von Benutzern, Netzwerkinformationen und verfügbaren Diensten. Allerdings ist NIS auf die Nutzung mit UNIX-Systemen beschränkt.

Da der Nutzen dieses Dienstes auch auf anderen Plattformen zum Einsatz kommen sollte, wurden diverse Projekte gestartet, von denen sich eines als „Standard“ durchsetzen konnte: X.500.

X.500 wurde als Dienst 1988 von der CCITT (Comité Consultativ International Télégraphique et Téléphonique) geboren und seitdem stetig weiterentwickelt und steht heute Pate für die abgespeckte Version des X.500 – LDAP. Die ersten LDAP-Funktionen wurden 1993 von der Universität in Michigan erstellt und beinhalten alle wesentlichen Funktionen von X.500, Funktionen, die für die Zusammenführung, das Auffinden und Verwalten von Ressourcen verschiedener Betriebssysteme oder Plattformen benötigt werden.

Betrachtet man X.500, dann kann man durchaus sagen, dass es keine Informationen gibt, die nicht in ihm gespeichert werden können. Es handelt sich also um eine Mega-Objektdatenbank, die in der Lage ist, alle Arten von Informationen aus unterschiedlichsten IT-Welten aufzunehmen und zentral für die weitere Verwendung verfügbar zu halten.

Dabei werden die zu speichernden Informationen in so genannten Objektklassen abgelegt.

Diese Objektklassen enthalten die folgenden Angaben:

- Attributnamen
- Typen
- Wertebereiche

Die am häufigsten verwendeten Objektklassen sind z. B.:

- Alias
- Country
- Locality
- Organization
- Organizational Unit

- Person
- etc.

Die am häufigsten verwendeten Attribute sind z. B.:

- Common Name (CN)
- Organization Name (O)
- Organizational Unit Name (OU)
- Locality Name (L)
- Street Address (SA)
- State of Province Name (S)
- Country (C)

Natürlich stellt sich jetzt die Frage, ob der Einsatz von X.500 nicht sinnvoll erscheinen würde. Denn wenn alle Informationen verarbeitet werden könnten, würde der „Traum" eines jeden IT-lers in Erfüllung gehen.

Leider hat dieses Verfahren auch einen entscheidenden Nachteil: denn die Implementierung ist äußert komplex und erfordert tief greifende Fachkenntnis. Zudem ist die Kommunikation zwischen den Clients und dem Server, auf dem dieser Dienst ausgeführt wird, sehr komplex und belastet das Netzwerk über die Maßen.

Damit für den „allgemeinen" Einsatz die Vorteile überwiegen und die Nutzung der Dienste trotzdem keinen wesentlichen Einschränkungen unterliegt, wurde eine verschlankte Version von X.500 entwickelt – eben LDAP.

LDAP steht in der Welt der verteilten Informationshaltung und Verarbeitung allerdings nicht nur für das schnelle Auffinden von Informationen und Daten sowie den diesbezüglich optimierten Zugriff, sondern ist auch Basis für eine Vielzahl von Funktionen und Anwendungen, die auf den LDAP-Funktionalitäten aufsetzen.

LDAP

Wie bereits beschrieben, handelt es sich bei LDAP um eine verschlankte Version des X.500 Standards, die für die Optimierung des Auffindens und Verarbeitens von Daten in einer verteilten Umgebung genutzt wird. Ursprünglich sollte LDAP lediglich als „Mittler" zwischen Clients und X.500 Servern dienen.

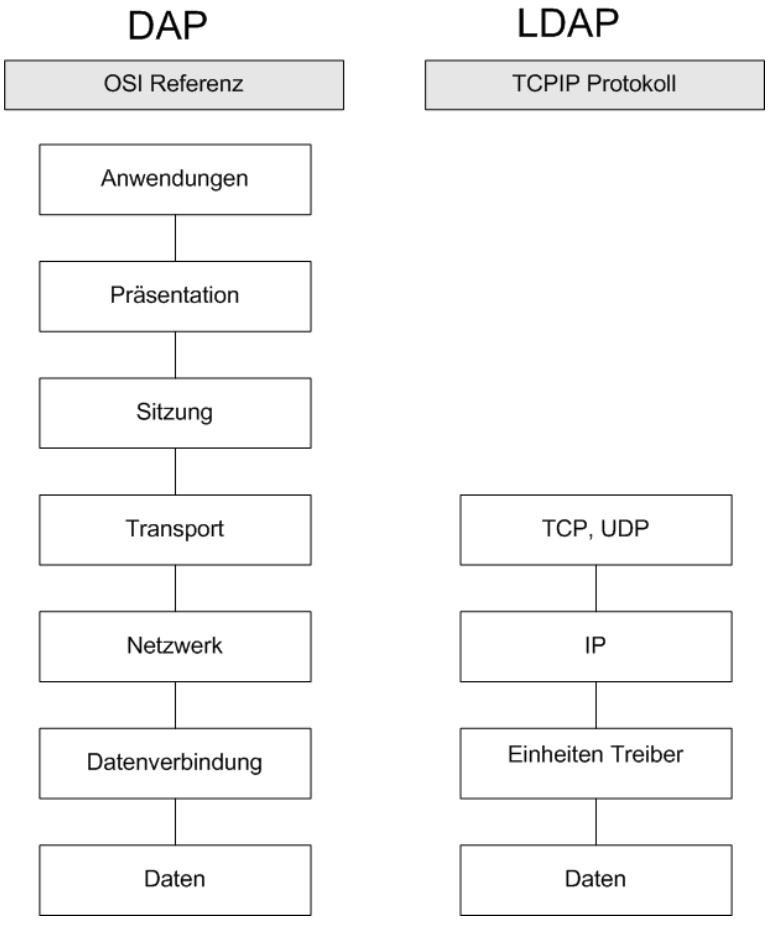
Mit der Zeit wurden auch Anforderungen definiert, die einen Einsatz über die Verwendung der X.500 Server hinaus erforderlich machten. Diese Zugriffe auf Nicht-X.500 Server werden über Gateways realisiert und erlauben die Einbindung der gängigsten Server-Versionen.

Die nachfolgende Abbildung soll die Funktion von LDAP verdeutlichen.

LDAP-Anlaufschema

Bei LDAP handelt es sich um ein Protokoll – genauer, um ein Kommunikationsprotokoll – das Informationen zwischen einem Client und einem X.500 Server (oder einem X.500 ähnlichen Server) austauscht. LDAP selbst definiert allerdings nicht das Verzeichnis (oder Directory) – dies muss an anderer Stelle geschehen.

Die nächsten Abbildungen zeigen schematisch Aufbau und Ablauf sowohl im „großen Bruder" von LDAP – dem DAP – sowie dem LDAP selbst.



DAP- und LDAP-Ablauf

LDAP ist in der Lage, Informationen zwischen unterschiedlichen Systemen auszutauschen und diese Informationen jeweils für weitere Systeme verwendbar zu machen. Damit entfallen aufwändige und zeitintensive Umsetz- und Sucharbeiten. Außerdem besteht die Möglichkeit, Standard-Informationen, die im Rahmen der Kommunikation zwischen unterschiedlichen IT-Systemen ausgetauscht werden müssen, und die zum Teil die Basis für weitere Anwendungen bilden, ohne Umformatieren und Anpassen auf diesen Systemen nutzen zu können. Die Basis für diesen Informationsaustausch bilden dabei unter anderem:

- Benutzer- und Gruppeninformationen
- IP-Protokoll-Informationen
- Netzwerkgruppen
- Anhanginformationen für Dateisysteme
- etc.

Diese Funktionen ermöglichen den Einsatz von LDAP als Verzeichnisdienst. Unter anderem bildet LDAP die Basis für Single Signon Funktionen (kurz: SSO) oder Enterprise Identity Mapping. Mit dieser Funktion, bei der ein Anwender lediglich eine oder wenige Anmeldungen benötigt, um mit den unterschiedlichsten Plattformen und Anwendungen arbeiten zu können, müssen alle Informationen von LDAP verarbeitet werden können, die für Authentifizierung und Authorisierung benötigt werden. Als da wären:

- Benutzerinformationen, erweitert um Authentifizierung und Autorisierungsinformationen
- Verwendung von Namens- und Adressbüchern zum Speichern dieser Informationen
- Ressourcen-Verwaltung, um Informationen zu bearbeiten
- Unterstützung verschiedener Software- und Hardwareversionen

Damit LDAP auf den unterschiedlichsten Plattformen verwendet und auch als Kommunikator zwischen den Plattformen eingesetzt werden kann, müssen dessen Funktionen einem gewissen Standard entsprechen, der von allen unterstützten Plattformen verstanden und verarbeitet wird. Die Informationen, die LDAP für seine Aufgaben verarbeitet, werden in Form von Objekten realisiert. Diese Objekte sind in Aufbau und Verwendung standardisiert und bilden somit die Grundlage für den Einsatz von LDAP.

Wir wollen an dieser Stelle nicht auf die Spezifikationen der LDAP-Objekte eingehen – denn dies würde den Rahmen dieses Buches sprengen. Sie sind zunächst auch nicht von großer Bedeutung für die Einführung und Einrichtung von LDAP.

Da in der Regel in jedem Unternehmen Anforderungen bestehen, deren Realisierung sich mittels LDAP wesentlich vereinfachen lassen, wollen wir uns in den nächsten Abschnitten näher mit dem Thema LDAP beschäftigen.

Die Anforderungen an LDAP lassen sich wie folgt auflisten:

- Optimierung für Lesezugriff
  - Da mittels LDAP schnell auf bestimmte Informationen zugegriffen werden soll, müssen die Lesezugriffe ebenso schnell und performant ablaufen. Es finden keine Veränderungen an den Einträgen statt.
- Organisation der Information in einer Baumstruktur.
  - Durch diese Struktur ist eine gezielte Suche nach Informationen in verschiedenen Ebenen und nach verschiedenen Vorgaben oder Sortierungen möglich. Denken wir hier an ein einfaches Beispiel aus dem Domino-Umfeld: Die Suche nach Einträgen in dem Adressbuch. Sind die Einträge zu Gruppen (zum Beispiel Firmen) zusammengefasst, dann haben wir ein hervorragendes Beispiel für eine Verzeichnisstruktur, wie wir sie mit LDAP verwenden.
- Replikationen
  - Beim Einsatz von verschiedenen Verzeichnis-Servern, auf denen LDAP-Dienste laufen, müssen diese repliziert werden können, um auf jedem der Server die erforderlichen aktuellen Informationen verfügbar zu haben.

Neben diesen Grundanforderungen an LDAP und die Dienste, die LDAP bietet, müssen besondere Funktionen realisiert sein, die für den Einsatz von LDAP optimiert sind.

In LDAP sind die folgenden Funktionen enthalten:

- Abfrageoperationen
  - Suchen
  - Lesen
  - Vergleichen
- Update-Operationen
  - Hinzufügen
  - Löschen
  - Ändern
  - Umbenennen

- Authentifizierung
  - Anmelden
  - Authentifizieren
  - Abmelden

Diese Anforderungen lassen sich mit Hilfe von speziellen LDAP-Servern realisieren. Diese können wahlweise dediziert auf einem eigenen Server betrieben werden oder aber auf einem bereits bestehenden Server mit ausgeführt werden. IBM hat bereits seit einigen Release-Versionen die LDAP-Server-Funktionen im i5/OS in Form des IBM Directory Servers integriert. Die dafür erforderlichen Funktionen sind in den Basisauslieferungen des Betriebssystems mit enthalten. Die Basis der Konfiguration bildet dabei der http-Administrationsbereich und in Teilbereichen auch der iSeries Navigator. Der Einrichtung und Verwaltung des LDAP-Dienstes auf der i5 werden wir uns in einem speziellen Kapitel widmen.

LDAP-Einsatz

Der Einsatz von LDAP ermöglicht – wie bereits beschrieben – den optimierten Zugriff auf Daten, die in einer verteilten Informationsumgebung verwendet werden. Gerade in Umgebungen, in denen über das lokale Netzwerk hinaus auf Informationen zugegriffen werden muss, macht der Einsatz von LDAP besonderen Sinn.

LDAP ermöglicht zunächst einmal einfach und schnell:

- zentrale Verwaltung von Informationen
- normalisierte Datenhaltung
- konsistente Schnittstelle zu den Anwendern
- konsistente Verwendung innerhalb des Netzwerkmanagements
- Verwendung von Security Policies

Damit diese Anforderungen erfüllt werden können, muss LDAP über einige Voraussetzungen verfügen:

- LDAP stellt eine Art Universalcontainer für Verzeichnisdienste dar
  - LDAP arbeitet mit so genannten Schemas. Dabei handelt es sich um ein objektorientiertes Konzept mit Eigenschaften, die wir auch aus anderen Bereichen der Objektorientierung kennen – „ Vererbung“ ist nur ein Beispiel dafür. Das ist auch der Grund, weshalb Ableitungen von Schemas recht einfach zu realisieren sind.
- LDAP ist ein einfaches Protokoll
  - Die abgespeckte Version von X.500 erlaubt eine schnelle und einfache Implementierung und einen performanten Einsatz.

- LDAP-Verwendung in Programmiersprachen
  - Durch die Eigenschaften von LDAP (Vererbung, Standardisierung) kann LDAP in den meisten Programmierumgebungen verwendet und eingebunden werden.
- LDAP ist ein offener Standard
  - Die Entwicklung an diesem Standard wird nicht nur primär von einem Unternehmen bestimmt, wie es sonst häufig in der IT der Fall ist, sondern wird vielseitig und vielschichtig weiterentwickelt. Kontrollmechanismen erlauben einen hohen Standard in den jeweiligen LDAP-Versionen.
- LDAP kann international eingesetzt werden
  - Gerade durch den Einsatz in weltweiten Umgebungen ist die Möglichkeit einer universellen Zeichensatzunterstützung eine Basisvoraussetzung für den Einsatz von LDAP. LDAP verwendet UTF-8. Damit wird die Unterstützung aller Zeichensätze und Sprachen sichergestellt.

Trotz aller Funktionen, die LDAP bietet – es ist für die folgenden Einsatzbereiche nicht ausgelegt und sollte auch nicht für diese Zwecke „missbraucht" werden:

- Keine Verwendung als „Allgemeine Datenbank"
  - LDAP unterstützt keine Transaktionsmechanismen.
  - Aktualisierungen von Einträgen sind nur schwer möglich.
  - LDAP ist keine relationale Datenbank.
  - LDAP unterstützt keine Stored Procedures.
- Das Verzeichnis ist kein Dateisystem
  - Keine Speicherung von „großen" Objekten.
- Keine lokale Speicherstelle für Dateien
  - Verzeichnisse im Zusammenhang mit LDAP sollen den Zugriff auf Informationen unterschiedlichster Herkunft optimieren.
- LDAP ist kein Netzwerkmanagement-Tool

Wenn Sie keine dieser Anforderungen haben, dann macht der Einsatz von LDAP sicher (oder vielleicht) auch in Ihrem Umfeld Sinn.

Nachfolgend schauen wir uns noch einige weitere Begriffe an, die im Zusammenhang mit LDAP immer wieder vorkommen und nicht jedem auf Anhieb bekannt sind. Sollten Sie sich bereits mit der LDAP-Thematik auseinandergesetzt haben, dann können Sie diesen erklärenden Abschnitt überspringen, denn wir werden nur Basis-Definitionen erläutern.

5.4.1
Seite 10

Directory Information Base (DIB)

Die Einträge, die in LDAP genutzt werden, stehen in einem Verzeichnis, dem Directory. Diese Informationen werden dem Endbenutzer in der Regel nicht vollständig angezeigt, sondern es werden nur die Informationen zur Verarbeitung bereitgestellt, die der Anwender auch wirklich benötigt. Die Gesamtmenge aller Informationen, die sich in einem Directory befinden, wird als „Directory Information Base" bezeichnet.

Entries (Verzeichniseinträge)

Das DIB selbst besteht aus einzelnen Einträgen, den so genannten „Entries". Ein Entry stellt dabei die Menge an Informationen zu einem Objekt dar. Wie so oft bei der Objektorientierung, stehen auch in diesem Fall Entries häufig für reale Objekte. Ein einfaches Beispiel dazu ist wieder das Telefonbuch. Dieses enthält Einträge von Telefonanschlüssen.

Alias

Bei einem Alias handelt es sich um einen Verzeichniseintrag, der auf ein anderes Verzeichnis verweist.

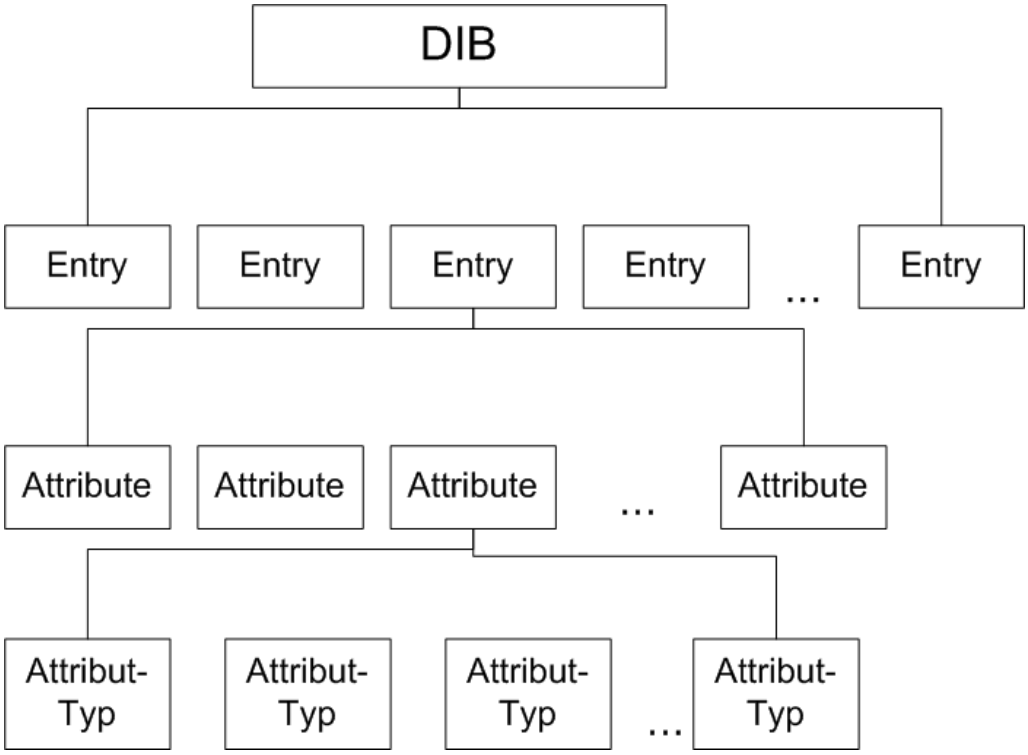
Attribute

Entries bestehen aus einer Reihe von Attributen. Jedes dieser Attribute besitzt einen Attributtyp. Der Attributtyp selbst kann einen oder mehrere Werte beinhalten.

Objektklassen

Alle möglichen Typen von Einträgen in einem LDAP-Schema werden als „Objektklassen" bezeichnet.

Nehmen wir diese „neuen" Begriffe und schauen sie uns in einem einfachen Schaubild an, um deren Bedeutung noch ein wenig genauer zu erfassen:



DIB

LDAP-Dienste werden auf Basis von verschiedenen Produkten angeboten. Im Bereich des System i finden wir dabei am häufigsten die folgenden LDAP-Server:

- Domino Directory
- IBM Tivoli Directory Server
- IBM Directory Server
- Microsoft Active Directory

Sicher könnte diese Auflistung auch noch erweitert werden. Wir wollen uns an dieser Stelle bewusst auf den Einsatz des IBM Directory Servers beschränken, der als Bestandteil des Systems i eingesetzt werden kann.

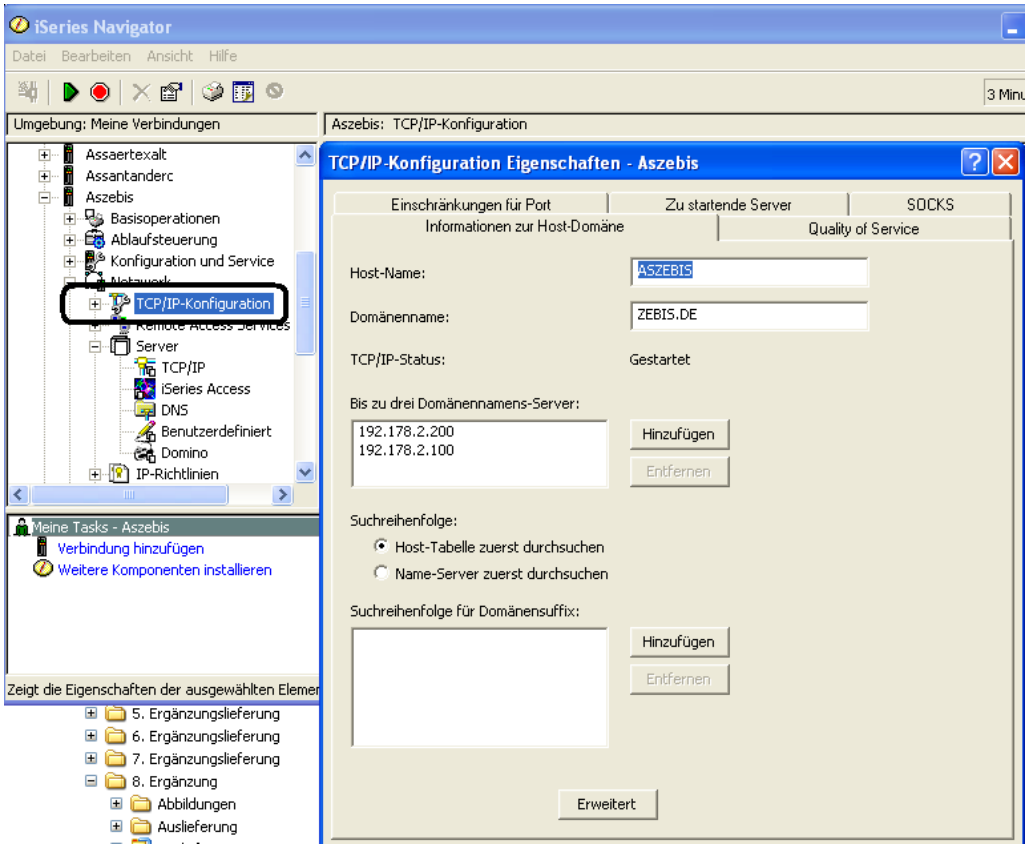
5.4.1
Seite 12

IBM Directory Server konfigurieren

Neben den verschiedenen LDAP-Servern, welche im weiteren Umfeld des System i zur Verfügung stehen, bietet IBM auch die Möglichkeit, die Verzeichnisdienste mit Hilfe des IBM Directory Servers auszuführen. Dieser IBM Directory Server steht als TCP/IP-Server nativ auf dem System i zur Verfügung und kann als zentraler LDAP-Server eingesetzt werden. In diesem Kapitel möchte ich Ihnen die Einrichtung des IBM Directory Servers kurz vorstellen.

Die Installation – oder besser die Konfiguration – des IBM Directory Servers erfolgt mit Hilfe des Web Administrators. Bevor wir diesen jedoch anwenden, wollen wir uns die Voraussetzungen anschauen, welche für die erfolgreiche Konfiguration benötigt werden.

Als erstes ist in diesem Zusammenhang die korrekte Einstellung des TCP/IP-Bereichs erforderlich. Dazu schauen wir uns zunächst die Domäneneinstellungen im TCP/IP-Bereich an. In der folgenden Abbildung sehen Sie diese in dem iSeries Navigator. Natürlich können Sie die Einstellungen auch mit der 5250-Version in dem Menü „GO CFGTCP“ und der Auswahl 12 (TCP/IP Domäneninformationen ändern) einsehen bzw. auch anpassen.



TCPIP Domänen-Einstellungen

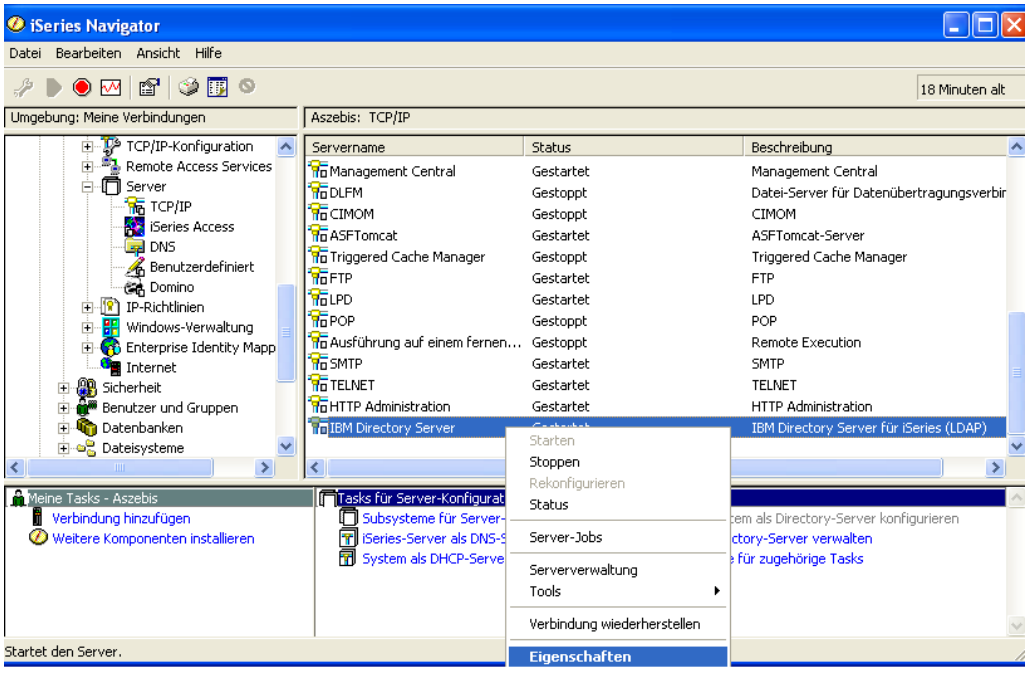
5.4.1.1
Seite 2

In diesem Bereich ist für die folgende Konfiguration des IBM Directory Servers der Domänenname von Bedeutung. Dieser ist in unserem Beispiel „ZEBIS.DE“. Dieser Eintrag sollte dem voll qualifizierten Namen der i5 entsprechen.

Anmerkung

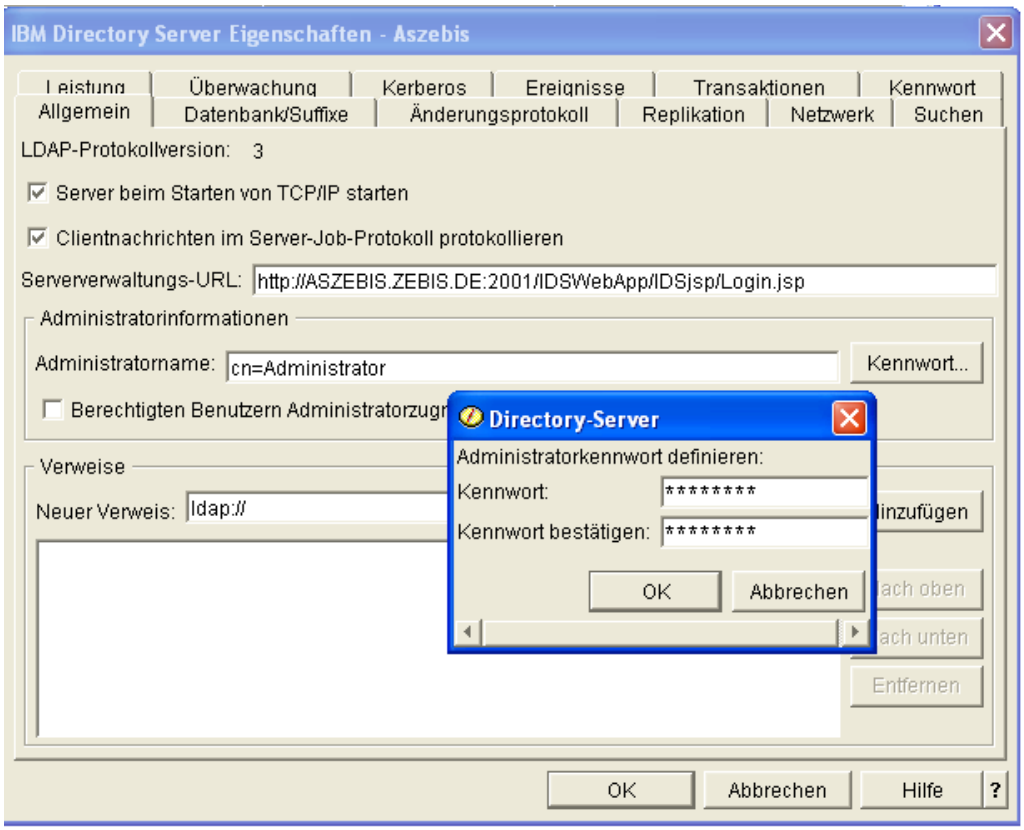
Der voll qualifizierte Name des Systems ist nicht nur für die Ausführung des IBM Directory Servers, sondern für eine Vielzahl anderer Anwendungen von Bedeutung, welche auf oder im Zusammenhang mit dem System i5 ausgeführt werden können. Deshalb ist es von immenser Bedeutung, den qualifizierten Namen des Systems korrekt zu definieren!

Wenn der voll qualifizierte Hostname des Systems geprüft und gegebenenfalls geändert wurde (Achtung: Nach dem Ändern des Namens ist ein Neustart von TCP/IP erforderlich!), schauen wir uns anschließend die Eigenschaften des Domino Directory Servers an. Diese finden wir ebenfalls in dem iSeries Navigator in dem Bereich Netzwerk / Server / TCPIP. Wenn Sie diesen Bereich selektieren, erscheint in dem Hauptbereich des iSeries Navigators eine Auflistung aller TCP/IP-Serverdienste. Unter anderem finden wir hier auch den IBM Directory Server, den wir mit der rechten Maustaste selektieren und anschließend die Option „Eigenschaften“ wählen.



Eigenschaften des IBM Directory Servers

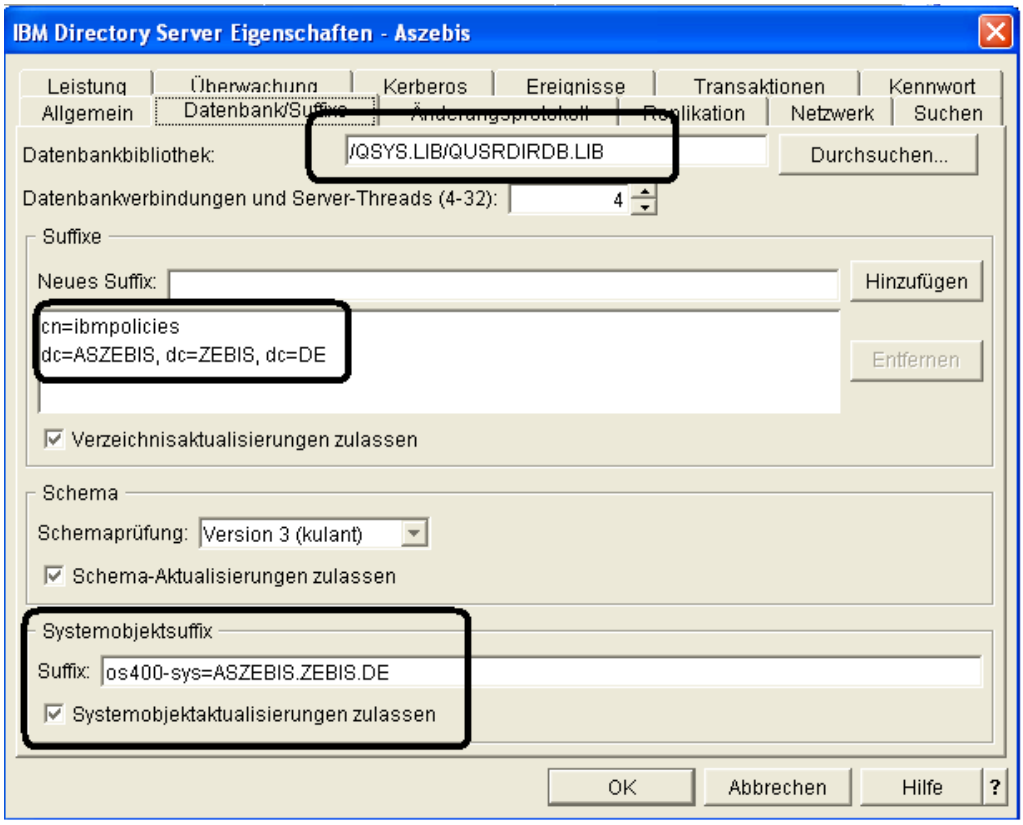
Neben der Vielzahl unterschiedlicher Parameter, welche für die Ausführung des IBM Directory Servers angegeben werden können, wollen wir uns zu diesem Zeitpunkt lediglich mit dem Administrator und dessen Kennwort auseinandersetzen. Hier liegt häufig eine „Unschärfe", welche bei dem Einsatz des IBM Directory Servers immer wieder zu Problemen führt – denn wurde an dieser Stelle kein Kennwort zugewiesen, dann schlägt auch die Authentifizierung mit dem Administrator fehl!



Kennwortangabe für den Administrator

Ein weiterer wichtiger Punkt verbirgt sich hinter dem Bereich „Datenbank / Suffixe".

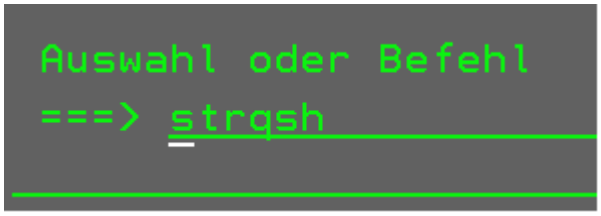
5.4.1.1
Seite 4



Suffixangaben

Achten Sie bei den Suffixangaben auf die korrekten Einträge in den Bereichen „Datenbankbibliothek“, „Suffixe“ und „Systemobjektsuffix“. Während in dem Bereich „Datenbankbibliothek“ der Name der Bibliothek stehen sollte, welche für den IBM Directory Dienst verwendet werden soll (der Standardwert ist die Bibliothek QUSRDIRDB), sollten in den beiden anderen Abschnitten als Elemente die einzelnen Teile des qualifizierten Hostnamens enthalten sein. Meist werden diese Einstellungen bei der Installation des Betriebssystems i5 durchgeführt – wenn Sie den qualifizierten Namen des Systems jedoch zwischenzeitlich verändert haben, kann das Anpassen der Einstellungen in diesem Bereich notwendig sein.

Wenn alle Angaben korrekt sind, dann können wir einen ersten Versuch tätigen und den LDAP Server (genauer: den IBM Directory Server) testen. Dazu verwenden wir die QSHell Eingabe. Um diese zu starten, geben wir in einer 5250-Befehlseingabezeile den Befehl STRQSH ein.



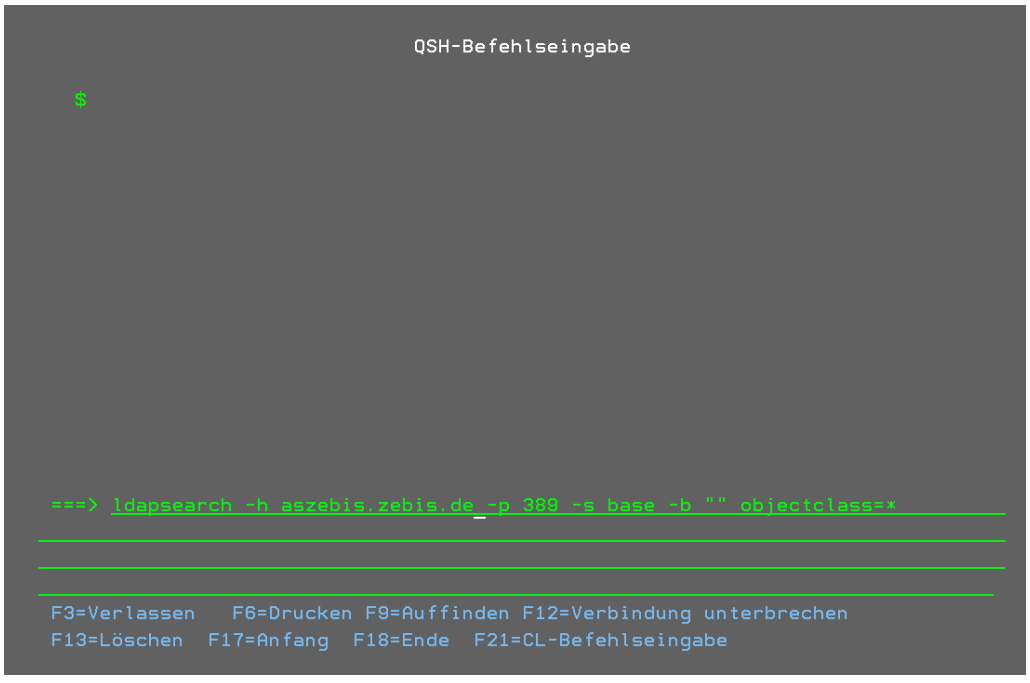
STRQSH

Damit gelangen wir in den Eingabebereich von QSH. Jetzt führen wir einen Test durch, ob der IBM Directory Server korrekt konfiguriert worden ist. Dazu geben wir einen LDAP-Suchbefehl ein:

ldapsearch -h server.domäne.endung -p 389 -s base -b “” objectclass=*

Ersetzen Sie in dem Befehl „Server.domäne.endung” durch die Elemente des voll qualifizierten Hostnamens, den wir zuvor in den TCP/IP-Einstellungen ermittelt haben. Damit erhalten wir dann einen Befehl wie das folgende Beispiel zeigt:

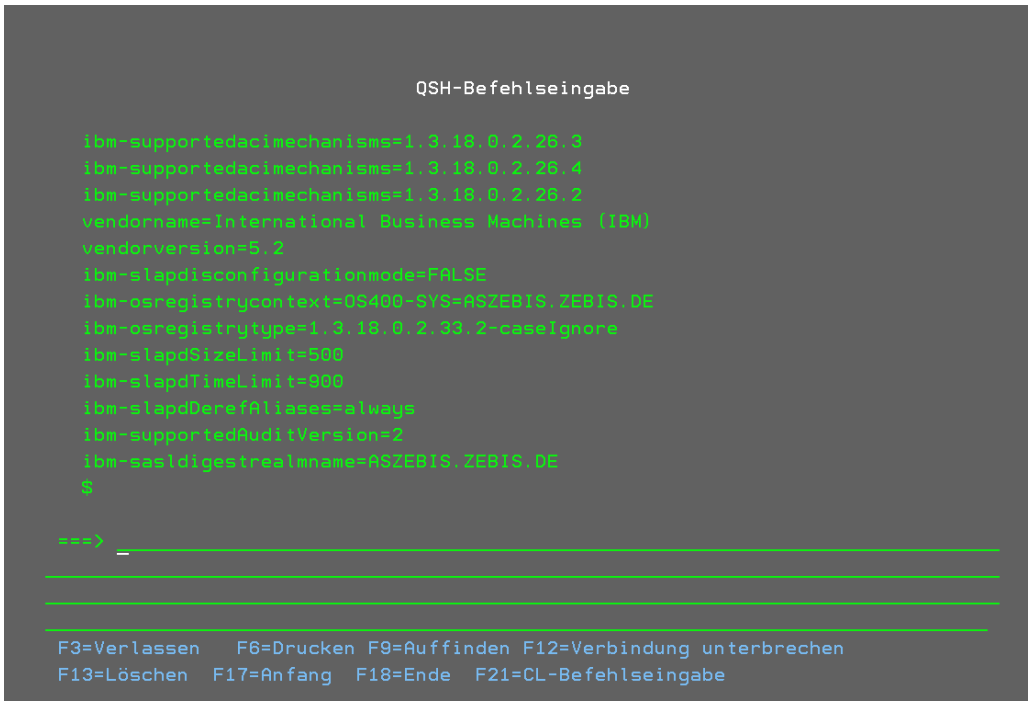
ldapsearch -h aszebis.zebis.de -p 389 -s base -b „“ objectclass=*



LDAP Suche in QSH

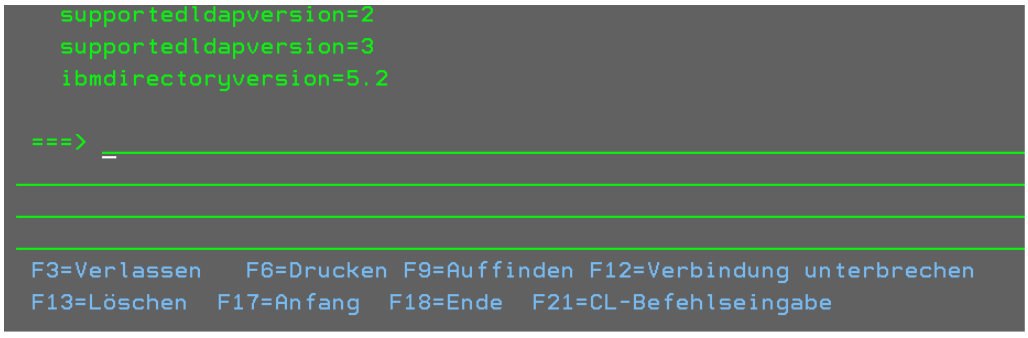
5.4.1.1
Seite 6

Bestätigen wir jetzt die Befehlseingabe mit der Enter-Taste, dann wird eine Suche innerhalb des LDAP-Verzeichnisses durchgeführt. Das Ergebnis finden wir wieder in der QSH Anzeige:



LDAP-Suche erfolgreich durchgeführt

Wichtig ist in dieser Anzeige unter anderem die Anzeige des Eintrags „ibmdirectoryversion=line“. Blättern Sie gegebenenfalls in der Anzeige, um den Eintrag zu finden.



Der Eintrag ibmdirectoryversion

Fehlt dieser Eintrag, dann können einige Anwendungen wie zum Beispiel IBM Workplace nicht korrekt ausgeführt werden. Das Fehlen des Eintrag deutet nämlich darauf hin, dass ein so genannter anonymer Zugriff auf den LDAP-Dienst unterbunden ist – eine Funktion, welche allerdings für einige Anwendungen (wie den zuvor IBM Workplace) benötigt wird. Sollten Sie den Eintrag nicht finden, dann schauen Sie sich noch einmal die Eigenschaften des IBM Directory Servers in dem iSeries Navigator an.

Tivoli Directory Server

Dem Meister der vielen Namen und Umbenennungen (gemeint ist hier IBM) ist es auch gelungen, in dem Bereich der Directory Server für einige Verwirrung zu sorgen, die ich an dieser Stelle ein wenig entwirren möchte.

Bis zum Erscheinen der Version V5R3 des Betriebssystems i5/OS war der IBM Directory Server der Standard-Server auf dem System i, wenn es um die Bereitstellung von Verzeichnis-Serverdiensten ging. Im Rahmen der Produktvereinheitlichung über die unterschiedlichen eServer-Plattformen hinweg wurde seit der Version V5R3 dieser Server durch den Tivoli Directory Server ersetzt. Während die „alten" Versionen des Verzeichnis-Servers auf der iSeries bzw. System i5 mit Hilfe des iSeries Navigators administriert werden konnten, wurden die Verwaltungsaufgaben mit dem Tivoli Directory Server in den http-Administrationsbereich verlagert. Damit bedingt der Tivoli Directory Server auch den Einsatz des WebSphere Application Servers, der als Ausführungsebene für den Verzeichnisdienst bzw. dessen Administration benötigt wird.

Den http-Administrationsbereich haben wir bereits in den vorhergehenden Kapiteln ausgiebig behandelt. Deshalb setze ich an dieser Stelle die Kenntnis über den Start und die grundlegenden Bedienungskenntnisse als gegeben voraus.

Um den Tivoli Directory Server zu administrieren, starten wir zunächst den http-Administrations-Client in einem Browser, indem wir den Hostnamen des System i und den dem Administrationsbereich zugeordneten Port (dies ist in der Regel der Port 2001) angeben:

http://hostname.domain:2001

In unserem Beispiel verwende ich dazu die folgende Beispiel-URL:

http://aszebis:2001

5.4.1.2
Seite 2

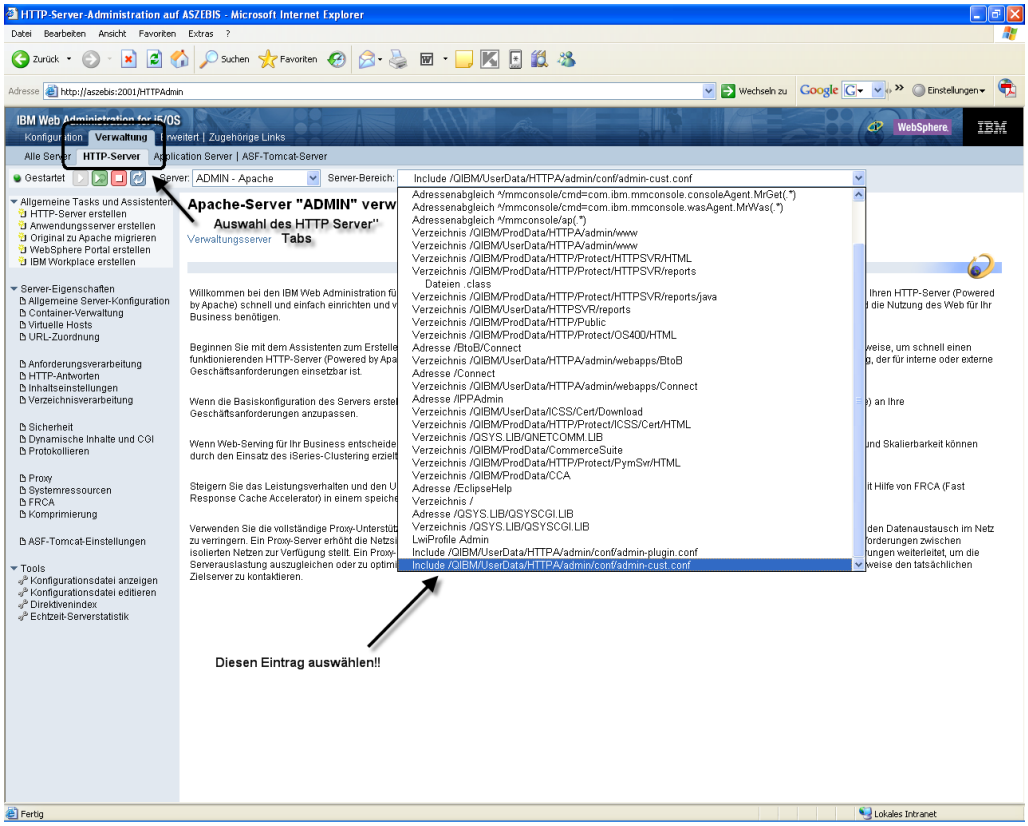
Damit gelangen wir auf die Hauptseite des http-Administrations-Clients:

Die Einstiegsmaske zum http-Administrator

Bitte wählen Sie an dieser Stelle nicht den Eintrag „IBM Directory Server for i5/OS“ aus, sondern selektieren Sie zunächst den obersten Eintrag in der Maske „IBM Web Administration for i5/OS“. In diesem Bereich müssen wir zunächst noch einige Einstellungen überprüfen bzw. anpassen, welche für die Ausführung des LDAP-Dienstes erforderlich sind.

In der Folgemaske selektieren wir nun den Tabellenbereich „Verwaltung“ und darin anschließend den Bereich „http-Server“.

Die Verwaltung des Application Servers wird ebenfalls in dem http-Administrations-Client ausgeführt.



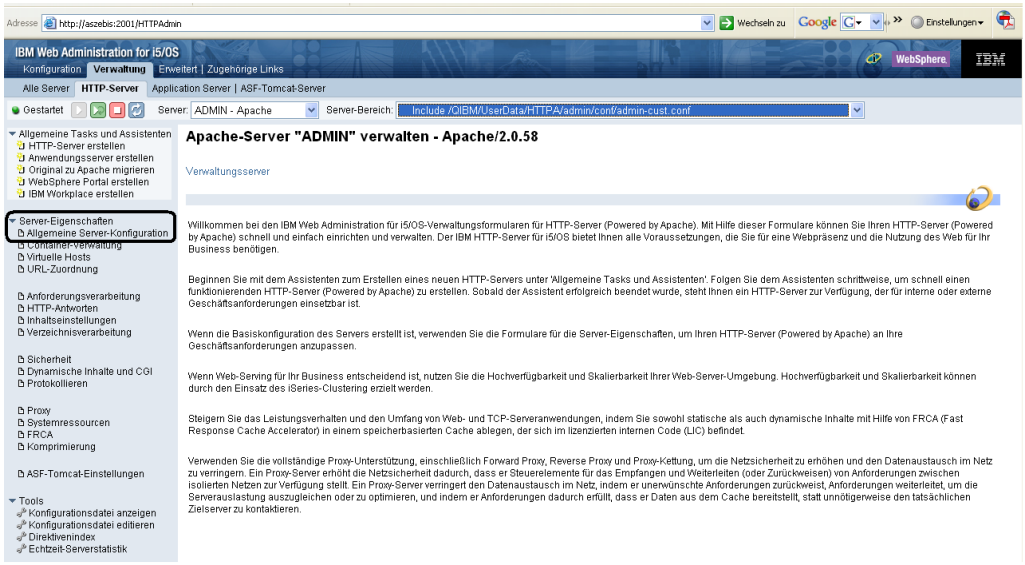
Auswahl des Web-Servers

Achten Sie darauf, dass Sie den Server „Admin Apache“ in dem Auswahlfeld selektiert haben.

In dem daneben befindlichen Feld „Server-Bereich“ können wir jetzt die Ebene auswählen, welche verarbeitet werden soll. Für unser Vorhaben ist dies der Eintrag „Include /QIBM/UserData/HTTPA/admin/config/admin-cust.conf“, den wir in der Auflistung der Bereiche markieren.

5.4.1.2
Seite 4

Damit sollte die Anzeige in etwa den Inhalt haben, wie er auch in der folgenden Abbildung zu sehen ist:



Auswahl der allgemeinen Server-Konfiguration

Wenn nicht bereits die allgemeine Server-Konfiguration angezeigt wird, selektieren wir in dem linken Navigationsbereich den entsprechenden Eintrag:

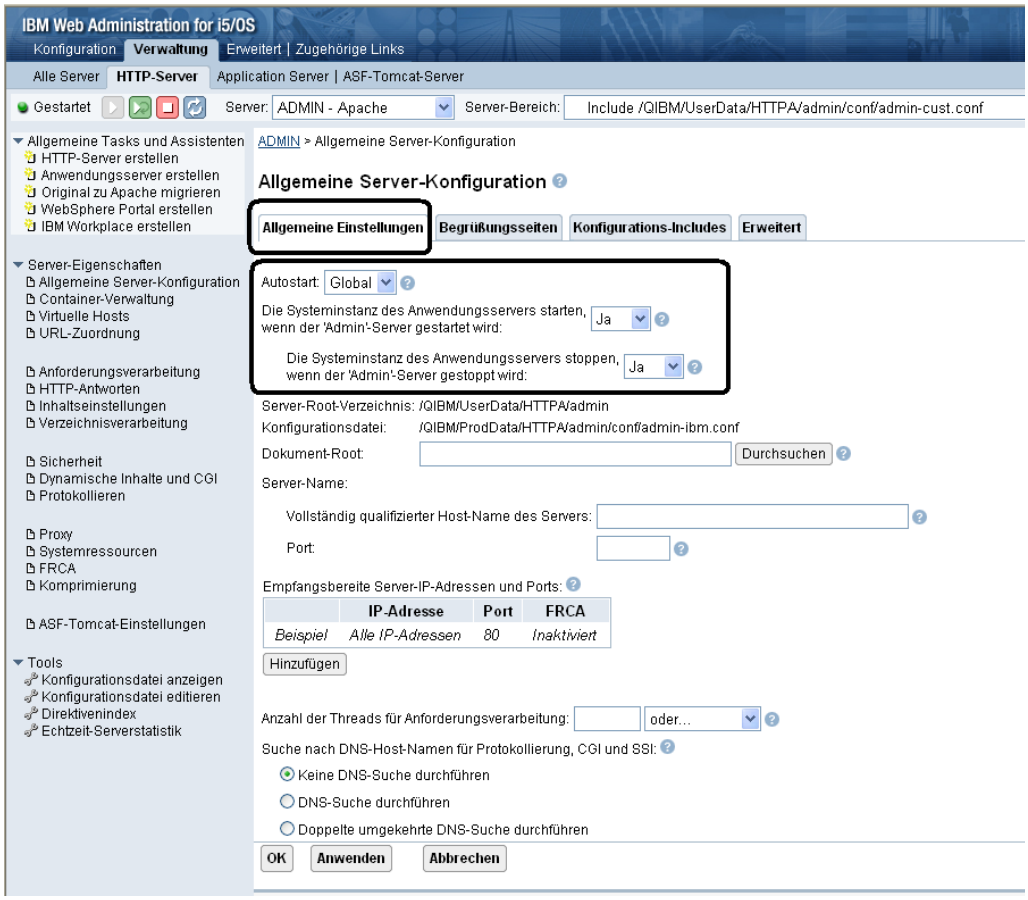
In dem ersten Tabellenbereich „Allgemeine Einstellungen“ befinden sich verschiedene Einstellungen, welche wir für den Betrieb der LDAP-Verzeichnisfunktionen anpassen müssen. Ändern Sie diese ab, sollten sie nicht bereits die erforderlichen Werte aufweisen:

Ausführungsschritte

- „Die Systeminstanz des Anwendungsservers starten, wenn der ‚Admin’-Server gestartet wird:“
  - Dieser Parameter muss auf „Ja“ stehen.
- „Die Systeminstanz des Anwendungsservers stoppen, wenn der ‚Admin’-Server gestoppt wird:“
  - Dieser Parameter muss auf „Ja“ stehen.

Achtung

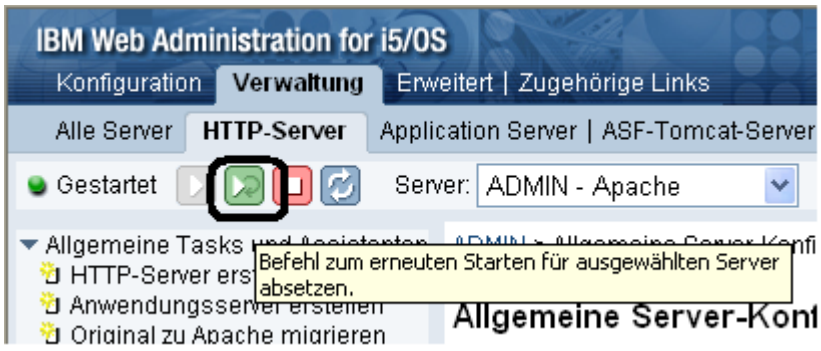
Der zweite Parameter „Die Systeminstanz des Anwendungsservers stoppen, wenn der ‚Admin’-Server gestoppt wird“ erscheint erst dann, wenn der erste Parameter auf „Ja“ gestellt worden ist!



Anpassen der globalen Einstellungen

Bestätigen Sie die Änderungen mit einem Klick auf „OK“ im unteren Bereich der Anzeige.

Die soeben durchgeführten Änderungen wirken sich erst nach einem Neustart des Web-Servers aus. Diesen Neustart können wir mit einem Klick auf das entsprechende Symbol in der oberen linken Anzeige initiieren:



Neustart des Web-Servers

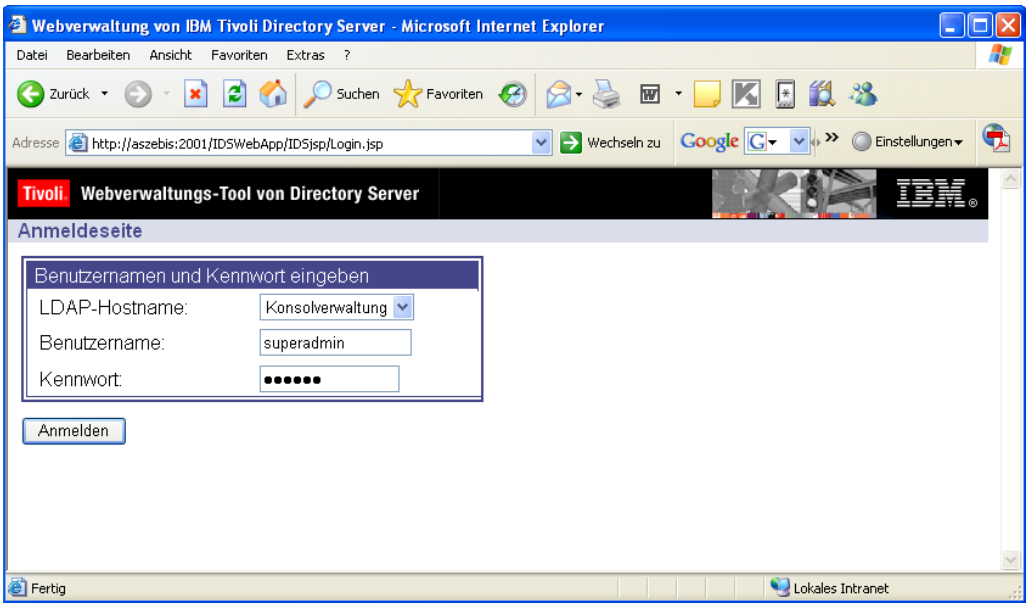
5.4.1.2
Seite 6

Ein solcher Neustart ist natürlich nur dann erforderlich, wenn Sie die Parameter angepasst haben!

Nachdem diese Voraussetzungen geprüft oder auch angepasst worden sind, können wir mit der Arbeit des Tivoli Directory Servers beginnen. Dieser wird in der Startseite des http-Administrators als „IBM Directory Server for i5/OS“ zur Auswahl angeboten:

Auswahl zur Konfiguration des IBM Directory Servers

Es kann je nach System i-Leistung einen kurzen Augenblick dauern. Danach werden wir aufgefordert, uns in der Anzeige „Tivoli Server Web Administrator“ anzumelden.



Anmeldeinformationen für Tivoli Directory Server

Geben Sie in dieser Anzeige die folgenden Parameter ein:

- LDAP-Hostname
  - Console Admin
- Benutzername
  - superadmin
- Kennwort
  - secret

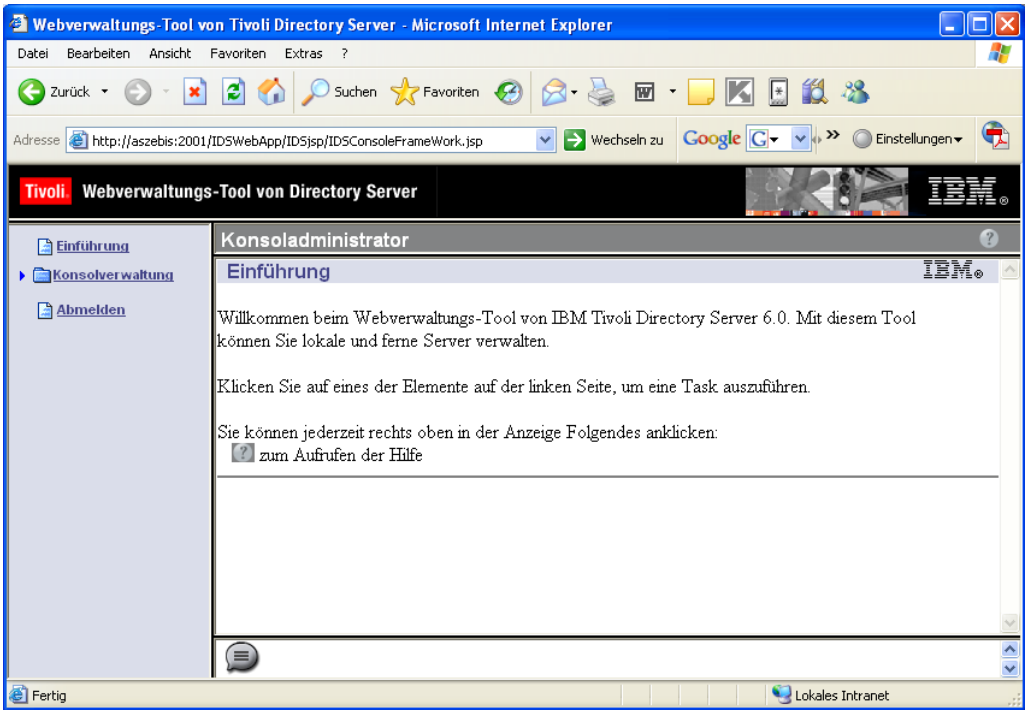
Anmerkung

Der Benutzer „superadmin“ und das zugehörige Kennwort „secret“ sind als Standardwerte von IBM vorbelegt. Wenn Sie diese bisher noch nicht genutzt haben, dann können Sie diese Werte für die erste Anmeldung verwenden.

Achten Sie in jedem Fall auf die Groß-/Kleinschreibung! Sowohl bei der Eingabe des Benutzernamens als auch bei dem Kennwort wird strikt zwischen Groß- und Kleinschreibung differenziert!

5.4.1.2
Seite 8

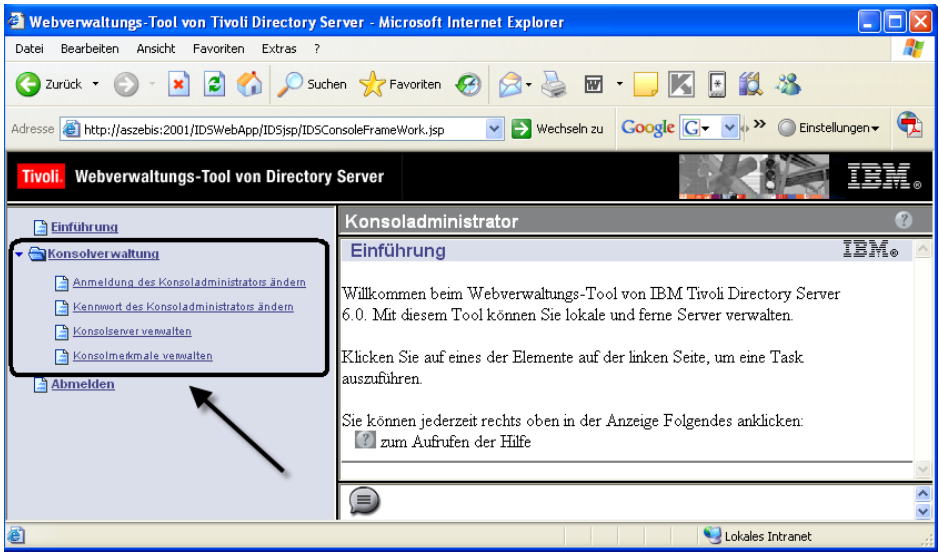
Bestätigen Sie Ihre Eingaben mit einem Klick auf die Schaltfläche „Anmelden“. Damit gelangen wir – vorausgesetzt die Login-Informationen waren korrekt – auf die Einführungsseite von Tivoli:



Einführungsseite zu Tivoli Directory

Wie fast alle Anwendungen im Web-Administrationsumfeld finden wir auch hier im linken Bereich der Anzeige den Navigationsbereich, der jetzt drei verschiedene Auswahlmöglichkeiten bietet. Per Standard ist die erste Option „Einführung“ als Startoption bereits durch das Anmelden geöffnet worden. Diese Option gibt einen Kurzeinblick auf beispielsweise die installierte Version des Tivoli Directory Servers.

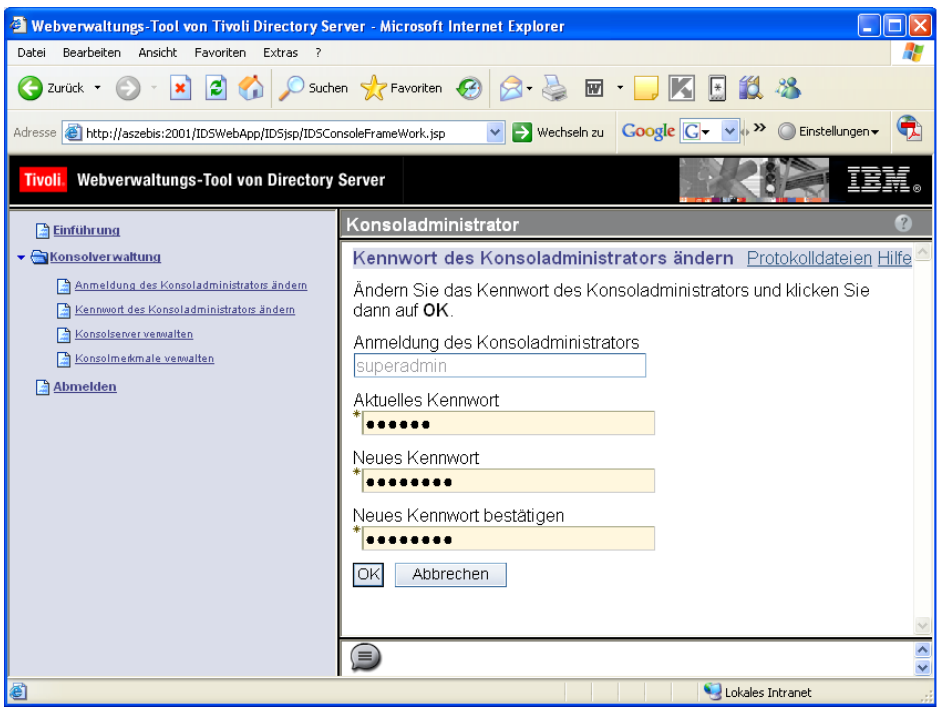
Schauen wir uns zunächst den Bereich „Konsolverwaltung“ einmal an und klicken dazu einmal auf den entsprechenden Eintrag im Navigationsbereich, um diesen zu erweitern:



Konsolverwaltung

Wie Sie sehen können, lassen sich in diesem Bereich unter anderem die Anmeldeinformationen zu dem Tivoli Directory Server ändern. Es ist dringend zu empfehlen, dass die Standard Anmelde-Informationen geändert werden. Anderenfalls kann jeder, der Kenntnis über die Standardbenutzer und Standardkennwörter hat, in den Verwaltungsbereich gelangen.

Um das Kennwort des Directory Administrators zu ändern, wählen wir die Option „Kennwort des Konsoladministrators ändern“. Damit gelangen wir in die folgende Anzeige:



Ändern des Kennworts

5.4.1.2
Seite 10

Der aktuelle Administrator „superadmin“ ist voreingestellt. Wie von anderen Anwendungen auch gewohnt, muss nun zunächst das aktuelle und anschließend zweifach das neue Kennwort eingegeben werden.

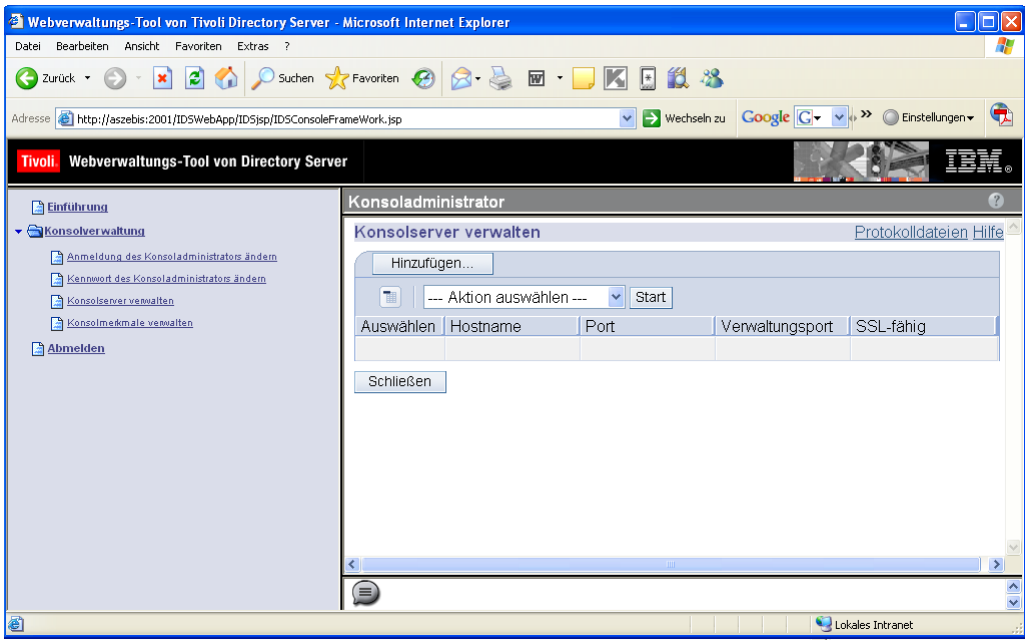
Denken Sie in jedem Fall wieder daran, dass die Eingabe der beiden Felder zwischen der Groß- und der Kleinschreibung unterscheidet!

Das Ändern des Kennworts bestätigen wir abschließend mit einem Klick auf die Schaltfläche „OK“.

Damit haben wir die Login-Informationen insofern „sicher“ gemacht, als das das Standardkennwort des Administrators angepasst wurde. Natürlich können Sie neben diesem Administrator auch noch weitere Administratoren definieren, welche zur Verwaltung des Verzeichnis-Servers berechtigt sind. Dies können Sie mit der Option „Anmeldung des Konsoladministrators ändern“ durchführen, die sich ebenfalls im Navigationsbereich befindet.

Kommen wir nun zu dem wichtigen Part – nämlich der eigentlichen Verwaltung des Verzeichnis-Servers. Es kann wahlweise einen oder auch mehrere solcher Server geben, die für unterschiedliche Aufgaben eingesetzt werden können. Um diese Server zu verwalten, selektieren wir zunächst im Navigationsbereich den Eintrag „Konsolserver verwalten“.

Damit gelangen wir in die folgende Anzeige:



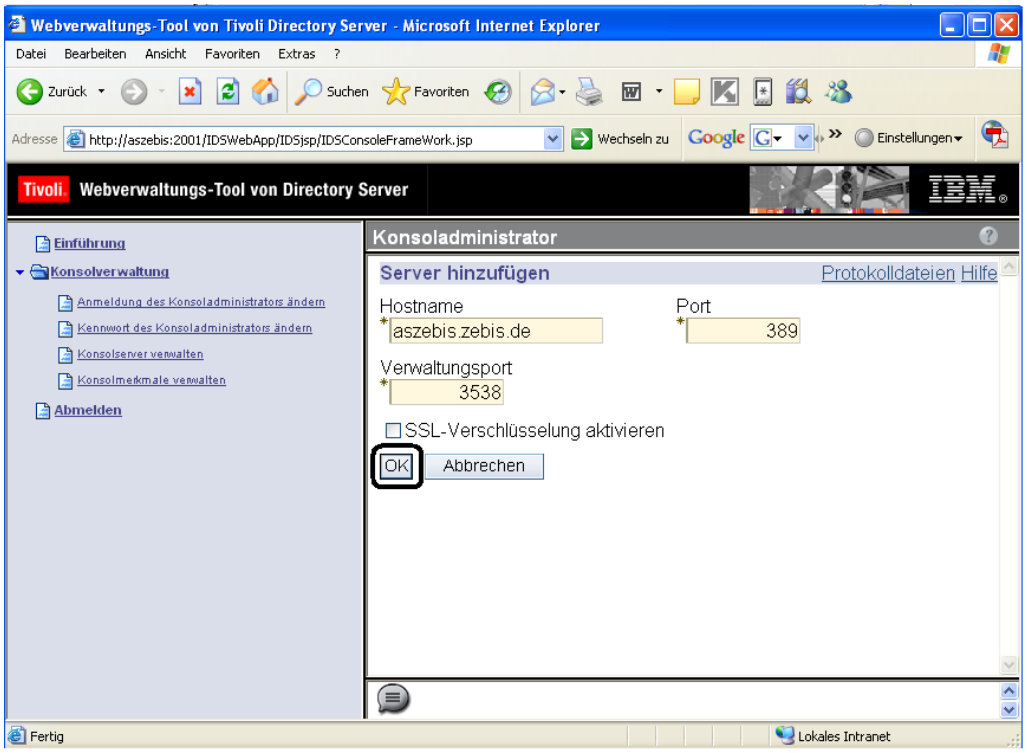
Konsolserver verwalten

Wenn Sie bisher noch keinen Directory Server konfiguriert haben, dann sehen Sie jetzt eine leere Seite, in der wir unseren ersten Verzeichnis-Server hinzufügen können. Mit einem Klick auf die Schaltfläche „Hinzufügen“ wird dieser Konfigurationsprozess ausgelöst.

Anmerkung

Sollte es bereits LDAP-Server geben, dann befindet sich neben der Schaltfläche „Hinzufügen“ eine Anzahl weiterer Schaltflächen, mit denen die vorhandenen Server verwaltet werden können.

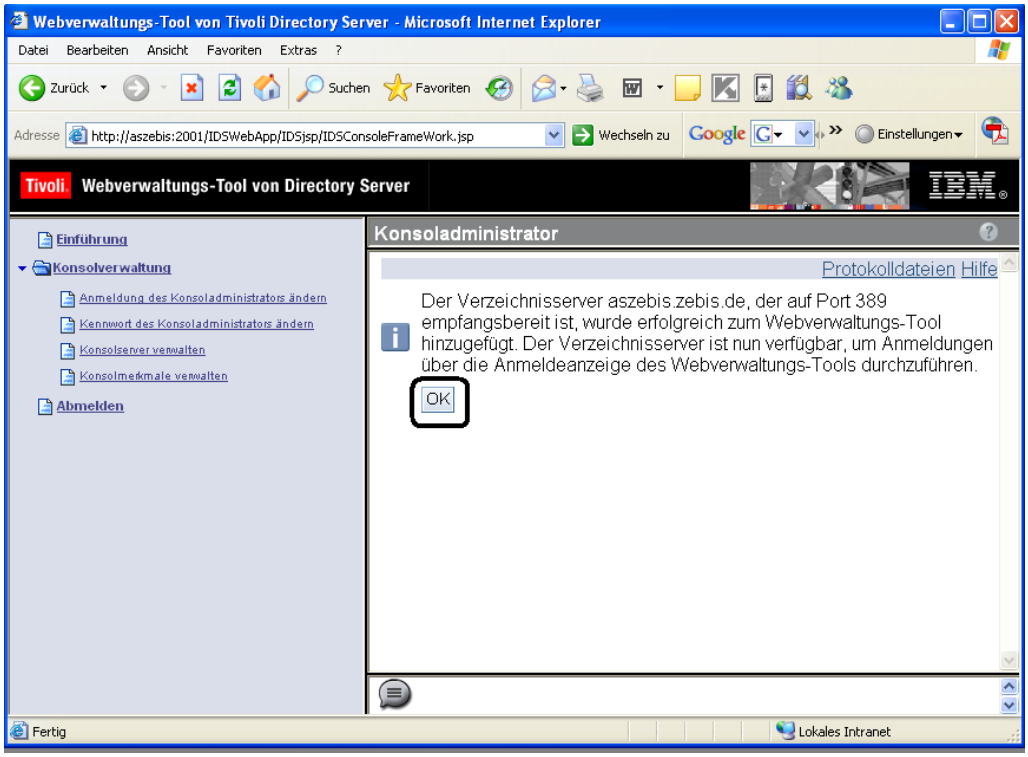
In der nächsten Anzeige werden wir zur Eingabe des qualifizierten Hostnamens des Servers aufgefordert, welcher für den Verzeichnisdienst verwendet werden soll. Hier geben wir den Namen ein, den wir in den TCP/IP-Einstellungen als den qualifizierten Namen zuvor ermittelt haben. Wenn er nicht bereits verwendet wird, können wir den per Standard vorgeschlagenen Port übernehmen. Wahlweise kann für die Kommunikation eine sichere SSL-Verbindung gewählt werden. Dazu selektieren Sie einfach die Option „SSL-Verschlüsselung aktivieren“.



Angabe der Serverparameter

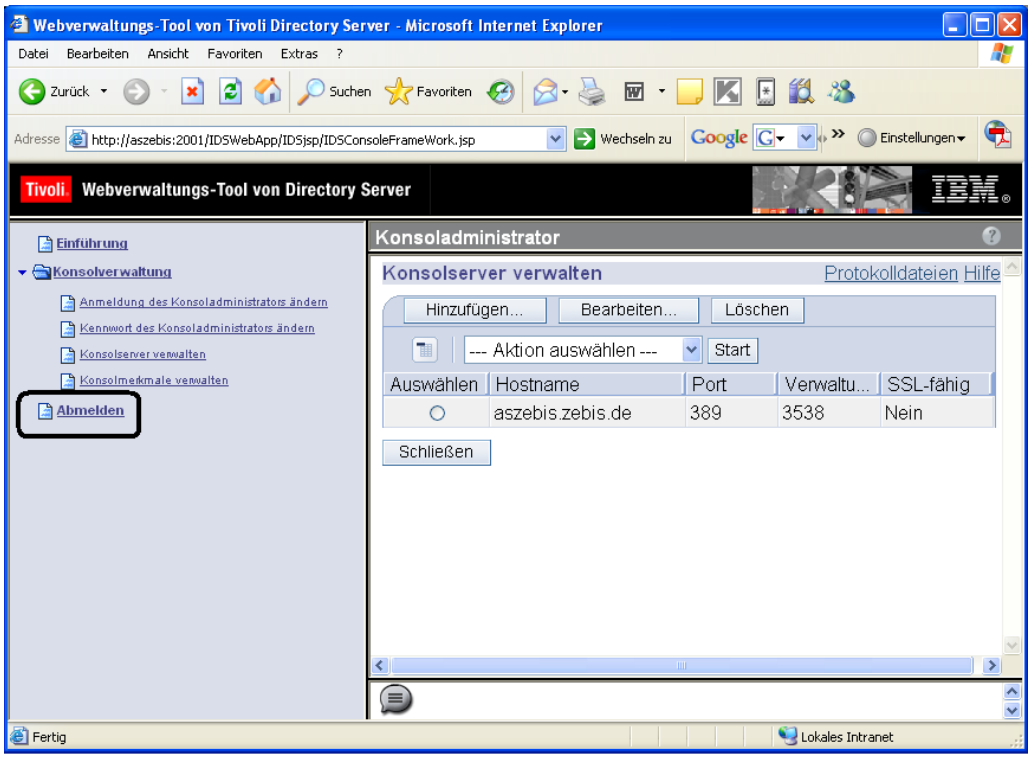
Nach Abschluss aller Eingaben bestätigen Sie diese mit einem Klick auf „OK“. Damit wird der Server konfiguriert und steht für den LDAP-Dienst zur Verfügung.

5.4.1.2
Seite 12



Erfolgreiche Einrichtung

Mit einem Klick auf „OK“ gelangen wir wieder in die Übersicht der definierten Server – hier sind jetzt weitere Schaltflächen zur Verwaltung der Server vorhanden.



Die Serverübersicht enthält den neuen Server

Damit haben wir die Grundinstallation des Directory Servers auf der i5 abgeschlossen. Zum Beenden dieser Tätigkeit melden wir uns mit einem Klick auf „Abmelden“ ab.

Abmeldung erfolgreich

Mit diesen Konfigurationsaufgaben haben wir nun einen einsatzfähigen LDAP Server auf dem System i zur Verfügung, den wir beliebig für die unterschiedlichen Anforderungen an Verzeichnisdienste einsetzen können.

5.4.1.2
Seite 14

Mail mit System i

Auch wenn es sicher keine Kernkompetenz des System i ist, so lassen sich auch einfache Mail-Komponenten mit Hilfe der Basiswerkzeuge der Server-Dienste des i5/OS abbilden und realisieren.

Damit besteht die Möglichkeit, TCP/IP-Funktionen der Maschine dazu einzusetzen, um Mails zu versenden und zu empfangen. Dabei stehen System i beide Standards für Versenden oder Empfangen von Mails in Form von POP-, IMAP- und SMTP-Diensten zur Verfügung.

Diese Dienste lassen sich dazu einsetzen, um Mails innerhalb des eigenen Netzwerks zu versenden bzw. zu empfangen, aber auch, um diese mit „fernen" Benutzern austauschen zu können.

Doch wozu können wir diese Mail-Funktionen native auf dem System i einsetzen? Die Frage ist relativ leicht zu beantworten: Lassen sich doch alle Mail-Funktionen bei Bedarf über die TCP/IP-Dienste abbilden, die auf dem System i zur Verfügung stehen. Allerdings fehlen dort natürlich solche Funktionen, die einen vollwertigen Mail-Server ausmachen. Diesen will IBM mit TCP/IP-Diensten, die im i5/OS integriert sind, auch keine Konkurrenz machen.

Es kann sicher im einen oder anderen Fall durchaus Sinn machen, e-Mail-Funktionen mit System i einzusetzen. Wenn beispielsweise Mails aufgrund von Systemüberwachungen erstellt und an einen zuständigen Mitarbeiter übertragen werden sollen, dann ist der Einsatz der Mail-Dienste auf dem System i durchaus sinnvoll. Allerdings ist das Umsetzen von komplexen Mail-Anwendungen native mit System-i-Funktionen eher fragwürdig. Dafür eignen sich spezielle Mail-Systeme – einige davon, wie zum Beispiel Lotus Domino, sind auch native auf dem System i verfügbar und können dort als zusätzliches Lizenzprogramm installiert werden. Doch mit diesen sehr speziellen Mail-Systemen wollen wir uns hier nicht beschäftigten.

Wenn Sie sich dazu entschieden haben, die nativen Mail-Funktionen im TCP/IP-Umfeld zu nutzen, dann müssen Sie folgende Informationen zusammentragen, die bei der Einrichtung der Mail-Dienste benötigt werden:

- Werden interne oder externe Mails oder ein Mix aus beiden Varianten benötigt?
- Wird eine Firewall eingesetzt? Wenn ja – dann benötigen Sie unbedingt die Konfigurationsdaten der Firewall, zumindest die IP Adresse.
- Wenn Sie DNS einsetzen, benötigen Sie die DNS-Konfigurationsinformationen – wie beispielsweise die IP-Adresse des DNS-Servers oder dessen Hostnamen.

5.4.2
Seite 2

- Wird Mail Proxy, Mail Relay oder Mail Router eingesetzt? Wenn ja, so benötigen wir auch in diesem Fall die IP-Adressen.

Doch bevor wir uns mit der eigentlichen Einrichtung der Mail-Dienste auf dem System i beschäftigen, sollten wir uns einmal die Standards ansehen, die für den Mail-Verkehr eingesetzt werden können.

SMTP

Bei SMTP (Simple Mail Transfer Protocol) handelt es sich um einen Standardprotokoll, mit dem wir Mails auf dem System i senden können. Der Standard-Port, der für diesen TCP/IP-Bereich verwendet wird, ist mit 25 vordefiniert.

Grundsätzlich ist der SMTP-Standard in der Lage, die Authentifizierung eines Clients mittels Benutzernamen und Kennwort durchzuführen.

Unterstützt wird neben dem „internen" Mail-Verkehr auch der Austausch von Mail-Informationen mit anderen Mail-Servern. In Abhängigkeit der SMTP-Einstellungen und der Mail-Empfängeradresse entscheidet das System selbständig, wie und an welchen Mail-Server die Mail zugestellt werden muss.

SMTP bedingt, dass der Mail-Versand vom Absender ausgelöst wird. Das Übertragen von Mails von einem Server – also auch dem System i – zu einem Client oder einem anderen Server wird nicht über SMTP realisiert. Sollten solche Übertragungen durchgeführt werden, dann stehen dazu alternativ das IMAP-Protokoll oder das POP3-Protokoll zur Verfügung.

SMTP ist ein recht einfaches Protokoll, das es uns erlaubt, Mails mit Hilfe von Programmen (CL, RPG oder ähnlichem) zu erzeugen und diese an den Empfänger zu versenden. Rein theoretisch ist auch das Versenden von Mails über eine entsprechende 5250-Eingabe möglich.

Für diese Form der „manuellen" Ausführung von SMTP stehen unterschiedliche Befehle zum Einsatz bereit, die Sie in der folgenden Tabelle auszugsweise finden:

SMTP-Kommando	Beschreibung
HELO/EHLO (Hello/Extended Hello)	HELO bzw. EHLO startet die SMTP-Sitzung und identifiziert den Client am Server.
MAIL	MAIL leitet die Mail-Übertragung ein und liefert gleich die Absenderadresse mit.
RCPT (Recipient)	RCPT gibt die Adresse eines oder mehrerer Empfänger an. Dieses Kommando kann deshalb mehrmals ausgeführt werden.
DATA	Mit DATA wird die Übermittlung der eigentlichen e-Mail-Nachricht mit „CRLF.CRLF" gekennzeichnet.

SMTP-Kommando	Beschreibung
RSET (Reset)	Mit RSET wird die bereits eingeleitete Mail-Übertragung abgebrochen. Die Verbindung zwischen Client und Server bleibt bestehen.
VRTY (Verify)	Mit VRFY kann die Empfängeradresse überprüft werden.
EXPN (Expand)	Die meisten MTAs behandeln EXPN wie VRFY.
NOOP	NOOP bewirkt eine Antwort vom Server. Damit wird die Verbindungstrennung durch Timeout verhindert.
QUIT	QUIT beendet die Verbindung zum SMTP-Server. Der Server liefert eine letzte Antwort zurück.

Jede SMTP Ausführung hat einen Returncode zur Folge, der Auskunft über die Transaktion gibt:

Status-Code	Beschreibung
211	System-Status oder System-Hilfe.
214	Hilfe-Informationen zum Ausführen eines Kommandos.
220	Server bereit.
221	Server beendet Verbindung.
250	Kommando ausgeführt.
251	Keine lokale Mailbox; Weiterleitung an „forward-path".
252	Überprüfung der Empfängeradresse nicht möglich; Die Nachricht wird dennoch versendet.
253	
354	Starten des Empfangs der Mail; Beenden mit „CRLF", „CRLF".
355	
421	Service nicht verfügbar; Verbindung wird beendet.
432	
450	Aktion nicht ausgeführt – Mail-Box nicht verfügbar.
451	Aktion abgebrochen – Fehler beim Ausführen.
452	Aktion abgebrochen – Nicht genügend System-Speicher.
453	
454	
458	
459	
500	Syntax-Fehler – Kommando unbekannt.
501	Syntax-Fehler – Parameter oder Argument falsch.
502	Kommando unbekannt/nicht implementiert.
503	Falsche Reihenfolge der Kommandos.
504	Parameter unbekannt/nicht implementiert.

Status-Code	Beschreibung
521	
530	
534	
538	
550	Syntax-Fehler – Kommando unbekannt.
551	Mail-Box nicht lokal; versuchen „forward-path" auszuführen.
552	Aktion abgebrochen – Fehler bei Speicherzuweisung.
553	Aktion nicht ausgeführt – Mailbox-Name nicht erlaubt (Syntax unkorrekt).
554	Transaktion fehlgeschlagen (beim Verbindungsaufbau: Kein SMTP-Service verfügbar).

Beim Einsatz von SMTP-Befehlen auf dem System i ist allerdings zu beachten, dass nicht alle Standard-SMTP-Befehle unterstützt werden. Nachfolgend finden Sie deshalb einige Befehle, die auf dem System i nicht zur Verfügung stehen:

SMTP Befehle

SMTP Befehl	Auswirkung	Auf System i verfügbar
SEND (Send)	Stellt Mails an eine oder mehrere Arbeitsstationen zu.	Nein
SOML (Send or mail)	Stellt Mails an eine oder mehrere Arbeitsstationen oder Empfänger zu, wenn der Benutzer nicht angemeldet ist.	Nein
SAML (Send and mail)	Stellt Mails an eine oder mehrere Arbeitsstationen und Empfänger zu, wenn der Benutzer nicht angemeldet ist.	Nein
EXPN (Expand)	Ask the receiver to confirm that a mailing list has been identified.	Nein
TURN (Turn)	Ask the receiver to send a valid reply and then become the SMTP sender, or else ask the receiver to send a refusal reply and remain the SMTP receiver.	Nein

POP

Beim zweiten, für den Mail-Verkehr wichtigen Protokoll handelt es sich um das Post Office Protocol (POP). Dieses Protokoll dient dazu, Mails zu empfangen bzw. zu lesen.

POP3 ist ein Protokoll für den Zugriff auf ein Postfach und ist im RFC 1225 definiert. Bei POP3 erfolgt der Datenaustausch per Standard über den TCP/IP-Port 110.

POP3 ist eine Erweiterung von POP2, das noch das SMTP-Protokoll zwingend erforderte, während POP3 auch ohne SMTP verwendet werden kann.

Einer der Vorteile von POP3 ist die Option, Mails nach dem Herunterladen noch auf dem Server zu belassen beziehungsweise zu entscheiden, nach welcher Zeit diese dort gelöscht werden sollen.

Wir können den POP3-Dienst bei gewisser Kenntnis mit folgenden Parametern beeinflussen. Diese lassen sich in CL-Programmen einbinden. Nachfolgend finden Sie einen Auszug aus den verfügbaren POP3-Parametern:

POP3-Befehl	Beschreibung
USER	Dieser Befehl identifiziert die Mail-Box.
PASS	Nach der Anmeldung muss der Benutzer mit einem Passwort authentifiziert werden.
QUIT	Mit diesem Befehl wird die Verbindung zum POP3-Server beendet. Alle gespeicherten Änderungen werden dann ausgeführt – zum Beispiel der Befehl DELE.
STAT	Dieser Befehl fordert vom Server die Anzahl der gespeicherten e-Mails an. Außerdem wird die Größe der Mail-Box zurückgeliefert.
LIST	Dieser Befehl fordert vom Server die Nummer und Größe aller e-Mails an. Es kann auch die Größe einzelner e-Mails abgefragt werden.
RETR	Dieser Befehl veranlasst den Server die angeforderte e-Mail zu liefern.
DELE	Dieser Befehl veranlasst den Server die angeforderte e-Mail zu löschen. Dieser Befehl wird aber erst nach dem Beenden der Verbindung ausgeführt.
NOOP	Dieser Befehl dient zur Aufrechterhaltung der Verbindung. Ein Timeout würde sonst die Verbindung trennen. Als Antwort liefert der Server „+OK".
RSET	Dieser Befehl setzt die aktive Verbindung zurück. Noch nicht ausgeführte Änderungen werden verworfen – zum Beispiel der Befehl DELE.

IMAP

Ein weiteres Protokoll darf bei der Behandlung der Mail-Funktionen auf dem System i nicht fehlen: das IMAP-Protokoll.

Bei IMAP handelt es sich um die Abkürzung: Internet Message Access Protocol.

Im Gegensatz zum POP3-Protokoll bleiben die Mails in der Regel auf dem Mail-Server; sie werden nur bei Bedarf auf den Client-Rechner übertragen. Damit werden die Clients entlastet, da die Mails nicht zu ihnen übertragen

werden. Allerdings belastet dieser Lösungsansatz unter Umständen das Netzwerk.

IMAP wurde mit dem Ziel entworfen, den Zugriff auf Mail-Boxen und Nachrichten so bereitzustellen, als wenn sich diese auf dem lokalen Rechner befänden.

Wie auch die übrigen Mail-Server-Dienste, so wird auch der IMAP-Dienst auf einem Standardport (143) ausgeführt.

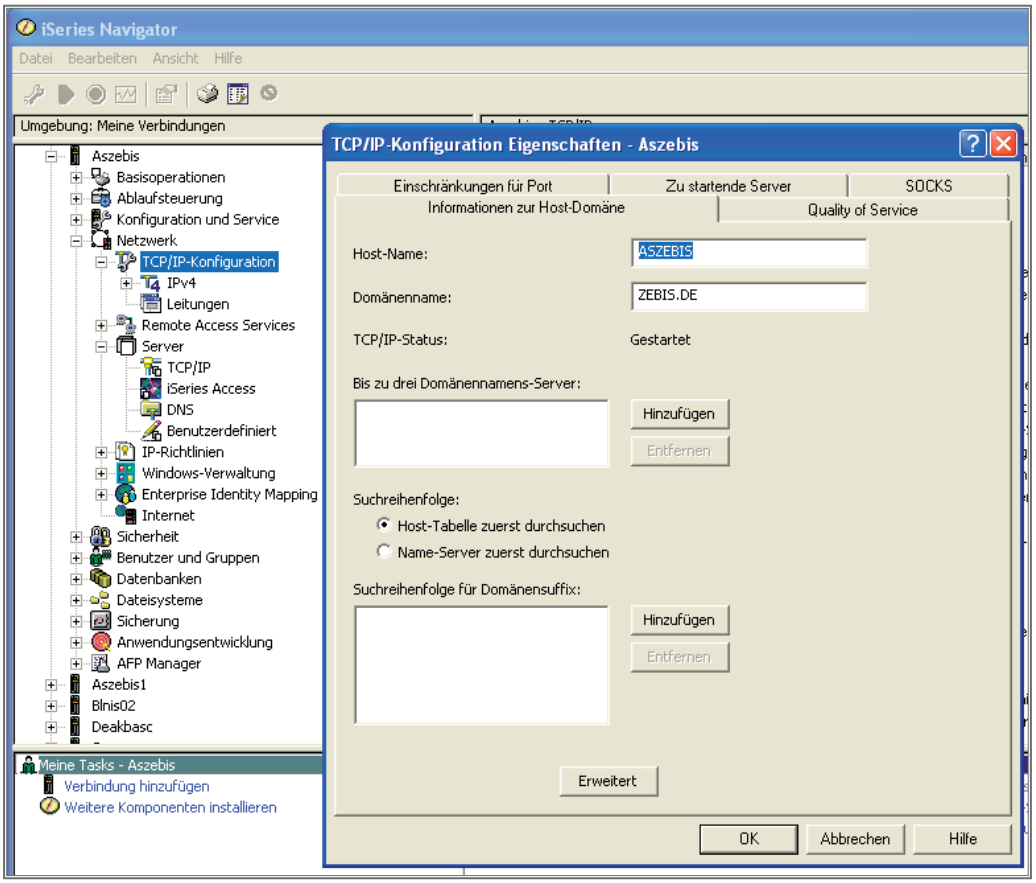
Einrichten der Mail-Funktionen auf System i

Die allgemeinen Einstellungen in Bezug auf die beiden TCP/IP-Mail-Protokolle können wir mit dem iSeries Navigator vornehmen. Dazu selektieren wir den Verbindungseintrag zum System i und darin den Untereintrag „Netzwerk / Server / TCPIP“ mit der rechten Maustaste. Damit gelangen wir in die Übersicht aller auf dem System i installierten TCP/IP Server. Darin enthalten sind unter anderem auch die beiden Mail-Server-Dienste POP und SMTP.

Die Basis für den Mail-Verkehr auf dem System i bildet TCP/IP. Deshalb muss dieser Bereich auch sauber konfiguriert sein. Details zum korrekten Installieren und Konfigurieren von TCP/IP sind bereits an anderer Stelle in diesem Buch detailliert beschrieben worden. Deshalb verweise ich hier lediglich auf die Bereiche, die die Grundkonfiguration von TCP/IP behandeln; ich beschränke mich lediglich auf die Prüfung der Informationen, die für den Mail-Verkehr relevant sind.

Auch wenn er sich noch nicht in allen Bereichen der Administration durchsetzen konnte, so bietet der iSeries Navigator die geeigneten Werkzeuge und Funktionen, um schnell und sicher den Mail-Bereich zu konfigurieren. Unter anderem werden wir ihn auch für die Prüfung der allgemeinen TCP/IP-Einstellungen verwenden.

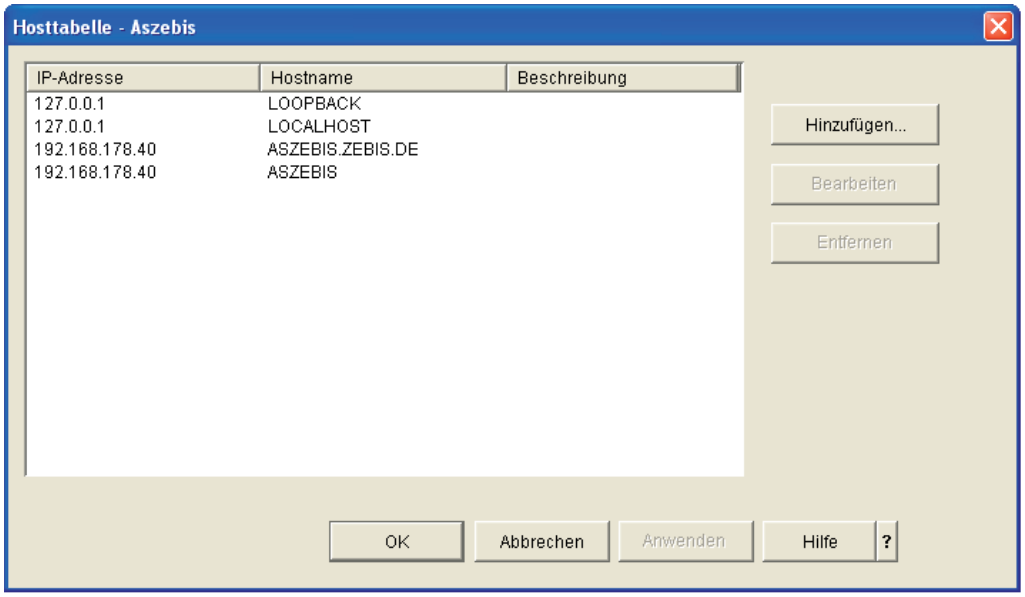
Dazu stellen wir mit dem iSeries Navigator eine Verbindung zum System i her und wählen den Bereich „Netzwerk / TCP/IP Konfiguration“, wie es die folgende Abbildung zeigt:



TCP/IP-Eigenschaften

Wichtig ist, dass Host- und Domänen-Name korrekt angegeben worden sind, denn diese beiden Werte dienen als Grundlage für den gesamten TCP/IP-Datenverkehr.

Beachten Sie, dass SMTP für die Kommunikation den voll qualifizierten Domänen-Namen benötigt. Dieser besteht aus Host- und Domänen-Name. Den Host-Namen können wir mit der Option „Host-Tabelle“ verwalten. Er wird als Option mit einem Klick der rechten Maustaste auf den Eintrag „TCP/IP-Konfiguration“ zur Auswahl angeboten:



Host-Tabelle

Achten Sie bitte darauf, dass der voll qualifizierte Domänen-Name als erster Eintrag für die zu verwendende IP-Adresse erscheint. Sollte dies nicht der Fall sein, dann ändern Sie bitte die Reihenfolge der Host-Einträge entsprechend ab.

Anmerkung

In der vorhergehenden Abbildung ist die Reihenfolge für die IP Adresse 192.168.178.40 korrekt angegeben:

40.40.40.40 ASZEBIS.ZEBIS.DE

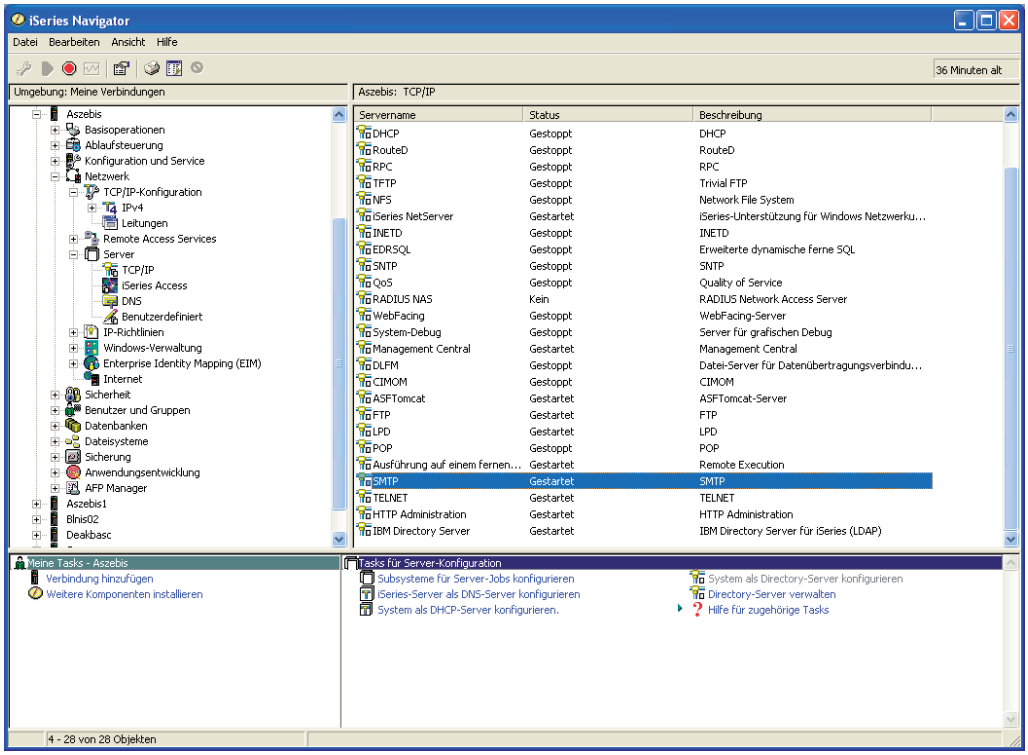
192.168.178.40 ASZEBIS

Sollte bei Ihnen der nicht qualifizierte „kurze“ Host-Name zuerst stehen, dann passen Sie bitte die Reihenfolge so an, dass der „lange“ Host-Name für diese Adresse zuerst aufgeführt wird.

SMTP Dienst konfigurieren

Schauen wir uns nun einmal an, wie man SMTP-Dienste auf dem System i konfiguriert. Dazu wählen wir im iSeries Navigator den Bereich „Netzwerk / Server /TCPIP“ mit einem Doppelklick aus.

Damit erscheint im rechten Hauptanzeigenbereich des iSeries Navigators eine Auflistung aller auf dem System i installierten TCP/IP-Dienste.

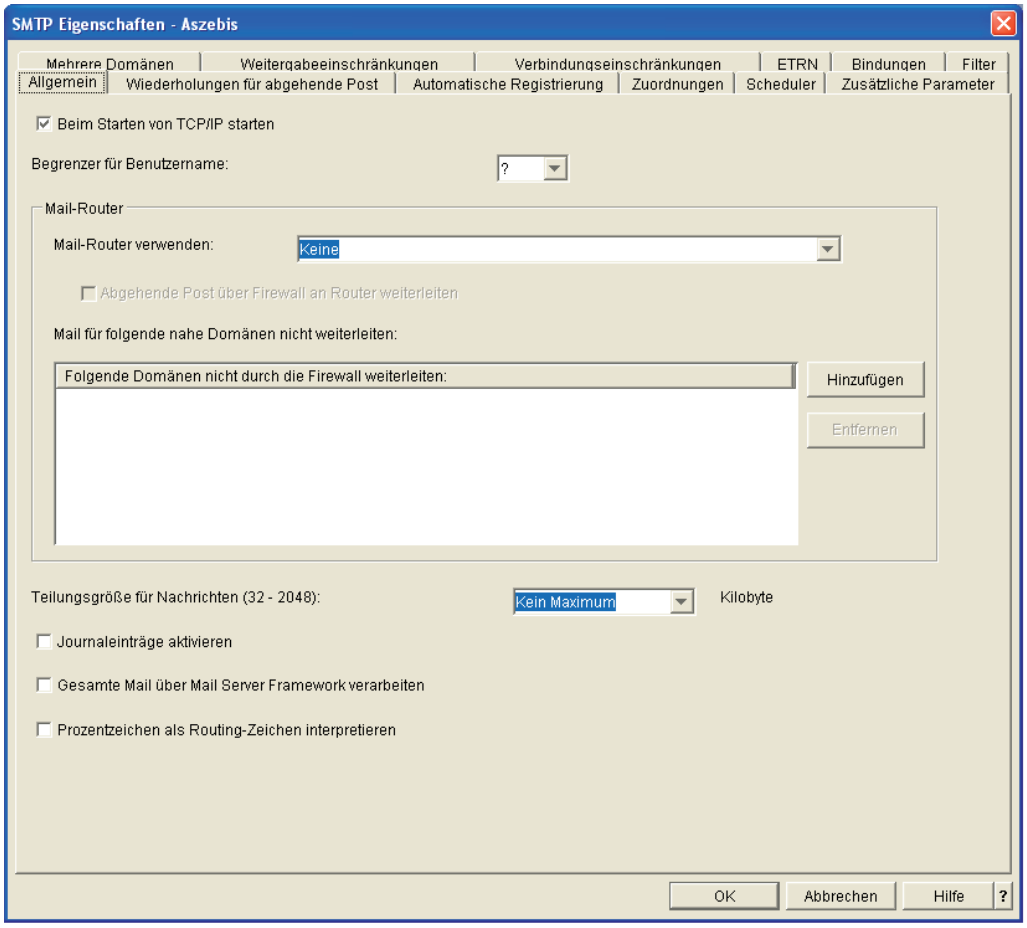


TCP/IP-Dienste

In der Auflistung finden wir unter anderem den SMTP-Eintrag, den wir mit der rechten Maustaste auswählen. Im darauf sich öffnenden Auswahlfenster selektieren wir den Eintrag „Eigenschaften“.

Damit öffnet sich der Definitionsbereich für den SMTP-Server.

5.4.2
Seite 10



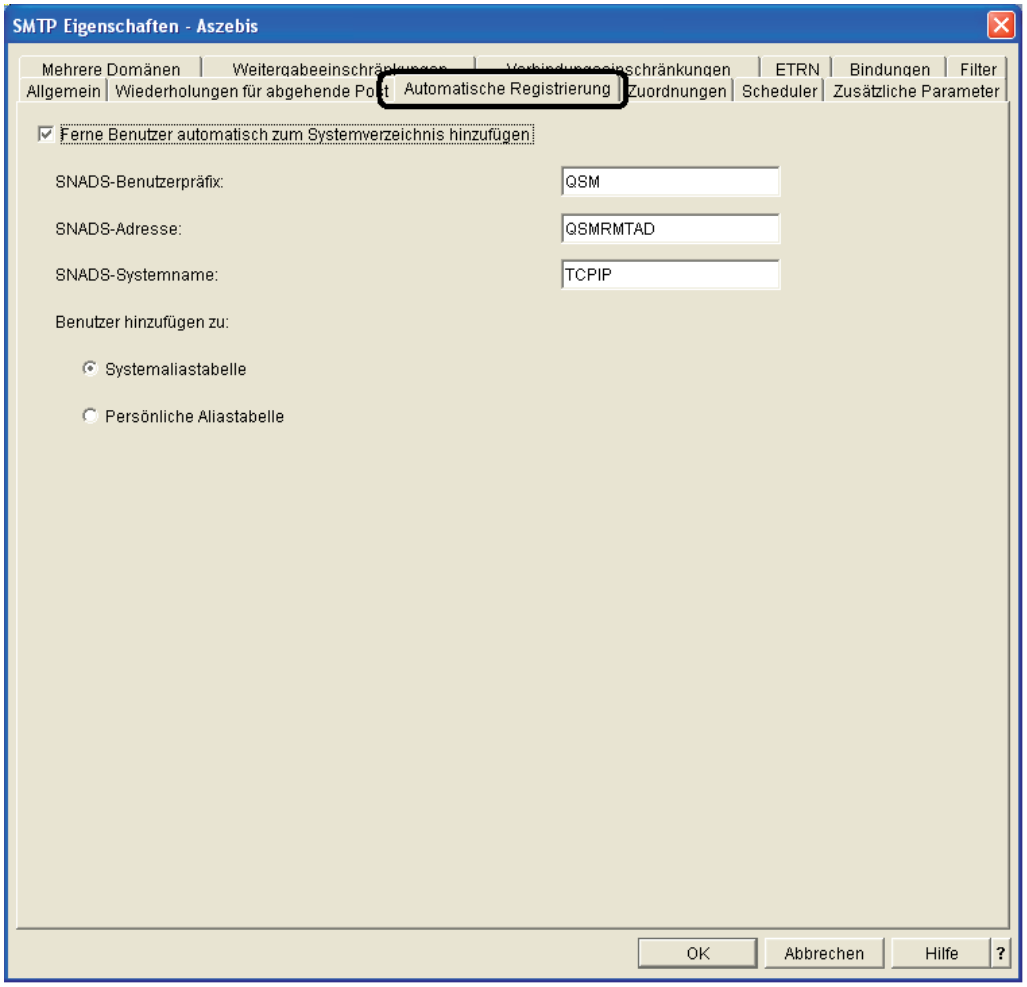
SMTP-Einstellungen

In diesem Fenster wählen wir zunächst den Abschnitt „Allgemein“. Prüfen Sie in dieser Anzeige die folgenden Einstellungen:

- Beim Starten von TCP/IP starten
  - Diese Option sollten Sie auswählen, wenn SMTP automatisch beim STRTCP mit hochgefahren werden soll. Diese Einstellung empfiehlt sich, wenn SMTP auf dem System i genutzt werden soll.
- Wenn Sie einen Mail-Router verwenden, dann geben Sie diesen im dafür vorgesehenen Feld ein.
- Im Bereich „Mail für folgende Domänen nicht weiterleiten“ tragen Sie die Domänen-Namen ein, die nicht durch die Firewall versendet werden sollen.

Sollten Sie den Befehl SNDDST (Verteilung senden) für das Übertragen und Empfangen von Mails verwenden, dann müssen neben den allgemeinen Einstellungen im SMTP-Bereich noch weitere Angaben getätigt werden. Diese sind erforderlich, wenn an Stelle der Internet-Adressierung SNADS-Adressen genutzt werden sollen. Um diese Einstellungen durchzuführen, wählen wir

den Tabellenbereich „Automatische Registrierung“, den Sie in der folgenden Abbildung finden:



Automatische Registrierung

In dieser Anzeige sollten Sie die Option „Ferne Benutzer automatisch zum Systemverzeichnis hinzufügen“ selektieren sowie die Auswahl „Benutzer hinzufügen zu:“ mit der Option „Systemaliastabelle“ auswählen.

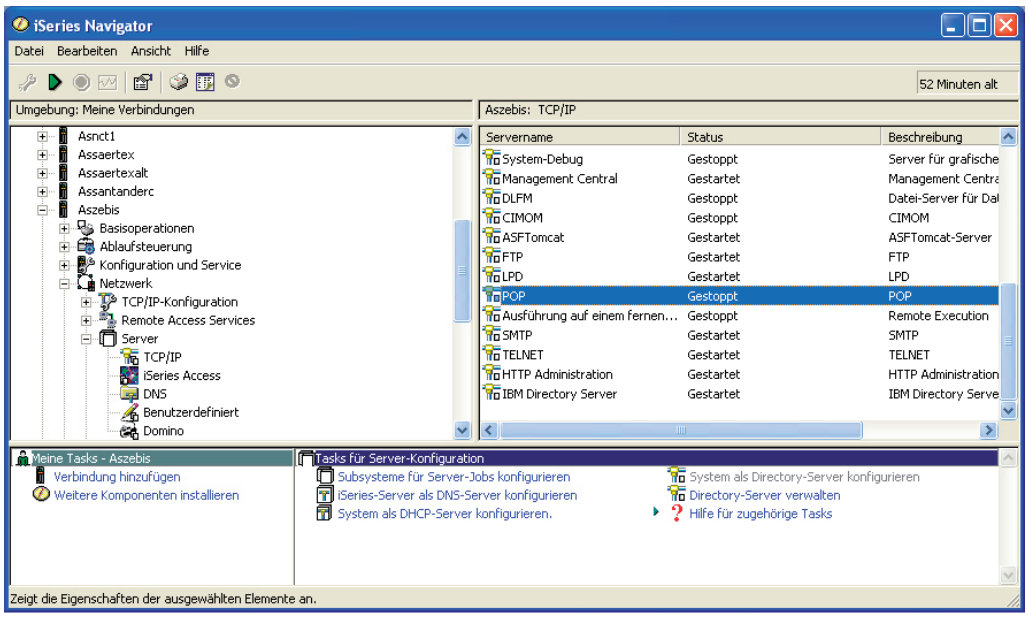
Damit sind bereits die erforderlichen Einstellungen für den SMTP-Bereich abgeschlossen.

Neben dem SMTP-Dienst empfiehlt sich der Einsatz des POP-Dienstes, der dann benötigt wird, wenn die zu versendenden Mails an einen POP-Client übertragen werden sollen.

5.4.2
Seite 12

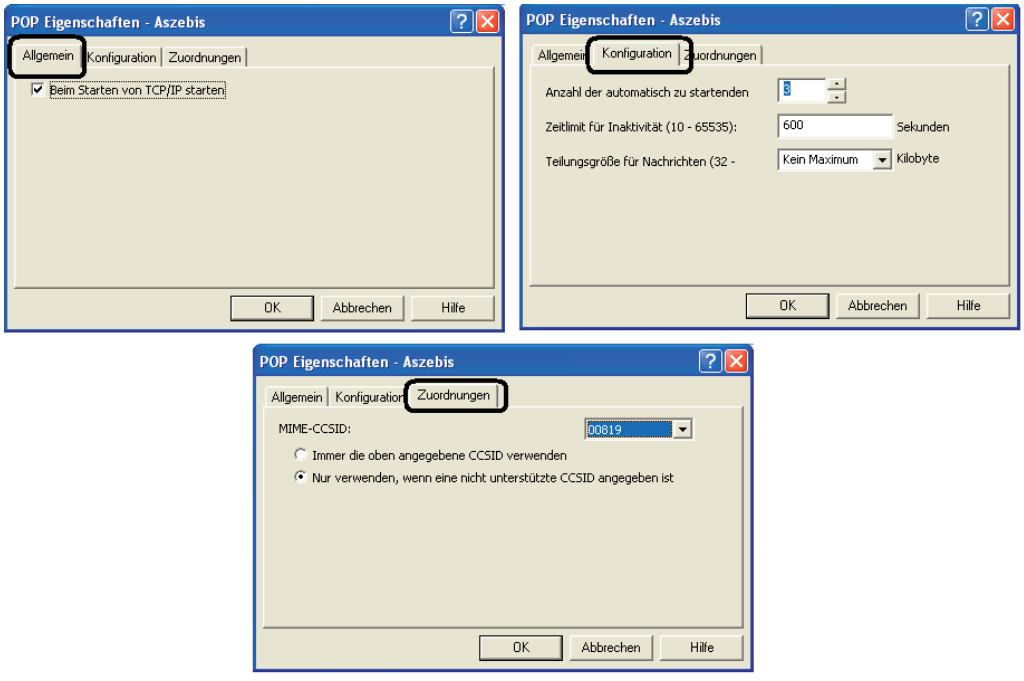
POP-Dienst konfigurieren

Auch dieser TPC/IP-Serverdienst wird mit dem iSeries Navigator definiert und konfiguriert. Dazu selektieren wir den POP-Eintrag im Bereich „iSeries Server / Netzwerk / Server / TCPIP“.



POP-Dienst auswählen

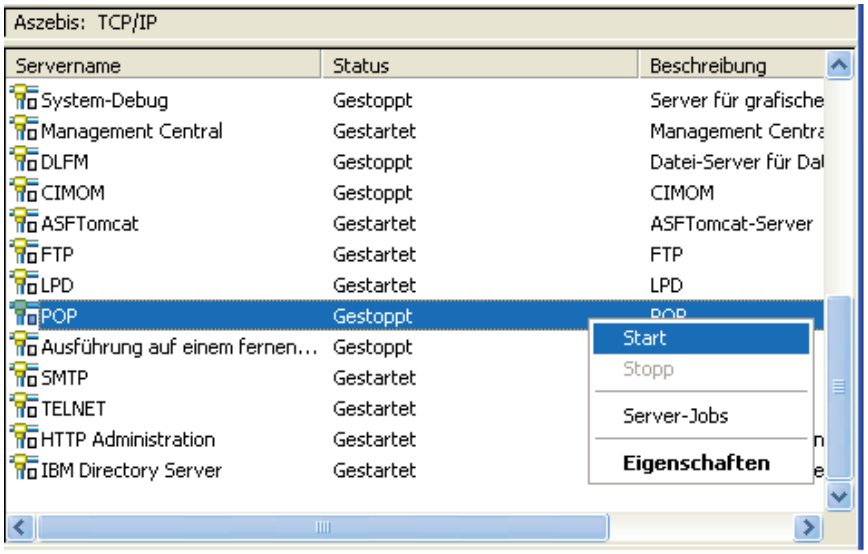
Es bietet sich – ähnlich wie bei der Konfiguration des SMTP-Bereichs – an, den POP-Serverdienst automatisch beim Hochfahren von TCP/IP mitzustarten. Diese Option können wir in den Eigenschaften des POP-Dienstes festlegen.



POP-Eigenschaften

Im Tabellenbereich „Konfiguration“ können wir die maximale Größe für einzelne Mails festlegen. In der Regel können die hier als Unterlassungswerte enthaltenen Einstellungen übernommen werden. Gerade bei Verwendung von internationalen Konfigurationen muss unter Umständen die Zuordnung der Zeichensätze angepasst werden. Damit lässt sich sicherstellen, dass Zeichenumsetzungen erfolgreich durchgeführt werden.

Starten Sie nach den geänderten Konfigurationseinstellungen den POP-Dienst, indem Sie ihn in der Diensteauflistung mit der rechten Maustaste selektieren und die Option „Starten“ wählen.



Starten des POP-Servers

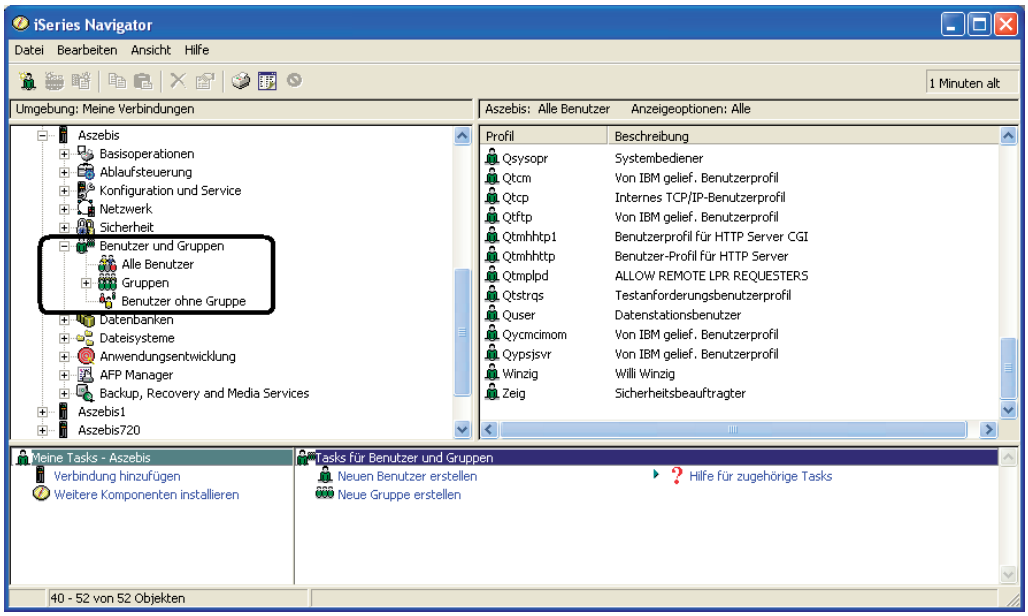
Damit sind die wesentlichen allgemeinen Einstellungen für das Versenden und Empfangen von Mails native mit dem System i definiert.

Als Nächstes gilt es, die Benutzer zu definieren, die am Mail-Verfahren teilnehmen sollen. Die Basis dafür bildet das i5/OS-Benutzerprofil. Schauen wir dazu als einfaches Beispiel die Konfigurationsschritte an, die benötigt werden.

5.4.2
Seite 14

Benutzer für Mail konfigurieren

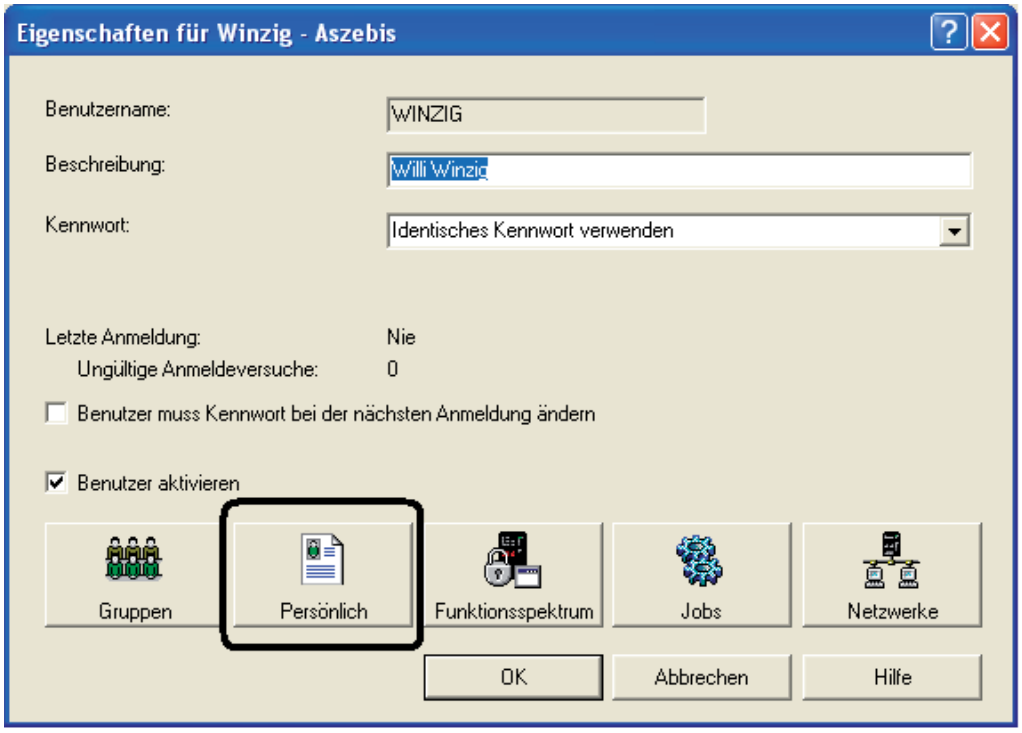
Die Konfiguration bietet sich unter Verwendung des iSeries Navigators an. Dort finden wir einen separaten Definitionsbereich, mit dem wir Benutzer und Gruppen auf dem System i verwalten können.



Benutzerverwaltung im iSeries Navigator

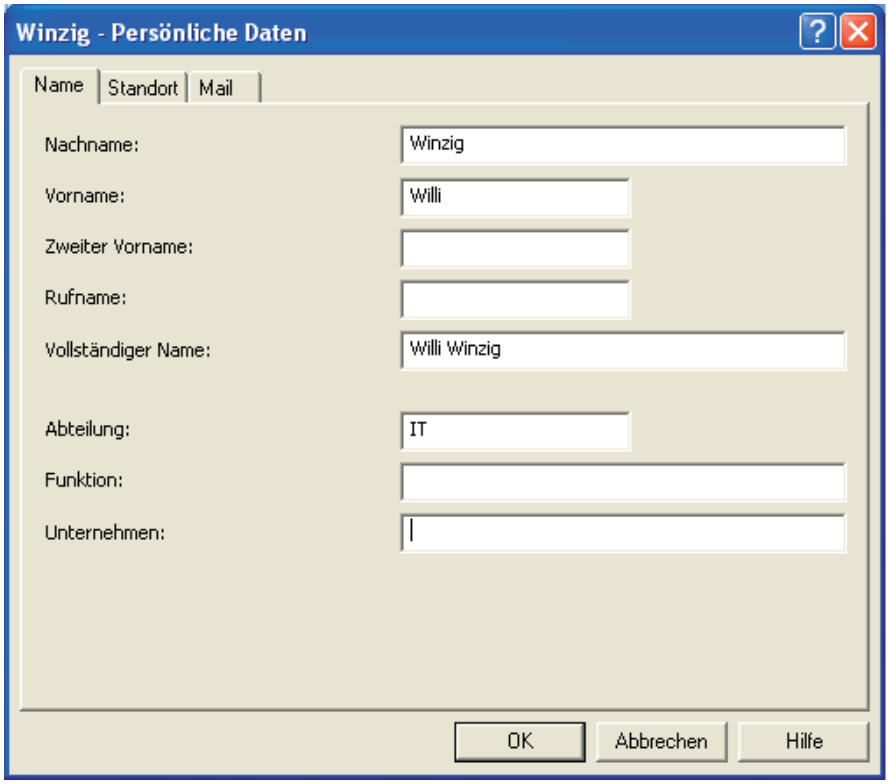
Wählen Sie im iSeries Navigator den Bereich „Benutzer und Gruppen“ und selektieren Sie in der rechten Auflistung den Eintrag, der verwaltet werden soll. Sie können aber auch mit der rechten Maustaste die Option „Neuer Benutzer“ wählen, wenn Sie einen vollkommen neuen Benutzer anlegen möchten. Unabhängig davon, ob ein bestehender Benutzer verwaltet, oder ein neuer Benutzer hingefügt werden soll, die Konfigurationseinstellungen für die Mail-Unterstützung sind gleich.

Ich markiere nun einen bestehenden Benutzereintrag mit der rechten Maustaste und wähle die Option „Eigenschaften“. Damit gelangen wir in folgende Anzeige:



Eigenschaften des Benutzers

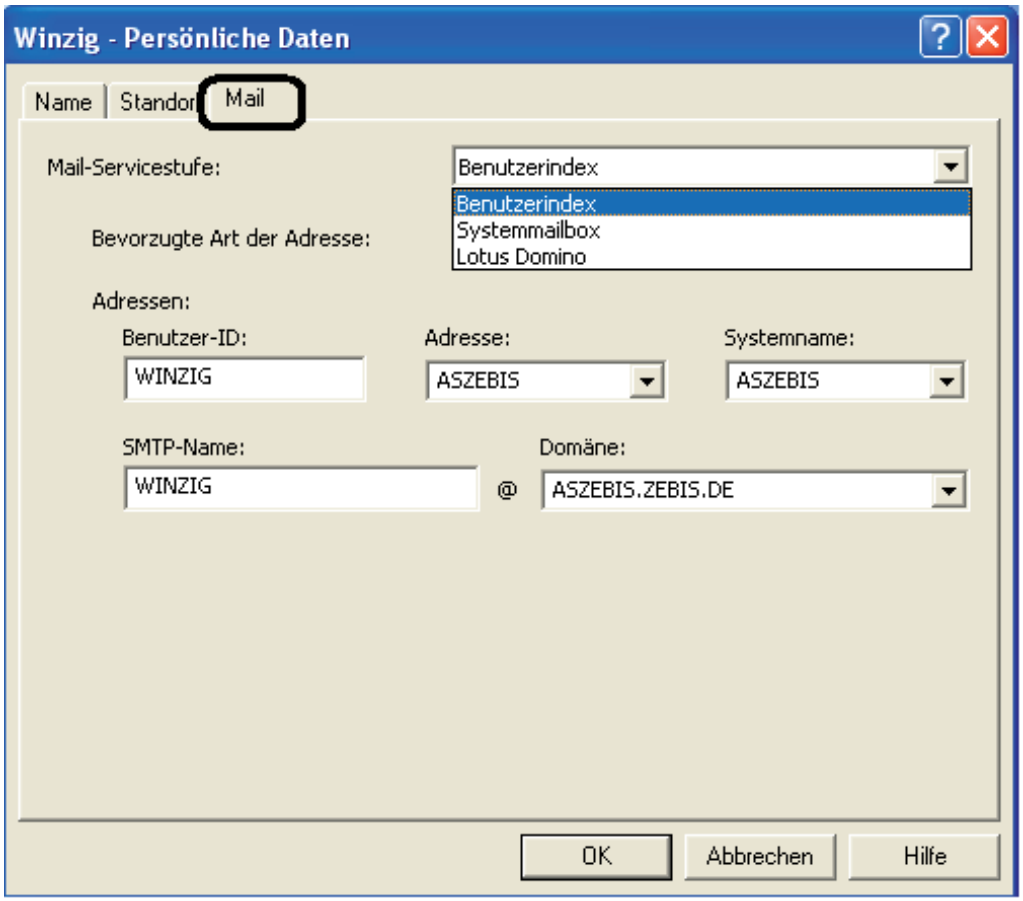
In den allgemeinen Eigenschaften sind diverse Informationen enthalten, die sich zwar nicht auf die Mail-Funktionen auswirken, die aber dennoch korrekt gefüllt sein sollten.



Namensangaben

5.4.2
Seite 16

Der Bereich „Mail“ ist für die Mail-Funktionen wichtig. Dort geben wir beispielsweise an, welche Mail-Adresse für das i5/OS-Benutzerprofil zu verwenden ist.



Mail-Angaben

Wie Sie anhand der Abbildung erkennen können, lassen sich verschiedene Mail-Servicestufen selektieren. Dahinter verbergen sich unterschiedliche „Standards“ – wie zum Beispiel die Integration von Lotus Domino, der als Ziel-Mail-System angegeben werden kann. Geben Sie hier das Mail-System an, das für den Benutzer verwendet wird. Handelt es sich um einen i5/OS-Benutzer, der lediglich die nativen Mail-Funktionen nutzen muss, dann geben Sie hier „Systemmailbox“ an.

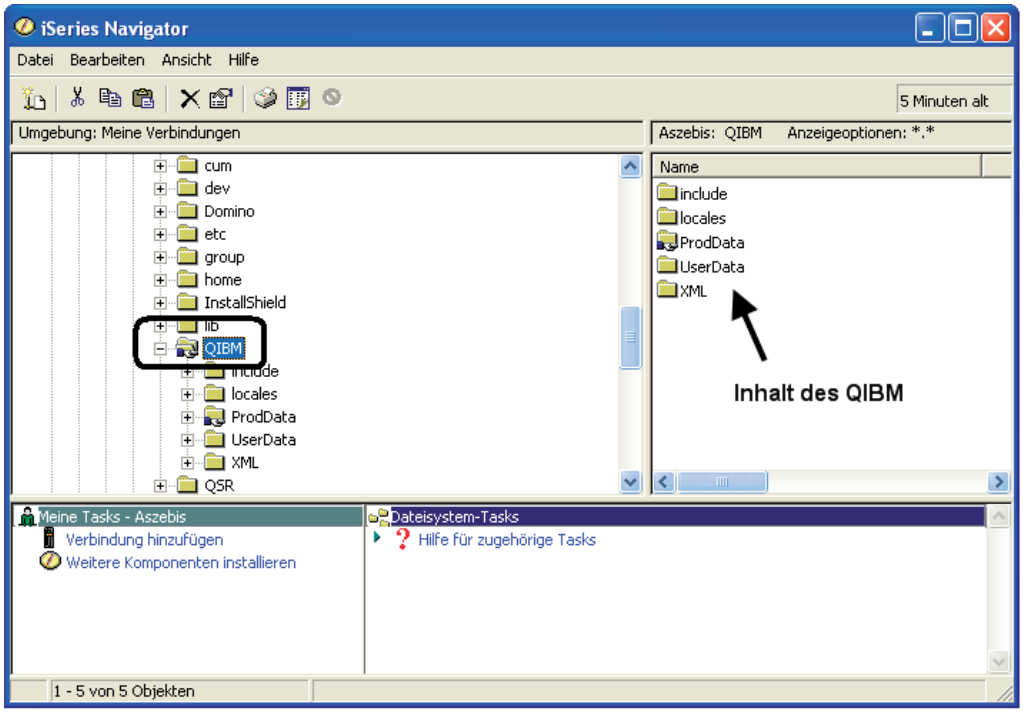
iSeries NetServer

In diesem Kapitel möchte ich Ihnen einen weiteren Standard-Server-Dienst auf dem System i vorstellen. Es handelt sich um den iSeries NetServer, der die System i-Welt mit der Windows-Welt verbindet.

Dieser Server-Dienst, der im TCP/IP-Bereich des Systems zu finden ist, dient Windows-Client- und Server-Systemen als Basis für den Zugriff auf die Netzlaufwerke, die sich auf dem System i befinden, sowie den darin enthaltenen Verzeichnissen und Dateien. Außerdem werden Windows-Betriebssysteme damit in die Lage versetzt, Ausgabewarteschlangen anzusprechen.

Zugriffe auf die Inhalte der Dateien und Verzeichnisse mit PC-Werkzeugen – wie beispielsweise dem Windows-Explorer – basieren auf dem Einsatz des NetServers. Der Vorteil beim NetServer-Einsatz für den Zugriff auf die System i-Inhalte, die sich in der Verzeichnisstruktur befinden, liegt darin, dass nach Konfiguration der NetServer-Funktionen auf dem System (mit 5250-Befehlen und iSeries-Navigator-Werkzeugen) keine spezielle Software auf Client-Seite benötigt wird, um auf die Freigabe des NetServers zuzugreifen.

Basis für diese Funktion bildet das Verzeichnis QIBM, das als Bestandteil des integrierten Dateisystems von i5/OS bereitgestellt wird.



QIBM im iSeries Navigator

Die gemeinsame Dateiverwaltung und der Druckereinsatz zwischen Windows und i5/OS bilden in vielen Unternehmen eine wichtige Voraussetzung für die Bewältigung der täglichen Arbeit. So liefert der iSeries NetServer bei-

spielsweise auch die Möglichkeit, mit Hilfe der Windows-Suchfunktionen den NetServer in einem Netzwerk zu finden.

Während die iSeries NetServer-Funktionen zwar unter dem Oberbegriff „TCP/IP-Server" geführt werden, ist deren Funktionsumfang nicht im Standardumfang des Lizenzprogramms 5722-TC1 enthalten. Dies wird mit einer separaten Option des 5722-SS1 abgebildet. Wenn Sie also die NetServer-Funktionen nutzen möchten, dann benötigen Sie auf Ihrem System i die beiden Lizenzprogramme beziehungsweise Lizenzprogrammoptionen:

- 5722-TC1 (TCP/IP)
- 5722-SS1 Option 12 (Host Servers)

Zwar stehen auf der 5250-Oberfläche einige Befehle für die Administration des NetServers zur Verfügung, aber für spezielle Funktionen sind die Tools des im Lizenzprogramm „iSeries Access for Windows" enthaltenen iSeries Navigators erforderlich.

Der NetServer-Dienst wird, wenn alle erforderlichen Software-Komponenten installiert sind, als eigener TCP/IP-Server-Dienst auf dem System ausgeführt. Bevor Sie die Datei- und/oder Druckerfreigaben nutzen können, muss der iSeries NetServer gestartet sein. Dieser Start kann – je nach Vorgabe – automatisch mit dem Starten des TCP/IP-Bereichs oder manuell über den Befehl STRTCPSVR *NETSVR erfolgen.

In der Regel arbeitet der NetServer zuverlässig. Er startet bei neueren Release-Versionen des Betriebssystems bereits beim Aktivieren von TCP/IP automatisch – je nach Benutzervorgaben.

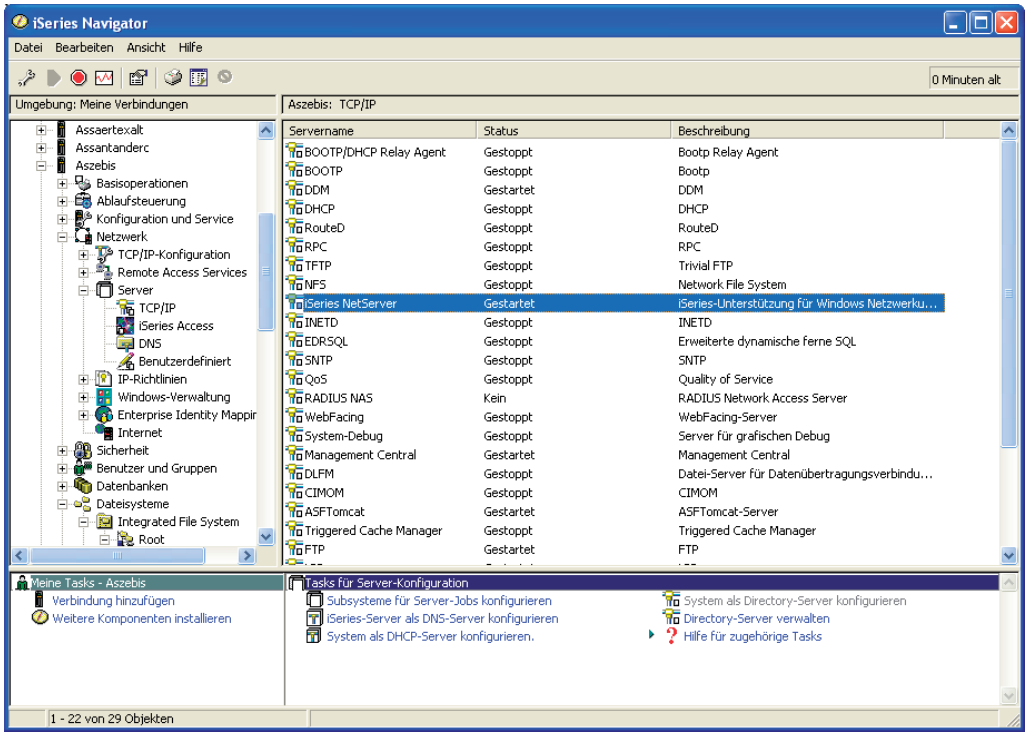
Allerdings kann es aus Sicherheitsüberlegungen auch sinnvoll, wenn nicht gar erforderlich sein, dass die Standardeinstellungen im NetServer-Bereich angepasst werden. Stellt man sich in diesem Zusammenhang zum Beispiel vor, dass ein Benutzer direkt oder indirekt über eine Anwendung auf den Inhalt des Dateisystems zugreifen muss, er aber über keine i5/OS-Benutzerregistrierung verfügt, dann muss ein Gastprofil verwendet werden, mit dem der Basiszugriff auf die Dateien und deren Inhalte möglich wird. Die Einrichtung und Aktivierung der Gastprofilzugriffe sollte behutsam erfolgen, da dies unter Umständen eine Sicherheitslücke darstellen kann.

Aber schauen wir uns zunächst einmal die Integration und Basisverwaltung des iSeries NetServers genauer an.

Die folgende Abbildung zeigt die Übersicht der TCP/IP-Dienste und den gestarteten iSeries NetServer, der auch mit der rechten Maustaste aus dieser Ansicht heraus gestartet, gestoppt und verwaltet werden kann.

Anmerkung

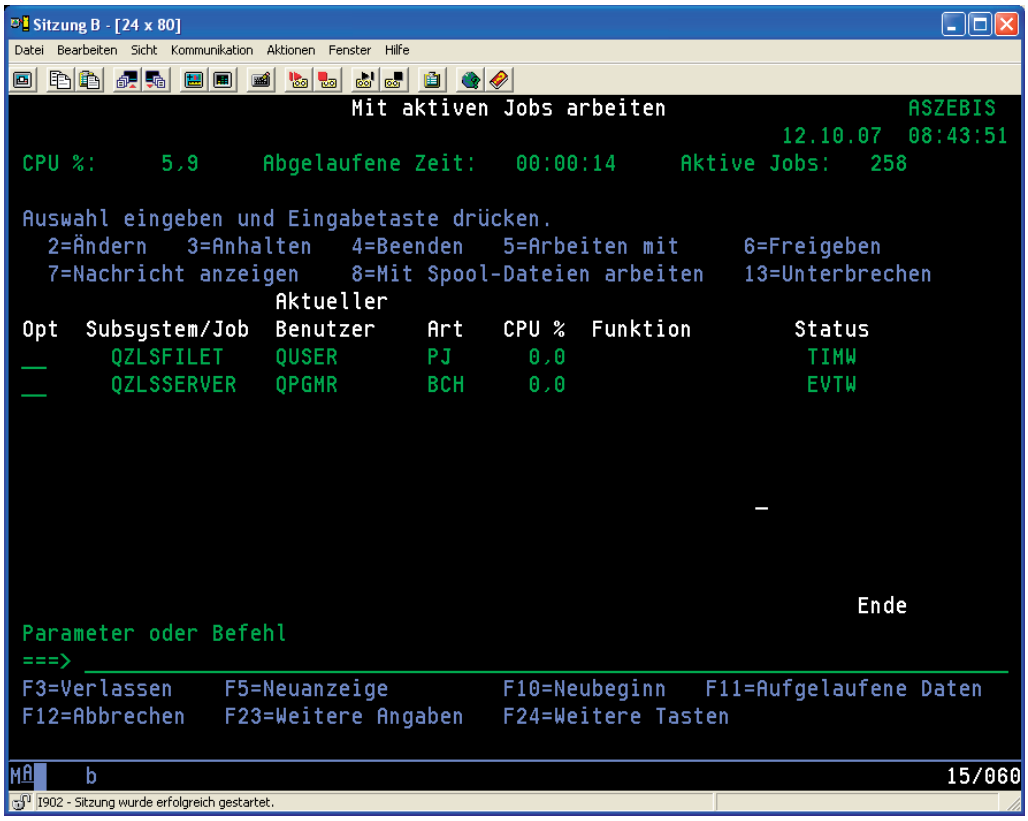
Sollten Sie den NetServer noch nie genutzt haben, dann führen Sie den Start bitte nicht aus, sondern konfigurieren den NetServer zunächst, wie es im Verlauf dieses Kapitels beschrieben wurde.



Server-Verwaltung

Dass der NetServer gestartet worden ist, kann man auch anhand der aktiven Jobs ermitteln.

5.4.3
Seite 4



QZLSSERVER

Ohne auf die einzelnen Jobs, die im Subsystem QSERVER ausgeführt werden, innerhalb des NetServer-Bereichs einzugehen, sind folgende Jobs für die Ausführung und den Einsatz des NetServers wichtig:

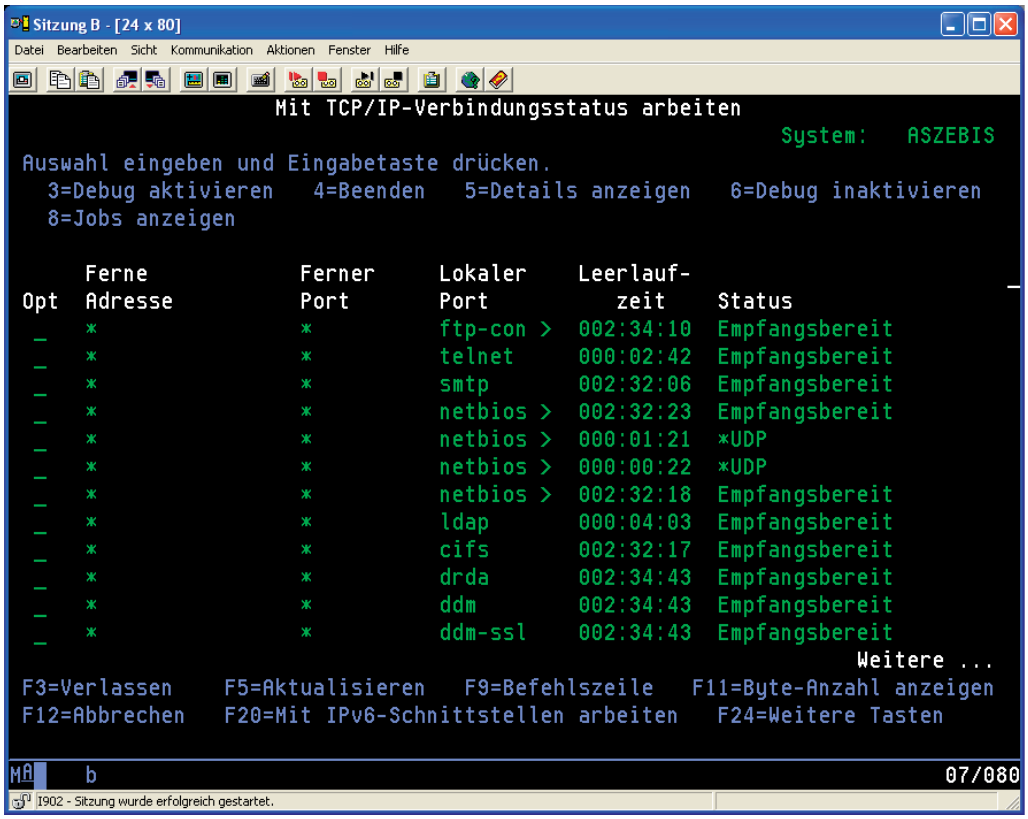
- QZLSSERVER
  Dies ist der Hauptjob, der für den NetServer ausgeführt wird. Dieser Job befindet sich innerhalb des Subsystems QSERVER und muss für die Verwendung der NetServer-Funktionen gestartet sein. Der Start des Jobs erfolgt im Normalfall über den allgemeinen Startbefehl des iSeries NetServers (STRTCPSVR *NETSVR).
- QZLSFILET oder QZLSFILE
  Dieser Job wird für jeden Benutzer gestartet, der die Dateifreigabe des NetServers aktiv nutzt.
- QNPSERVD
  Dieser Job steht für den Network PrintServer. Dieser Job wird im Subsystem QSYSWRK ausgeführt.

Anmerkung

Es ist durchaus möglich, die Jobs des iSeries NetServers in einem vom QSERVER abweichenden Subsystem auszuführen. Das kann auch aus administrativer Sicht Sinn machen, um zum Beispiel Clients einem anderen Subsystem, in dem die NetServer-Funktionen ausgeführt werden, zuzuordnen.

Eine weitere Kontrolle, die sich dann anbietet, wenn der NetServer beziehungsweise dessen Funktionen nicht korrekt ausgeführt werden, ist der Befehl NETSTAT *CNN. Dieser liefert als Ergebnis eine Anzeige, wie sie auch in der folgenden Abbildung zu sehen ist. Darin sind folgende Einträge von Bedeutung:

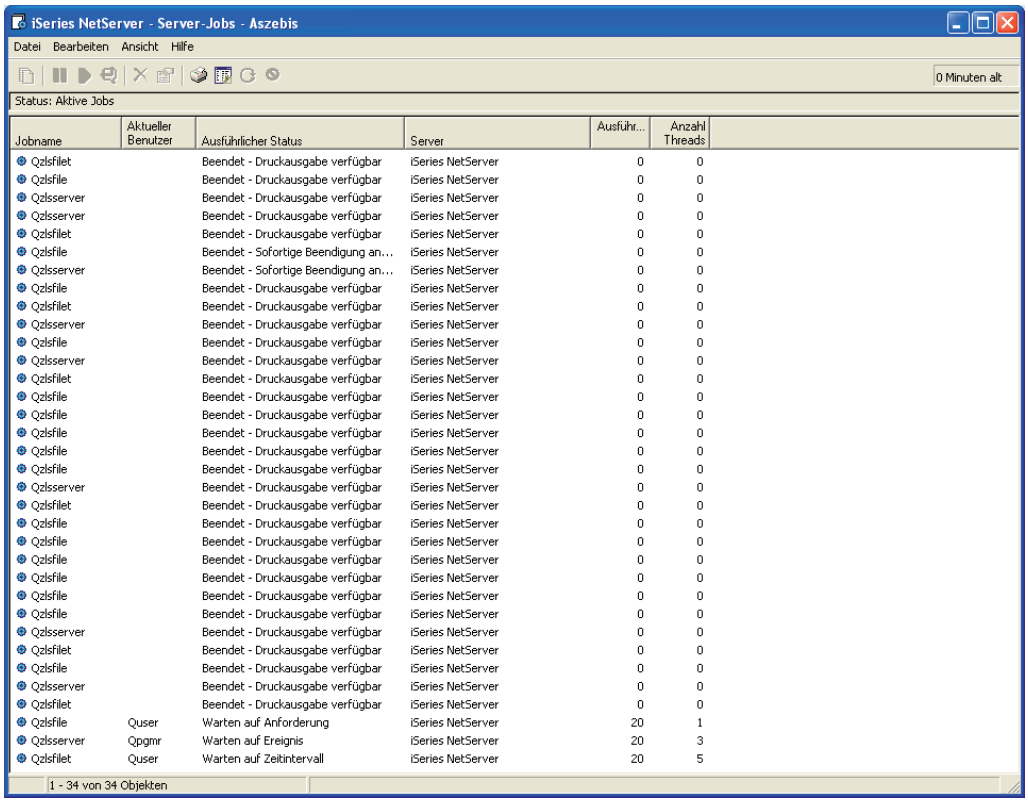
```
** netbios>   002:32:23 Empfangsbereit
** netbios>   000:01:21 *UDP
** netbios>   000:00:22 *UDP
** netbios>   002:32:18 Empfangsbereit
** cifs>   002:32:17 Empfangsbereit
```



NETSTAT CNN

5.4.3
Seite 6

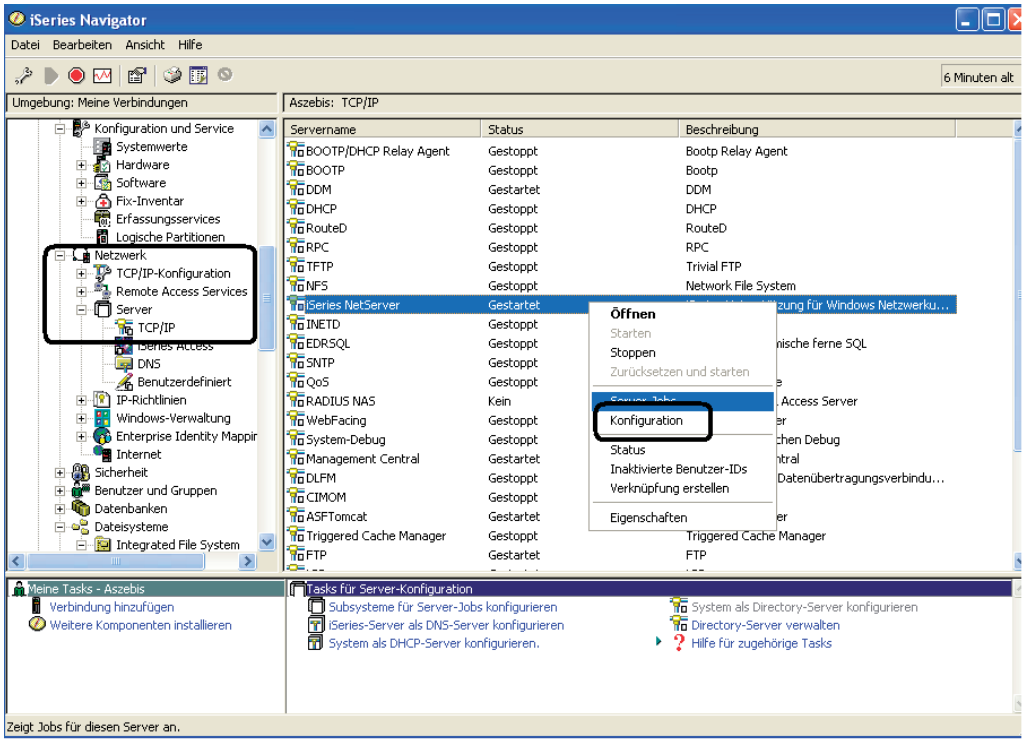
Natürlich lassen sich die Jobs des NetServers auch mit dem iSeries Navigator verwalten. Der Vergleich zeigt eine übersichtliche Auflistung der Jobs, die im Zusammenhang mit Drucker- und Dateifreigaben stehen.



NetServer-Jobs

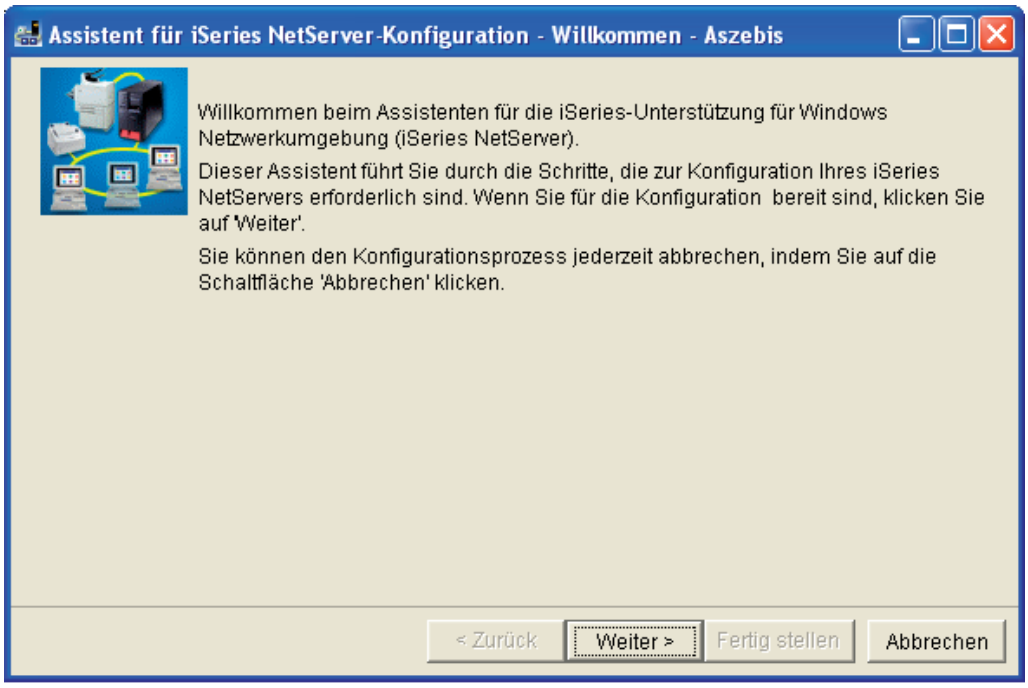
Ebenso wie die Auflistung der NetServer-Jobs ist die Verwaltung der Einstellungen des NetServers unter Verwendung des iSeries Navigators einfacher als das Einsetzen der auf der 5250-Ebene zur Verfügung stehenden Befehle. Denn der iSeries Navigator führt durch alle für die Konfiguration erforderlichen Teilbereiche.

Um die Konfiguration des NetServers zu erstellen oder zu ändern, wählen wir im iSeries Navigator den Bereich „Netzwerk → Server → TCP/IP“ mit einem Doppelklick aus. In der Auflistung der Server-Dienste markieren wir den Eintrag „iSeries NetServer“ mit der rechten Maustaste und selektieren die Auswahl „Konfiguration“.



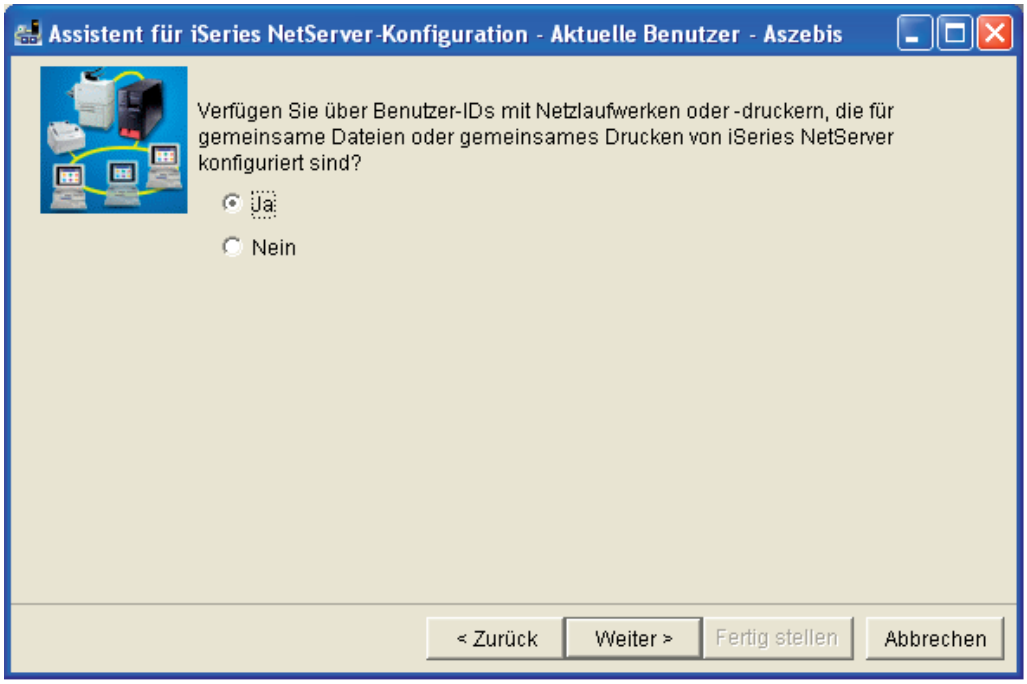
Konfiguration starten

Mit dem Start der Konfiguration initiieren wir den Assistenten für die Net-Server-Konfiguration; er wird uns durch die einzelnen Schritte der möglichen Einstellungen führen. Mit einem Klick auf die Schaltfläche „Weiter“ wird der Konfigurationsvorgang gestartet.



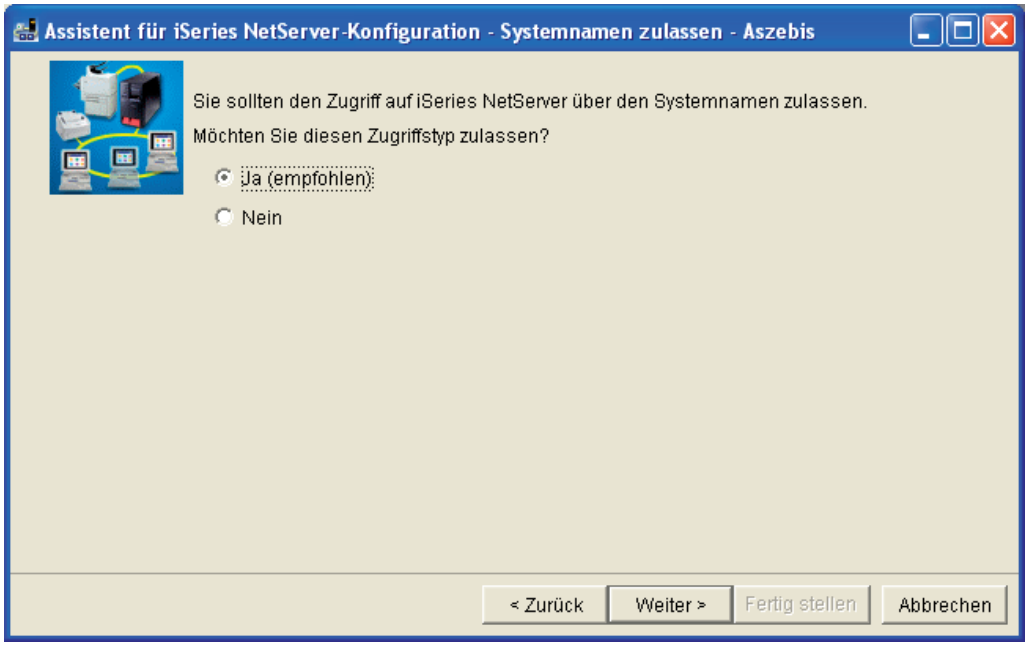
Konfigurationsstart mit iSeries Navigator

5.4.3
Seite 8



Freigaben

Es ist beinahe Standard, dass der Systemname des Systems i als Zugriffsname für den NetServer genutzt wird. Dieser „Standard“ kann jedoch in seltenen Fällen zu Komplikationen führen. In älteren Betriebssystemversionen durfte der Name für den NetServer-Zugriff nicht gleich der Systemnamen sein. Deshalb wurde in der Regel der Systemname (aus Anzeige DSPNETA) mit einem anführenden „Q“ versehen und als Name für den NetServer-Zugriff verwendet. Diese Einstellung ist jedoch kein Muss; sie kann nach Belieben – allerdings unter Berücksichtigung einiger Einschränkungen, die auch im Zusammenhang mit älteren Windows-Betriebssystemversionen stehen – angepasst werden.



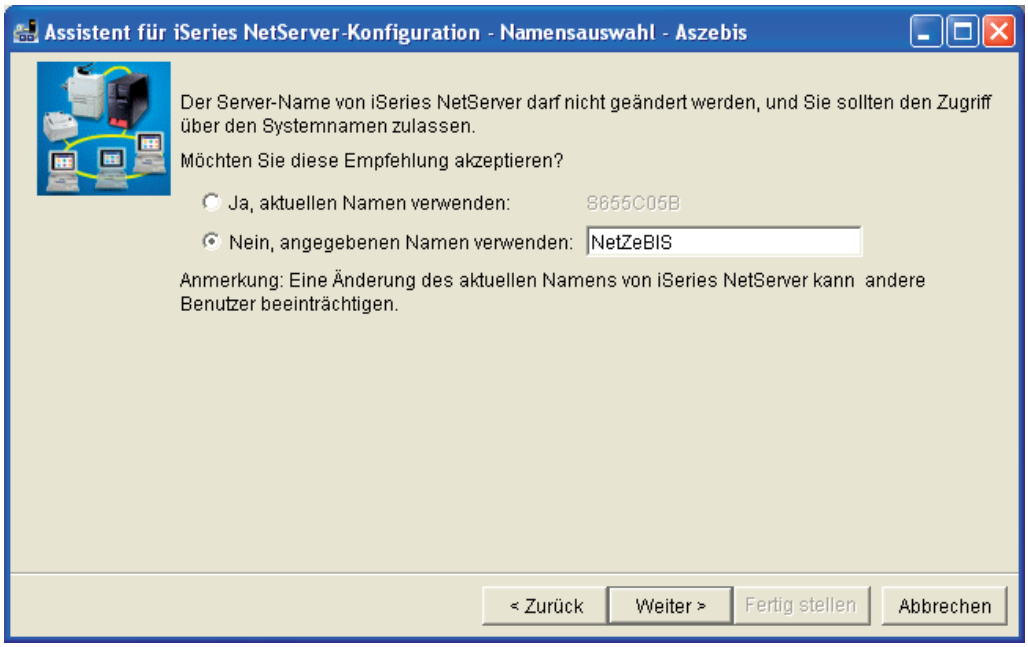
Freigabe über Systemnamen

Anmerkung

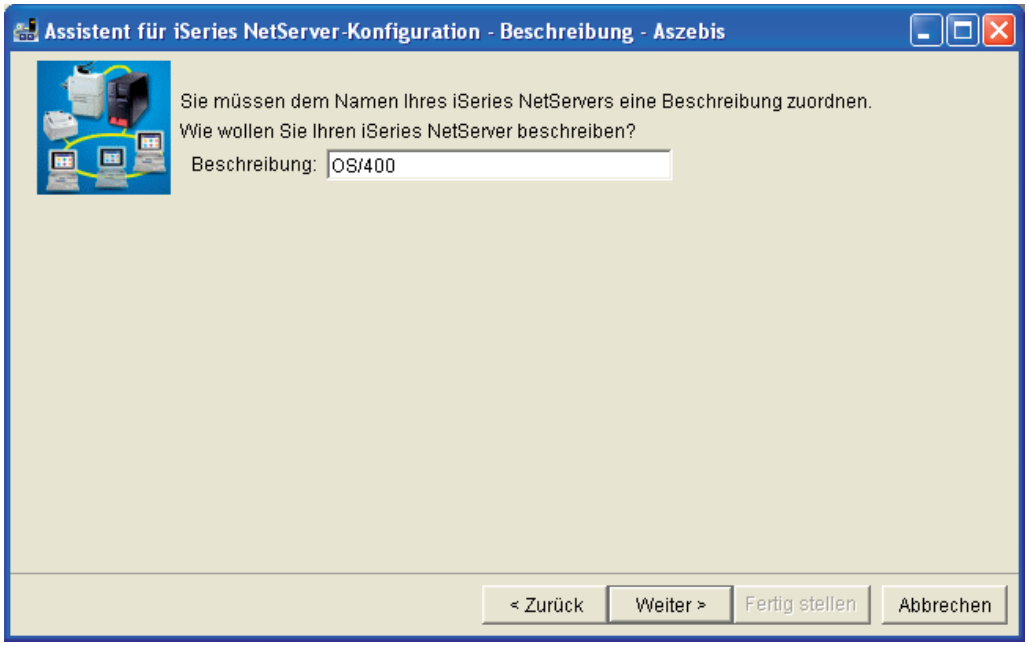
Nachträgliche Änderungen am Namen für NetServer-Zugriffe können zu Problemen führen. Deshalb sollten Sie bereits bei der Erstkonfiguration einen Namen verwenden, der Bestand hat und sich nach Möglichkeit nicht verändert.

Wenn Sie einen abweichenden Namen für den Zugriff verwenden möchten, dann wählen Sie in dieser Anzeige die Option „Nein“ und bestätigen Sie die Auswahl mit einem Klick auf „Weiter“.

5.4.3
Seite 10

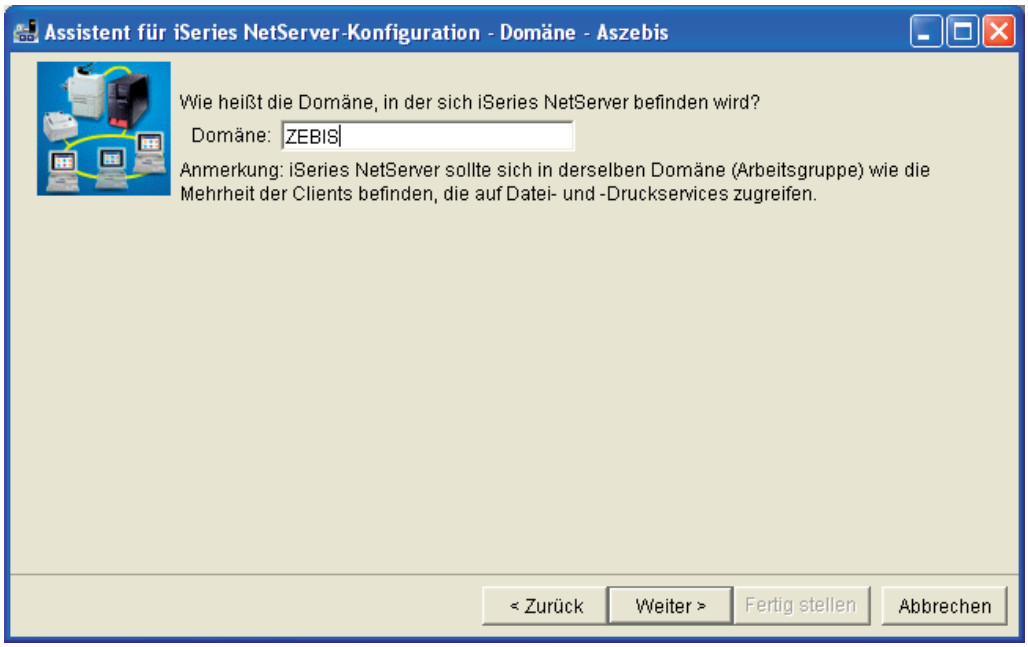


Freigabename



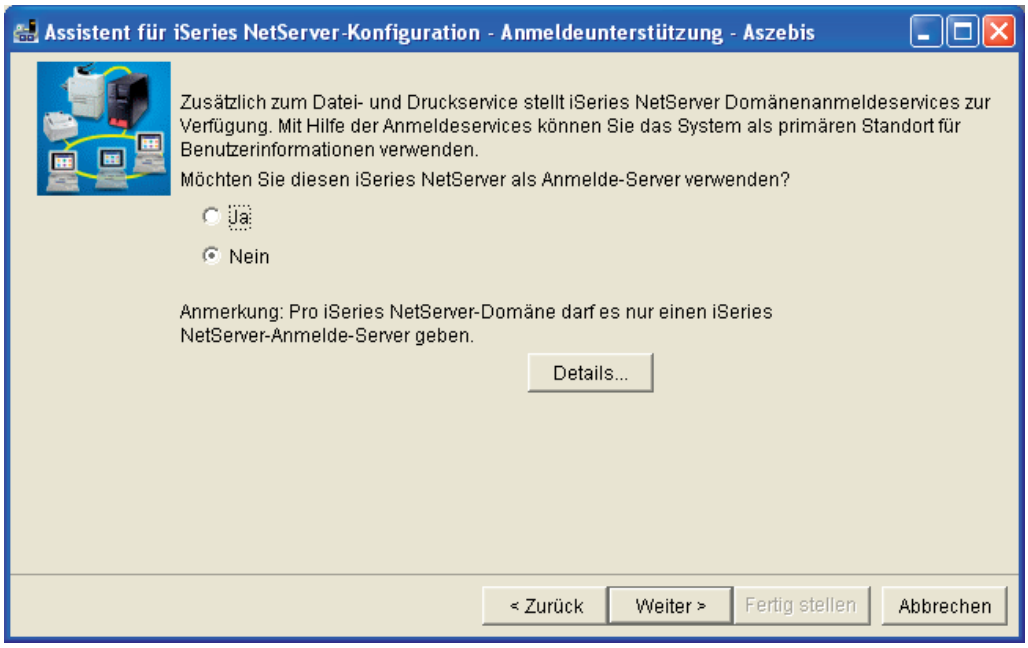
Beschreibung NetServer

Damit Freigaben und auch Suchfunktionen in den Microsoft-Betriebssystemen mit den Dateifreigaben des NetServers arbeiten können, sollte sich der NetServer nach Möglichkeit in derselben Domäne befinden, in der auch die Microsoft-Clients definiert worden sind. Geben Sie in der folgenden Anzeige den Namen dieser Domäne ein.



Domänenangaben

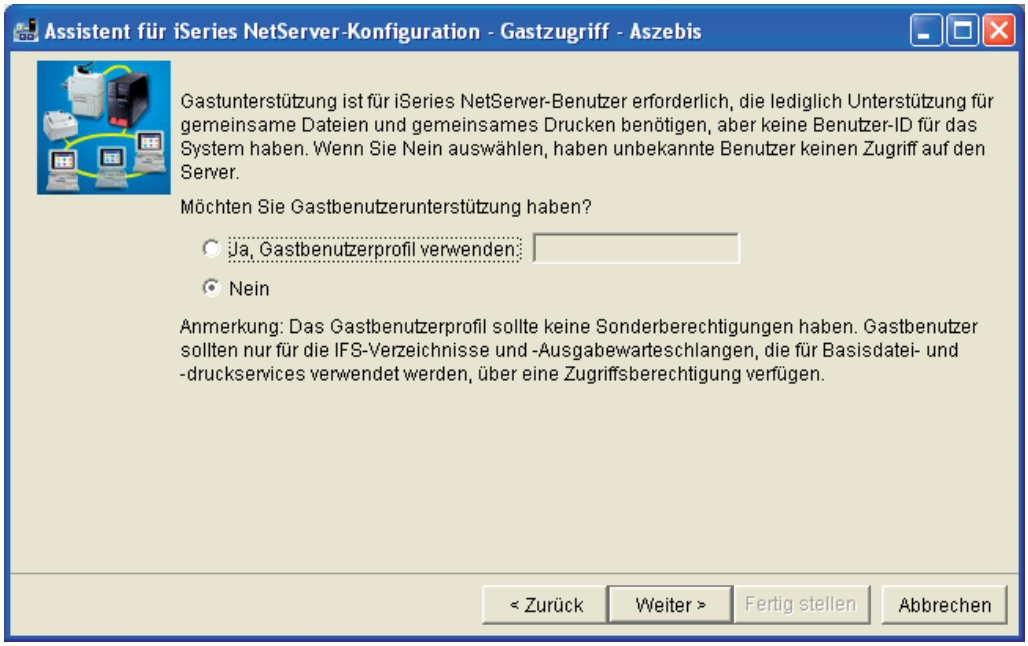
Optional lässt sich der NetServer auch als Anmelde-Server für die Domäne verwenden. Dabei sind jedoch einige Bedingungen zu beachten. Beispielsweise müssen Sie sicherstellen, dass es pro Domäne nur einen Anmelde-Server gibt. Anderenfalls sind Freigaben beziehungsweise Zugriffe über das Windows-Netzwerk nicht fehlerfrei möglich.



Anmeldevorgaben

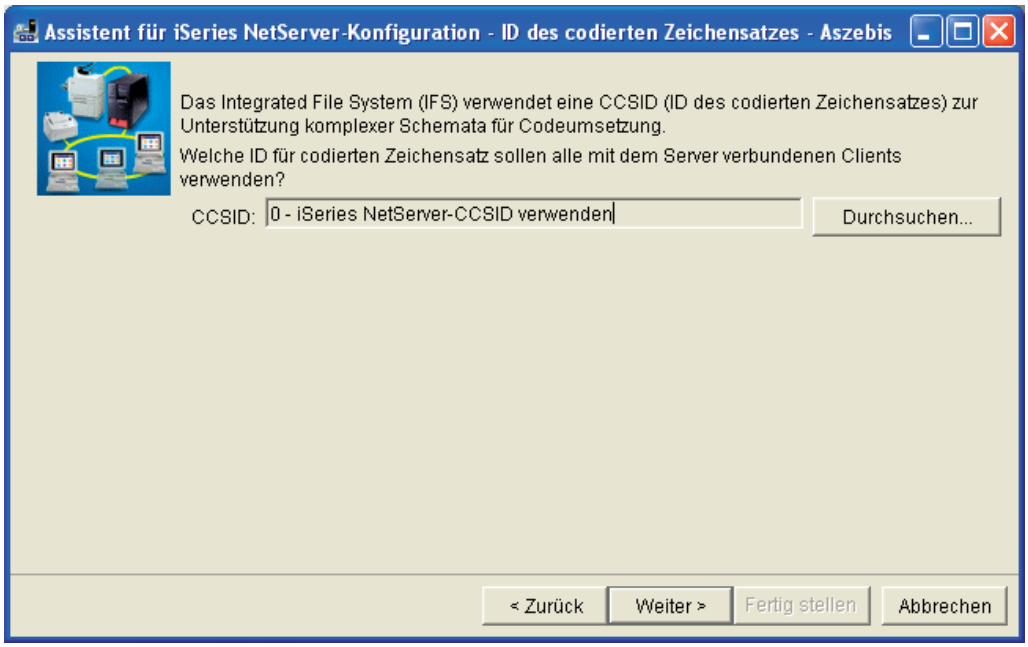
5.4.3
Seite 12

In der nächsten Abbildung können Sie sehen, wie wir einen Bereich öffnen, der es erlaubt, Unberechtigten Zugriff auf freigegebene Dateien und Drucker zu ermöglichen (auf keine weiteren Systemressourcen!). Ein Gastprofil beziehungsweise die Möglichkeit des Zugriffs mittels Gastprofil ist dann erforderlich, wenn Anwendungen mit NetServer-Freigaben arbeiten müssen, bei denen kein i5/OS-Benutzer verwendet werden soll. Die Hinweismeldung in der folgenden Abbildung weist auf mögliche Zugriffe „unbekannter Benutzer" hin. Bedenken Sie die potentielle Gefahr und lassen Sie nicht ohne Weiteres einen Zugriff mit einem Gastprofil zu!



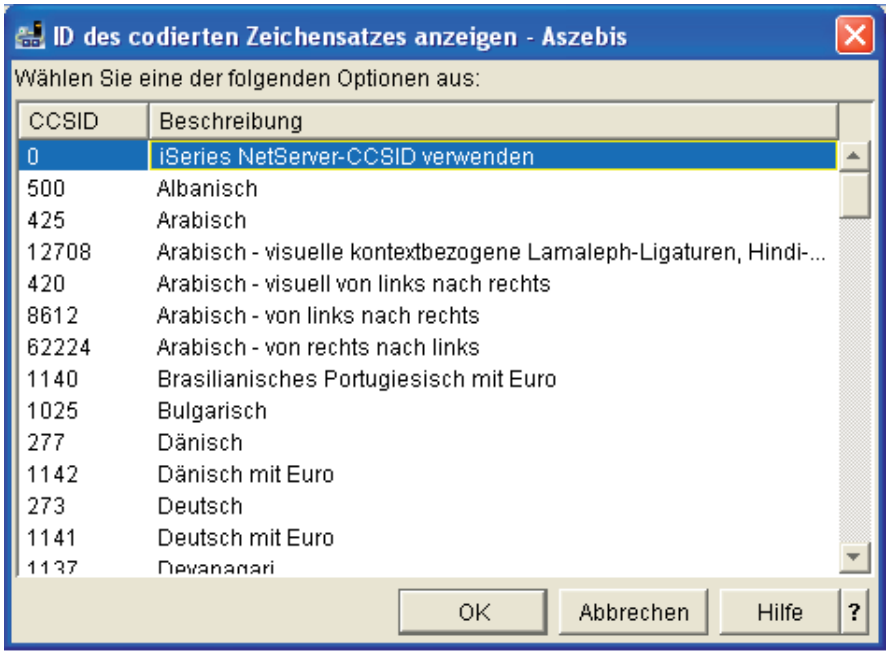
Gastprofil zulässig

Trotz aller Warnhinweise sollte aber auch nicht verschwiegen werden, dass ein Gastbenutzer nur über eine sehr begrenze Berechtigung und einen ebenso begrenzen Funktionsumfang verfügt. Wenn Sie ein Gastprofil verwenden möchten, dann geben Sie es, wie in der Abbildung gezeigt, als „Gastzugriff" im dafür vorgesehenen Feld ein.



Zeichensatz vorgeben

Das leidige Thema der unterschiedlichen Zeichensätze setzt sich auch bei Verwendung der NetServer-Funktionen fort. Heterogene Lösungen in unterschiedlichen PC-Anwendungen, Betriebssysteme, die in einem Netzwerk vorkommen, sowie einige weitere Kriterien erfordern bestimmte Einstellungen der Zeichensätze, die für die Darstellung der gespeicherten Informationen verantwortlich sind.



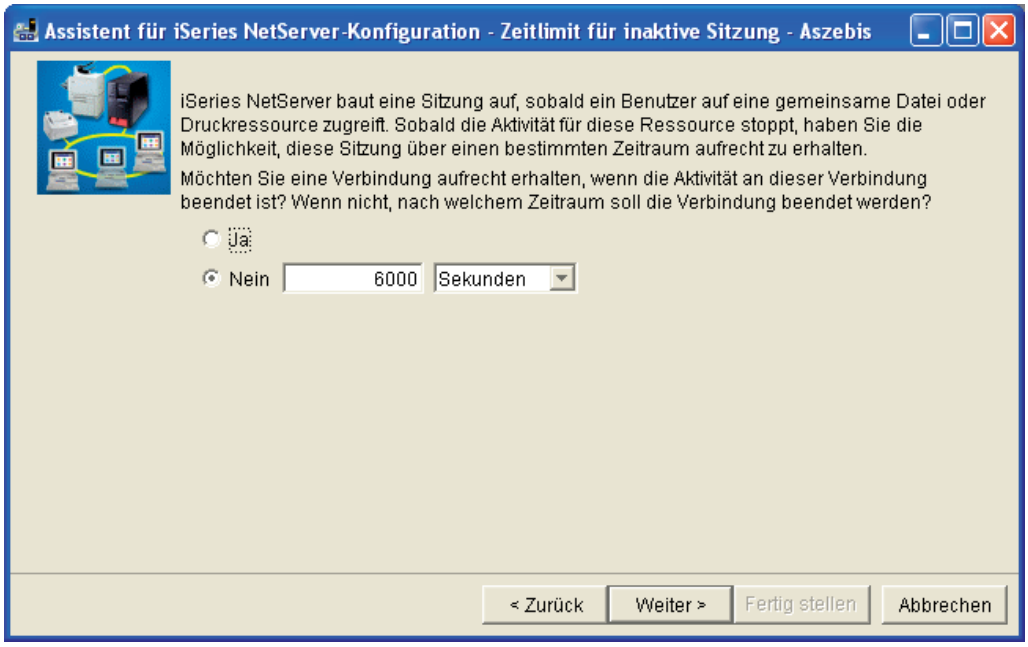
Mögliche Zeichensätze

5.4.3
Seite 14

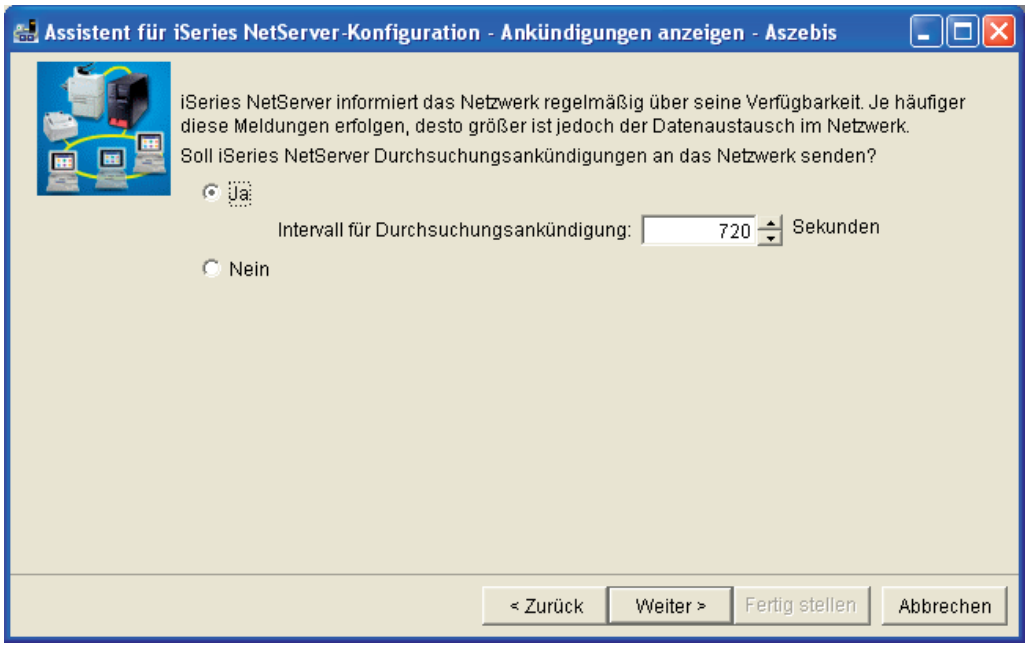
Durch die Globalisierung ist die Nutzung von „exotischen“ Zeichensätzen mittlerweile in einigen Unternehmen Alltag. Für die Standardverwendung des NetServers müssen wir den Zeichensatz vorgeben, den alle Clients nutzen sollen, die mit NetServer-Ressourcen arbeiten.

Jeder Aufbau eines Zugriffs auf den Inhalt einer Datei- oder Druckerfreigabe baut eine Verbindung über den NetServer auf. Diese Verbindung ist Grundlage für den Datentransfer, die Authentifizierung und einiges mehr.

Es sind Geschwindigkeitsüberlegungen, die es uns erlauben, eine einmal aufgebaute Verbindung nach Beendigung der auslösenden Anforderung für einen vorzugebenden Zeitraum aufrecht zu erhalten. Die dazu möglichen Einstellungen lassen sich in folgender Anzeige vornehmen.

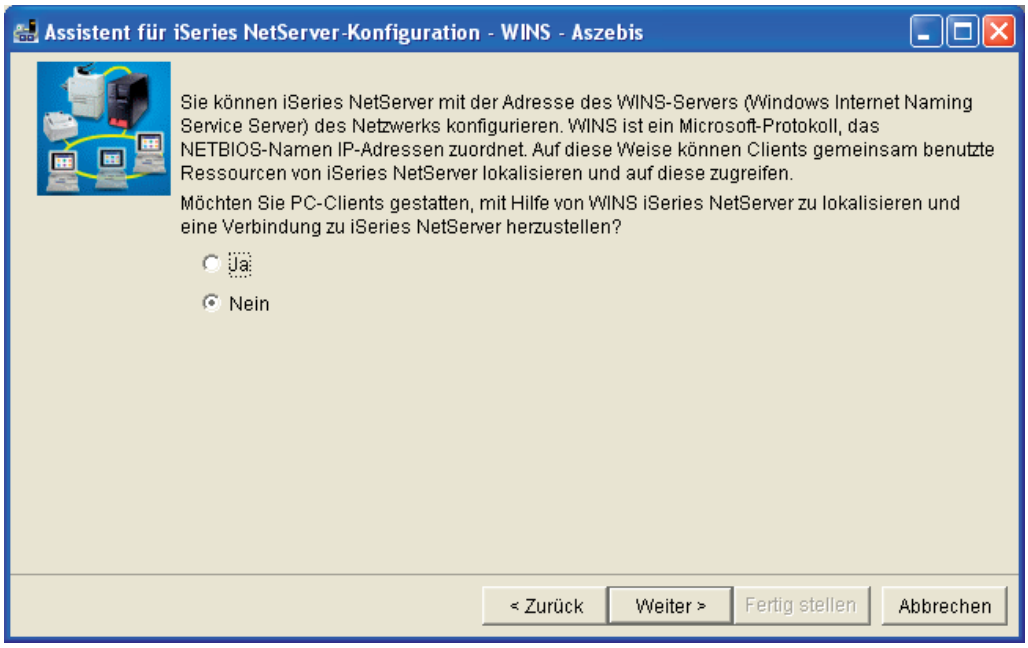


Zeitliche Sitzungsbegrenzung



Intervall-Vorgabe

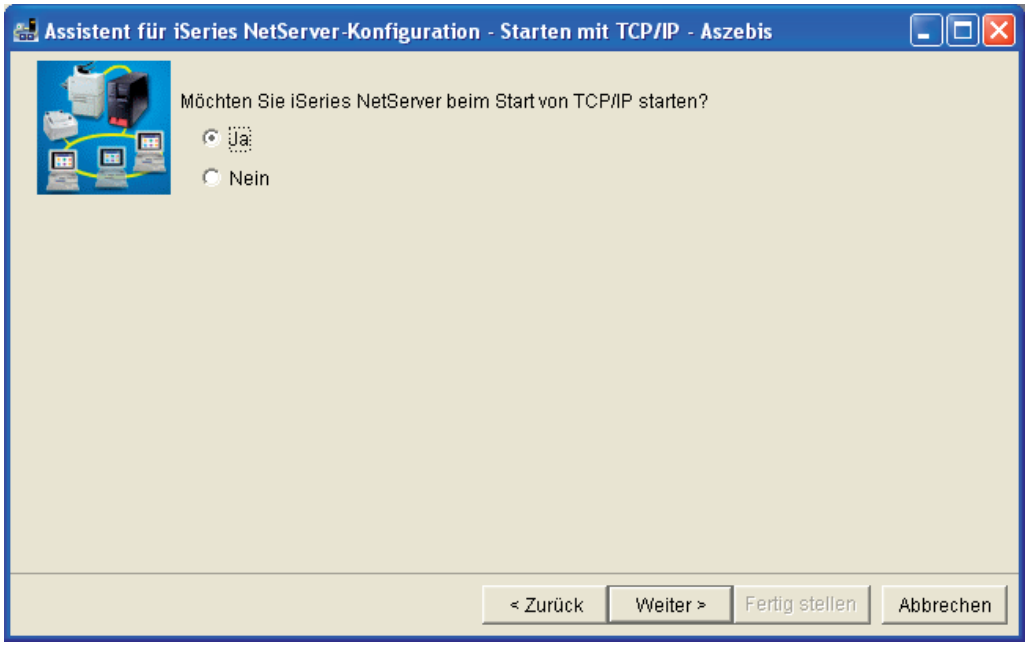
Die gemeinsame Nutzung der Ressourcen des NetServers mit Windows-Werkzeugen erlaubt auch den Einsatz des Microsoft-Protokolls WINS. Mit diesem speziellen Protokoll lassen sich Standardfunktionen von Microsoft auch auf Dateifreigaben des NetServers anwenden. Wenn Sie diese nutzen wollen, wählen Sie dazu in der folgenden Anzeige die entsprechende Option.



WINS-Konfiguration

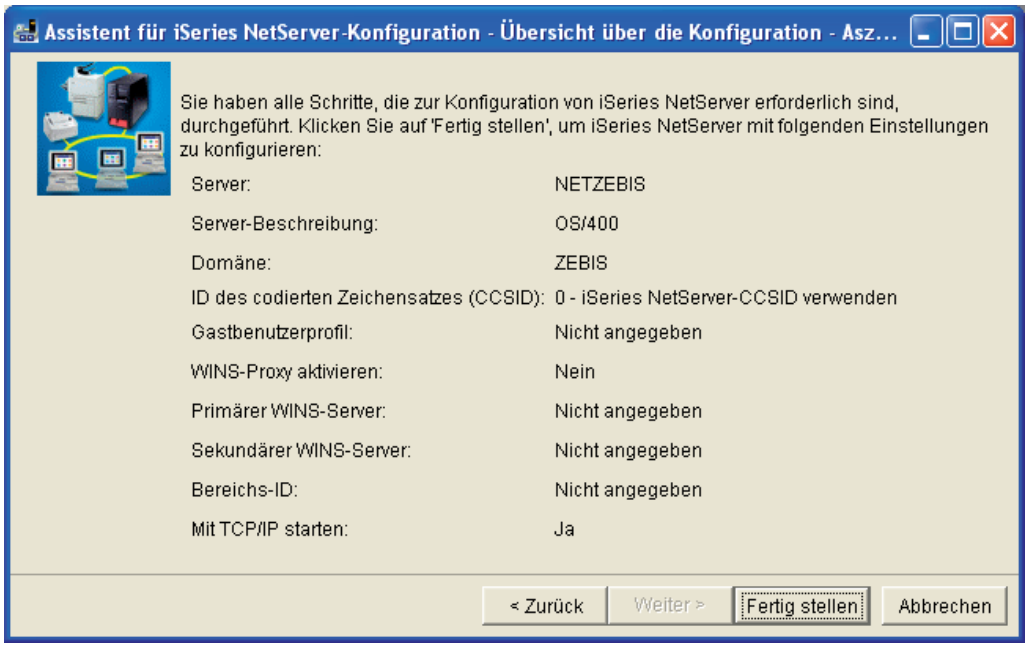
5.4.3
Seite 16

In der Regel macht es Sinn, die NetServer-Funktionen automatisch beim Start von TCP/IP zu aktivieren. Ist dies gewünscht, dann treffen Sie in der folgenden Anzeige die entsprechende Auswahl. Natürlich können Sie jederzeit mit dem Befehl STRTCPSVR *NETSVR oder den entsprechenden Funktionen im iSeries Navigator den Start des NetServers manuell durchführen.



Automatischer Start

Die in den vorhergehenden Abbildungen zu tätigenden Angaben wurden vom Assistenten gesammelt; sie werden nun nochmals vor der Bestätigung zur Kontrolle aufgelistet.



Zusammenfassung

Bestätigen Sie Ihre Angaben mit einem Klick auf die Schaltfläche „Fertig stellen“. Damit werden die Einstellungen des NetServers vorgenommen. Diese wirken sich erst nach einem Start beziehungsweise Neustart des NetServers aus, der automatisch nach Abschluss der Konfiguration durchgeführt werden kann.

Hinweis auf Neustart

5.4.3
Seite 18

NetServer-Sicherheitseinstellungen

Der Einsatz des iSeries NetServers birgt unweigerlich auch die Frage der Sicherheit. Wie kann mit dem Bereitstellen der Daten in einem Netzwerk garantiert werden, dass nur wirklich berechtigte Benutzer an den Inhalt der freigegebenen Dateien und Verzeichnisse gelangen können?

Betrachten wir zunächst einmal die Kommunikation zwischen der i5/OS-Welt mit der Windows-Umgebung in Form der mit dem NetServer zu verwaltenden Ressourcen wie beispielsweise Dateifreigaben oder dem Zugriff auf die Drucker-Ressourcen.

Für den Zugriff auf den NetServer und dessen freigegebenen Ressourcen benötigen wir ein Benutzerprofil, das sowohl in der i5/OS-Umgebung und auch in der Windows-Ebene verfügbar ist. Zusätzlich muss dieses Benutzerprofil über ein identisches Kennwort verfügen. Diese Besonderheit wird bei jedem Zugriff auf den NetServer geprüft. Existiert nun ein Benutzer in beiden Welten und wurde in einem Bereich ein Kennwort geändert, dann wird der Benutzer bei dem Einsatz von NetServer-Funktionen darauf hingewiesen und aufgefordert, die Kennworte zu synchronisieren.

Ein solches Verfahren ist aus Sicherheitsüberlegungen heraus sicher sinnvoll – wenngleich es den heute zur Verfügung stehenden Techniken wie EIM und SSO nicht wirklich entspricht. Letztere sind in Verbindung mit dem iSeries NetServer in Form der Kerberos-Identifizierung ebenfalls einsetzbar.

Anmerkung

Wenn Sie die Kerberos-Identifizierung für den Zugriff auf die iSeries NetServer-Ressourcen verwenden, dann können Sie dort unterschiedliche Prinzipals definieren. Diese werden der Chiffrierschlüsseldatei hinzugefügt und dienen der Identifizierung der Clients, welche mit dem iSeries NetServer eingesetzt werden können. Diese Prinzipalnamen variieren je nach eingesetztem Client-Betriebssystem. Für Windows XP können Sie die folgenden Prinzipals für die EIM-Integration mit dem NetServer einsetzen:

- cifs/iSeries vollständig qualifizierter Domänenname,
- cifs/iSeries Hostname,
- cifs/QiSeries Hostname,
- cifs/qiSeries Hostname,
- cifs/IP-Adresse.

5.4.3.1
Seite 2

Anonymer Zugriff

Es wird neben der Absicherung der Freigaben in Form einer Authentifizierung sicher auch Fälle geben, bei denen „Anwender" – ohne vorherige Anmeldung – auf den Inhalt unkritischer Dateien zugreifen können. Eine solche Form bezeichnen wir als „anonymen Zugriff". Bei einem solchen wird für den Zugriff auf den Inhalt im IFS keine Berechtigungsabfrage gestellt. Das bedeutet, dass ein Benutzer bzw. eine Anwendung ohne Benutzeranmeldung auf die freigegebenen Ressourcen zugreifen kann. Die Freigabe der Ressourcen in Form von Dateien oder Verzeichnissen lässt sich gezielt definieren – damit besteht die Möglichkeit, einige Dateien für einen anonymen Zugriff freizugeben, während andere Bereiche gegen einen solchen allgemeinen Zugriff geschützt sind.

Der Zugriff auf diese Bereiche erfolgt mit dem Benutzer „Anonymous". Dieser verfügt über eingeschränkte Berechtigungen und kann die FTP-Server-Bereiche nur eingeschränkt verwenden. In der Regel kann dieser Benutzer auf den FTP-Serverbereich zugreifen, die Inhalte auflisten und freigegebene Dateien kopieren bzw. übertragen. Meist darf dieser Benutzer keine Daten des FTP-Servers verändern oder diesem Inhalte zufügen.

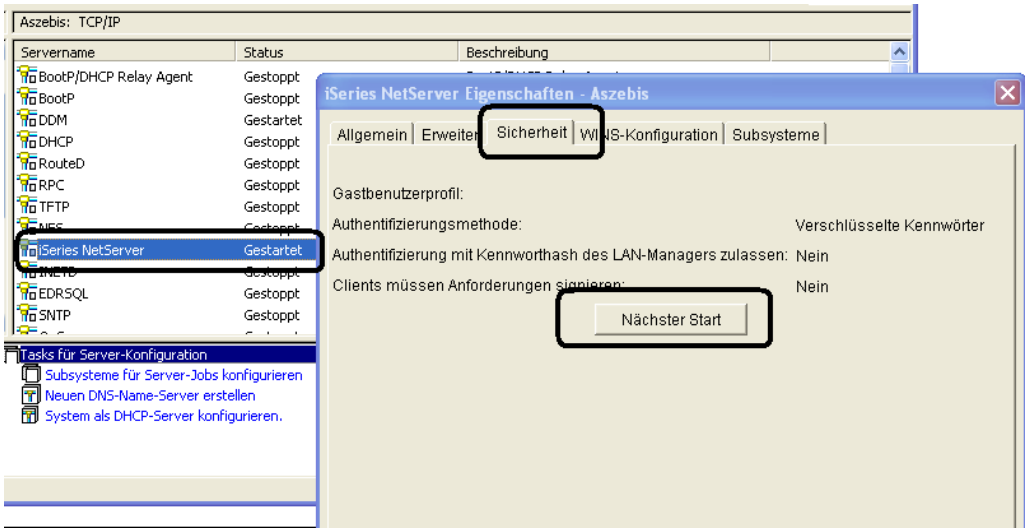
Diese Form des Zugriffs wird bei der Verwendung des NetServers über ein sogenanntes „Gastbenutzerprofil" realisiert.

Gastzugriffe
Besonderheiten im Zugriff auf Inhalte des NetServers können unter bestimmten Voraussetzungen mit Hilfe eines sogenannten Gastbenutzerprofils geschickt umgangen werden. Mittels eines solchen Profils ermöglichen Sie Anwendern den Durchgriff auf freigegebene Ressourcen ohne die Notwendigkeit, dass die Kennwörter auf beiden Ebenen identisch sein müssen. Sicher kennen Sie solche Formen des Zugriffs auch aus anderen Bereichen – dort werden hin und wieder auch „Anonymous"-Zugriffe realisiert. Das Gastprofil ist mit einem solchen „Anonymous Benutzer" vergleichbar. Bei einem solchen Gastzugriff wird ein Benutzerprofil akzeptiert, dessen Kennwort von den Windows-Einstellungen abweicht.

Natürlich sollte man sich im Klaren darüber sein, dass damit unter Umständen eine Sicherheitslücke geöffnet wird. Denn unachtsam freigegebene Ressourcen können mit einem solchen „Scheunentor" genutzt werden!

Die Einrichtung eines solchen Gastprofils geschieht über den iSeries Navigator.

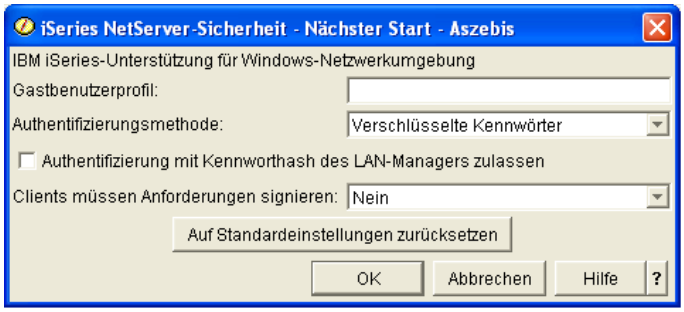
Dort öffnen wir zunächst den Bereich „Netzwerk / Server / TCPIP“ und wählen den Eintrag „iSeries NetServer“ mit der rechten Maustaste zur Verwaltung aus.



NetServer verwalten

In dem sich öffnenden Verwaltungsfenster klicken wir nun auf den Bereich „Sicherheit“. Diesen Bereich werden wir später bei der weiteren Betrachtung der Sicherheitseinstellungen noch genauer kennenlernen.

Mit einem Klick auf die Schaltfläche „Nächster Start“ gelangen wir in den Sicherheitsdefinitionsbereich des NetServers. Wenn noch kein Gastprofil eingerichtet worden ist, dann ist die erste Zeile „Gastbenutzerprofil“ leer.



Sicherheitsdefinitionsbereich

Geben Sie in dem Feld „Gastbenutzerprofil“ nun den gewünschten Namen des Benutzerprofils ein, mit dem ein anonymer Zugriff ermöglicht werden soll.

5.4.3.1
Seite 4

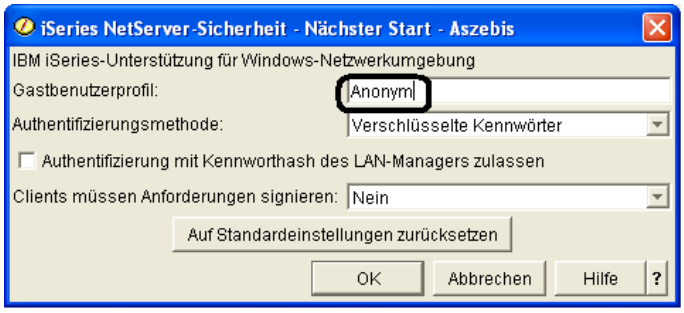
Achten Sie darauf, dass der hier eingegebene Name mit dem Namen des Benutzerprofils übereinstimmt, welches auf dem System i zur Authentifizierung genutzt wird. Sollten Sie noch kein solches Profil haben, dann definieren Sie ein solches wahlweise mit dem Befehl CRTUSRPRF oder mittels des iSeries Navigators.

Benutzerprofil i5/OS

Anmerkung

Achten Sie auf die Beschränkungen der Benutzernamen in den unterschiedlichen Systemen. Häufig sind die Längen der Benutzernamen in den einzelnen Systemen beschränkt. Natürlich gilt auch bei dem Einsatz des NetServers die Beschränkung der maximalen Länge eines Benutzerprofils im i5/OS-Umfeld. Achten Sie deshalb grundsätzlich darauf, dass die Benutzernamen bei dem Einsatz des NetServers maximal 10-stellig sind!!

Ein solcher „Benutzer“ verfügt über die Berechtigungen, welche von einem Administrator vorgegeben worden sind. Achten Sie deshalb unter anderem auch bei der Definition eines solchen Pseudobenutzers darauf, dass die Berechtigungen für die Verwendung entsprechend vergeben werden!



Gastbenutzerprofil

Sollte das Gastbenutzerprofil kein übereinstimmendes i5/OS-Profil vorfinden, dann erscheint eine entsprechende Fehlermeldung. Anderenfalls wird nach dem nächsten Start des NetServers der Zugriff mit dem Benutzer „Anonym“ möglich sein.

Nicht vorhandenes Benutzerprofil

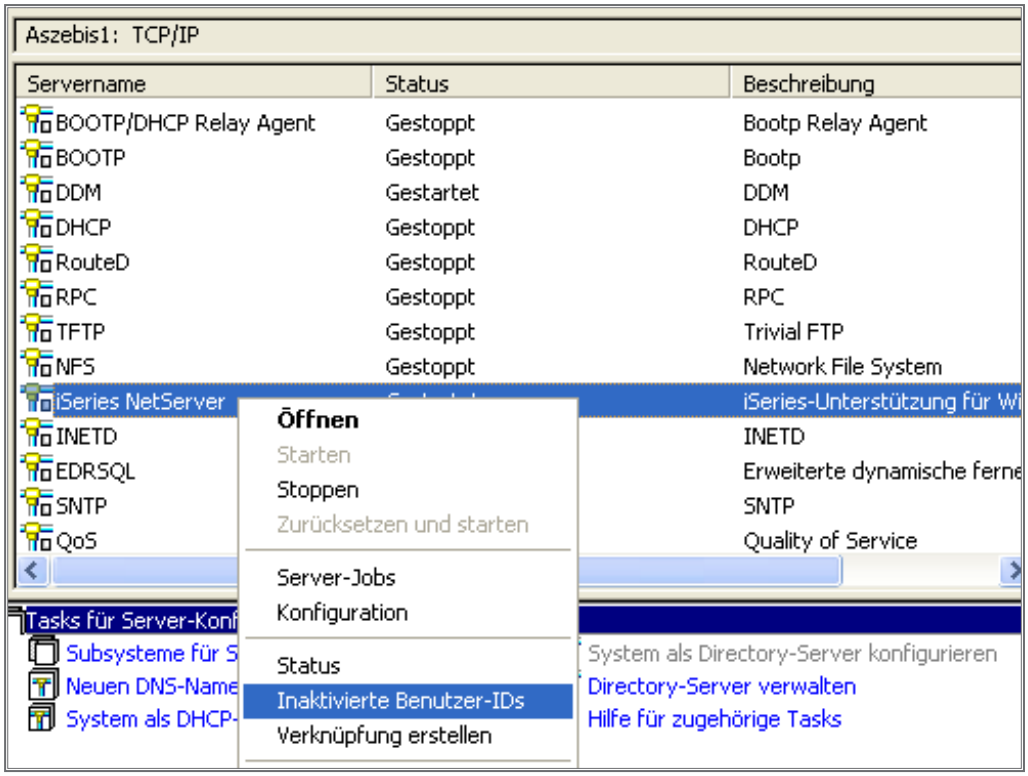
Wenn das Gastbenutzerprofil angelegt wurde, muss der NetServer neu gestartet werden. Danach kann mit den Basisberechtigungen auf die Freigaben innerhalb des NetServers zugegriffen werden.

Die Verwendung des NetServers basiert, wie wir soeben gesehen haben, auf einem gültigen Benutzerprofil oder einem „Anonymous“-Benutzer. Letzterer kann für Standardzugriffe verwendet werden, ist jedoch hinsichtlich der verfügbaren Berechtigungen bewusst eingeschränkt.

Im Regelfall können System i-Anwender mit deren Benutzerprofil auf die Ressourcen des NetServers zugreifen. Hier werden dann die System i-spezifischen Sicherheitskriterien angewendet. Dies bedeutet, dass beispielsweise das Kennwort abgefragt wird und dieses, bei mehrfacher Falscheingabe, inaktiviert werden kann.

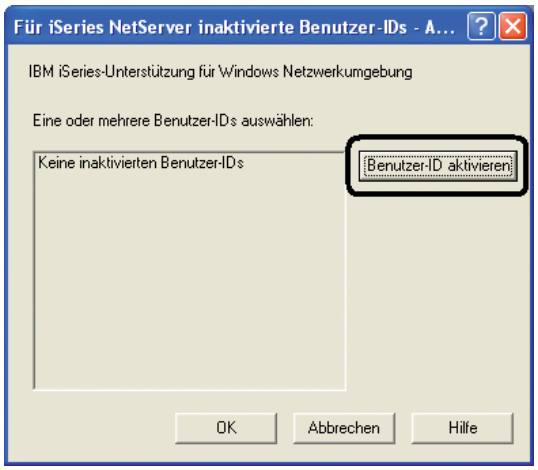
5.4.3.1
Seite 6

Ein auf diese Weise deaktiviertes Benutzerprofil kann im iSeries Navigator in dem Verwaltungsbereich des NetServers angezeigt werden. Dazu verwenden wir die Auswahl „Netzwerk / Server / TCPIP“ und wählen in der Auflistung der TCP/IP-Server den NetServer mit der rechten Maustaste aus. In der Auswahlliste befindet sich unter anderem auch der Eintrag „Inaktivierte Benutzer-IDs“. Ein Beispiel dazu finden Sie in der folgenden Abbildung.



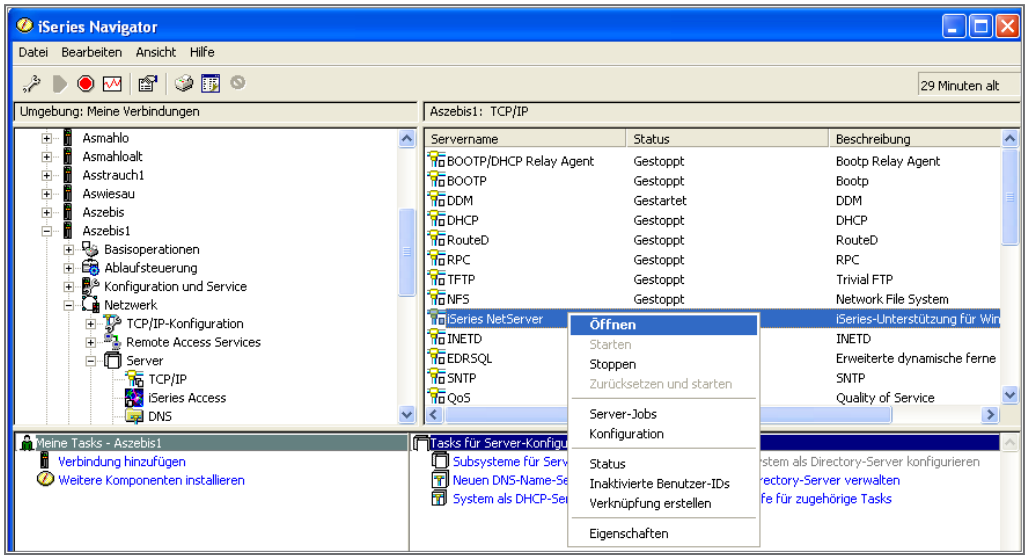
Inaktivierte Benutzer-IDs

Sollte in der Auflistung der so genannten Inaktivierten Benutzer Einträge vorhanden sein, dann können Sie diese mittels der Option „Benutzer-ID aktivieren“ wieder für die Verwendung freischalten.



Aktivieren von Benutzer-IDs

Einen weiteren Bereich der Sicherheitseinstellung finden wir in den individuellen Einstellungen der Freigaben. Unabhängig davon, ob eine Freigabe auf Dateiebene oder für Drucker erfolgt ist, können die Zugriffsberechtigungen in dem iSeries NetServer-Verwaltungsbereich definiert werden. Diese Einstellungen lassen sich in dem Freigabenbereich des NetServers verwalten. Diese befinden sich in der Auswahl „Öffnen“, welche mit einem rechten Mausklick auf den Eintrag „iSeries NetServer“ innerhalb der Auflistung der TCP/IP-Server erscheint. Ein Beispiel dazu finden Sie in der folgenden Abbildung.

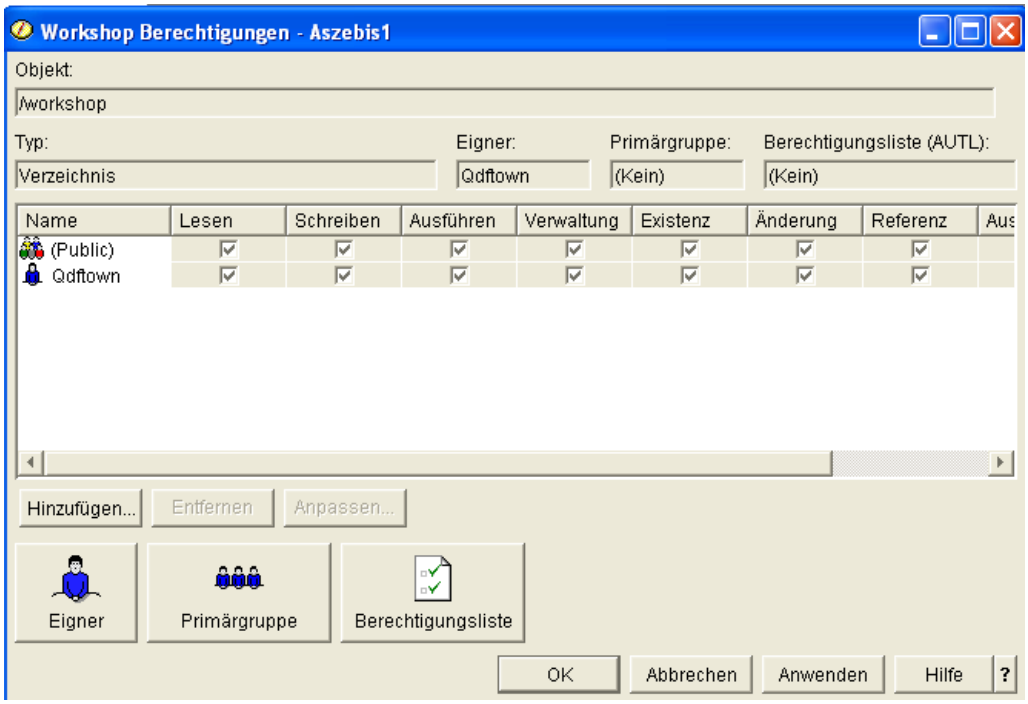


Starten der Konfiguration

Mit dieser Auswahl gelangen wir in die Übersicht der Freigaben des iSeries NetServers. Wählen Sie dort den gewünschten Eintrag mit der rechten Maustaste aus. In der sich öffnenden Auswahlliste befindet sich unter anderem auch der Eintrag „Berechtigungen“, den wir nun selektieren.

In der Folgeanzeige erhalten wir die Objektberechtigungen, welche für den ausgewählten Bereich definiert sind.

5.4.3.1
Seite 8



Berechtigungen

An dieser Stelle können Sie die gewünschten Anpassungen vornehmen. Die Berechtigungssteuerung unterscheidet sich dort nicht von der allgemeinen Berechtigungsverwaltung, wie sie auch von anderen Bereichen des System i bekannt sein dürfte.

NetServer-Freigaben

Der iSeries NetServer dient im Wesentlichen der Bereitstellung von Ressourcen des System i für die Verwendung mit anderen Plattformen wie Windows-Systemen. Dabei unterscheiden wir die Freigabe nach

- Dateifreigaben und
- Druckerfreigaben.

Während die Dateifreigaben den Zugriff auf die Inhalte des IFS ermöglichen, übernimmt der iSeries NetServer bei der Druckerfreigabe die Rolle eines Print-Servers, welcher die Druck-Ressourcen des System i für PC-Benutzer und PC-Anwendungen verfügbar macht.

5.4.3.2

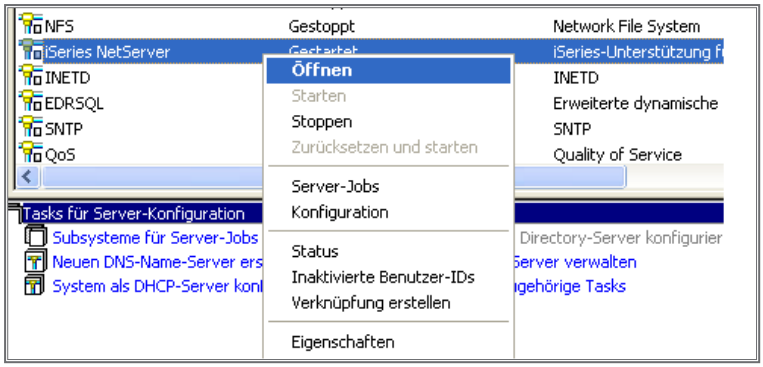
Seite 2

NetServer-Dateifreigaben

Die Möglichkeit der Dateifreigaben stellt sicher eine der Hauptaufgaben für den NetServer dar. Mit der Dateifreigabe kann ein beliebiges Verzeichnis innerhalb des integrierten Dateisystems des System i (kurz: IFS) freigegeben werden. Damit kann ein Windows-Benutzer auf die in dem IFS liegenden Ressourcen zugreifen.

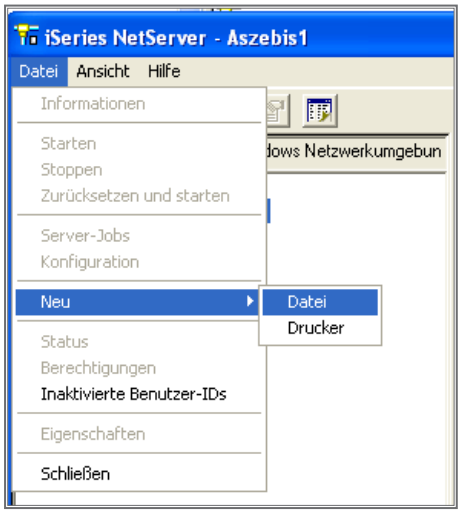
Bevor wir ein Verzeichnis innerhalb des IFS über den NetServer nutzen können, muss dieses zunächst freigegeben werden. Eine solche Freigabe erfolgt über den iSeries Navigator. Dort selektieren wir den Eintrag „Netzwerk / Server / TCPIP“.

Klicken Sie anschließend mit der rechten Maustaste auf den Eintrag „NetServer“ und wählen Sie die Option „Öffnen“.



Freigaben erstellen

Selektieren Sie nun den Menüeintrag „Datei / Neu“, wie es die nachfolgende Abbildung zeigt.



Neue Dateifreigabe

In dem Abschnitt „Allgemeine“ geben wir nun den Namen der Freigabe ein. Eine solche Freigabe bezieht sich auf ein Verzeichnis innerhalb des NetServers. Neben der Freigabebezeichnung und dem Namen des Verzeich-

5.4.3.2.1
Seite 2

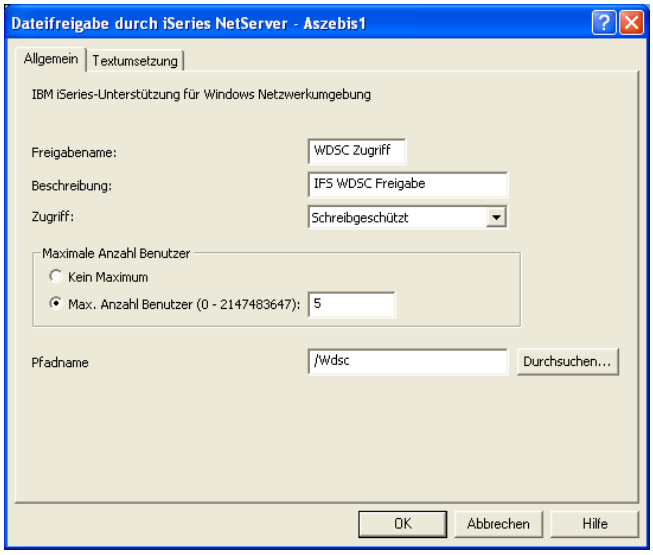
nisses lässt sich auch die maximale Anzahl zeitgleicher Benutzer und auch die Form der Freigabe (z. B. schreibgeschützt) definieren.

Der Bereich „Pfadangabe“ erlaubt die Zuordnung des IFS-Bereiches, welcher der NetServer-Freigabe zugeordnet werden soll. Mit einem Klick auf „Durchsuchen“ erhalten wir eine Auflistung aller auf dem System i befindlichen IFS-Verzeichnisse, welche für die Freigabe genutzt werden können. Aus der Auflistung kann nun das gewünschte Verzeichnis ausgewählt werden.

Auswahl des freizugebenden Ordners

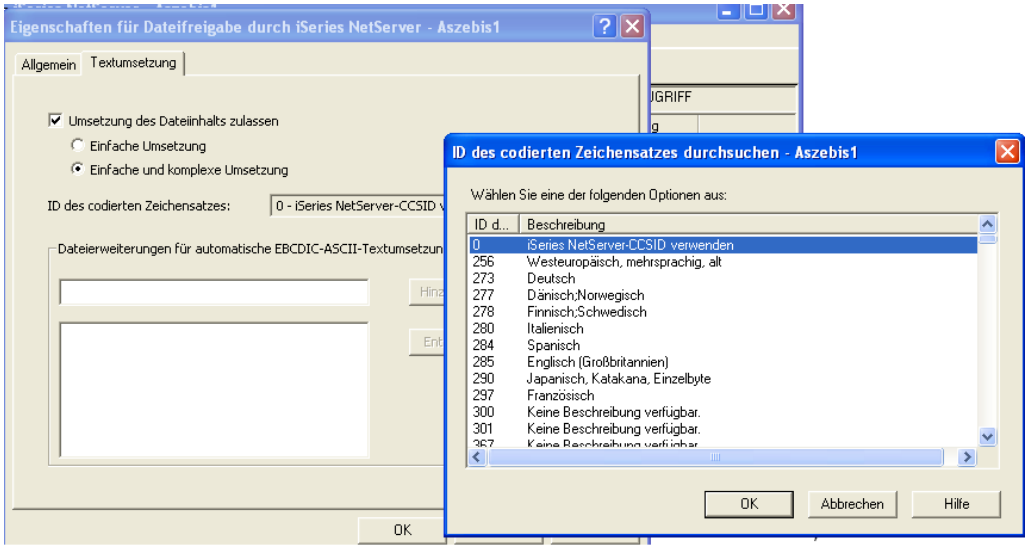
Wenn Sie den Namen des freizugebenden Bereichs kennen, dann können Sie diesen natürlich auch direkt angeben. Dabei ist zu beachten, dass der Name mit einem führenden Schrägstrich „/“ eingegeben wird. Die Direkteingabe des IFS-Verzeichnisses aus der vorhergehenden Abbildung würde also in Form von „/wdsc“ erfolgen.

Damit sind die Grundangaben für die neue Freigabe fast abgeschlossen. In unserem Beispiel haben wir das IFS-Verzeichnis „WDSC“ unter dem Freigabenamen „WDSC Zugriff“ schreibgeschützt für bis zu fünf zeitgleiche Benutzer freigegeben.



Definitionsbereiche der Freigabe

Besondere Bedeutung kommt der Zeichen- bzw. Textumsetzung zu. Diese ist dann sinnvoll, wenn die originären i5/OS-Daten in ein anderes Zeichenformat umgesetzt werden sollen, das für andere Systeme, wie zum Beispiel Windows, benötigt wird. Mit der Angabe der CCSID lassen sich die Standardumsetzungen festlegen.

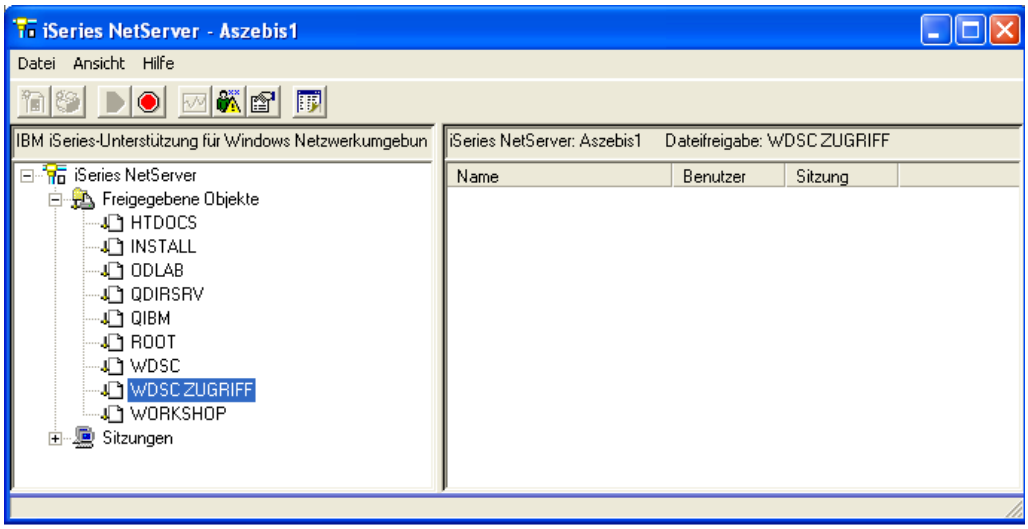


Zeichenumsetzung

Der Zeichenumsetzung kommt besonders dann eine wesentliche Bedeutung zu, wenn die ausgetauschten Daten zwischen beiden Systemwelten weiter verarbeitet werden sollen. Nicht selten kommt es bei falsch verwendeten Zeichensätzen bzw. Zeichenumsetzungen zu Problemen. Achten Sie deshalb darauf, dass für den individuellen Fall der korrekte Zeichensatz genutzt wird.

5.4.3.2.1
Seite 4

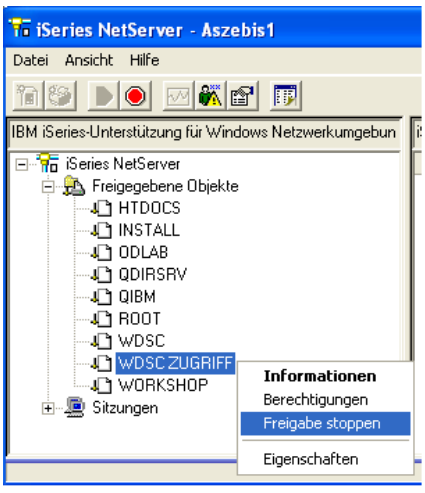
Wenn alle gewünschten Parameter eingestellt sind, kann die Freigabe durchgeführt werden. Damit wird das ausgewählte Verzeichnis für die gemeinsame Verwendung freigegeben und erscheint in der Auflistung der freigegebenen iSeries NetServer Ressourcen.



Die freigegebenen Objekte

Anmerkung

Dieser Verwaltungsbereich der freigegebenen Objekte erlaubt beispielsweise auch das Stoppen einer Freigabe. Soll eine Freigabe auf einen Bereich zurückgenommen werden, dann markieren Sie diesen Bereich mit der rechten Maustaste und selektieren die Option „Freigabe stoppen“.



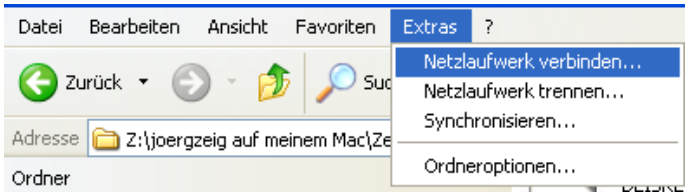
Freigabe stoppen

Damit erhalten Clients oder Programme, welche nach dem Stoppen versuchen, mit dem Inhalt des Verzeichnisses zu arbeiten, keinen Zugriff. Bereits angemeldete Clients oder Programme, die vor dem Stoppen eine Verbindung zu dem Verzeichnis aufgebaut haben, können auch nach dem Stoppen so lange mit dem Inhalt des Verzeichnisses arbeiten, bis sie sich abmelden. Die erneute Anmeldung würde dann für ein nicht freigegebenes Verzeichnis bzw. für eine gestoppte Freigabe dazu führen, dass der Zugriff auf den Inhalt verwehrt wird.

Ein freigegebenes Objekt verbinden

Die Verzeichnisfreigabe innerhalb des NetServers bildet beispielsweise auch die Grundlage für den Zugriff auf den Inhalt des IFS mittels der Standardwerkzeuge von Windows. So können wir beispielsweise ein freigegebenes Verzeichnis auch mittels des Windows Explorers verbinden.

Eine solche Laufwerksverbindung wird wie von anderen Laufwerken auch schon bekannt, über die Funktion „Extras / Netzlaufwerk verbinden“ realisiert.



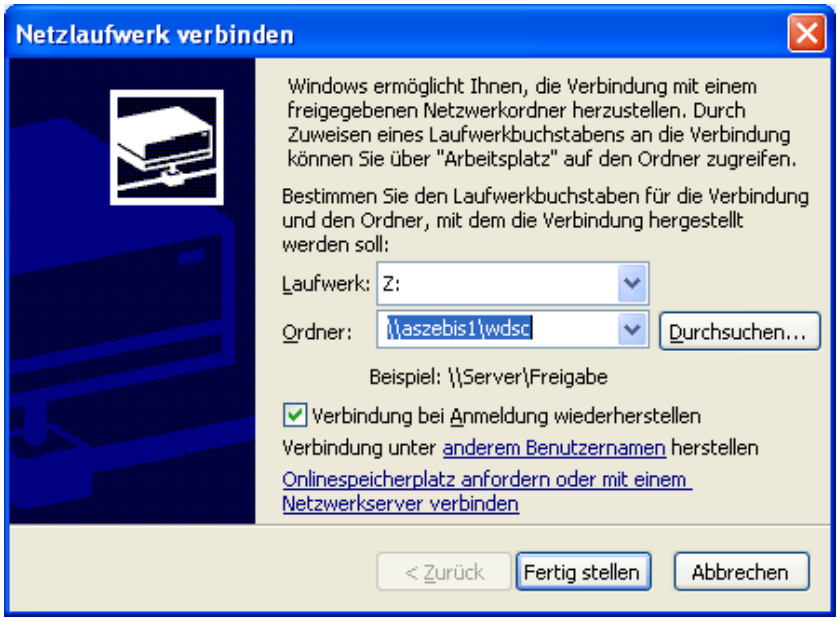
Netzlaufwerk verbinden

Damit öffnet sich das nächste Definitionsfenster, in dem wir den Laufwerksbuchstaben angeben bzw. aus der Auflistung der freien Bereiche auswählen können. Neben der Angabe des Laufwerks muss noch der Freigabe-Name in Form des NetServer-Namens gefolgt vom Freigabe-Namen angegeben werden. Achten Sie darauf, dass vor dem NetServer-Namen zwei Backslash-Zeichen angegeben werden. Die Trennung zwischen iSeries NetServer-Namen und Freigabe-Namen ist ebenfalls in Form eines Backslash vorzunehmen.

Ein Beispiel für eine solche Freigabe wäre

\\NetServername\freigabename

5.4.3.2.1
Seite 6



Laufwerkszuordnung

Mit einem Klick auf „Fertig stellen“ wird die Netserver-Freigabe dem Laufwerk zugeordnet. Danach kann auf den Inhalt des freigegebenen Verzeichnisses wie gewohnt mit Windows-Mitteln zugegriffen werden.

Name des NetServers ermitteln

Bei der Laufwerkszuordnung muss, wie wir zuvor gesehen haben, der Name des iSeries NetServers angegeben werden. Diesen Namen kann man unter anderem auch mittels der Standard Windows-Funktionen ermitteln. Ist der Name nicht bekannt, dann öffnet man den Windows Explorer und erweitert den Eintrag „Netzwerkumgebung“. Dort erweitern wir den Eintrag „Gesamtes Netzwerk / Microsoft Windows Netzwerk“. Innerhalb des Bereichs „Workgroup“ finden wir unter anderem die NetServer, auf die Zugriff besteht. Ein Beispiel dazu finden Sie in der folgenden Abbildung.

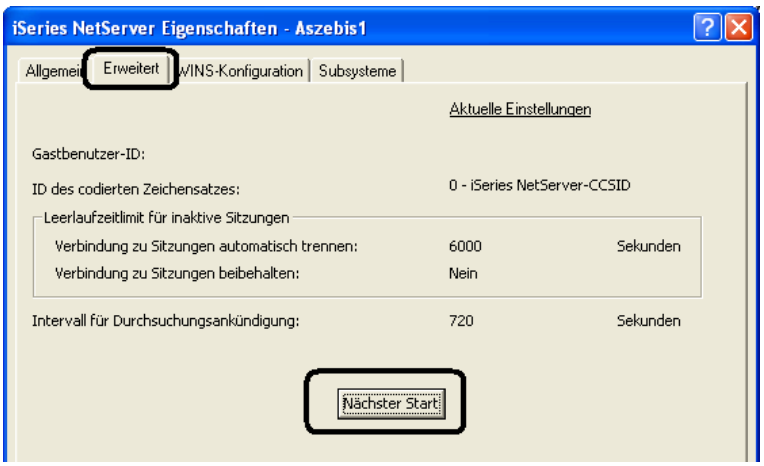
NetServer finden

Anmerkung

Es kann unter Umständen der Fall sein, dass der iSeries NetServer aus Sicherheitsgründen nicht in der Auflistung der in dem Netzwerk befindlichen Server dargestellt wird.

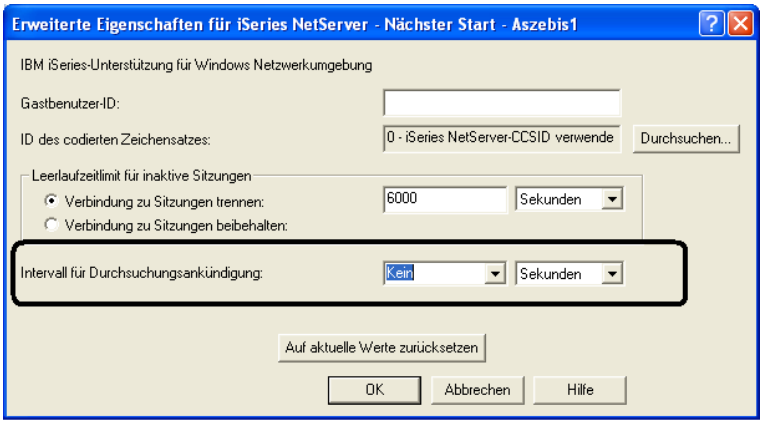
Sollten Sie den iSeries NetServer auch „verdeckt“ freigeben wollen, dann können Sie das auf folgende Weise umsetzen:

Wählen Sie den iSeries NetServer-Eintrag in der Auflistung der TCP/IP-Server mit der rechten Maustaste aus und klicken Sie auf den Eintrag „Eigenschaften“. Wählen Sie anschließend den Bereich „Erweitert“ und klicken Sie auf die Schaltfläche „Nächster Start“.



Erweiterte Einstellungen

Suchen Sie die Option „Intervall für Durchsuchungsankündigung“ und ändern Sie den zugehörigen Wert auf „Kein“ ab.



Durchsuchungsintervall

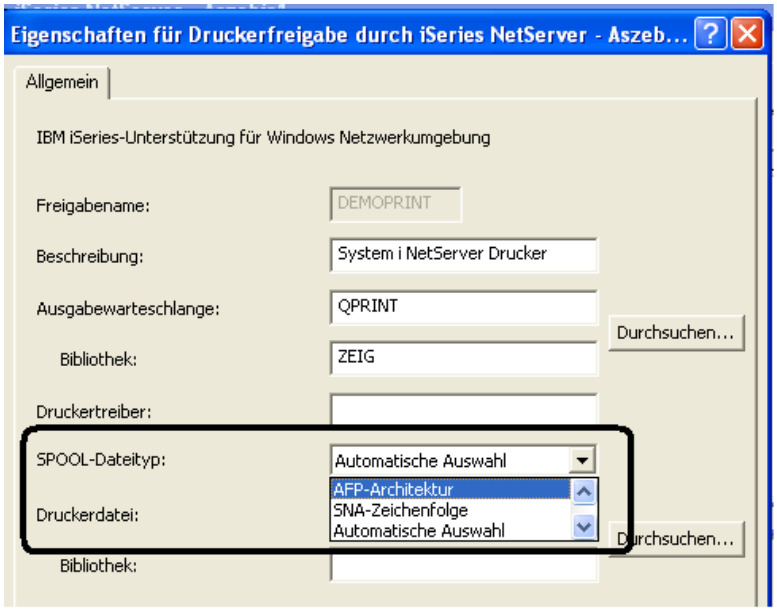
5.4.3.2.1

Seite 8

NetServer-Druckfreigaben

NetServer bietet nicht nur die Möglichkeit der Freigabe von Verzeichnissen des System i, um diese beispielsweise mit Windows-Werkzeugen und Windows-Programmen verarbeiten zu können, sondern liefert auch die Möglichkeit, auf die Druck-Ressourcen des Systems mit Windows-Ressourcen zuzugreifen. Diese Form der gemeinsamen Benutzung bezeichnen wir als Druckfreigaben. Dabei agiert der NetServer als Print-Server für die Windows-Umgebung und stellt damit die System i-Druck-Ressourcen für die Verwendung mit Windows bereit. Diese Bereitstellung kann bei dem Vorhandensein entsprechender Treiber dazu eingesetzt werden, dass Ausdrucke in die Ausgabewarteschlangen des System i gestellt werden. Die Basis für den Zugriff von Windows auf die Druckausgaben des System i bildet die Ausgabewarteschlange (OUTQ), welche als Grundlage für die Freigabe mit Hilfe des iSeries Navigators genutzt wird. Derzeit werden die folgenden Spool-Dateitypen unterstützt:

- Benutzer ASCII,
- AFP,
- SNA,
- Automatische Auswahl.



Unterstützte Spool-Dateitypen

Achten Sie unbedingt darauf, dass der korrekte Spool-Dateityp verwendet wird!

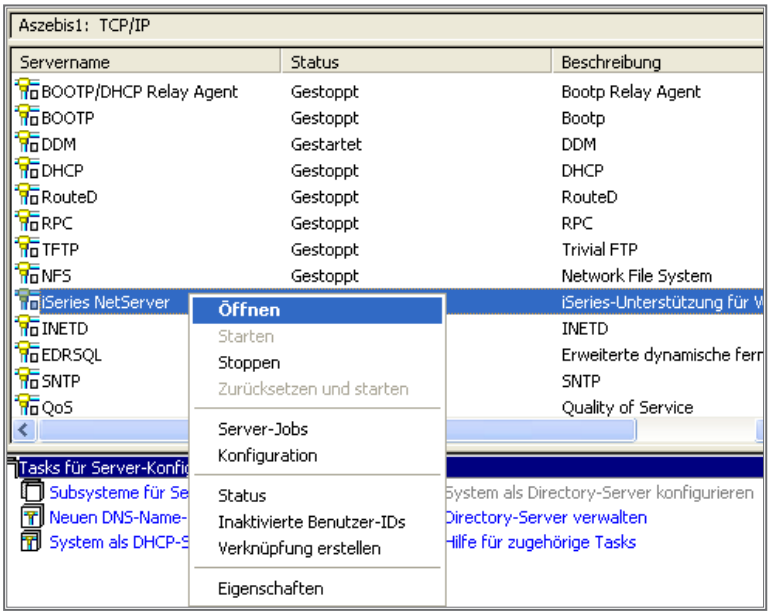
5.4.3.2.2
Seite 2

Ähnlich wie bei der Freigabe von Netzlaufwerken werden die Druckfreigaben ebenfalls über die iSeries Navigator Funktion „Netzwerk / Server / TCPIP“ definiert. Wählen Sie in der Auflistung der TCP/IP-Server den NetServer mit der rechten Maustaste aus und selektieren Sie anschließend die Option „Öffnen“.

In dem sich öffnenden Bereich wählen wir jetzt den Eintrag „Shared Objects“ mit der rechten Maustaste aus. Mittels der Option „Neu“ und „Drucker“ starten wir die Freigabe der Ausgabewarteschlange. Eine solche Freigabe besteht aus den folgenden Komponenten:

- Name der Freigabe,
- Beschreibung,
- Ausgabewarteschlange,
- Bibliothek
- Druckertreiber,
- Typ der Spool-Datei,
- Druckerdatei.

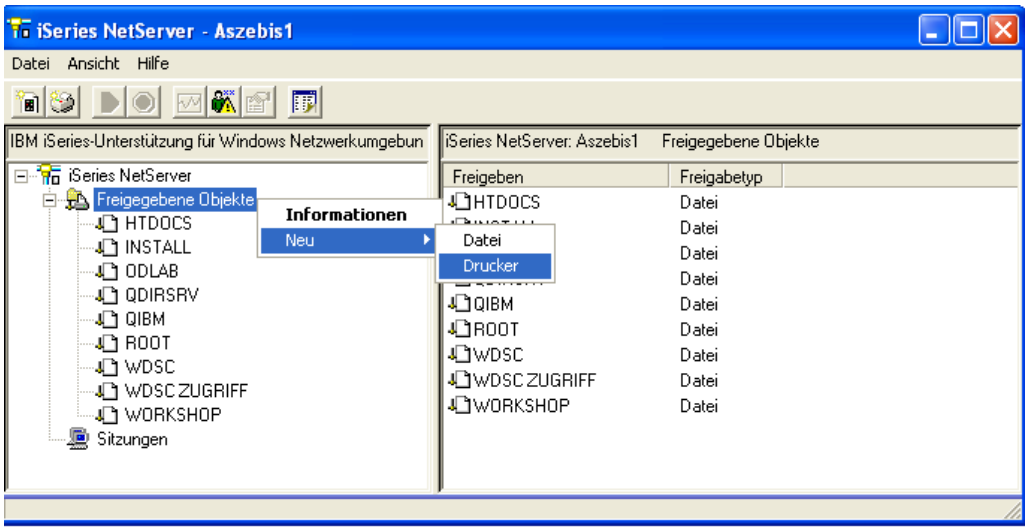
Die Druckerfreigaben können, wie auch die Dateifreigaben, über die Auswahl „Öffnen“ definiert werden. Wählen Sie dazu in der Auflistung der TCP/IP-Server den Eintrag „iSeries NetServer“ mit der rechten Maustaste aus und selektieren Sie die Option „Öffnen“.



Option „Öffnen“

In der sich öffnenden Anzeige selektieren wir nun den Eintrag „Freigegebene Ordner“, der sich in dem linken Anzeigenbereich befindet. Klicken Sie auf diesen Eintrag mit der rechten Maustaste.

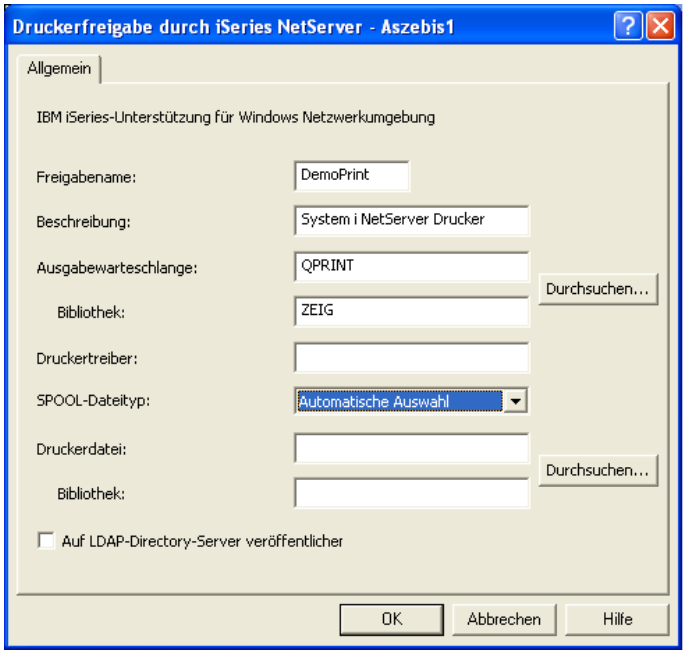
Anschließend wählen wir die Option „Neu / Drucker“, wie es die folgende Abbildung zeigt.



Neue Druckerfreigabe

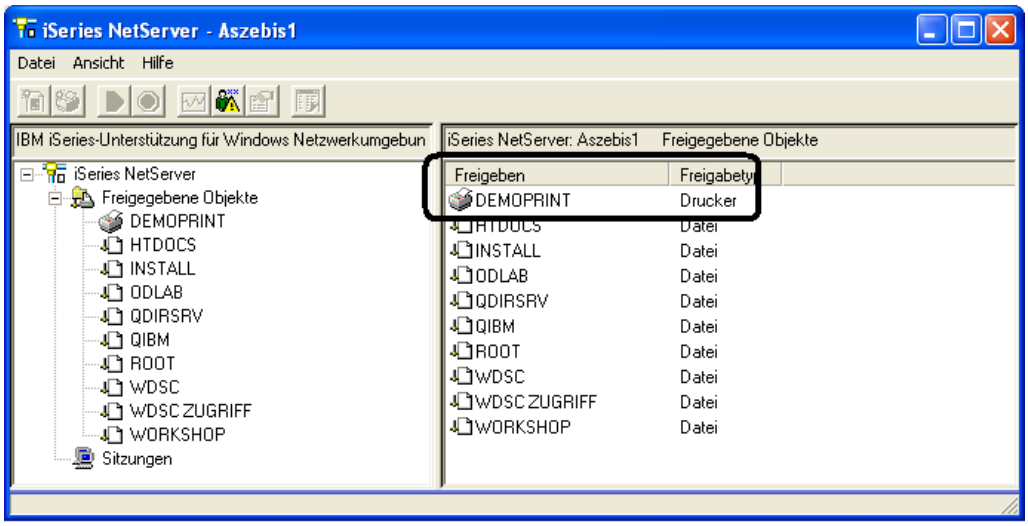
Die Freigabe eines Druckers erfolgt auf Basis eines frei wählbaren Freigabe-Namens. Eine Beschreibung ist ebenfalls definierbar. Die Basis für die Freigabe bildet eine Ausgabewarteschlange des System i, welche qualifiziert, dass bedeutet zusammen mit der zugehörigen Bibliothek angegeben werden muss. Optional kann ein Druckertreiber definiert werden, um Druckzuordnungen ordnungsgemäß umzusetzen.

5.4.3.2.2
Seite 4



Details der Druckerfreigabe

Nach der Angabe der erforderlichen Werte erscheint diese Druckerfreigabe in der Übersicht der freigegebenen NetServer-Ressourcen und kann, ähnlich wie die Dateifreigaben auch, von dieser Stelle aus verwaltet werden.



Freigegebener Drucker

Damit ist die Freigabe auf dem System i beendet.

Die Verfügbarkeit dieses Druckers in einer Windows-Umgebung kann über den gewohnten Windows-Prozess abgebildet werden. Über das Hinzufügen eines Netzwerkdruckers kann in Windows der System i-Drucker für die PC-Anwendungen verfügbar gemacht werden.

NetServer und SSO

Sicherheit wird beim Einsatz von sensiblen Daten großgeschrieben, und nicht selten finden wir sensible Daten auch im NetServer des System i.

NetServer bieten Windows-Benutzern die Option, Informationen geschützt und hochverfügbar auf dem System i abzulegen. Wichtig ist dabei natürlich ein entsprechendes Verfahren, mit dem der Zugriff auf die sensiblen Daten gesteuert werden kann. Hier hat sich die Verwendung der Authentifizierung per Kennwort etabliert. Dieses Verfahren ist in der Regel auch bei AS/400-Anwendern in Gebrauch. Allerdings kann für den Einsatz mit dem NetServer auch die Funktion des Single Sign-on verwendet werden. Da diese Funktion in Verbindung mit dem NetServer jedoch gewisse Besonderheiten aufweist, möchte ich nachfolgend auf sie eingehen.

Seit geraumer Zeit bietet IBM ihren iSeries- und System i-Kunden Single Sign-on-Funktionen in Verbindung mit unterschiedlichen Diensten an. Single Sign-on – oder kurz SSO ist in einigen Bereichen auch als „Einzelanmeldung“ bekannt. Mit Hilfe von Single Sign-on lassen sich Mehrfacheingaben von Benutzerinformationen und Kennwörtern für die Anmeldung an unterschiedlichen Servern, Anwendungen oder Diensten umgehen. Mit Hilfe von standardisierten Techniken können so plattformübergreifend Authentifizierungsfunktionen genutzt werden, mit denen Anwender sich per einmaliger Anmeldung alle Systeme und Anwendungen freischalten können, für die sie eine Berechtigung haben.

IBM hat viele Bereiche des i5/OS bzw. des IBM i für die Verwendung der Einzelanmeldung ausgelegt. Während die meisten dieser Bereiche problemlos einen Parallelbetrieb von Kennwortauthentifizierung und SSO unterstützen, ist dies bei anderen Funktionen leider nicht oder nur recht umständlich möglich.

Einer der Dienste, die auf der System i mit einer Einmalanmeldung ausgestattet werden können, ist – wie gesagt – der NetServer, also der Bereich, der System i-Ressourcen beispielsweise für die Verwendung mit Microsoft-Dateien zur Verfügung stellt. Single Sign-on bedeutet in diesem Zusammenhang die Nutzung von Kerberos-Standards, die mit dem System i in Form der Netzwerk-Authentifizierungsservices (NAS) im Basisbetriebssystem verankert sind.

So einfach es grundsätzlich ist, Single Sign-on mit dem System i zu betreiben, so wichtig ist insbesondere beim Einsatz von SSO zusammen mit dem NetServer eine optimale Planung – denn ein nur partielles Umstellen auf den Einsatz mit Kerberos und EIM kann in Verbindung mit dem NetServer nicht nur unerwünschte, sondern katastrophale Folgen haben.

5.4.3.3
Seite 2

Das Aktivieren der Single Sign-on-Funktionen kann sowohl mit 5250 – bzw. Qshell-Befehlen oder alternativ dazu mit den Assistenten innerhalb des System i Navigators durchgeführt werden. Hier lässt sich quasi per Mausklick das System i für den Einsatz mit SSO freischalten. Die Auswahl des NetServers für das Erstellen der Single Sign-on-Konfiguration ist eine Vorstufe der eigentlichen Aktivierung. Werden die Konfigurationseinstellungen dann auch noch dem LDAP bzw. dem Active Directory bekannt gemacht, dann geht der NetServer bei der Authentifizierung zunächst einmal davon aus, dass Single Sign-on durchgängig für die Anmeldung verwendet wird. Sind nicht alle Benutzer dem EIM bzw. dem Single Sign-on zugeordnet, können die nicht zugeordneten Benutzer unter Umständen nicht mehr auf den NetServer zugreifen!

Wird erst einmal ein Service Prinzipal mit Hilfe des Befehls KTPASS erstellt und dem Windows Active Directory bekannt gemacht, so geht Kerberos davon aus, dass SSO verwendet werden soll – und das gilt zunächst für alle Anwender in der Windows-Domäne!

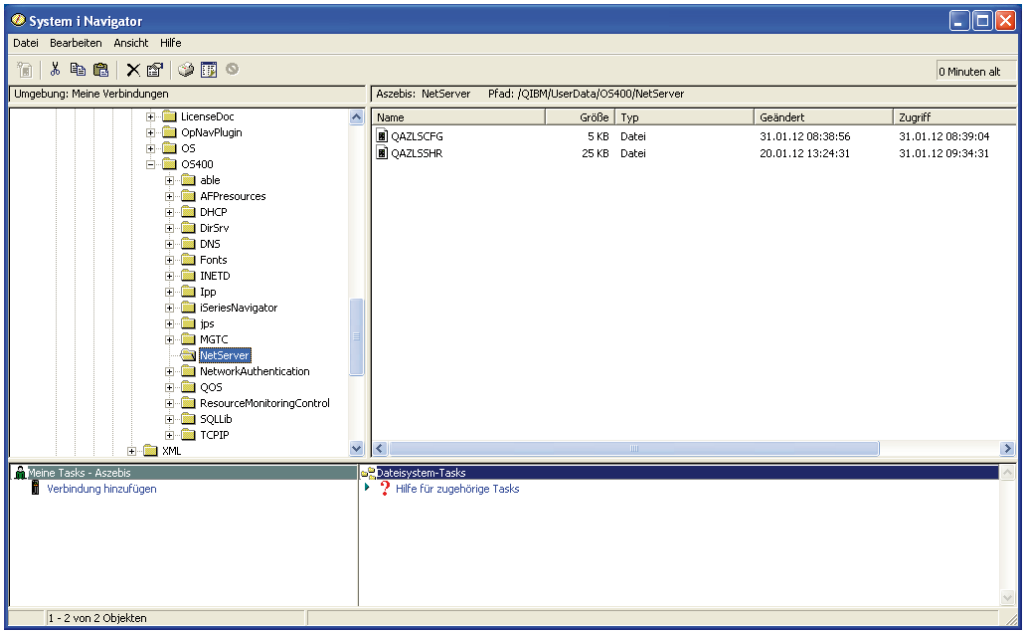
Grundsätzlich gilt als Empfehlung, SSO in Verbindung mit dem NetServer nur dann zu aktivieren, wenn alle Benutzer, die NetServer-Funktionen nutzen sollen, in der SSO-Umgebung definiert sind – in diesem Fall bedeutet dies, dass für einen IBM i-Benutzer die Zuordnung zum Benutzerkonto im Active Directory erfolgt ist.

Sicher ist sicher – und entsprechend sollte man die wesentlichen Konfigurationskomponenten des NetServers sichern, bevor maßgebliche Veränderungen, wie beispielsweise das Einrichten von Single Sign-on für den NetServer, durchgeführt werden.

Die Konfigurationseinstellungen für den NetServer finden wir im IFS des System i wieder. Dort befindet sich u. a. das Verzeichnis /QIBM/UserData/OS400/NetServer. Dieses Verzeichnis sollte regelmäßig gesichert werden, denn darin befinden sich die Grundeinstellungen für den Betrieb des NetServers.

Die einzelnen Bereiche innerhalb des Ordners NetServer beinhalten die folgenden Komponenten:

– Qazlscfg
  Diese Datei beinhaltet die Konfigurationseinstellungen des NetServers.
– Qazlsshr
  Diese Datei beinhaltet die verteilten Informationen.
– Qazlsextxxx
  Diese Datei beinhaltet die Informationen zur Textumsetzung für die verschiedenen Freigabenamen. Diese Datei ist nicht auf jedem System vorhanden.



Konfigurationsdateien im IFS

In besonders großen Umgebungen kann unter Umständen nicht sichergestellt werden, dass alle Anwender zum Zeitpunkt der SSO-Aktivierung für den NetServer konfiguriert sind. In diesem „Ausnahmeszenario" kann man mit einigen Zusatzschritten die SSO-Aktivierung für den NetServer-Bereich so durchführen, dass eine Verwendung von verschlüsselten Kennworten oder die Kerberos-Authentifizierung genutzt werden.

Ich empfehle, diese Option nur für i5/OS-Versionen ab V5R3 zu verwenden!

Ab V5R3 ist im System i Navigator die Option enthalten, die Authentifizierung des NetServers sowohl über verschlüsselte Kennwörter als auch über Kerberos durchzuführen. Aber Vorsicht – ein unbedachter Versuch, mal eben SSO auch für den NetServer zu aktivieren kann im schlimmsten Fall dazu führen, dass Anwender nicht mehr mit den Daten des NetServers arbeiten können!

Wie kann nun der NetServer für den Gebrauch von verschlüsselten Kennwörtern und SSO konfiguriert werden? Die Lösung liegt in einer der Grundtechniken, die SSO bzw. EIM im System i-Umfeld nutzen – der Namensauflösung im Netzwerk.

Um einen Parallelprozess der Authentifizierung für den NetServer zu implementieren, ist die Verwendung eines Namens notwendig, der noch nicht verwendet wird. Damit kann dann eine Authentifizierung über SSO erfolgen, während die bereits verwendeten NetServer-Zugriffe wie gewohnt über die Kennwortauthentifizierung erfolgen. Dieses Verfahren erfordert jedoch einige Zusatzarbeit – müssen doch z. B. die Netzlaufwerkfreigaben auf den Clients unter Umständen angepasst werden.

5.4.3.3
Seite 4

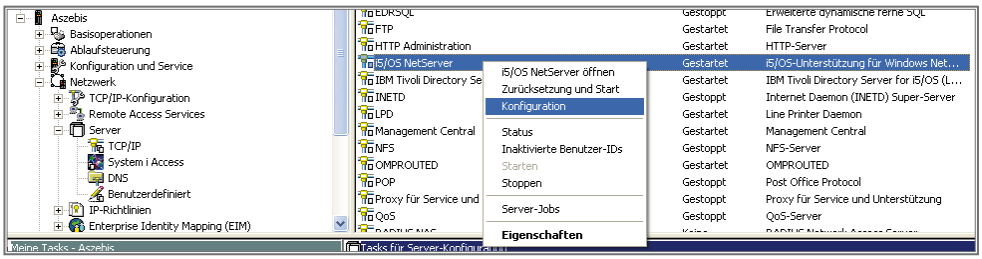
Wichtig ist in diesem Zusammenhang, dass für den eigentlichen NetServer-Namen kein Prinzipal erstellt wird. Das Prinzipal kann beispielsweise über den Assistenten im System i Navigator generiert werden. Würde für den NetServer mit dessen ursprünglichem Namen ein solches Prinzipal erstellt und mit der AD konfiguriert, dann geht Windows „gnadenlos“ davon aus, dass SSO für die Authentifizierung genutzt wird – ein nicht korrekt für SSO konfiguriertes Benutzerprofil wäre somit nicht mehr in der Lage, auf die Ressourcen des NetServers zuzugreifen.

Die Lösung für den Parallelprozess der Authentifizierung liegt also in der DNS-Namensauflösung. Sollten Sie keinen neuen Namen vergeben können, können Sie alternativ dazu auch einen DNS Alias verwenden. Mit diesem Namen können anschließend die erforderlichen SSO-Komponenten in Form von beispielsweise dem Service-Prinzipal-Namen erstellt werden. Der DNS Alias dient zur Authentifizierung des NetServers über Kerberos bzw. Single Sign-on, während der ursprüngliche Definitionsbereich des NetServers wie gewohnt mittels Kennworten verwendet werden kann.

Die Basis für den Einsatz der SSO- bzw. EIM-Verwendung zusammen mit dem NetServer ist die korrekte Konfiguration von EIM auf dem System i. IBM hat im System i Navigator ein Prüfprogramm implementiert, mit dem die Voraussetzungen für das Einrichten von SSO zusammen mit dem NetServer geprüft werden können. Nur wenn diese Prüfung erfolgreich verläuft, sollten die weiteren Konfigurationsschritte für das Einrichten von Kerberos und EIM zusammen mit dem NetServer fortgesetzt werden!

Testen wir zunächst die EIM-Einrichtung auf dem System i:

– Öffnen Sie den System i Navigator und stellen Sie eine Verbindung zum gewünschten System her.
– Wählen Sie die Option „Netzwerk / Server / TCPIP“.
– Wählen Sie in der Serverübersicht den Eintrag „NetServer“ mit der rechten Maustaste aus.
– Wählen Sie die Option „Konfigurieren“.



NetServer konfigurieren

Mit dieser Auswahl startet der Konfigurationsassistent für den NetServer.

Konfigurationsassistent für den NetServer

– Klicken Sie so lange auf „Weiter“, bis Sie in die Anzeige zur Auswahl der Authentifizierungsmethode gelangen. Die nachfolgende Abbildung zeigt dazu ein Beispiel:

Authentifizierungsmethode

5.4.3.3
Seite 6

– Selektieren Sie hier (aber nur probeweise!!!) die Option „Kennwörter/ Netzwerkauthentifizierung“
– Wählen Sie in der Folgeanzeige die Option „Konfiguration prüfen“:

Konfiguration prüfen

– Ist das Prüfergebnis positiv, dann kann mit der Einrichtung von EIM für den NetServer-Bereich begonnen werden – aber erst später. Jetzt brechen wir den Konfigurationsassistenten zunächst wieder ab – denn es gibt noch weitere Vorarbeiten, die wir erledigen müssen!

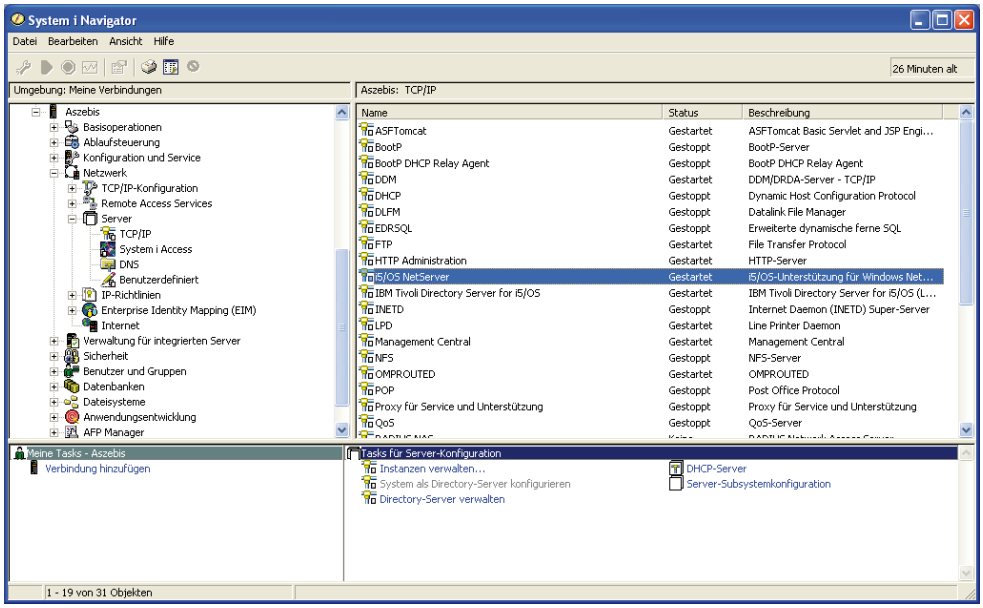
Sollte das Prüfergebnis zeigen, dass erforderliche Einstellungen noch nicht vorhanden sind, dann wird ein zusätzlicher Assistent gestartet:

Hinweis auf erforderliche Änderungen

Um sowohl „normale“ Kennwortauthentifizierung als auch SSO zusammen mit dem NetServer verwenden zu können, konfigurieren wir nach der Sicherung der NetServer-Konfiguration die Einstellungen mittels Assistent im System i Navigator.

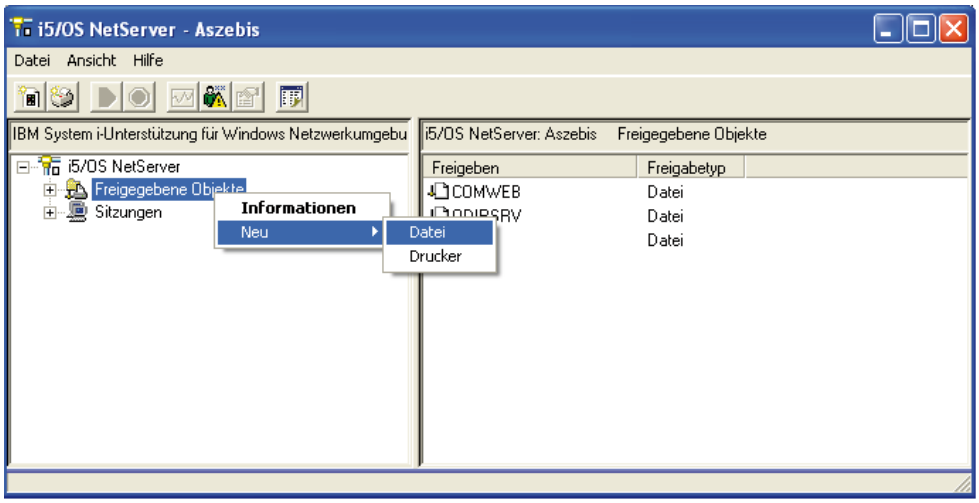
– Wählen Sie im System i Navigator den Eintrag „Server / Netzwerk / Server / TCPIP“.

– Selektieren Sie den Eintrag „i5/OS NetServer“ durch einen Doppelklick.



Auswahl NetServer

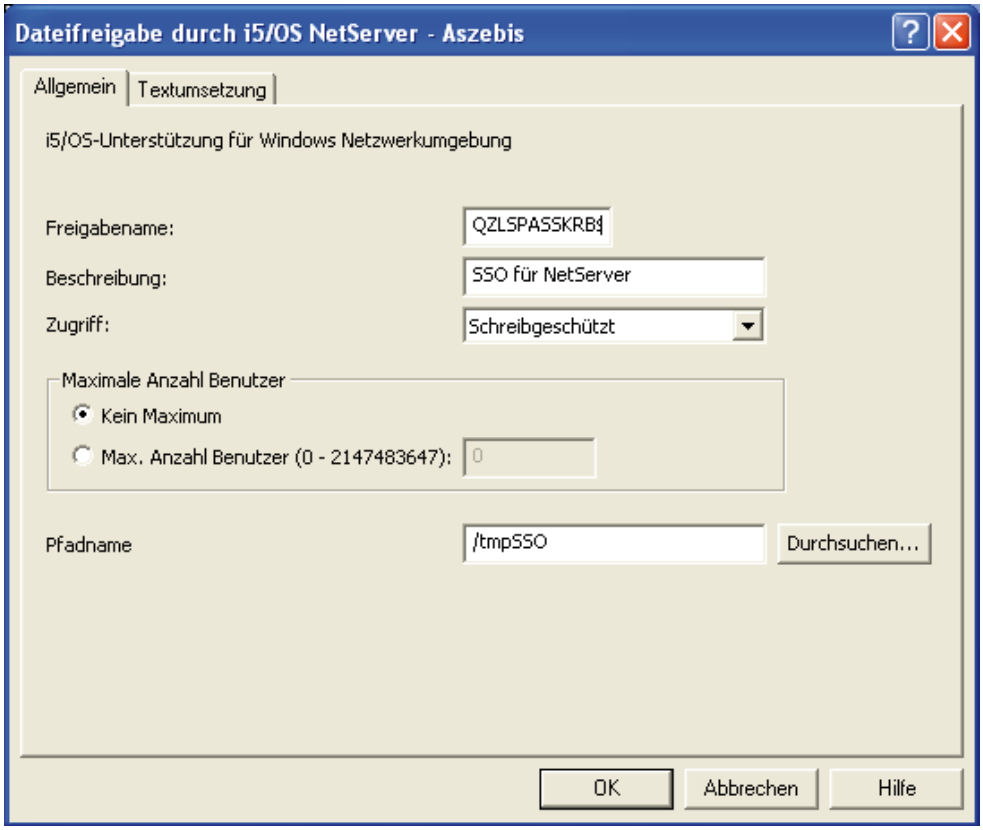
– In dem sich öffnenden Fensterbereich selektieren wir nun auf der linken Seite den Eintrag „Freigegebene Objekte“ mit der rechten Maustaste und wählen anschließend die Option „Neu / Datei“.



Auswahl NEU

5.4.3.3
Seite 8

– In dem sich öffnenden Definitionsbereich erfolgen nun die Angaben für die neue Freigabe. Dazu geben wir die folgenden Werte ein:
  - Namen für die Freigabe (im Beispiel QZLSPASSKRB$)
    Dieser Freigabename ist ein Sondername, über den der NetServer veranlasst wird, sowohl verschlüsselte Kennwörter als auch Kerberos-Verbindungen zu akzeptieren.
  - eine Beschreibung
  - Zugriff (schreibgeschützt)
  - einen Pfadnamen (dieser darf nicht existieren!)

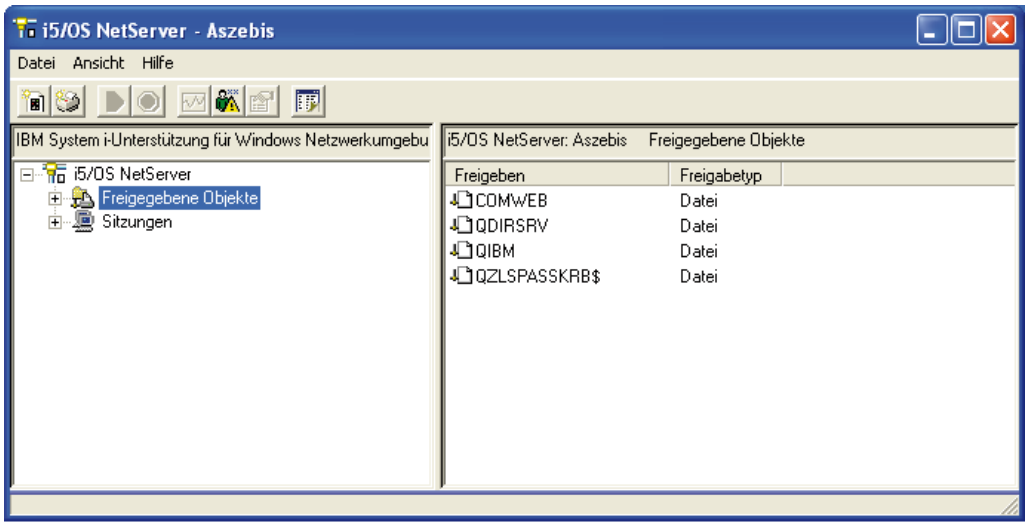


Neue Dateifreigabe

Achtung!

Für den Parallelbetrieb der Authentifizierung am NetServer mit verschlüsselten Kennwörtern und SSO ist der Freigabename QZLPASS-KRB$ zwingend vorgeschrieben!!

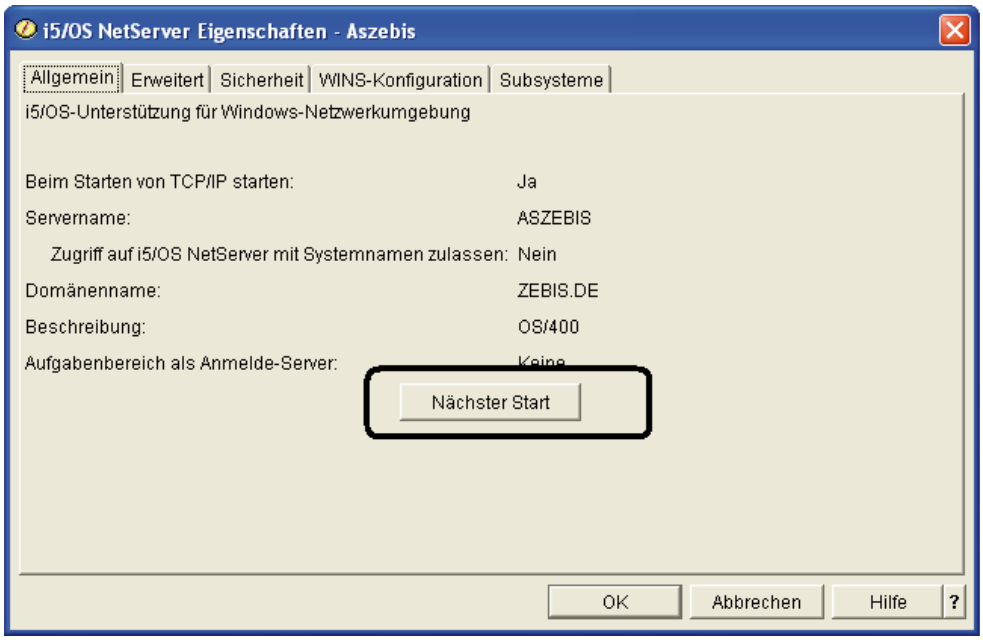
Nach der Erstellung der neuen Freigabe sollte diese in der Übersicht der freigegebenen Objekte erscheinen:



Freigegebene Objekte

Um nun den NetServer für die Verwendung beider Authentifizierungsverfahren nutzen zu können, bedarf es noch einer Anpassung am NetServer selbst. Auch diese nehmen wir im System i Navigator vor:

– Wählen Sie den Eintrag „Netzwerk / Server / TCPIP“.
– Klicken Sie mit der rechten Maustaste auf den Eintrag „NetServer“.
– Wählen Sie die Option „Eigenschaften“.



NetServer-Eigenschaften

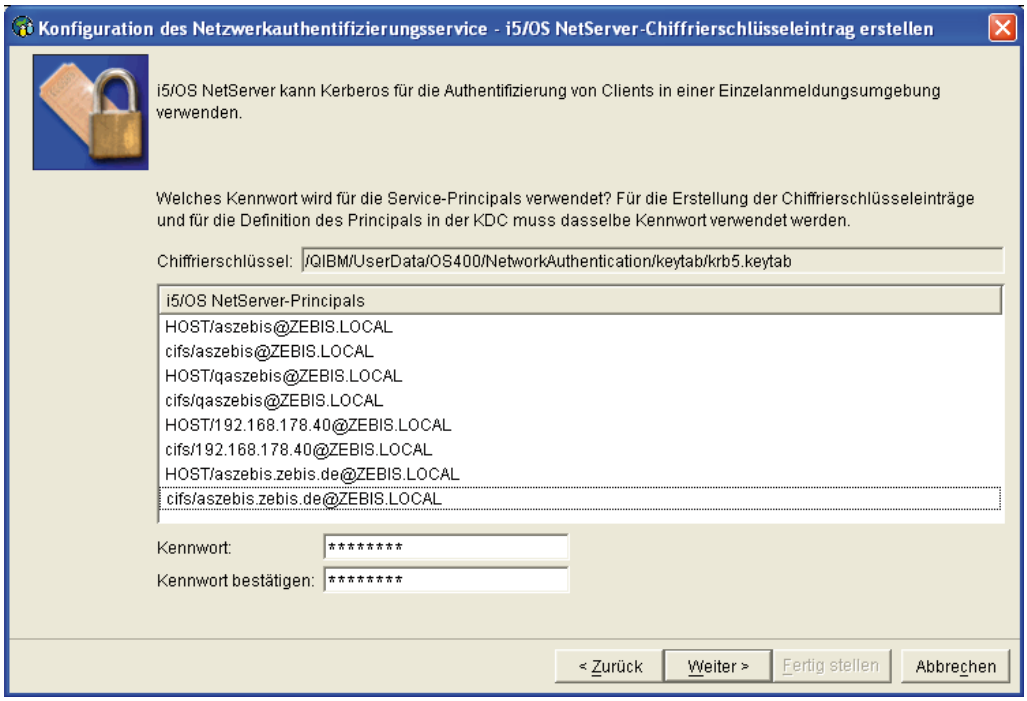
– Wählen Sie die Option „Nächster Start“.

5.4.3.3
Seite 10

Wird EIM für den NetServer aktiviert, dann wird eine ganze Anzahl von möglichen NetServer-Namen generiert, die mit Kerberos genutzt werden können. Dies können z. B. sein:

– HOST/ASZEBIS (für die Verwendung mit z.B. Windows XP
– cifs/ASZEBIS (für die Verwendung mit z.B. WIN2000)
– HOST/192.168.178.4@ASZEBIS.ZEBIS.DE

Die folgende Abbildung zeigt eine Auflistung einiger der erzeugten Prinzipals für die Verwendung mit dem NetServer:



Prinzipals für den NetServer

Sollten die Prinzipals für die Verwendung mit dem NetServer fehlen, dann können sie wahlweise über den System i Navigator, den 5250-Befehl oder über PASE erzeugt werden.

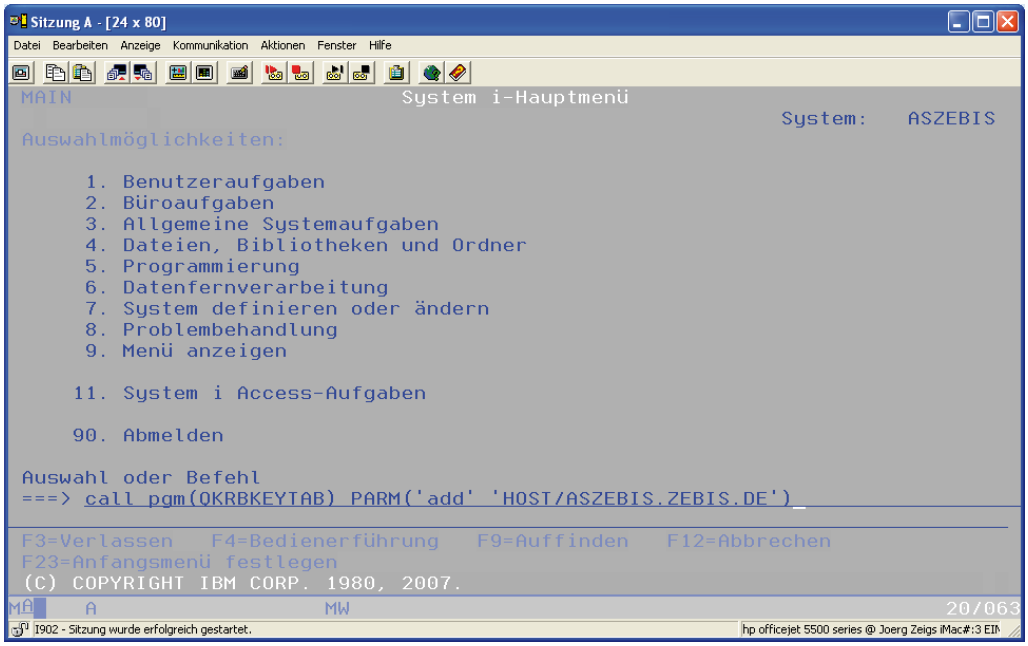
Um ein Prinzipal mit 5250-Befehlen zu erstellen, gehen wir wie folgt vor:

Geben Sie in der 5250-Befehlszeile den folgenden Befehl ein:

```
CALL PGM(QKRBKEYTAB) PARM('add' 'HOST/hostname')
```

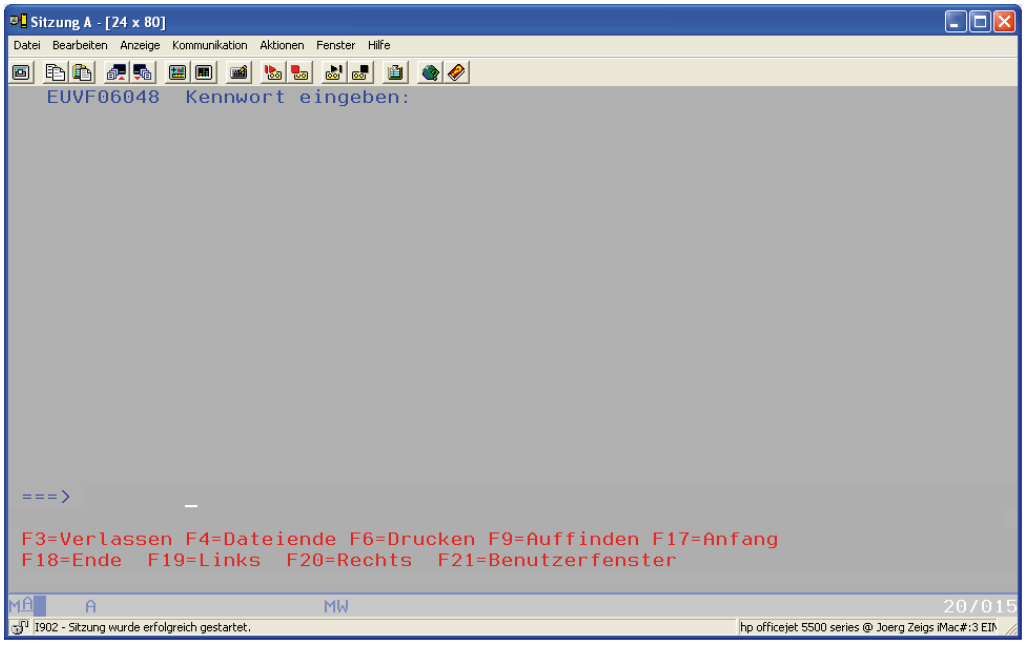
Ersetzen Sie hierbei den Platzhalter „hostname“ durch den voll qualifizierten Hostnamen oder die IP-Adresse des System i. Ein Beispiel dazu zeigt der folgende Befehl:

```
call pgm(QKRBKEYTAB) PARM('add' 'HOST/ASZEBIS.ZEBIS.DE')
```



CALL QKRBKEYTAB

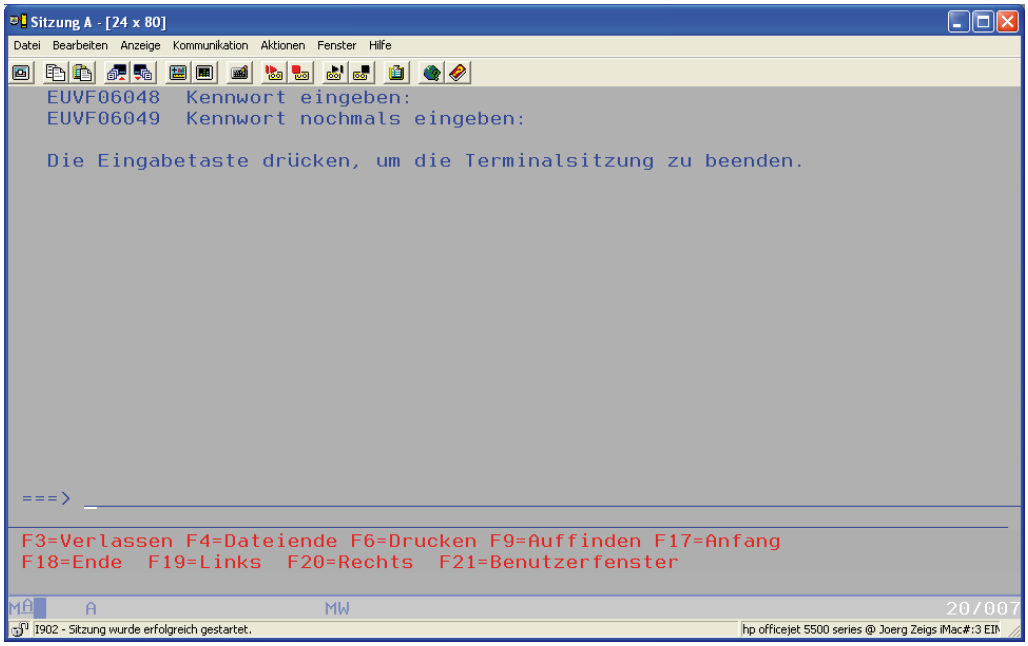
Mit dem Aufruf des Programms QKRBKEYTAB gelangen wir in ein Eingabefenster, in dem wir das Kennwort für das Generieren der Prinzipals zweifach eingeben müssen:



Kennworteingabe

Nachdem das Kennwort das zweite Mal eingegeben wurde, ist das Prinzipal erstellt und die Terminalsitzung kann durch Drücken der Eingabetaste wieder verlassen werden.

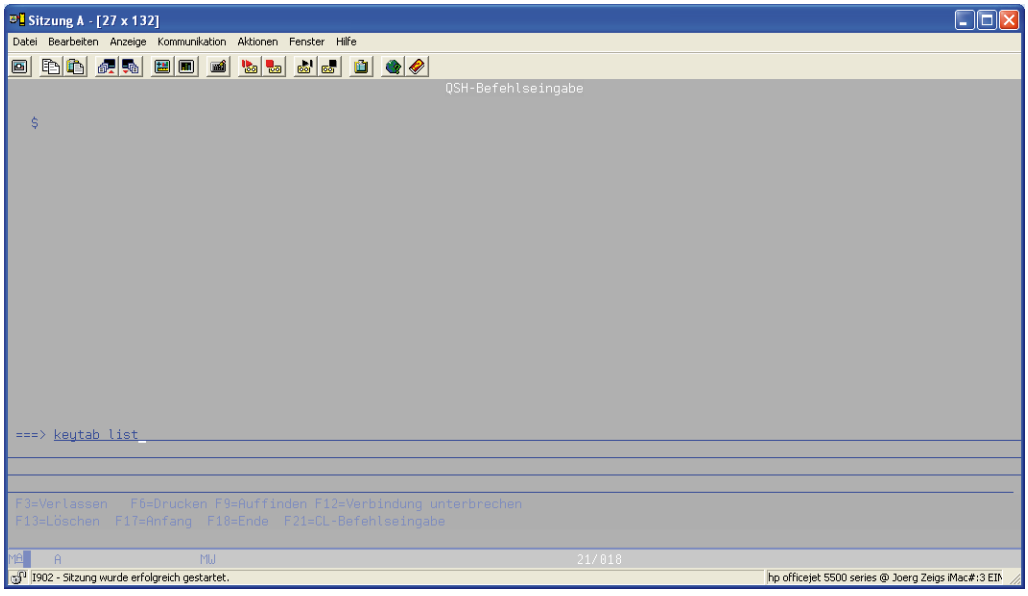
5.4.3.3
Seite 12



Abschluss der Eingabe

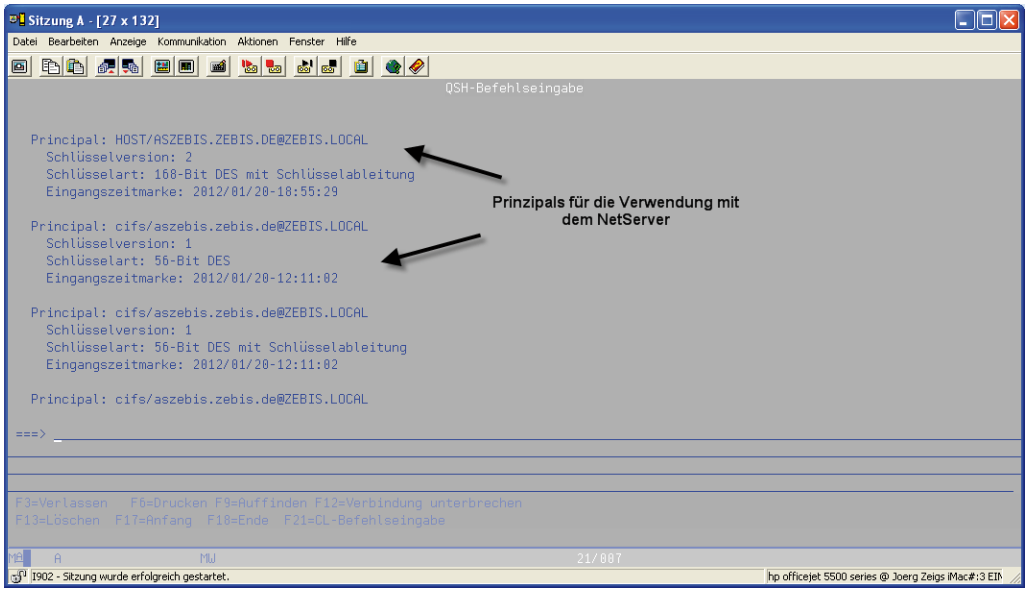
Mit Hilfe des QSHELL-Befehls keytab können wir das Ergebnis prüfen. Dazu starten wir zunächst QSHELL durch Eingabe des Befehls QSH in der Befehlszeile.

In die QSHELL-Befehlszeile geben wir nun den Befehl zur Auflistung der Prinzipals ein. Er lautet *keytab list:*



Keytab list

Das Ergebnis der Ausführung von keytab list ist eine Auflistung aller konfigurierten Prinzipals. Suchen Sie nach den Einträgen HOST und cifs – dieses sind die Prinzipals für die Verwendung mit dem NetServer:



Auflistung Prinzipals

Sind die betreffenden Prinzipals in der Auflistung enthalten, dann können sie nach Abstimmung mit dem Active Directory für Single Sign-on für den NetServer verwendet werden.

Noch einige Tipps zur Verwendung des NetServers:

In der Regel empfiehlt es sich, EIM schrittweise einzurichten und solche Anwendungen, die problemlos einen Parallelbetrieb der kennwortgestützten Anmeldung und des SSO bieten, zuerst testweise für SSO und Kerberos zu konfigurieren. Die im System i Navigator erstellte BAT-Datei kann bei Bedarf um die zusätzlichen SSO-Dienste erweitert werden und lässt sich in der Regel problemlos in der AD (Windows Active Directory) einspielen.

IBM hat in der Vergangenheit einige Sonderfreigabenamen eingeführt, die unterschiedliche Aufgaben übernehmen. Dies sind u. a.:

QZLSPASSKRB$
Dieser Name für eine NetServer-Freigabe erlaubt den Parallelbetrieb von Authentifizierungen über verschlüsselte Kennwörter und Kerberos (SSO/EIM). Nur wenn beide Verfahren genutzt werden sollen (z. B. zur Übergangszeit während der Einrichtung von SSO), dann sollte diese Sonderfreigabe konfiguriert und eingesetzt werden).

5.4.3.3
Seite 14

QZLSPWDANY$
Mit dieser Sonderfreigabe akzeptiert der NetServer auch Kennwörter im Windows NT-Format.

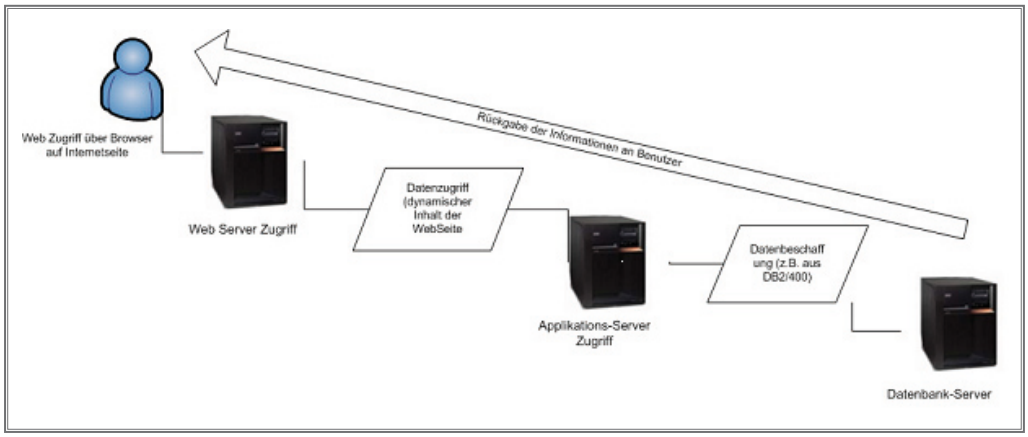
QZLSVUSRS$
Dieser Sondername wurde/wird für Multi-User-Unterstützung im Windows NT-Bereich verwendet.

Datenaustausch/ferner Datenzugriff

Lange Zeit war die ehemalige AS/400 ein in sich geschlossenes System mit nur wenigen Möglichkeiten des Datenaustausches mit der Außenwelt. Diese Aussage ist heute sicher in keiner Weise mehr zutreffend – leider jedoch immer wieder Gegenstand von Diskussionen, wenn man mit „Kennern" redet, die vor 20 Jahren einmal Kontakt mit den ersten Modellen der AS/400 Familie inklusive deren verschiedenen Nachfolgern hatten.

Dass die Kommunikation mit anderen Systemen und auch der Datenaustausch mit den heterogenen Datenbanken heute Standard ist, braucht sicher nicht erwähnt oder gar vertieft zu werden.

Die Verfügbarkeit von flexiblen Anwendungen, der Einsatz des System i als Anwendungsserver, als Datenbankserver oder auch als Webserver zeigen, wie weit die Maschine mitsamt den Anwendungen und Daten einsetzbar ist. Mit genau diesen Fähigkeiten wurde dem System auch eine Notwendigkeit mitgegeben, welche die Kommunikation mit anderen Systemen ermöglicht. Diese Kommunikation beinhaltet natürlich auch den Zugriff auf die Daten und den Datenaustausch.



Datenzugriff schematisch

Dezentralisierung hat in einigen Bereichen dazu geführt, dass ein System i nicht mehr als „Eine-für-Alles-Maschine" zum Einsatz kommt, sondern der Einsatzbereich auf eine Teilfunktion beschränkt wird. So finden wir zum Beispiel System i Installationen, bei denen auf einer Maschine lediglich Datenbankfunktionen ausgeführt werden. Anwendungen, gleich welcher Art, werden auf einem anderen System i verarbeitet. Dies kann aus Sicherheitsgründen und auch im Sinne der Systemarchitektur sicher sinnvoll sein. Dennoch ist natürlich der Einsatz eines System i für all die in der vorgehergehenden Abbildung gezeigten Teilbereiche in den meisten Fällen sinnvoll und zu empfehlen.

Aber nicht nur in einem solchen Szenario, bei dem Webanwendungen auf die Daten einer fremden Maschine zugreifen, sondern bei der Übertragung von Informationen von einem auf ein anderes System sind verschiedene Kriterien entscheidend.

Kompatibilität, Sicherheit, Verfügbarkeit und Schnelligkeit sind nur einige Anforderungen, die an den Datenaustausch gestellt werden.

Meist sind es Standards, die von den gängigen Betriebssystemen mehr oder weniger übernommen und individuell umgesetzt sind. TCP/IP stellt in nahezu jedem Fall die Basis für das Übertragen von Daten von einem System auf ein anderes System dar.

Datenaustausch muss dabei nicht das Übertragen von Informationen von einem System auf das andere bedeuten, sondern kann bei Bedarf auch den Zugriff auf Daten von einer Anwendung, die auf einem externen System ausgeführt wird, bedeuten.

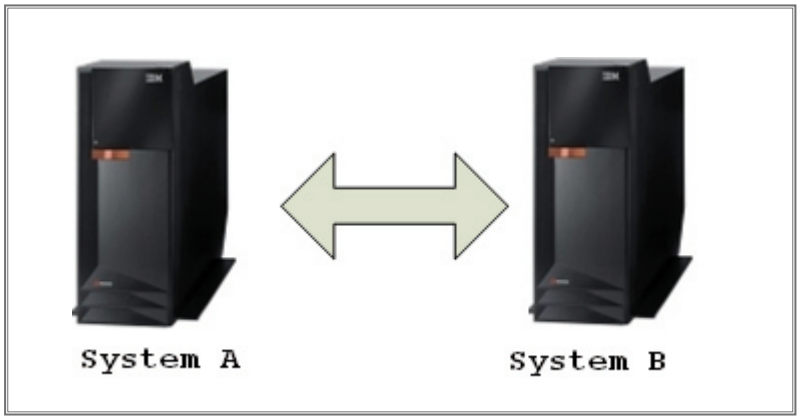
Bezogen auf das System i stehen verschiedene Techniken zur Verfügung, mittels deren wir Informationen von einem System auf ein anderes System übertragen können.

Als Beispiele kommen in diesem Zusammenhang in Frage:

- Verwendung des NetServers
- Nutzung von NFS
- Einsatz des QFileSvr.400 Dateisystems
- Datenaustausch mittels FTP

Schauen wir uns den Datenaustausch bzw. den fernen Datenzugriff einmal anhand einiger Fallbeispiele an.

Ausgehend davon, dass auf einem System A eine Anwendung ausgeführt wird, die auf die Daten eines „fernen“ Systems B zugreifen muss, dann kann dieser Durchgriff auf unterschiedliche Arten und Weisen realisiert werden.



Datenaustausch zwischen zwei Maschinen

Es kommt letztlich auch auf die Form der zu verarbeitenden Daten an und wie bzw. wo diese gespeichert werden. Dieses Faktum als auch die übrigen Eckdaten bestimmen letztlich welche Form des Zugriffs und welche Technologie eingesetzt werden können.

Nachfolgend finden Sie einige Beispiele für das Übertragen von Informationen zwischen verschiedenen Systemen bzw. den fernen Zugriff auf diese Daten.

5.4.4
Seite 4

QFileSvr.400

Eine bereits seit langem zur Verfügung stehende Lösung (Seit V3R1M0 des OS/400 Betriebssystems) basiert auf dem Einsatz von QFileSvr.400. QFileSvr.400 ist jedoch eine Lösung, die heute kaum noch zum Einsatz kommt – zu speziell sind die Einschränkungen oder Einsatzbereiche.

Bei QFileSvr.400 handelt es sich genau genommen um eine spezielle Form des Dateisystems, welches in der System i Welt unter anderem im 5250 Bereich als auch in dem Systems Navigator implementiert ist.

Während die Verbindung über QFileSvr.400 recht einfach zu etablieren und der Datenaustausch zuverlässig durchgeführt werden kann, ist die Geschwindigkeit, die für den Zugriff auf die Daten des fernen Systems bereitsteht, nicht wirklich zufriedenstellend. Deshalb muss man bei einer Entscheidung für eine Datenaustauschmethode natürlich auch die Leistungsfähigkeit der zur Verfügung stehenden Alternativen sehr wohl abwägen.

Die Basis für den Datenaustausch mit QFileSvr.400 bildet TCP/IP mit den verschiedensten Standards, die auch mit dem System i bereitstehen.

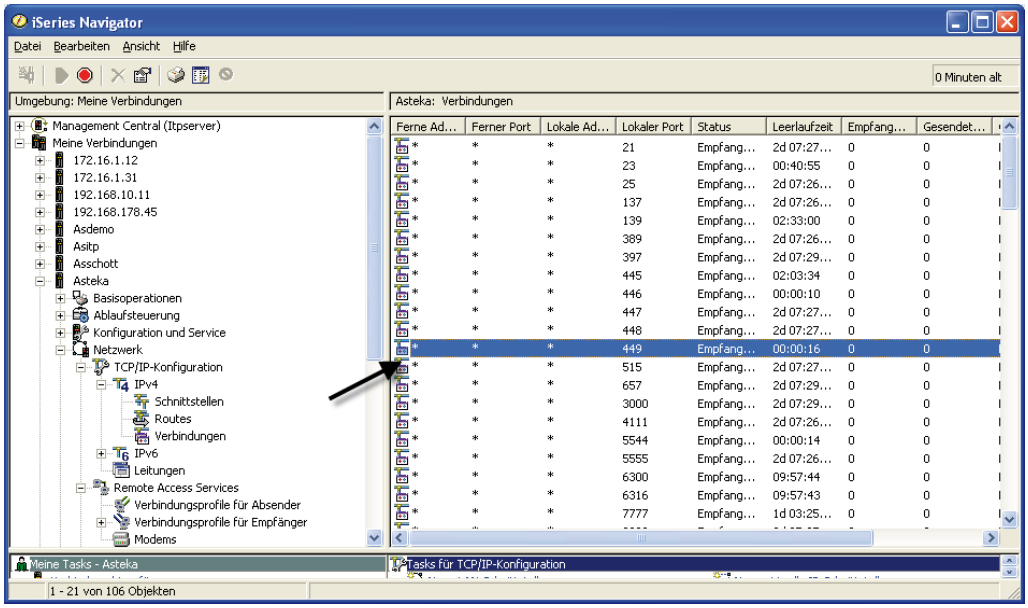
Anmerkung

Durch die Erweiterungen im TCP/IP Bereich durch IPv6 können mit den neueren Betriebssystemversionen die neusten Standards und Funktionen für TCP/IP Kommunikation genutzt werden. Auch wenn QFileSvr.400 eine mittlerweile in die Jahre gekommene Lösung ist, so wird auch hier IPv6 unterstützt!

Die Kommunikation wird über den Port 449 durchgeführt. Bevor man den QFileSvr.400 Bereich konfiguriert, sollte geprüft werden, ob dieser Port anderweitig genutzt wird.

Die folgende Abbildung zeigt die Übersicht der Ports in dem System i Navigator.

5.4.4.1
Seite 2



Portübersicht

Der Zugriff auf den Inhalt des Dateisystems eines System i von einem anderen System i lässt sich beispielsweise über QFileSvr.400 realisieren. Wo liegen nun die Vor- oder auch Nachteile bei dem Einsatz dieser Datenaustauschmethode?

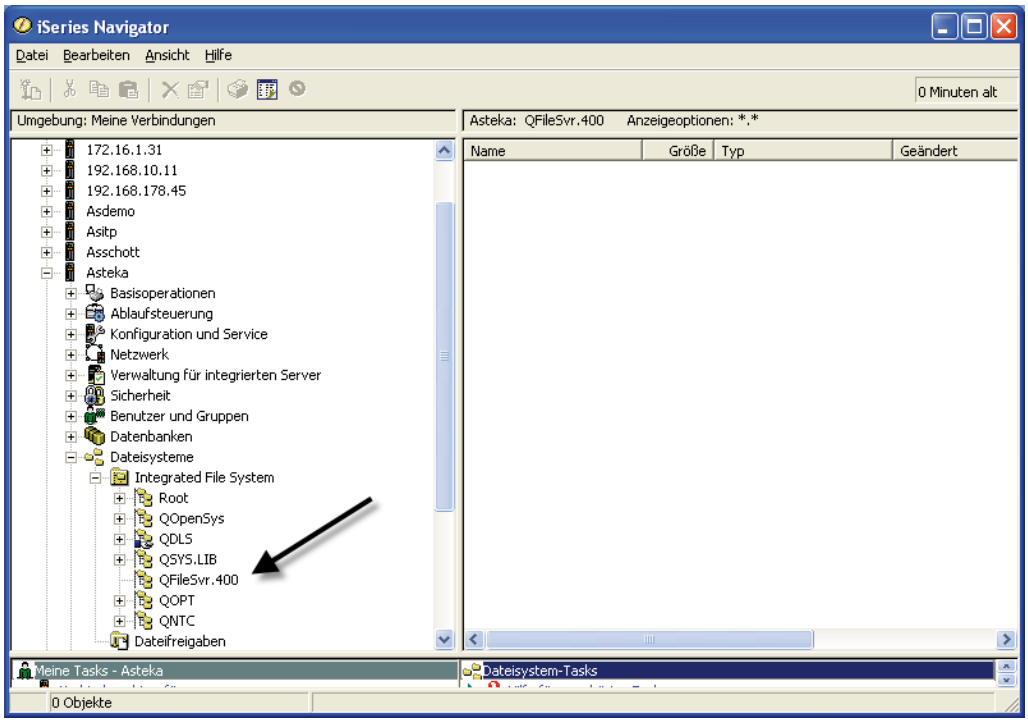
Grundsätzlich birgt QFileSvr.400 sicherheitsrelevante Besonderheiten: Einmal konfiguriert, wird über QFileSvr.400 der gesamte Inhalt des IFS für ein fernes System sichtbar und nutzbar. Ein beschränkter Zugriff auf einzelne Verzeichnisse oder IFS Inhalte ist mit QFileSvr.400 nicht möglich. Damit verbunden ist ein nicht unerheblicher administrativer Aufwand, um die Sicherheitstore, die mit dieser „Lösung“ geöffnet werden, so zu schließen, dass die Daten auch tatsächlich sicher sind und nicht Dritten ungewollt in die Hände fallen können.

Eine weitere Besonderheit bei der Verwendung von QFileSvr.400:

Mit dieser Lösung lässt sich der Inhalt des IFS für fremde Systeme nutzbar machen. Allerdings ist damit der Zugriff auf folgende IFS Inhalte beschränkt:

- Root
- QOpenSys
- QLANSrv
- QDLS
- QOPT

Die folgende Abbildung zeigt den Aufbau des Zugriffs auf den Inhalt eines i5 in dem System i Navigator. Neben verschiedenen anderen Einträgen befindet sich hier auch der Bereich „QFileSvr.400“. In diesen Bereich gelangen wir über den Navigationseintrag „Dateisysteme / Integrated File System“.



QFileSvr400

Unterhalb des Eintrages „QFileSvr.400“ befindet sich ein Verzeichnis mit dem Namen des fernen Systems. Dieser Eintrag dient quasi als eine Art Brücke zwischen den beiden System i Installationen.

Ein wichtiger Teilbereich ist in der Auflistung nicht enthalten – QSYS.LIB. Damit fällt der Einsatz von QFileSvr.400 für das Arbeiten mit den Inhalten der Bibliotheken oder Schemata eines i5 weg.

Und noch eine Besonderheit:

Für den Zugriff auf die Daten eines fernen Systems muss eine Authentifizierung erfolgen. Diese geschieht über das Benutzerprofil und ein Kennwort. Beide (!!!) müssen auf den zwei Maschinen identisch sein! Sicherheitsrichtlinien in einem Unternehmen können damit ein K. o.-Kriterium gegen den Einsatz von QFileSvr.400 sein – denn die abgestimmten Benutzerprofile und zugehörigen Kennworte sind ein Muss auf den beiden miteinander verbundenen Systemen. Eine sinnvolle Lösung des Abgleichs der Benutzerprofil- und Kennworteinstellungen kann der Einsatz von Single Signon Lösungen (SSO/EIM) sein. Mit Hilfe einer solchen Lösung kann die Authentifizierung auch im Bereich von QFileSvr.400 durchgeführt werden.

5.4.4.1
Seite 4

Abschließend sei noch erwähnt, dass zusammen mit der Nutzung von QFileSvr.400 viele der bekannten CL Befehle auf dem System i genutzt werden können. Es gibt aber auch eine Reihe von Befehlen, die nicht zusammen mit QFileSvr.400 zur Verfügung stehen. Datensicherungs- oder Wiederherstellungsbefehle lassen sich beispielsweise nicht auf einem fernen System absetzen.

Die folgende Auflistung zeigt die wesentlichen Befehle, die NICHT mit QFileSvr.400 genutzt werden können.

- ADDLNK
- APYJRNCHG
- CHGAUT
- CHGJRNOBJ
- CHGOWN
- DSPAUT
- DSPJRN
- ENDJRN
- RCLLNK
- RCVJRNE
- RST
- RTVJRNE
- SAV
- SNDJRNE
- STRJRN
- WRKOBJOWN
- WRKOBJPGP

Aber wie und wo kann nun diese Technologie eingesetzt werden?

Nehmen wir als Beispiel eine Webanwendung, die auf einer Partition einer i5 Installation ausgeführt wird. Nennen wir diese Partition „System A“. Diese Webanwendung läuft auf einer dedizierten Partition, um die Sicherheitseinstellungen und Zugriffe auf diese Webanwendungen regulieren zu können. Daten sind aus Sicherheitsgründen auf einer anderen Maschine abgelegt – dem „System B“. Damit kann man potentiellen Zugriffen auf die Daten entgegenwirken, da man beispielsweise den Zugriff auf die fernen Daten durch Sicherheitseinstellungen beschränken kann. Zwar bedeutet dies einen administrativen Zusatzaufwand – dieser kann jedoch, verglichen mit anderen Lösungen, durchaus akzeptabel und vertretbar sein!

Unsere Anwendung auf „System A“ nutzt folglich die Daten auf „System B“. Der Zugriff erfolgt mittels QFileSvr.400.

Datenaustausch mittels QFileSvr400

Auf dem System A muss sich in dem Bereich QFileSvr.400 ein Unterverzeichnis mit dem Namen „SYSTEM B“ befinden. Klickt beispielsweise ein Anwender auf diesen Eintrag, dann wird damit ein Prozess in Gang gesetzt, bei dem die Konfiguration ausgelesen und ein Zugriff auf den Inhalt des fernen Systems (System B) durchgeführt wird.

Die Basis für die Kommunikation und die Datenübertragung bzw. den Datenzugriff bilden besondere Jobs, die auf beiden Systemen ausgeführt werden müssen:

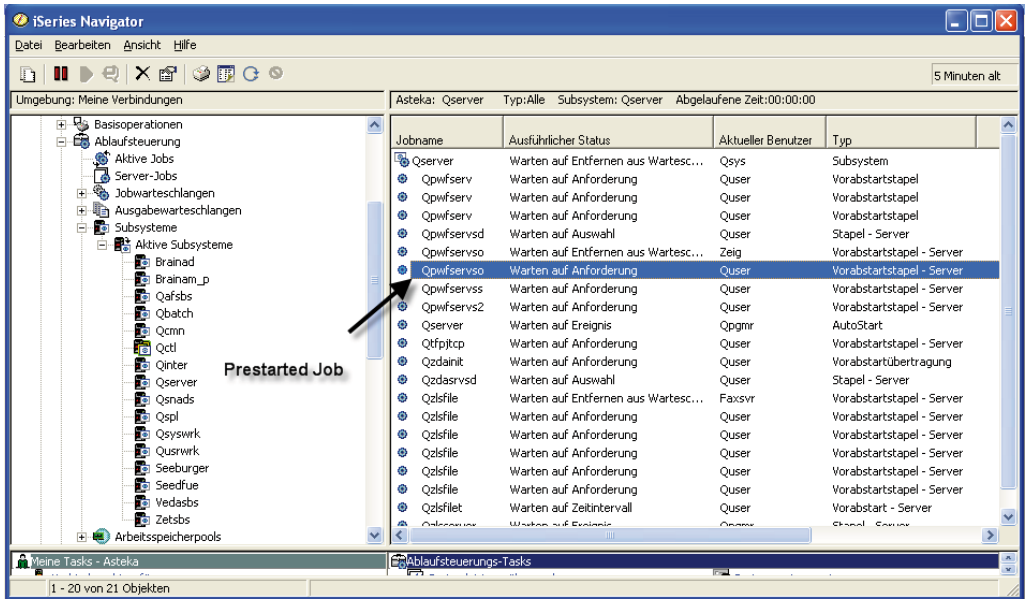
- QPWFSERVSO
- QPWFSERVSD

Diese Jobs laufen per Standard in dem Subsystem QSERVER. Das Subsystem QSERVER dient nicht nur der Ausführung dieser speziellen QFileSvr.400 Jobs, sondern ist die Grundlage für eine ganze Reihe von Serverjobs. Deshalb wird das Subsystem per Standard bei dem IPL des Systems mit gestartet, kann aber auch manuell über den Befehl STRSBS SBS(QSERVER) gestartet bzw. mit ENDSBS SBS(QSERVER) beendet werden.

Die beiden Jobs müssen sowohl auf dem System A als auch auf dem System B laufen!

Über den in dem Subsystem QSERVER laufenden Prestarted Job QPWFSERVSO wird im Zuge des Datenaustausches über QFileSvr.400 die Verbindung zwischen zwei Systemen aufgebaut.

5.4.4.1
Seite 6



QPWFSERVSO Prestarted Job

Anmerkung

In diesem Bereich werden auch einige Wartungsaufgaben für den Betrieb von QFileSvr.400 ausgeführt. Unter anderem wird alle zwei Stunden eine Prüfung durchgeführt, mit der bestehende Verbindungen in diesem Bereich analysiert werden. Wird eine Verbindung gefunden, die länger als zwei Stunden keine Aktivität aufweist, dann wird sie von dem System automatisch beendet.

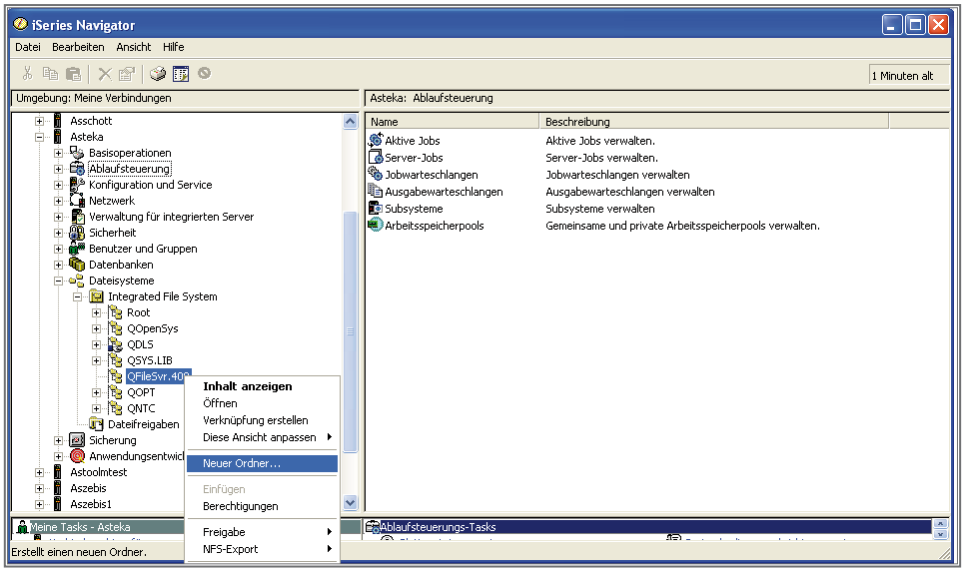
Bevor nun ein fremdes System auf den Inhalt einer Maschine zugreifen kann, muss auf der Maschine mit den Daten innerhalb des Verzeichnisses QFileSvr.400 ein Unterverzeichnis angelegt werden, das denselben Namen trägt wie der Hostname des fremden Systems.

Nehmen wir einmal an, wir wollen von unserem System A (das System mit der Webanwendung) auf die Daten des Systems B zugreifen. Der Hostname des Systems A lautet „SYSTEM A“. Folglich müssen wir auf dem System B innerhalb des Verzeichnisses QFileSvr.400 ein Unterverzeichnis mit dem Namen „SYSTEM A“ anlegen.

Das Anlegen eines Verzeichnisses kann auf unterschiedliche Weise erfolgen. Ideal für das Arbeiten mit dem Inhalt des IFS ist der System i Navigator (bzw. iSeries Navigator).

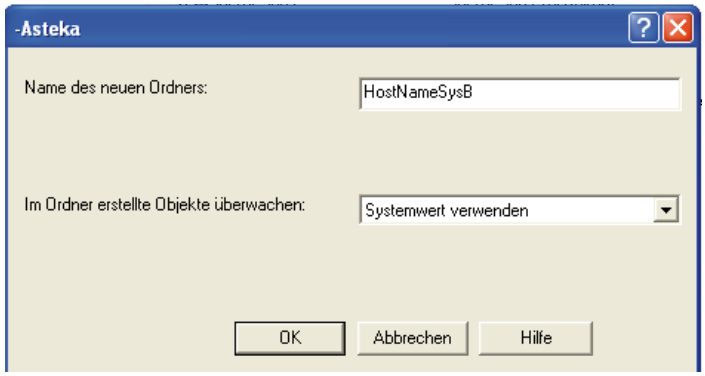
Ausführungsschritte

1. Stellen Sie eine Verbindung im iSeries Navigator zu dem System i her.
2. Erweitern Sie den Eintrag System i/Dateisystem/Integrated File System“.
3. Klicken Sie mit der rechten Maustaste auf den Eintrag „QFileSvr.400“.
4. Wählen Sie die Option „Neuer Ordner“.



Neuen Ordner anlegen

Verwenden Sie als Bezeichnung für den neu erstellten Ordner den Hostnamen des fernen Systems.

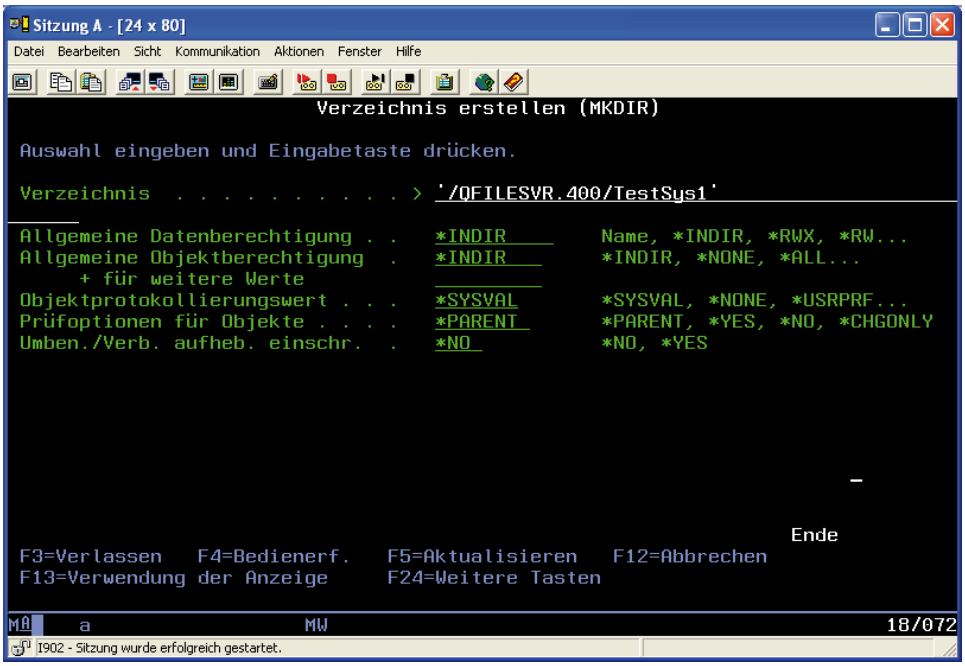


Details des neuen Ordners

5.4.4.1
Seite 8

Achtung!

Ein neues Verzeichnis können Sie auch mit dem 5250 Befehl „MKDIR“ erstellen. Dabei ist dann zu beachten, dass als Parameter „Verzeichnis“ der volle Verzeichnisname inklusive „/QFileSvr.400“ enthalten sein muss. Ein Beispiel dafür befindet sich in der nachfolgenden Abbildung.



MKDIR

Damit haben wir erst einmal die Grundvoraussetzungen auf dem „Datensystem“ geschaffen, damit das ferne System auf den Inhalt der Daten zugreifen kann.

Bevor die Daten jedoch von dem fernen System genutzt werden können, muss auf diesem System (System A) noch ein Verweis auf die fernen Daten angelegt werden. Mittels dieses Verweises ist das System dann in der Lage, die Daten in der fernen Quelle zu lokalisieren und abzurufen. Diesen Verweis bezeichnen wir auch als einen „symbolischen Link“.

Rufen wir uns nochmals die Konstellation in Erinnerung:

Auf dem System A läuft eine Webanwendung, welche auf die Daten des System B zugreift. Normalerweise würde das System A die Daten auch auf demselben System suchen und erwarten. Damit die Anwendung auf die fernen Daten zugreift, definierten wir den symbolischen Link in Form von „/application/webcontent/static“. Dieser Link verweist dann auf die Daten in Form von „/QFileSvr.400/SystemB/application/webcontent/static“.

Hinzufügen eines symbolischen Links:

Ausführungsschritte

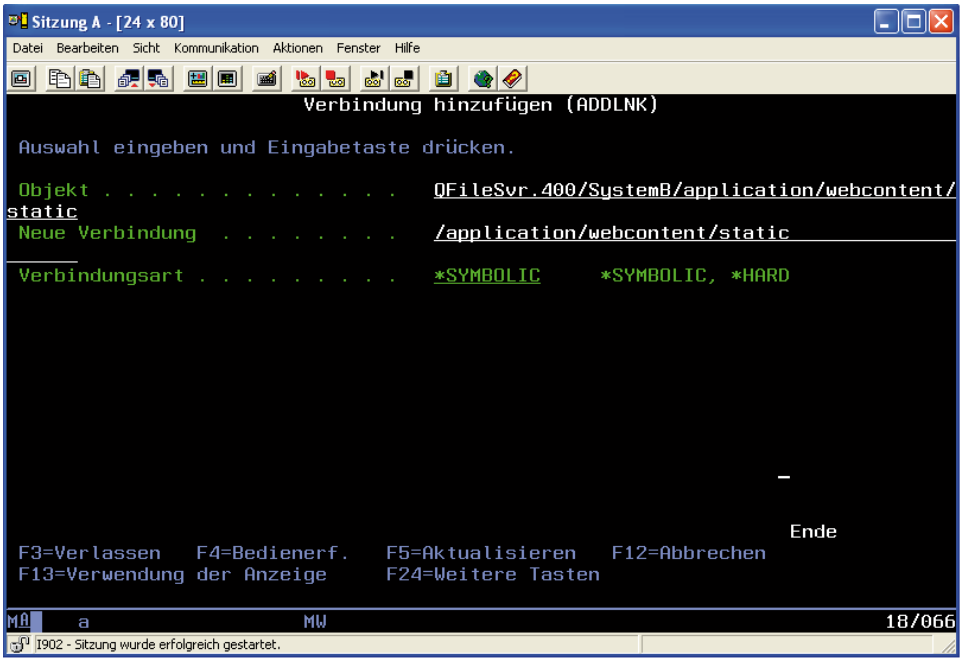
Geben Sie in der Befehlszeile den Befehl ADDLNK ein und rufen Sie dazu mit der F4 Taste die Bedienerhilfe auf.

Erfassen Sie in dem Parameter „Objekt“ den folgenden Wert: „/QFileSvr.400/SYSTEMB/application/webcontent/static“

In dem Parameter „Neue Verbindung“ wird der folgende Wert eingegeben: „/application/webcontent/static“

Als Verbindungsart geben wir „*SYMBOLIC an.

Verlassen Sie nun die Definitionseingabe für den symbolischen Link.



ADDLNK

Bevor wir nun die nächsten Schritte ausführen, sollte ein kleiner Test die vorherigen Definitionsschritte verifizieren:

Dazu stellen wir eine Testverbindung mit dem folgenden Befehl her:

```
WRKLNK ,/application/webcontent/static/*'
```

Damit sollte nun eine Auflistung des Inhalts des fernen Verzeichnisses ausgegeben werden. Ein Versuch, den Inhalt einer Datei einzusehen, sollte auch funktionieren. Können Sie diese Schritte erfolgreich ausführen, dann ist die Kommunikation zwischen den beiden Maschinen als Grundlage für den Einsatz der Webanwendung geschaffen.

5.4.4.1
Seite 10

OpenSSH

Sichere Verbindungen sind heute ein Muss im Bereich der Kommunikation innerhalb eines Unternehmens bzw. eines Unternehmens mit Partnern. Es gibt auf dem Markt eine Fülle von Techniken, mittels deren eine sichere Verbindung zu einem fernen Rechner aufgebaut werden kann. Natürlich spielen hier VPN-Verbindungen eine wichtige Rolle.

SSH (Secure Shell) ist eine weitere Technologie, mittels deren basierend auf TCP/IP sichere Verbindungen zu fernen Rechnern aufgebaut werden können. Ursprünglich vorwiegend im UNIX und Linux Umfeld eingesetzt, steht diese Technologie seit der Version V5R3 auch den IBM i Betreibern zur Verfügung. Über SSH können Verbindungen zwischen zwei Rechnern getunnelt werden. Auf Basis einer sicheren und verschlüsselten Verbindung lässt sich dann zwischen den Rechnern der Datenaustausch durchführen.

Diese sichere Verbindung basiert auf einem privaten und einem öffentlichen Schlüssel. Während der öffentliche Schlüssel auf dem Zielrechner verfügbar sein muss, benötigt man den privaten Schlüssel, um den Inhalt einer Übertragung zu entschlüsseln. Folglich können die Daten, welche mittels SSH übertragen werden, nur dann eingesehen und genutzt werden, wenn sie mit dem erforderlichen passenden Schlüssel entschlüsselt werden. Damit entfällt der Einsatz von Benutzerprofilen und Kennwörtern zur Datenübertragung.

Auf dem System i5 wird SSH in Form von OpenSSH unterstützt. Dabei handelt es sich um eine Open Source Version von SSH, in der einige unterschiedliche Werkzeuge und Funktionen implementiert sind, die mit dem System i eingesetzt werden können. SSH lässt sich im Umfeld der i5 dazu einsetzen, um Verbindungen zu anderen Servern und/oder Clients abzusichern. Betrachtet man die typischen Installationen von modernen Umgebungen, dann finden wir dort die unterschiedlichsten Einsatzbereiche für Datentransfer. Neben den klassischen Datenübertragungen sind dies verstärkt auch die HMC Installationen. Haben Sie auch daran gedacht, dass die Verbindung zur HMC und von dort zum IBM i abgesichert sein sollte? Auch in einem solchen Fall lässt sich OpenSSH einsetzen.

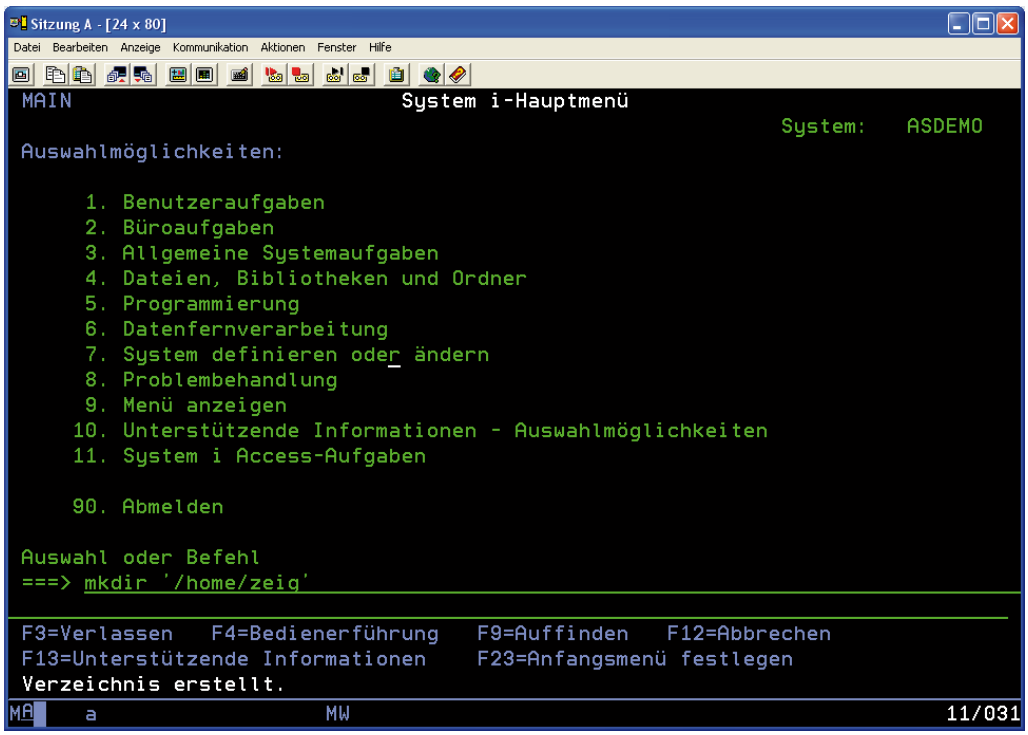
Wer sich mit der Übertragung von Daten mittels FTP auf i5 beschäftigt hat, der wird sicher die beiden Versionen des Einsatzes kennen: Interaktiver Datenaustausch und automatisierte Datenübertragung mittels Batchverarbeitung. Im letzteren Fall müssen die Informationen für den Aufbau der Verbindung in einer Datei enthalten sein. Leider beinhalten diese Informationen auch sicherheitskritische Elemente wie zum Beispiel Benutzer und Kennwort – in Klartext! Für wirklich sichere Datenübertragung wohl nur eingeschränkt geeignet! SSH kann hier eine sinnvolle Alternative darstellen.

5.4.4.2
Seite 2

Bei dem Einsatz von SSH sind in Bezug auf das IBM i folgende Besonderheiten zu berücksichtigen:

- Die für SSH eingesetzten Benutzerprofile dürfen maximal achtstellig sein!
- Jeder Benutzer muss über ein Homeverzeichnis in dem IFS verfügen.

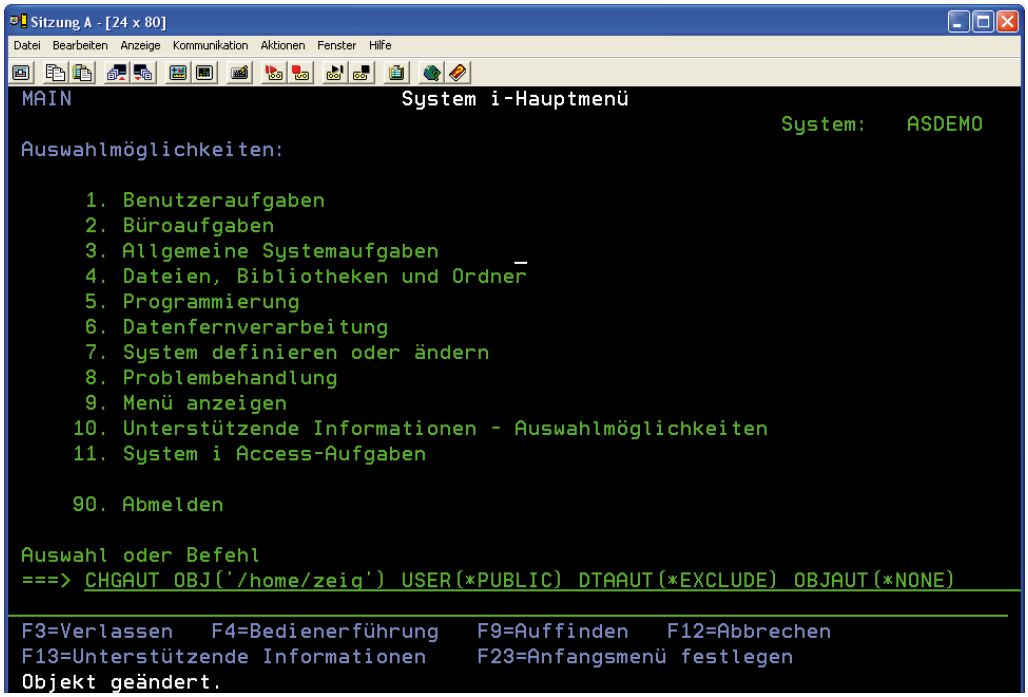
Das Homeverzeichnis wird über den Befehl MKDIR erstellt. Geben Sie dazu den Befehl wie folgt ein: mkdir ‚/home/benutzername' also z. B. mkdir ‚/home/zeig'.



Homeverzeichnis erstellen

Nachdem das Homeverzeichnis erstellt wurde, müssen wir noch die Berechtigungseinstellungen anpassen. Dazu geben wir den folgenden Befehl ein:

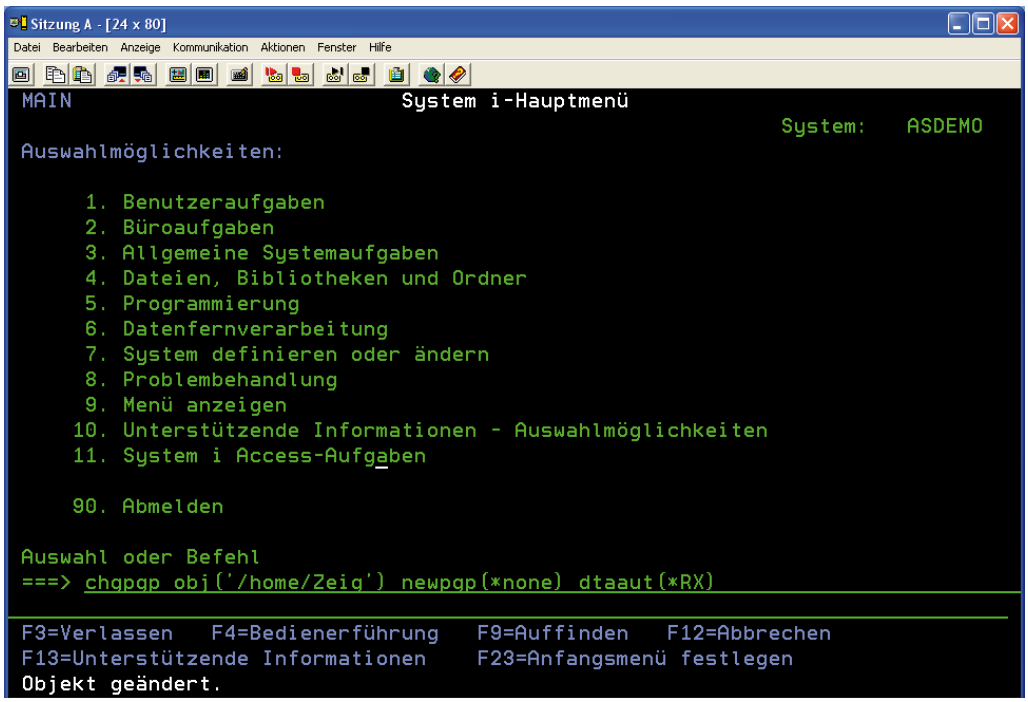
```
CHGAUT OBJ(,/home/benutzerprofil')
USER(*PUBLIC) DTAAUT(*EXCLUDE) OBJAUT(*NONE)
```



CHGAUT

Nun passen wir noch die Berechtigungen für das Verzeichnis an, indem wir die Primärgruppe ändern. Dazu verwenden wir den folgenden Befehl:

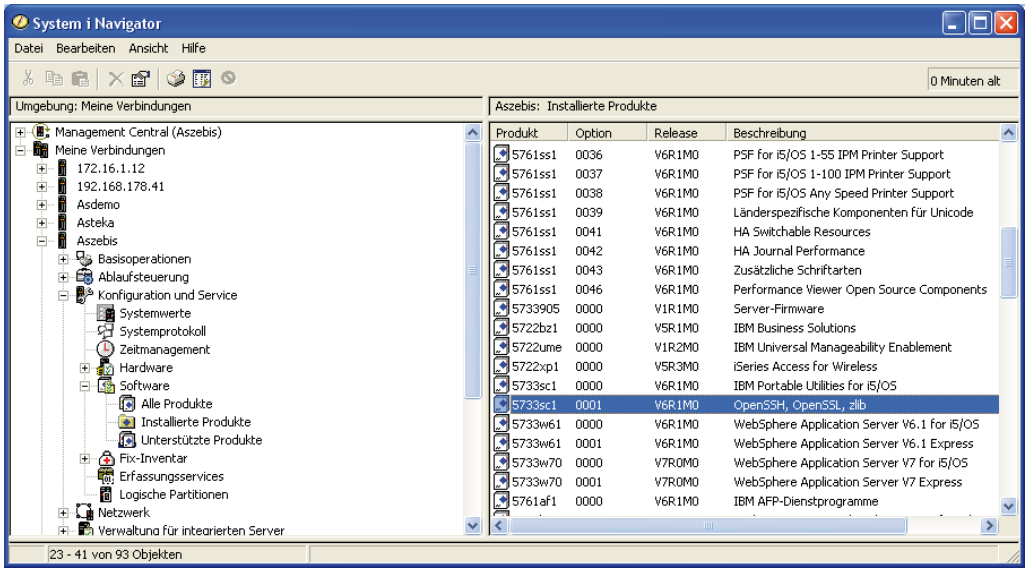
```
CHGPGP OBJ('/home/benutzerprofil')
NEWPGP(*NONE) DTAAUT(*RX)
```



CHGPGP

5.4.4.2
Seite 4

OpenSSH steht für das Betriebssystem i5/OS in Form des Lizenzprogramms 57xx-SC1 zur Verfügung und wird in der PASE Umgebung des i5 ausgeführt. Bevor man also OpenSSH nutzen kann, muss zunächst sichergestellt sein, dass das zuvor genannte Lizenzprogramm neben der PASE Umgebung installiert ist.



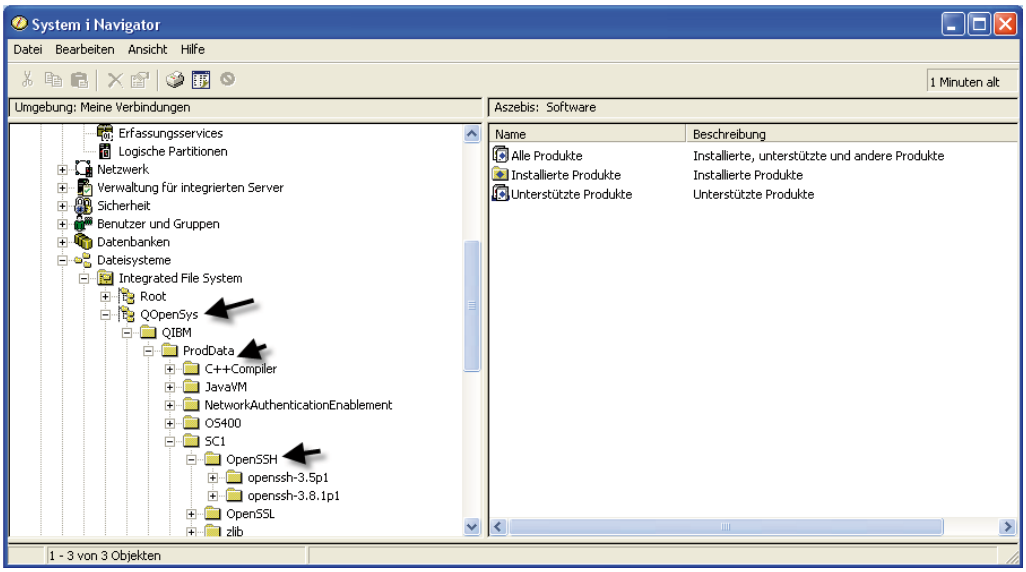
Installierte Lizenzprogramme

Sollte 57xx-SC1 noch nicht installiert sein, dann kann diese mit den folgenden Befehlen installiert werden:

```
RSTLICPGM LICPGM(5733SC1) DEV(OPTxx) OPTION(*BASE) RSTOBJ(*ALL)
LNG(2924)
RSTLICPGM LICPGM(5733SC1) DEV(OPTxx) OPTION(1) RSTOBJ(*PGM)
```

Neben dem Lizenzprogramm befinden sich die wichtigen Installations- und Konfigurationsdateien für die OpenSSH Umgebung in den folgenden Verzeichnissen der Maschine:

```
/QOpenSys/QIBM/ProdData/SC1/OpenSSH
/QOpenSys/QIBM/ProdData/SC1/OpenSSL
/QOpenSys/QIBM/ProdData/SC1/zlib
/QOpenSys/QIBM/UserData/SC1/
```



IFS Inhalt

Achtung!

Verwechseln Sie das Verzeichnis nicht mit dem „gewöhnlichen“ Verzeichnis, bei dem die Installation von Produkten in dem Verzeichnis QIBM vorgenommen wird!

5.4.4.2
Seite 6

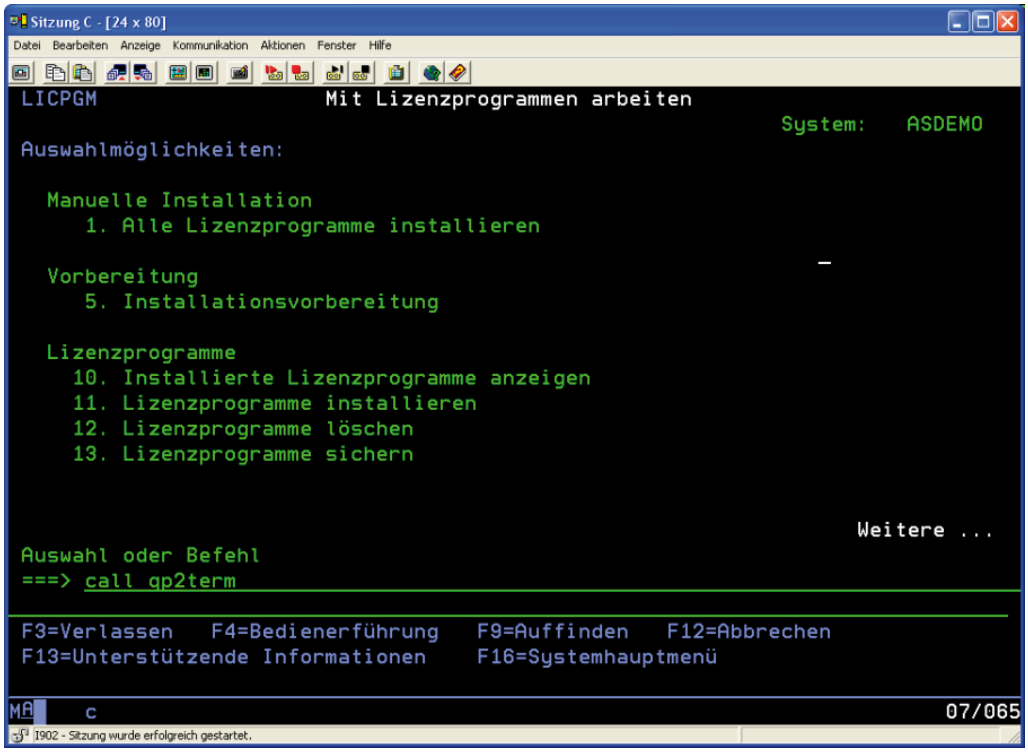
sshd Daemon

Der sshd Daemon ist eine zentrale Funktion, die beispielsweise auf eine sichere Netzwerkverbindung prüft, die Authentifizierung durchführt und die auszuführenden Befehle entgegennimmt. Diese Funktion wird unter IBM i als Serverprogramm für ssh ausgeführt.

Der sshd Daemon muss auf jedem IBM i konfiguriert werden, damit auf dieses System mittels ssh zugegriffen werden kann.

Die Konfiguration des ssh Daemon besteht im Wesentlichen aus dem Generieren von Schlüsseln. Diese lassen sich mit Hilfe des folgenden Befehls erstellen:

Aufruf des Programms CALL QP2TERM zum Starten der PASE Umgebung:

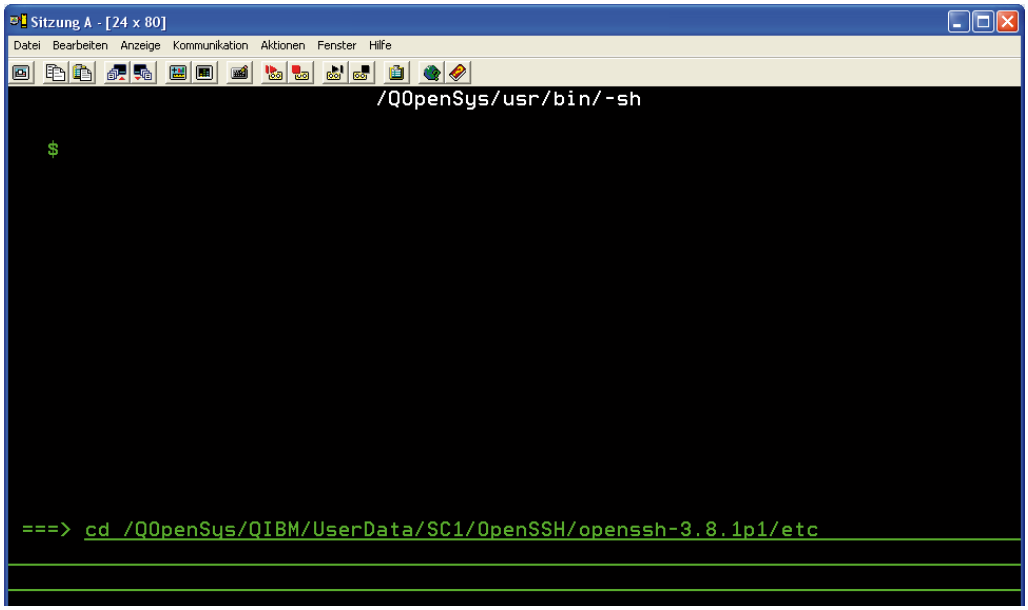


CALL QP2TERM

In der Pase Sitzung geben wir die folgenden Befehle ein:

```
cd /QOpenSys/QIBM/UserData/SC1/OpenSSH/openssh-3.8.1p1/etc
```

Mit diesem Befehl wechseln wir in das gewünschte Verzeichnis. Dieses kann je nach verwendeter Version variieren. Das korrekte Verzeichnis finden Sie in dem folgenden Pfad: /QOpenSys/QIBM/ProdData/SC1/OpenSSH.

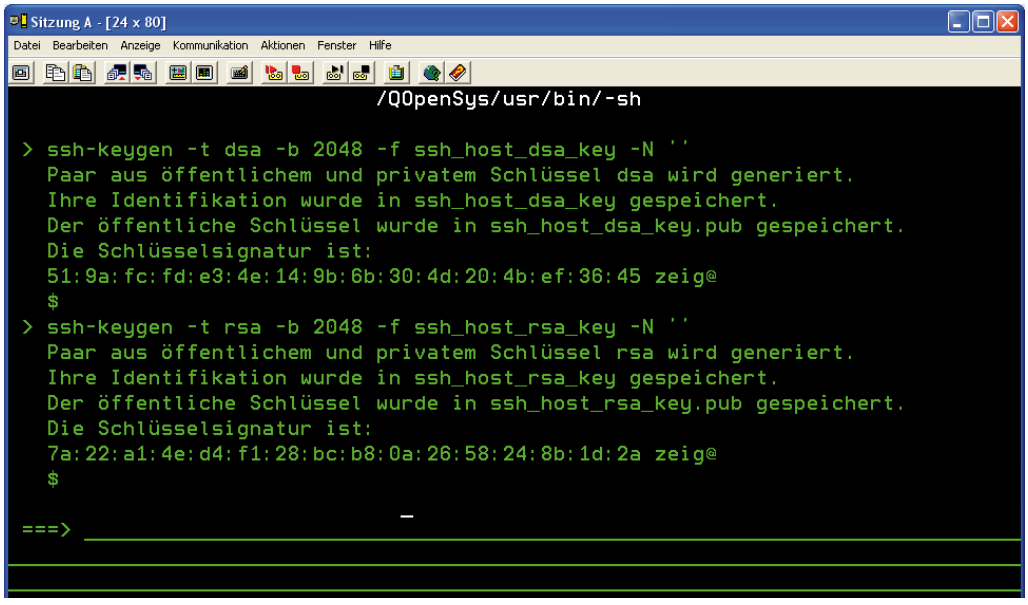


Wechseln des Verzeichnisses

Nun generieren wir die öffentlichen und privaten Schlüssel mit den nachfolgenden Befehlen:

```
ssh-keygen -t rsa1 -b 2048 -f ssh_host_key -N ,‘
ssh-keygen -t dsa -b 2048 -f ssh_host_dsa_key -N ,‘
ssh-keygen -t rsa -b 2048 -f ssh_host_rsa_key -N ,‘
```

Mit diesen Befehlen erzeugen wir die benötigten öffentlichen und privaten Schlüssel für die beiden SSH Protokolltypen.



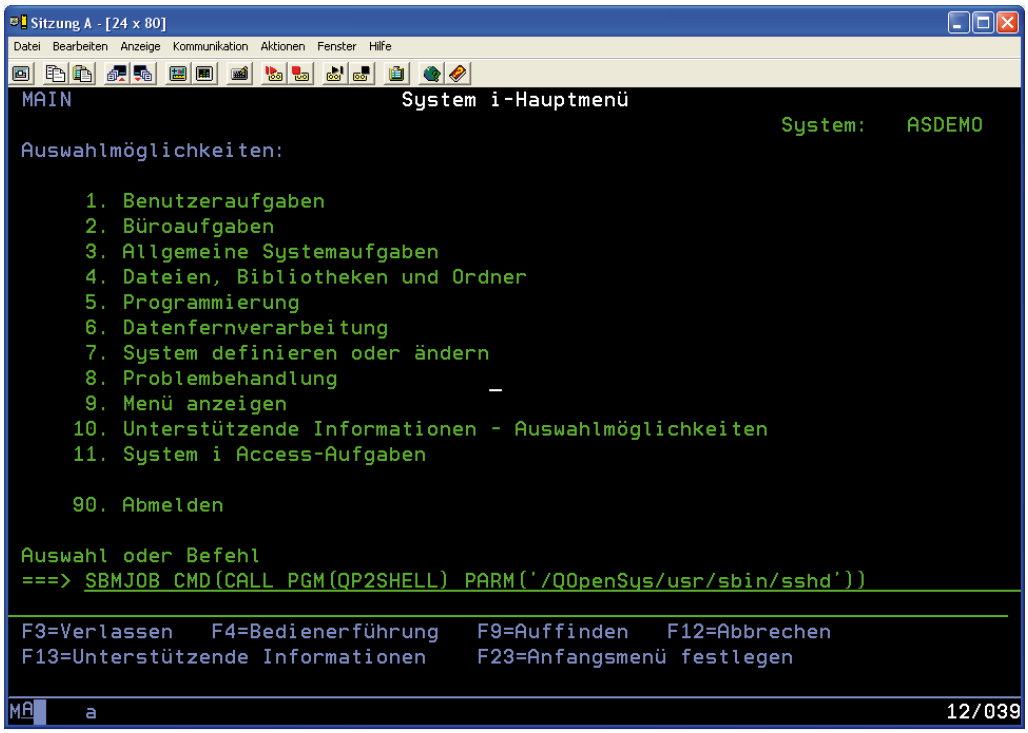
Erstellung der Schlüssel

5.4.4.2
Seite 8

Starten des sshd Daemon auf IBM i

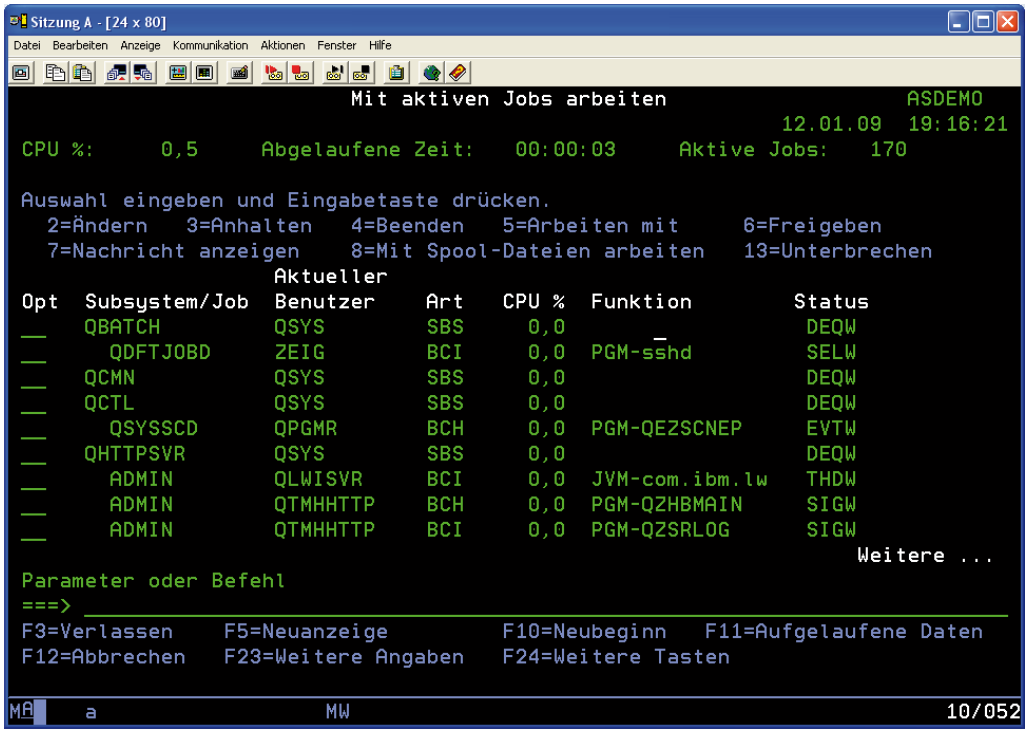
Nach dem Erstellen der Schlüssel können wir den ssh-Daemon auf dem IBM i aktivieren. Mit Eingabe des folgenden Befehls wird der sshd Daemon unter i5/OS gestartet:

```
SBMJOB CMD(CALL PGM(QP2SHELL) PARM(,/QOpenSys/usr/sbin/sshd‘))
```



ssh Daemon starten

Abschließend kontrollieren wir den erfolgreichen Start von ssh auf dem IBM i. Der Job „ssh“ wird per Standard dem Subsystem QBATCH aktiviert. Mittels WRKACTJOB kann beispielsweise der Status dieses Jobs eingesehen werden.



ssh ist aktiv

Damit kann eine SSH Sitzung aufgebaut werden.

5.4.4.2
Seite 10

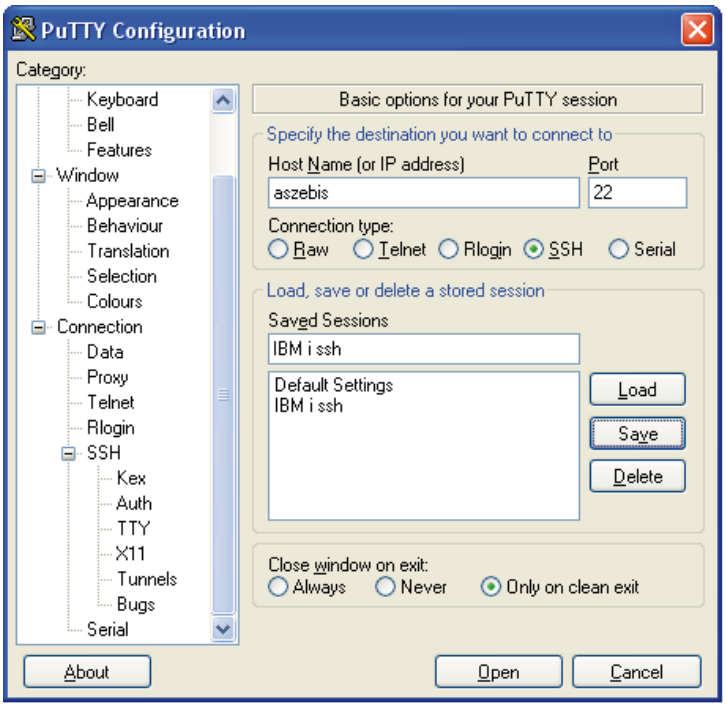
Aufbau einer SSH Sitzung

SSH Sitzungen können zwischen zwei i5/OS Systemen oder auch von Nicht-i5/OS Betriebssystemen zu einer i5/OS Umgebung hergestellt werden. Soll eine Verbindung mit Windows und i5/OS über SSH aufgebaut werden, dann können wir dazu unter Windows ein Freewaretool namens PUTTY verwenden. Dabei handelt es sich um ein für Windows optimiertes SSH Tool mit grafischer Oberfläche. Dieses kann über die folgende Internetadresse heruntergeladen werden: http://www.putty.nl/download.html

Putty selbst lässt sich durch einen Klick auf die .exe Datei starten. Anschließend befolgen Sie die nachfolgenden Schritte:

Ausführungsschritte

1. Geben Sie in dem Feld „Host Name" den Hostnamen des IBM i ein.
2. Geben Sie als Port „22" ein.
3. Wählen Sie SSH.



Putty

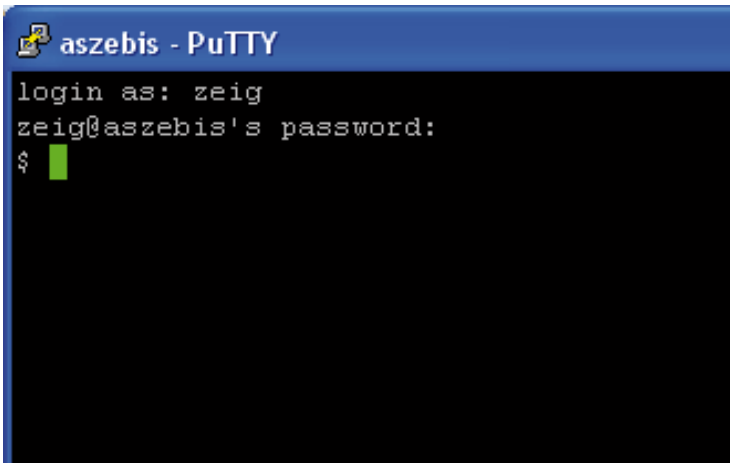
4. Sie können die Definition für beliebig viele Systeme durchführen und diese jeweils speichern. Dazu kann in dem Bereich „Saved Sessions" eine Angabe gemacht werden, unter der die Einstellungen für das spezifische System gesichert werden.

5. Klicken Sie nun auf die Schaltfläche „Open“.

   Unter Umständen kann ein Sicherheitshinweis ausgegeben werden. Schauen Sie sich die Meldung an und klicken Sie, nachdem Sie sich vergewissert haben, dass es sich bei der Meldung um einen Hinweis handelt, der akzeptiert werden kann, auf „Ja“.

   Sicherheitshinweis

6. Geben Sie in dem Putty-Loginbereich den i5 Benutzernamen und dessen Kennwort an. Damit gelangen wir auf das System i.



   Putty Login

5.4.4.2
Seite 12

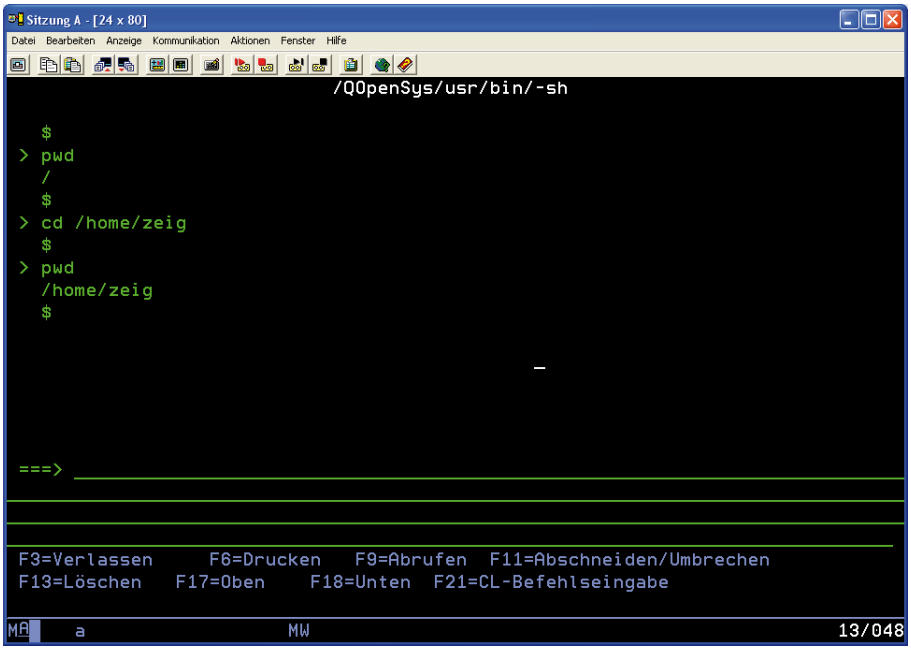
Datenaustausch mit SSH
Nachdem SSH eingerichtet ist, kann der sichere Datenaustausch über dieses Medium erfolgen. Für den Datenaustausch stehen mit SSH zwei Programme zur Verfügung:

- SFTP
- SCP

Während SFTP durchaus vergleichbar mit FTP ist, kann SCP schnell dazu eingesetzt werden, eine einzelne Datei von einem System auf ein anderes zu übertragen. Leider haben beide Werkzeuge einen Nachteil: Sie können nicht zusammen mit der Eingabe von Benutzerprofilen und Kennwörtern als „Sicherheitsauthentifizierung" genutzt werden. Stattdessen müssen beide Schlüssel genutzt werden. Diese Besonderheit hat aber auch ihr Gutes – denn damit entfallen die Eingabe, das Speichern und das Übertragen von sicherheitskritischen Informationen. Bevor wir allerdings eines der beiden Werkzeuge nutzen können, müssen die Schlüssel erstellt und ausgetauscht werden. Dazu befolgen Sie einfach die nachfolgenden Schritte:

Ausführungsschritte

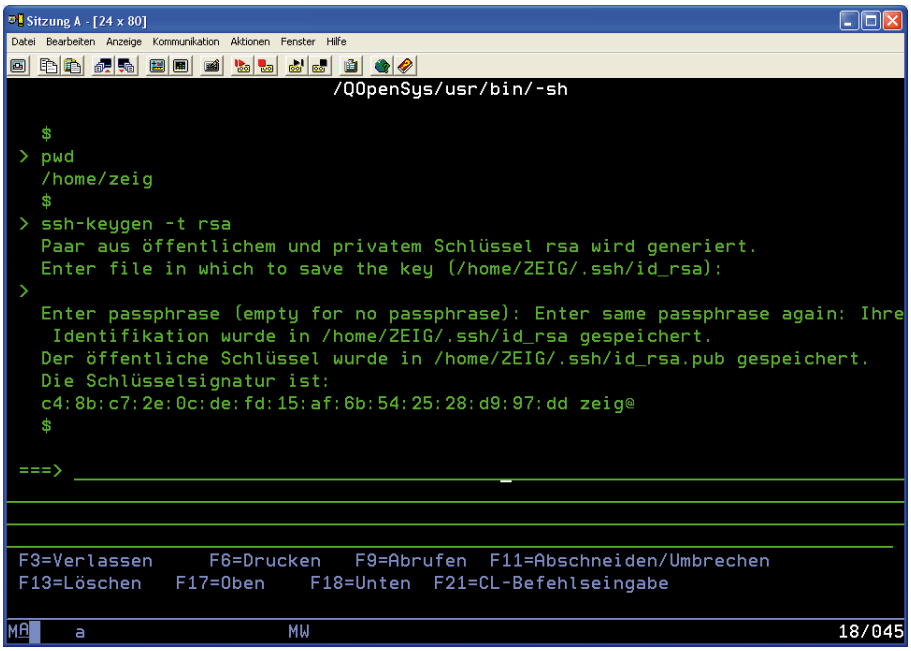
7. Rufen Sie PASE durch Eingabe des Befehls „CALL QP2TERM" in der i5 Befehlszeile auf.
8. Melden Sie sich mit Ihrem i5 Benutzer und dem zugehörigen Kennwort an.
9. Prüfen Sie das Homeverzeichnis durch Eingabe des Befehls pwd. Sollte anschließend nicht das Homeverzeichnis angezeigt werden, dann wechseln Sie zunächst in das Homeverzeichnis des Benutzers, mit dem Sie sich anmelden. Zum Wechseln des Verzeichnisses geben Sie einfach CD /home/benutzername an. Schauen Sie sich dazu auch die folgende Abbildung an:



Verzeichnis wechseln

10. Geben Sie nun den folgenden Befehl ein, um ein Paar aus öffentlichem und privatem Schlüssel zu generieren:

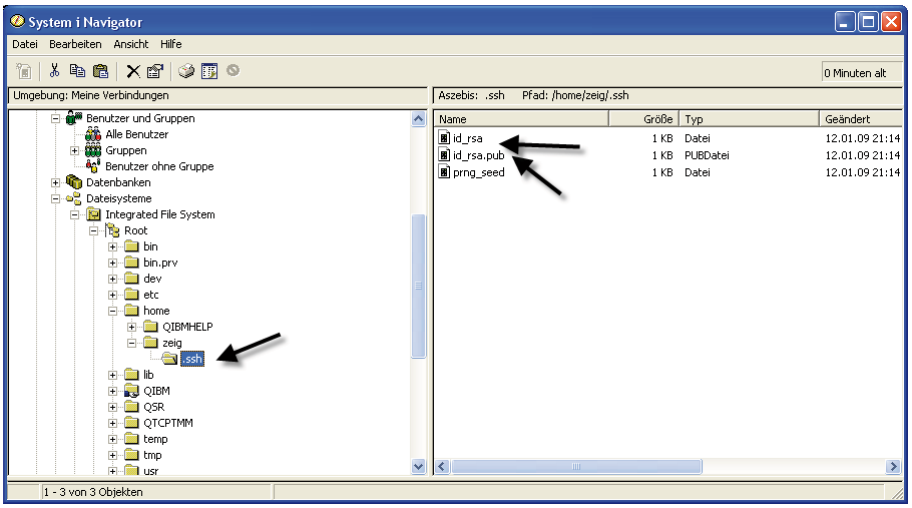
```
ssh-keygen -t rsa
```



Signaturen wurden erstellt

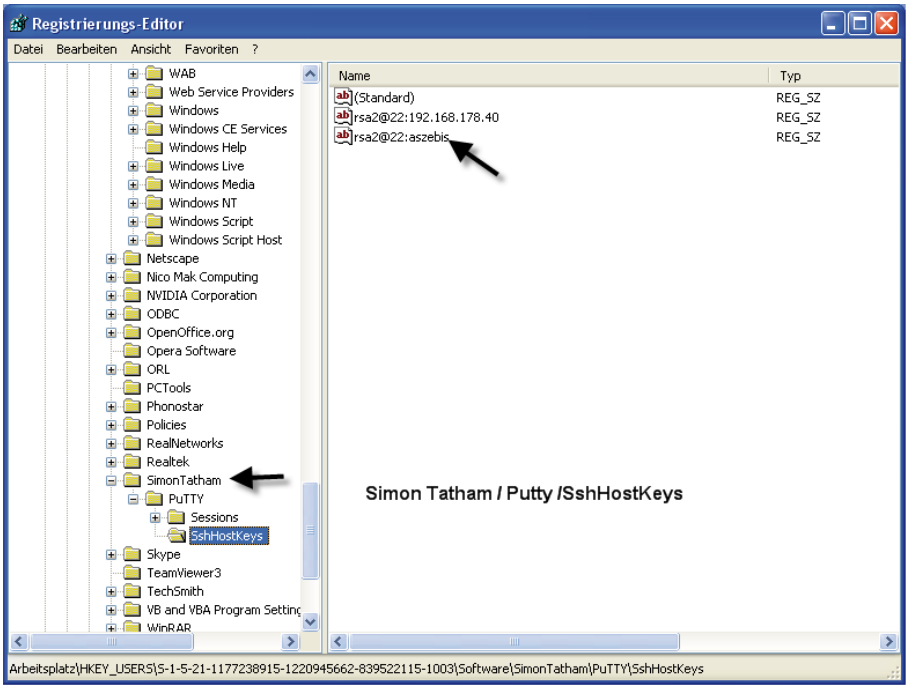
11. Bestätigen Sie den vorgeschlagenen Speicherort und geben Sie zweifach nacheinander ein beliebiges Kennwort ein.

    Damit werden zwei Dateien erstellt, welche den privaten und den öffentlichen Schlüssel beinhalten. Beide befinden sich in dem angegebenen Verzeichnis.



Erstelle Schlüsseldateien

12. Nachdem die Schlüssel erstellt wurden, muss der öffentliche Schlüssel (und nur dieser!) auf das System übertragen werden, von dem ein Datenaustausch mit dem IBM i durchgeführt werden soll. Diese Werte werden dann in die Registry von Windows abgelegt. Suchen Sie nach dem Eintrag „HKEY_USERS / Software / SimonTatham / Putty /SshHostKeys.



Registry Werte

Basierend auf diesen Schlüsseln kann nun ein sicherer Datenaustausch zwischen i5 und anderen Rechnern erfolgen.

5.4.4.2
Seite 16

REXEC

Nachdem das System i und auch dessen Vorgänger mit anderen Systemen verbunden werden können bzw. in der Lage sind, mit ihnen zu kommunizieren, gibt es die Forderung, Befehle von „Fernen Systemen“ (hier sind zum Beispiel PC-Anwendungen gemeint) auf dem System i auszuführen.

Ferne Ausführungen von Befehlen werden allgemein als „Remote Execution“ (kurz: REXEC) bezeichnet.

Dazu gehören nicht nur Java-Anwendungen, die beispielsweise die Anforderung haben, dass auf dem System i Server-Anwendungen remote gestartet werden können, sondern auch klassische Anwendungen, die durch eine PC-Aktion den Programmaufruf auf dem System i erforderlich machen. Beispiele für den Einsatzbereich solcher „Fernen Befehle“ sind weit verbreitet. Nachfolgend einige Musterbereiche, für die ein ferner Befehl sinnvoll sein kann:

- Ermittlung von Umsatzzahlen, die vom System i in ein PC-Programm (z. B. Excel) übertragen und dort aufbereitet werden können.
- Abrufen von Kalkulationen, um sie mit einem PC-Programm weiter verarbeiten zu können.
- Im Workflow eines Windows-Rechners erfolgt der Aufruf eines Batch-Programms, das auf dem System i bestimmte Berechnungen durchführt, die erst vorgenommen werden dürfen, wenn auf dem fernen Rechner die notwendigen Aktionen erfolgreich abgeschlossen wurden.
- Nach Abschluss bestimmter Arbeiten auf dem System i erfolgt der Anwendungsaufruf auf einem fernen System, bei dem die zuvor auf dem System i erstellten Daten benötigt werden.
- Umschalten von Ressourcen auf einem anderen System i.

Bei solchen Interaktionen müssen die spezifischen Eigenschaften und Anforderungen des Zielsystems (des fernen Systems) bedacht und implementiert werden. Darunter fallen auch die sicherheitsrelevanten Einstellungen, die das Ausführen von Programmen oder Befehlen auf dem System i erst ermöglichen.

Eventuell ist Ihnen die Ausführung von fernen Befehlen von PC-Seite auf dem System i bereits bekannt: Der Befehl RUNRMTCMD dient unter anderem dazu, von einem System aus einen Befehl auf dem System i abzusetzen bzw. auszuführen. Da der Befehl RUNRMTCMD auch im i5/OS zur Verfügung steht, können wir ihn vom System i aus nutzen, um einen Befehl auf einem anderen fernen System aufzurufen. Dabei kann es sich beim fernen System um einen beliebigen Rechner handeln, beispielsweise um ein anderes System i, einen Windows-Rechner oder ein UNIX-System.

5.4.5
Seite 2

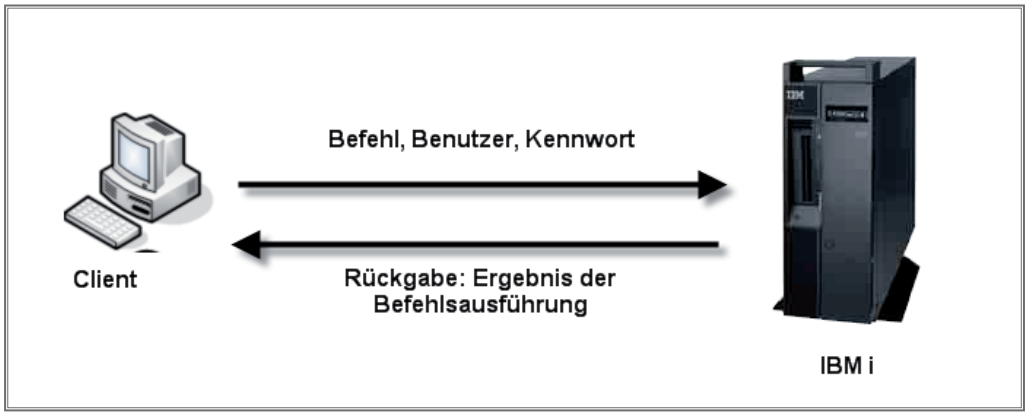
Basis für diese Funktionen ist ein TCP/IP-Standard, der nicht nur im i5/OS, sondern auch in anderen Betriebssystemen implementiert wurde: der REXEC-Standard. Dieser besondere TCP/IP-Bereich überträgt die von einem Client abgesetzten Befehle auf einen Ziel-Server, um sie dort auszuführen. Eine solche Übertragung besteht aus folgenden Komponenten:

– Auszuführender Befehl

– Benutzerprofil

– Kennwort für das Benutzerprofil

Als Rückgabe erhält der Client das Ergebnis der Befehlsausführung vom fernen System. Die nachfolgende Grafik verdeutlicht dieses Verfahren:



REXEC-Ausführung grafisch

In Bezug auf das System i bedeutet die REXEC-Anbindung und -Ausführung, dass auf der Maschine ein spezieller TCP/IP-Serverdienst gestartet werden muss, der die fernen Befehle entgegennimmt.

Die REXEC-Anbindung und -Realisierung mit dem System i bedeutet, dass unterschiedliche Befehlsformen an den Server übertragen werden können. Dies sind folgende Bereiche:

– Native System i-Befehle

– QShell-Befehle

Achtung!

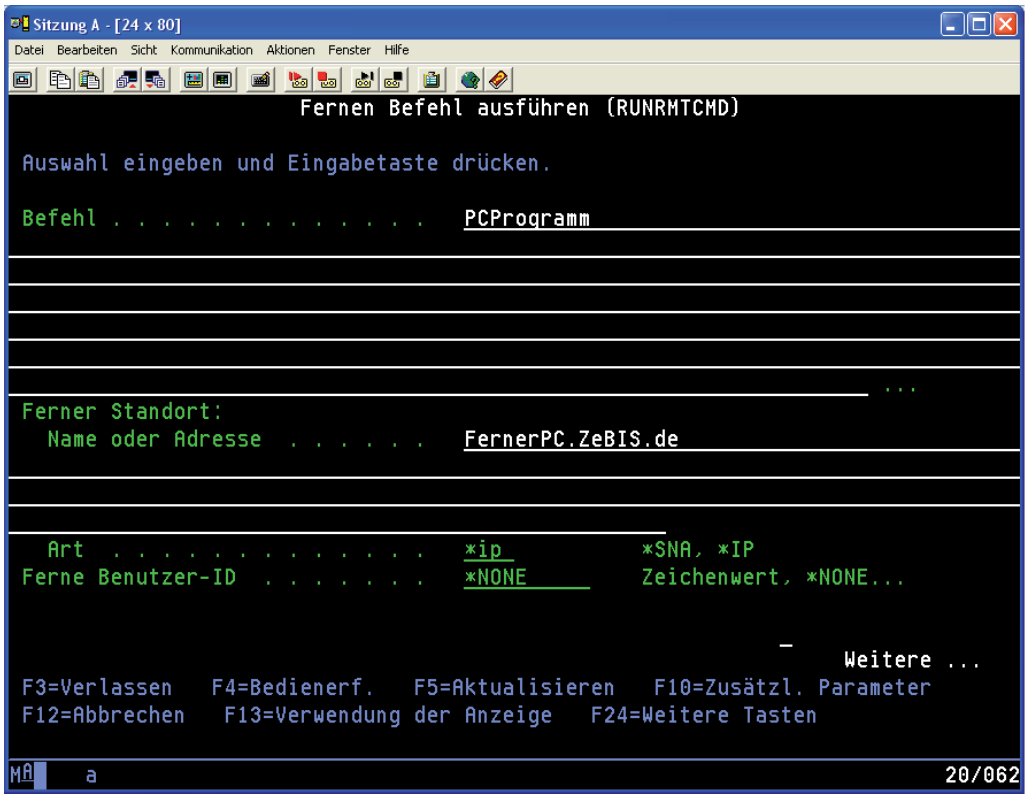
Bei Verwendung der REXEC-Funktionen zur Befehlsübergabe ans System i ist darauf zu achten, dass diese Befehle als Hintergrundjobs auf dem System ausgeführt werden.

Neben der Besonderheit der Batchjobs, die mit dem REXEC-Bereich des System i verarbeitet werden können, müssen auch Grenzwerte eingehalten werden, die zum Beispiel in der maximalen Länge des Befehls liegen. Dieser Befehl kann vom REXEC-Prozess verarbeitet werden, der auf 4000 Bytes limitiert ist.

Wenn von einem Client ein Befehl an den REXEC-Server übergeben wird, dann wird dazu der Befehl RUNRMTCMD verwendet. Dieser arbeitet auf Basis von zwei einzelnen Verbindungen, die dazu genutzt werden, um die Rückinformationen entgegenzunehmen. Nachfolgend finden Sie ein einfaches Beispiel für einen solchen Befehl:

```
RunRmtCmd Cmd(,Start E:\MyDirectory\MyProgram.exe‘) RmtLocName(1.2.3.4
*IP) RMTUSER(*None) RMTPWD()
```

Im Ausführungsfall des Befehls auf dem System i finden wir den Befehl RUNRMTCMD wie folgt vor:



RUNRMTCMD

5.4.5
Seite 4

Wenn man beispielsweise einen Windows-Client verwendet, dann wird mit dem Befehl RUNRMTCMD automatisch eine Verbindung zur Rückgabe der Fehlermeldungen sowie eine weitere Verbindung für die Rückgabe der Standardmeldungen genutzt. Diese stehen anschließend auf dem Client zur Auswertung zur Verfügung. Auf diese Weise kann man feststellen, ob der ferne Befehl ordnungsgemäß verarbeitet wurde.

Grundsätzlich gilt beim Einsatz von REXEC, dass ein Client (oder Server) auf einem fernen Rechner einen Befehl absetzen kann. Dabei spielt es keine Rolle, ob es sich beim Rechner, von dem der ferne Befehl abgesetzt wird, um einen PC, einen Server oder ein System i handelt. Klar ist damit auch, dass der ferne Aufruf von einem System i zu einem anderen System i eingesetzt werden kann.

Das Positive bei einem solchen Szenario ist, dass das System i die Automatisierung von CL-Aufrufen in einem CL-Programm gestattet. Da es sich beim Aufruf eines fernen Befehls um einen CL-Befehl handelt, kann man ihn auch problemlos in einem CL-Programm einbinden und damit die Verarbeitung automatisieren.

Die Ausführungsgrundlage eines fernen Befehls ist TCP/IP. Um einen fernen Befehl abzusetzen, müssen wir also zunächst eine TCP/IP-Verbindung zum fernen System herstellen. Unabhängig davon ist, um welches lokale bzw. um welches ferne System es sich handelt.

Nach der Verbindungsherstellung wird zunächst das Benutzerprofil mitsamt dem zugehörigen Kennwort und dem auszuführenden fernen Befehl übertragen. Als Rückgabewert erhalten wir nach der Befehlsausführung auf dem fernen System jede dort erstellte Ausgabe. Abschließend wird die Verbindung beendet.

FTP

Der Datenaustausch zwischen verschiedenen Rechnern bzw. Systemen ist durch die Integration heterogener Plattformen, aber auch durch das Differenzieren von Anwendungen auf verschiedenen Rechnern immens wichtig. Datenaustausch kann mittels verschiedenster Standards und Werkzeuge ausgeführt werden. FTP (File Transfer Protokoll) hat sich als TCP/IP Dienst als ein solcher Standard etablieren können.

Natürlich unterstützt auch das System i diesen TCP/IP Dienst. Dem Benutzer stehen dabei verschiedenste Varianten zur Verfügung, durch die er den Datenaustausch vornehmen kann. Dabei kann beispielsweise die Datenübertragung über eine interaktive Befehlseingabe oder auch durch einen grafischen Client erfolgen. Zum Automatisieren lassen sich FTP Funktionen auch als Batchverarbeitung ausführen, um somit bedienerlos Datenaustausch, beispielsweise in der Nacht, von einem System zu einem anderen durchführen zu können.

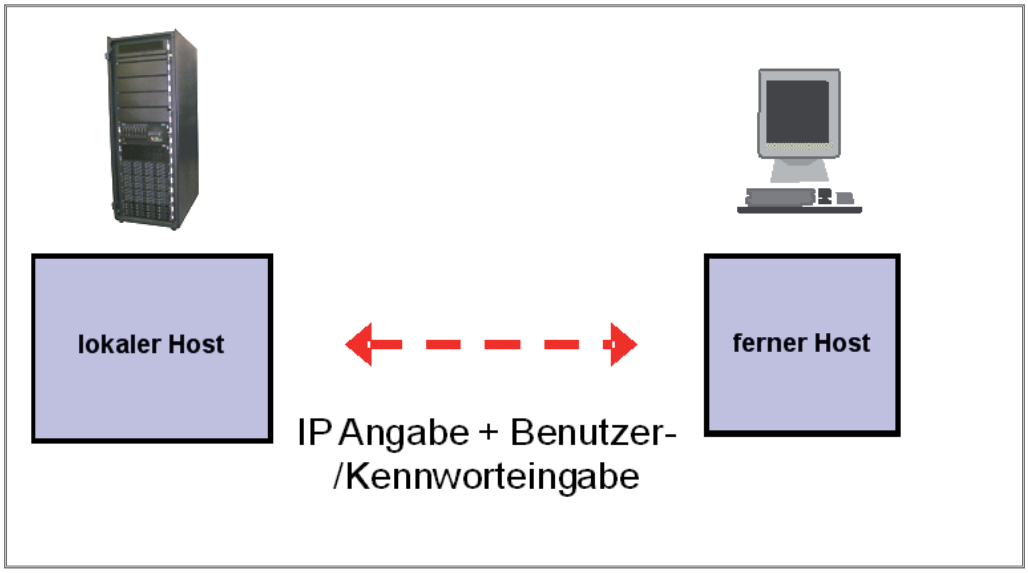
Besonderheiten wie zum Beispiel die fehlende Möglichkeit der nativen FTP Übertragung von Bibliotheken oder Objekten sind bei dem Einsatz von FTP natürlich zu berücksichtigen. Möchte man in System i Anwender den Inhalt dieser typischen System i Objekte auf ein anderes System übertragen, dann müssen diese Objekte zunächst in eine spezielle Sicherungsdatei (SAVF) übertragen werden. Diese Sicherungsdatei kann dann wiederum mittels FTP auf ein anderes System übertragen werden.

Voraussetzung für den Einsatz von FTP ist das Lizenzprogramm 57xx-TC1, das in der Regel auf jedem System als kostenfreies Lizenzprogramm installiert ist. Zudem lassen sich bei Bedarf Sicherheitsoptionen einsetzen, die beispielsweise in Form von verschlüsselten Übertragungen den FTP Datenaustausch sicherer machen. Die FTP Kommunikation wird per Standard über den TCP/IP Port 21 durchgeführt.

Der Vorteil von FTP liegt auf der Hand: Unterstützt doch nahezu jedes System auf dem Markt diesen FTP Dienst. Damit ist ein Datenaustausch zwischen unterschiedlichen Systemen schnell und einfach durchführbar.

5.4.6
Seite 2

In der Regel stellt sich der FTP Datenaustausch wie folgt dar:



FTP Szenario

Die Basis für die FTP Kommunikation bildet TCP/IP. Basierend auf einer IP Adresse oder dem Hostnamen des fernen Systems kann eine Verbindung hergestellt werden.

Anmerkung

Sollten Sie für die FTP Verbindung die IP Adresse angeben wollen, dann achten Sie darauf, dass diese in Hochkomma angegeben werden muss. Als Beispiel: ftp ‚192.168.148.1'.

Grundsätzlich kann der Datenaustausch zwischen Servern oder Clients ausgeführt werden. Häufig finden wir auch die Bezeichnungen „lokaler Host" und „ferner Host". Dabei handelt es sich dann um die beiden Gegenseiten, die bei einer FTP Sitzung zum Einsatz kommen. Die Integration von FTP auf dem System i basiert auf verschiedenen Jobs, die auf dem System i in einem spezifizierten Subsystem (QSYSWRK) ausgeführt werden.

Der FTP Bereich kann mit dem System i zur Übertragung von beispielsweise Inhalten des Bibliotheksbereiches oder auch zur Dateiübermittlung basierend auf IFS Inhalten genutzt werden.

Das System i unterstützt zur Leistungsoptimierung die zeitgleiche Ausführung von bis zu 20 FTP Servern. Per Standard werden auf jedem System zeitgleich drei solcher FTP Server ausgeführt. Das Erhöhen dieser Voreinstellung kann über FTP Vorgaben durchgeführt werden. Allerdings sollte man diese Anpassung wirklich nur dann vornehmen, wenn abzusehen ist, dass zeitgleich mehrere Benutzer bzw. Jobs FTP Übertragungen ausführen müssen.

Für einen Datenaustausch werden zwei Rechner benötigt. Dabei kann einer der beiden Rechner als FTP Server eingesetzt werden, während der andere Rechner als FTP Client genutzt wird. Bei diesem Szenario spielt es keine Rolle, um welche Arten von Rechnern bzw. Betriebssystemen es sich handelt. Wichtig ist nur, dass die eingesetzten Betriebssysteme den FTP Dienst unterstützen. So wäre es zum Beispiel denkbar, dass man den einen Bereich der Datenübertragung von einem IBM i ausführt, während der andere Rechner ein Microsoft oder Apple Betriebssystem besitzt. Genauso ist auch ein Datenaustausch mittels FTP von einem IBM i System zu einem anderen IBM i System möglich.

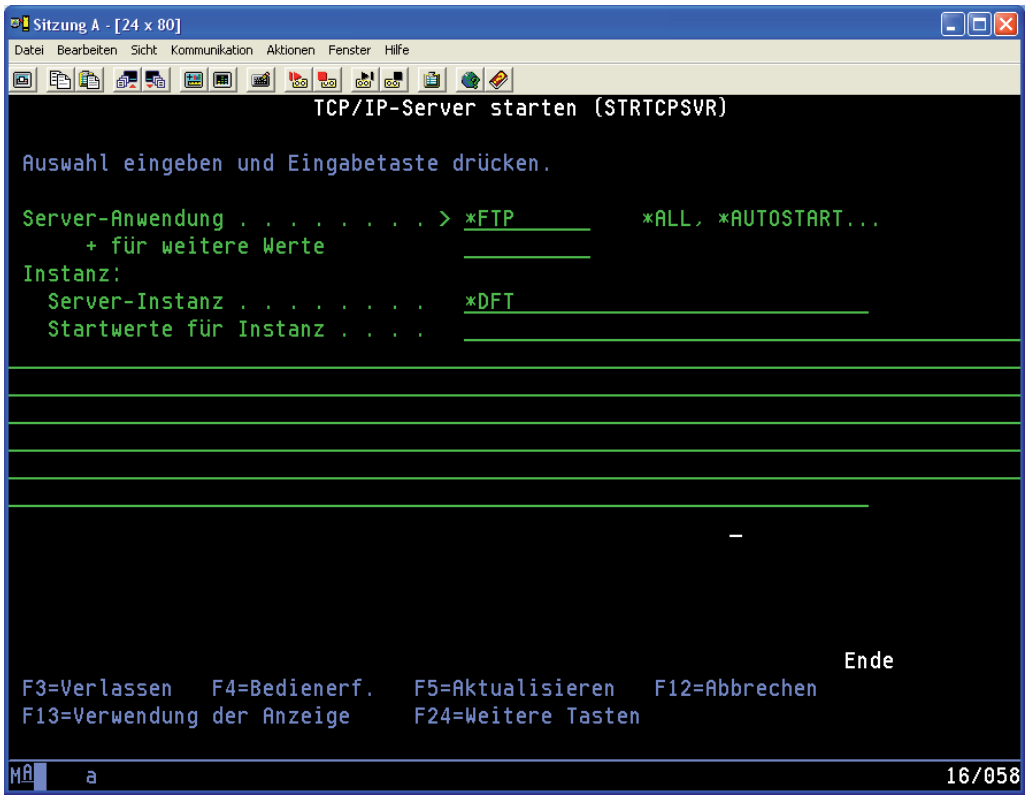
Für eine Datenübertragung werden die folgenden Informationen benötigt:

– IP Adresse oder Hostname des fernen Systems.
– Benutzernamen und Kennwort (wenn Mindestsicherheit definiert wurde)
– Name und Speicherort der zu übertragenden Datei(en)
– Dateiformat (z. B. ASCII oder binäres Dateiformat)

5.4.6
Seite 4

Starten/Stoppen des FTP Dienstes

Der FTP Bereich des IBM i wird über einen separaten TCP/IP Dienst ausgeführt. Dieser kann über den iSeries Navigator oder den Greenscreen verwaltet werden. Für letzteren verwenden wir zum Starten des FTP Bereiches den Befehl STRTCPSVR und geben als Server-Anwendung *FTP an, wie es die nachfolgende Abbildung zeigt.

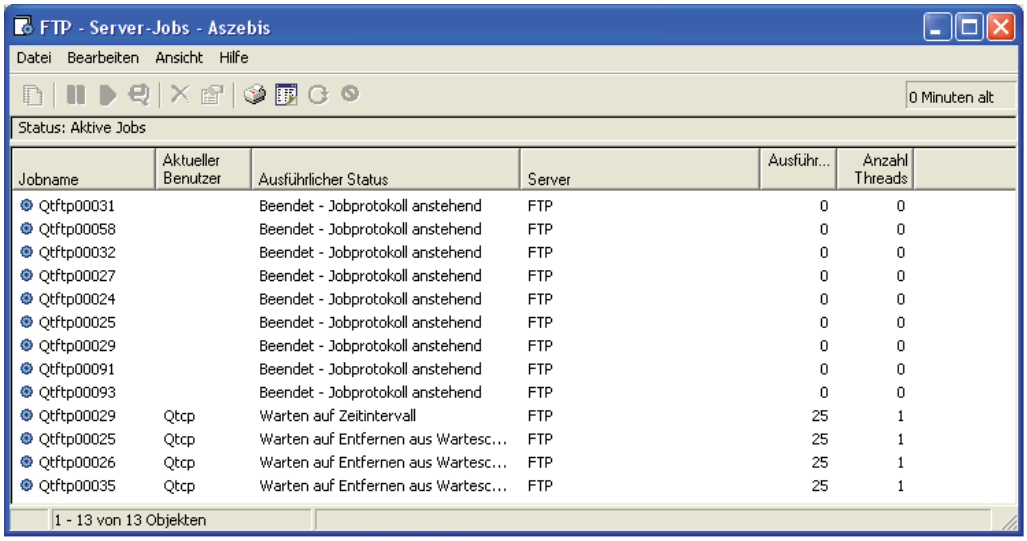


STRTCPSVR

Um den FTP Dienst auf dem System i zu beenden, verwenden wir den Befehl ENDTCPSVR mit der Option *FTP.

5.4.6.1
Seite 2

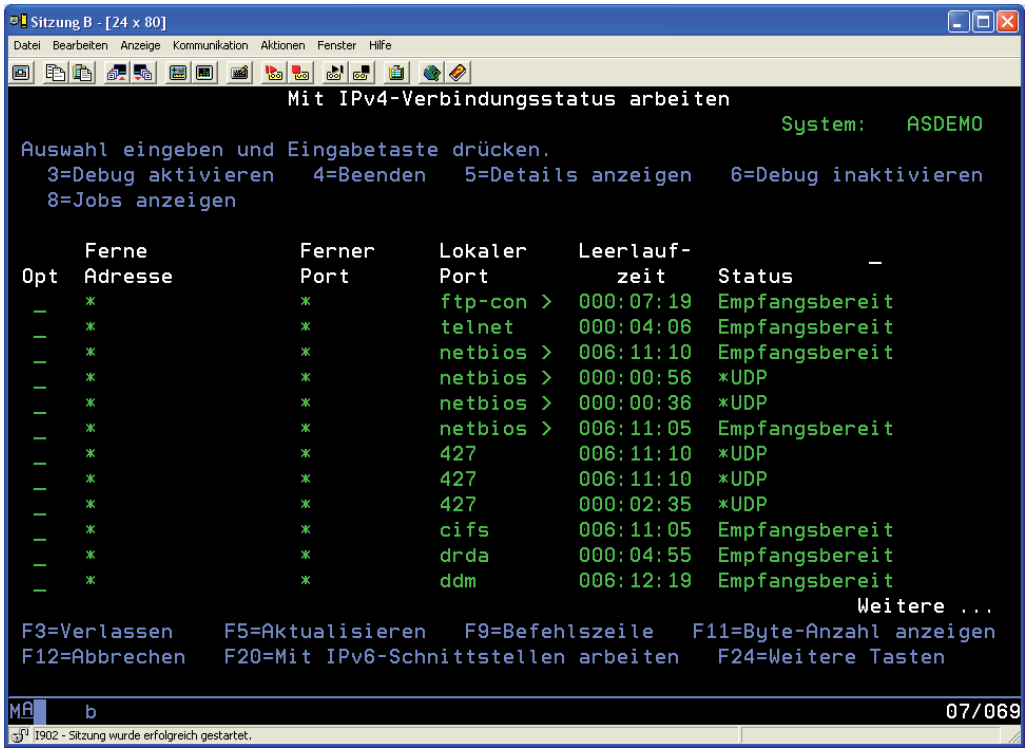
Ob FTP aktiv ist, kann über den iSeries Navigator oder die 5250-Befehlseingabe ermittelt werden. Die nachfolgende Abbildung zeigt die aktiven FTP Jobs auf dem System i.



Serverjobs

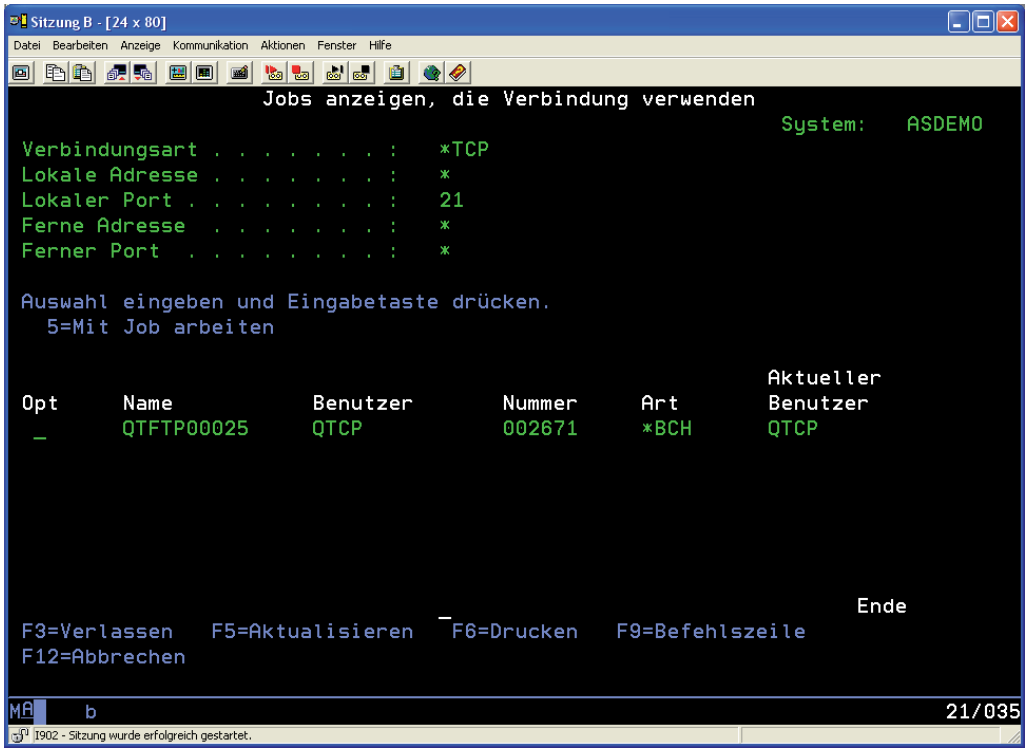
Eine ähnliche Anzeige erhalten wir mit der Eingabe des Befehls WRKACTJOB. Dort werden jedoch alle in dem Subsystem aktiven Jobs angezeigt, so dass man die speziellen FTP Jobs herausfiltern muss.

Mittels des Befehls netstat *cnn können wir die Ports auf dem System einsehen. Der für FTP genutzte Port 21 (ftp-con) befindet sich auch in der Auflistung der Ports auf dem System i. Ein Beispiel dafür finden Sie in der folgenden Abbildung.



netstat cnn

Der für ftp-con verantwortliche Job kann mittels der Option 8 eingesehen werden.



FTP Job

5.4.6.1

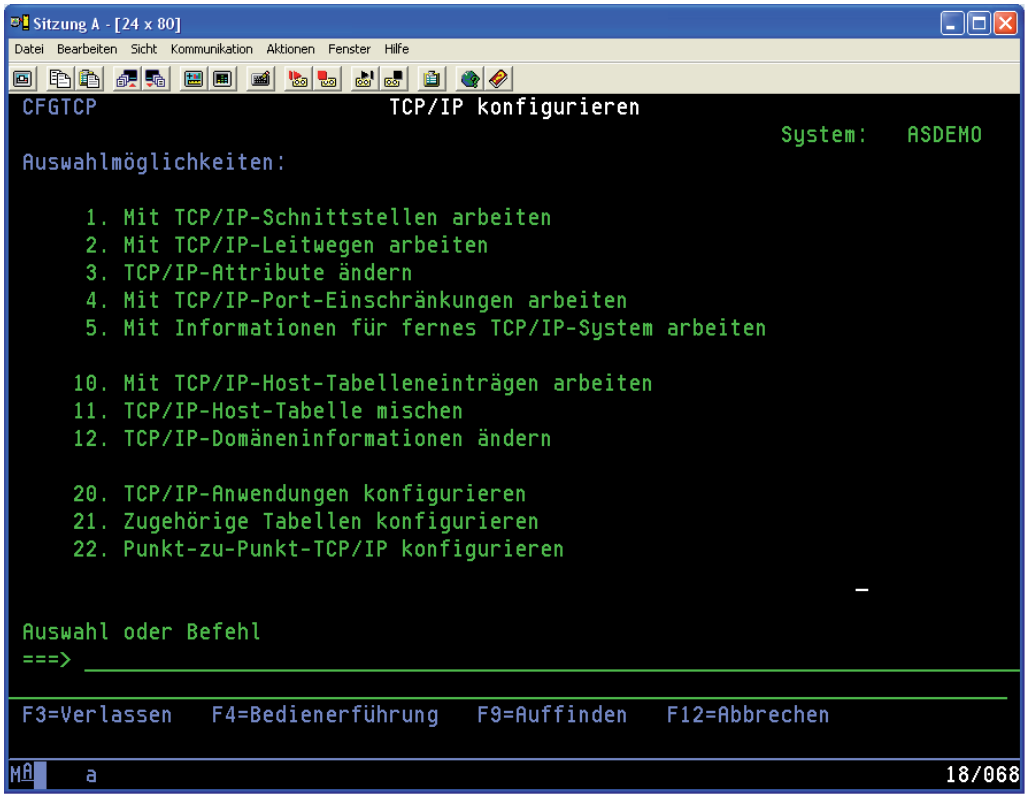
Seite 4

FTP konfigurieren

FTP unterstützt die verschiedensten „Standards“ bzw. Einstellungen. Bedingt durch die heterogenen Systeme, zwischen denen Daten ausgetauscht werden können, sind auch die Varianten der Voreinstellungen entsprechend verfügbar. Bei unterschiedlichen Systemen muss unter anderem bedacht werden, dass die Zeichendarstellung gegebenenfalls unterschiedlich durchgeführt werden kann. Wenn wir zum Beispiel Daten auf ein IBM i übertragen wollen, dann ist dazu je nach Daten das binäre Dateiformat zu verwenden. Auch die Zeichensätze sind für die Daten wichtig.

Es kann auch durchaus Sinn machen, dass man den FTP Serverdienst automatisch zusammen mit TCP/IP startet. Das ist dann sinnvoll, wenn man permanent den FTP Dienst zum Datenaustausch mit anderen Systemen nutzt. Allerdings sollte man in solch einem Fall die Sicherheitsaspekte bedenken, welche dann unbedingt definiert werden müssen, damit nicht etwa unberechtigte Übertragungen von Daten durchgeführt werden.

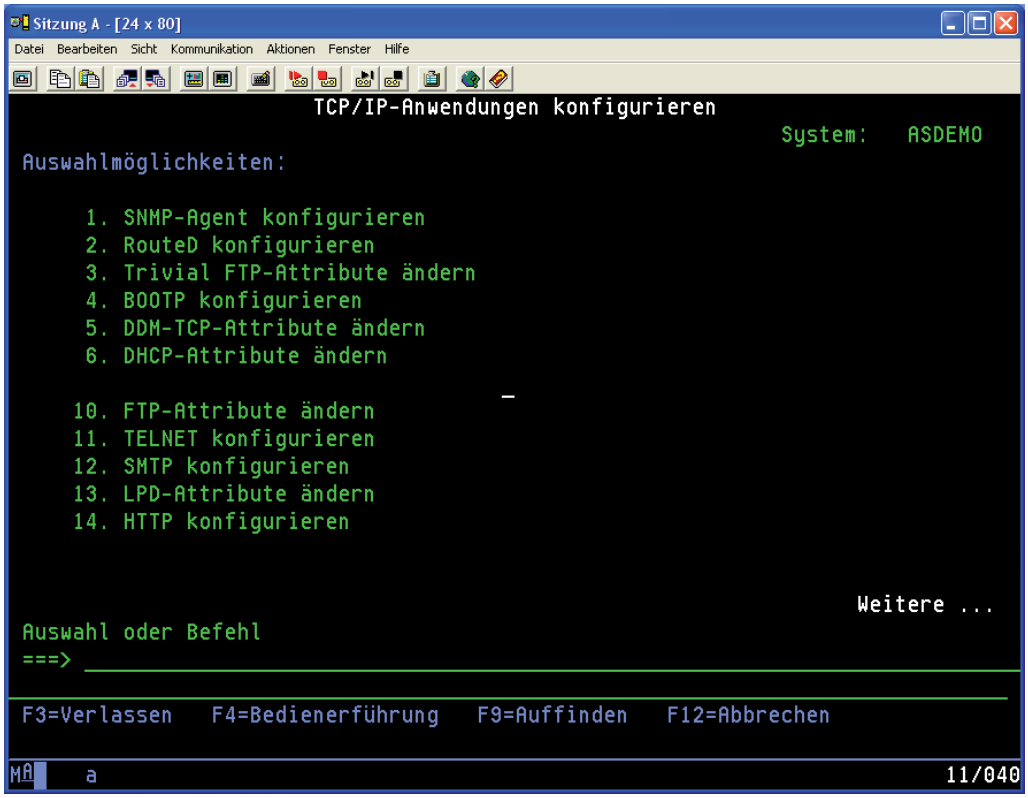
Schauen wir uns einmal die Vorgaben für die Definition der FTP Einstellungen an. Diese können wir in den TCP/IP Angaben festlegen. Geben Sie dazu in der Befehlszeile den Befehl CFGTCP an. Damit gelangen wir in das Menü der TCP/IP Konfiguration.



CFGTCP

5.4.6.2
Seite 2

Mit dem Menüpunkt 20 (TCP/IP Anwendungen konfigurieren) lassen sich die einzelnen FTP Dienste konfigurieren – so auch FTP. Die nachfolgende Abbildung zeigt die verschiedenen TCP/IP Anwendungen.



FTP Attribute ändern

Mit der Option 10 (FTP Attribute ändern) können wir die FTP Voreinstellungen nach Bedarf anpassen.

Mögliche Einstellungen sind unter anderem:

AUTOSTART
Automatischer Start des FTP Dienstes zusammen mit dem Starten von TCP/IP.

NBRSVR
Anzahl der zu startenden FTP Server. Der Standardwert lautet 3 Server. Wenn zeitgleich mehr Benutzer bzw. Jobs FTP Übertragungen ausführen müssen, dann kann dieser Parameter auf max. 20 parallel auszuführende FTP Server erhöht werden.

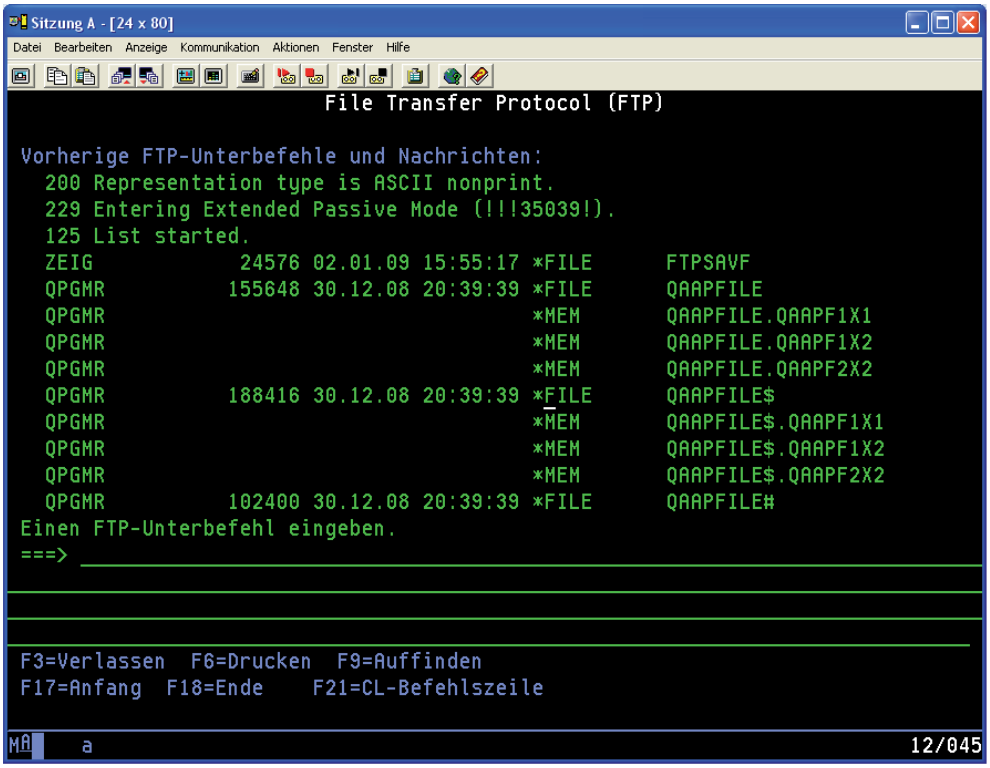
INACTTIMO
Zeitangabe in Sekunden für Inaktivität einer FTP Verbindung. Dieser Wert ist per Standard mit 300 Sekunden vorgegeben. Aus Sicherheitsgründen kann es sinnvoll sein, den Wert zu verringern. Bei diesem Wert ist zu beachten, dass es sich nicht um eine Inaktivität der FTP Transaktionen bzw. der Sitzung selbst handelt, sondern dass der FTP Job auch nach der Beendigung der FTP Ausführung noch eine gewisse Zeit auf dem System i bestehen bleibt. Diese Zeitangabe bezieht sich auf Kommunikation zwischen dem FTP Client und dem auf dem System i laufenden FTP Serverjob.

CCSID
Der Zeichensatz ist für den Datenaustausch zwischen verschiedenen Systemen immens wichtig. Mit diesem Parameter CCSID steuern wir den ASCII Zeichensatz. Dieser wird immer dann verwendet, wenn man nicht in der FTP Sitzung eine andere Angabe tätigt. Wählen Sie hier den korrekten Wert, der eine mögliche Zeichenumsetzung unterstützt, mit der die Daten verschiedener Systeme auch vollständig eingelesen bzw. umgesetzt werden können. In diesem Zusammenhang stehen auf dem System i verschiedene Zeichenumsetzungstabellen zur Verfügung, die als Subparameter für die CCSID Angaben definiert werden können.

5.4.6.2
Seite 4

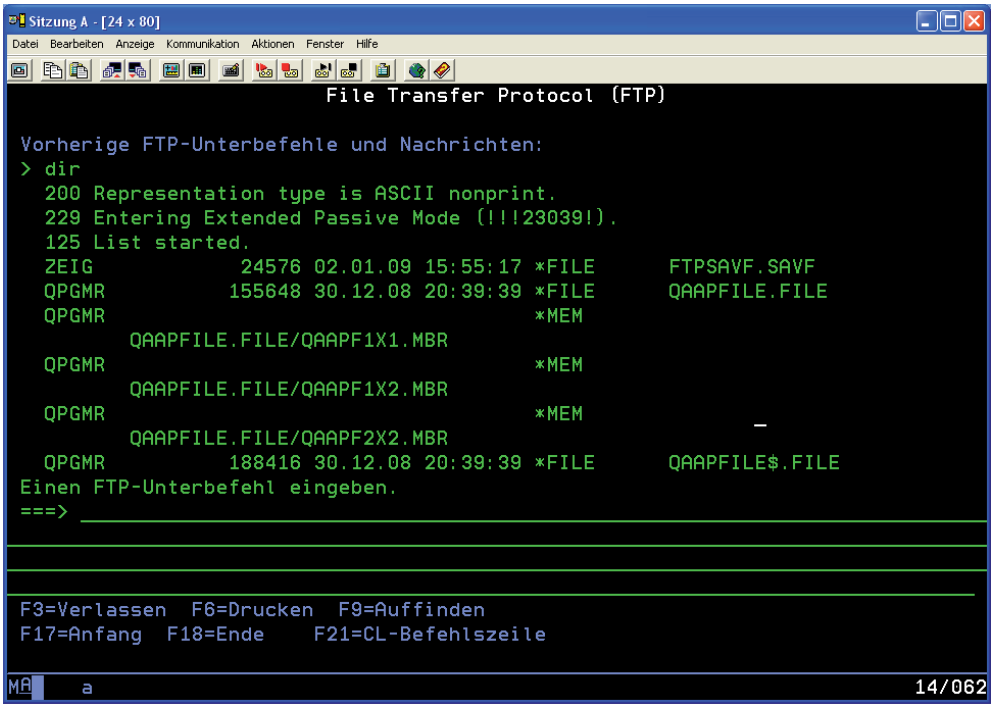
NAMEFMT
Mit diesem Parameter steuern wir die Verwendung des Dateiformates. Wird vorwiegend mit dem Inhalt von Bibliotheken gearbeitet, dann bietet sich der Wert „0“ als Standardwert an. Werden meist Daten im IFS verarbeitet, dann macht die Voreinstellung „1“ als NAMEFMT Sinn. Mögliche Werte sind 0 oder 1.

– **NAMEFMT 0**
Verwendung für Dateien im Bibliotheksumfeld. Das verwendete Format ist *[Bibliotheksname/] Dateiname[.Teildateiname]*.



NAMEFMT 0

– **NAMEFMT 1**
Verwendung für alle Dateisysteme. Das verwendete Format ist z. B. *[QSYS.LIB/] [Bibliotheksname.LIB/] Dateiname.FILE[/Teildateiname.MBR]*.



NAMEFMT 1

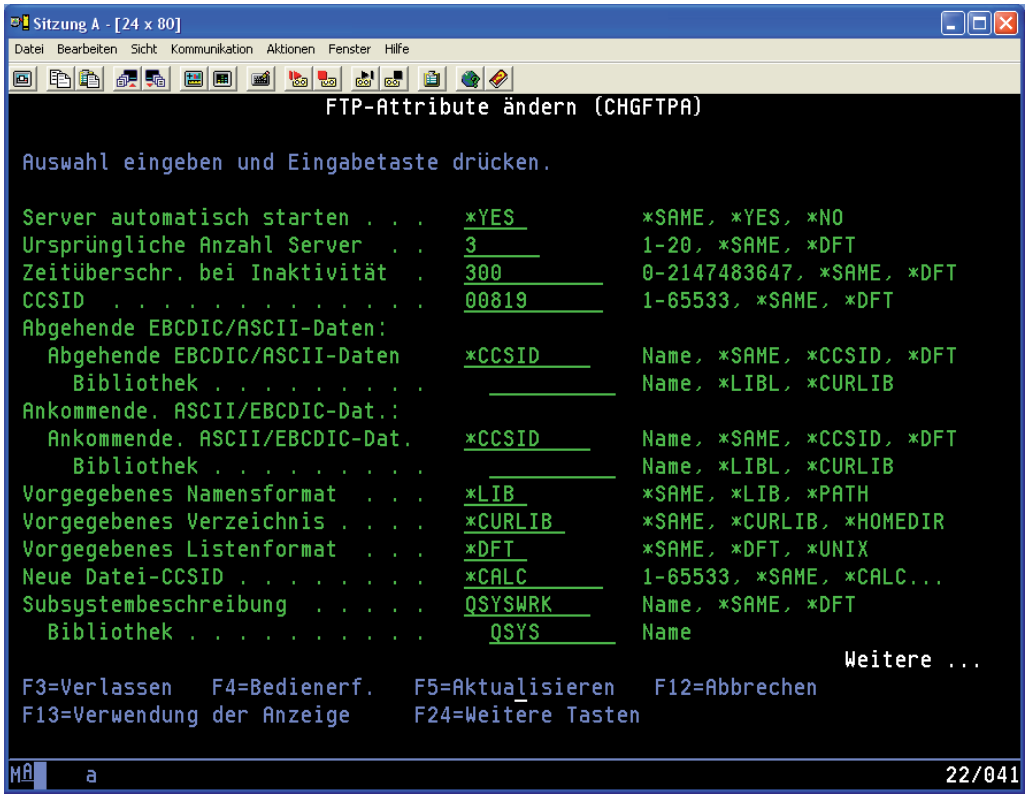
CURLIB
Dieser Parameter steuert das Standard-Verzeichnis, das dann verwendet wird, wenn die FTP Sitzung aufgebaut wird.

Der Sonderwert *CURLIB bedeutet, dass die aktuelle Bibliothek des Jobs, der FTP startet, als Standardverzeichnis für FTP genutzt wird. Wird als Sonderwert *HOMEDIR angegeben, dann nutzt der FTP Job das Homeverzeichnis, das für den Benutzer angegeben ist. Wenn Sie diese Einstellung verwenden wollen, dann kann damit nicht der Parameter NAMEFMT (*LIB) angegeben werden.

SBSD
Dieser unscheinbar wirkende Parameter kann mit dem richtigen Einsatz einen nicht unwesentlichen Einfluss auf die Leistungsfähigkeit von FTP haben. Per Standard wird die FTP Ausführung auf dem System i in dem Subsystem QSYSWRK ausgeführt. Dieses Subsystem wird jedoch auch für eine Reihe anderer Jobs genutzt, die auf der Maschine ausgeführt werden. Wenn ein System stark ausgelastet ist, dann kann es sich negativ auswirken, wenn FTP neben anderen Jobs in demselben Subsystem aktiviert wird. Deshalb sollte man in Abhängigkeit der Auslastung der Maschine die Anpassung des Subsystems erwägen und FTP in einem anderen Subsystem ausführen, dem man beispielsweise dann auch separate Leistungseinstellungen zuordnen kann.

5.4.6.2
Seite 6

ALWSSL
Dieser Parameter steuert, ob FTP zusammen mit SSL (Secure Socket Layer) genutzt werden soll. Der Vorgabewert lautet *YES. SSL ermöglicht das Verschlüsseln der mit FTP übertragenen Daten. In der Verschlüsselung enthalten ist dann neben den Daten auch die Übertragung des Kennwortes, welches bei dem Verbindungsaufbau anzugeben ist. Aus Sicherheitsgründen sollte man stets SSL Verschlüsselung verwenden, wenn die Daten außerhalb des firmeninternen Netzes übertragen werden. Wenn Sie diesen Parameter verwenden wollen, dann müssen Sie darauf achten, dass auch die Voraussetzungen für SSL gegeben sind. Werden diese nicht erfüllt, dann kann es vorkommen, dass der FTP Dienst nicht gestartet werden kann. In diesem Zusammenhang muss zum Beispiel das FTP Server Zertifikat in DCM (Digital Certificate Manager) konfiguriert worden sein.

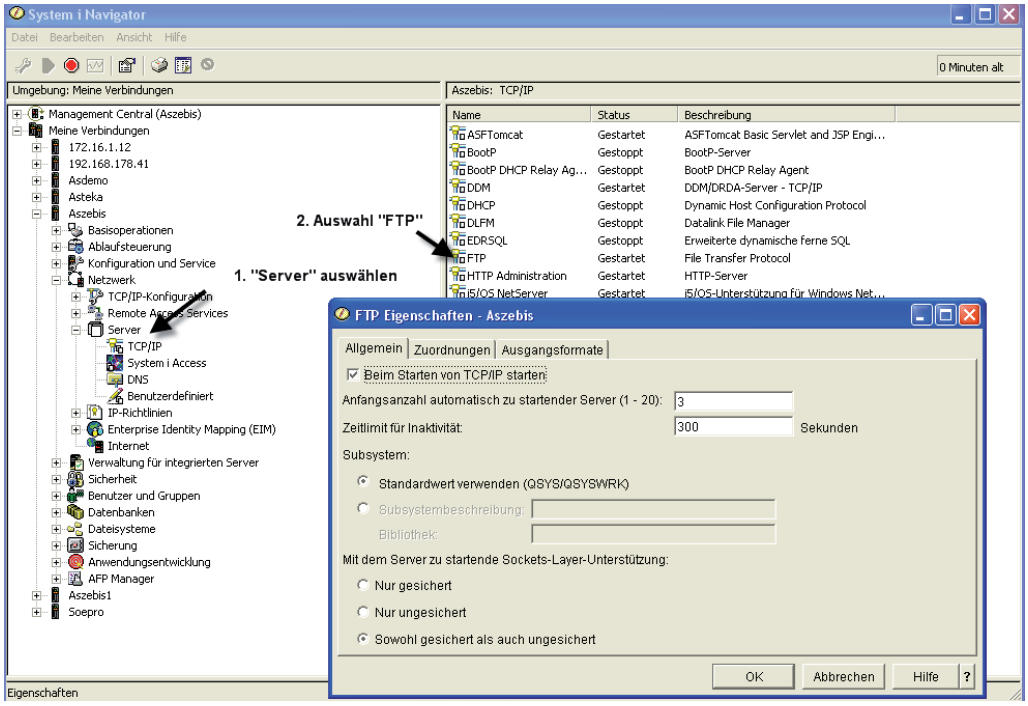


CHGFTPA

Alternativ zum Einsatz der 5250-Befehle können wir den FTP Dienst auch über den iSeries Navigator definieren. Dort wählen wir in dem linken Navigationsbereich den Eintrag „Netzwerk/Server“. Dort befindet sich unter anderem auch der Eintrag „FTP“.

Achtung!

Verwechseln Sie diesen nicht mit TFTP – dabei handelt es sich um einen weiteren TCP/IP Dienst, den wir später noch behandeln werden.



FTP Attribute anpassen

5.4.6.2

Seite 8

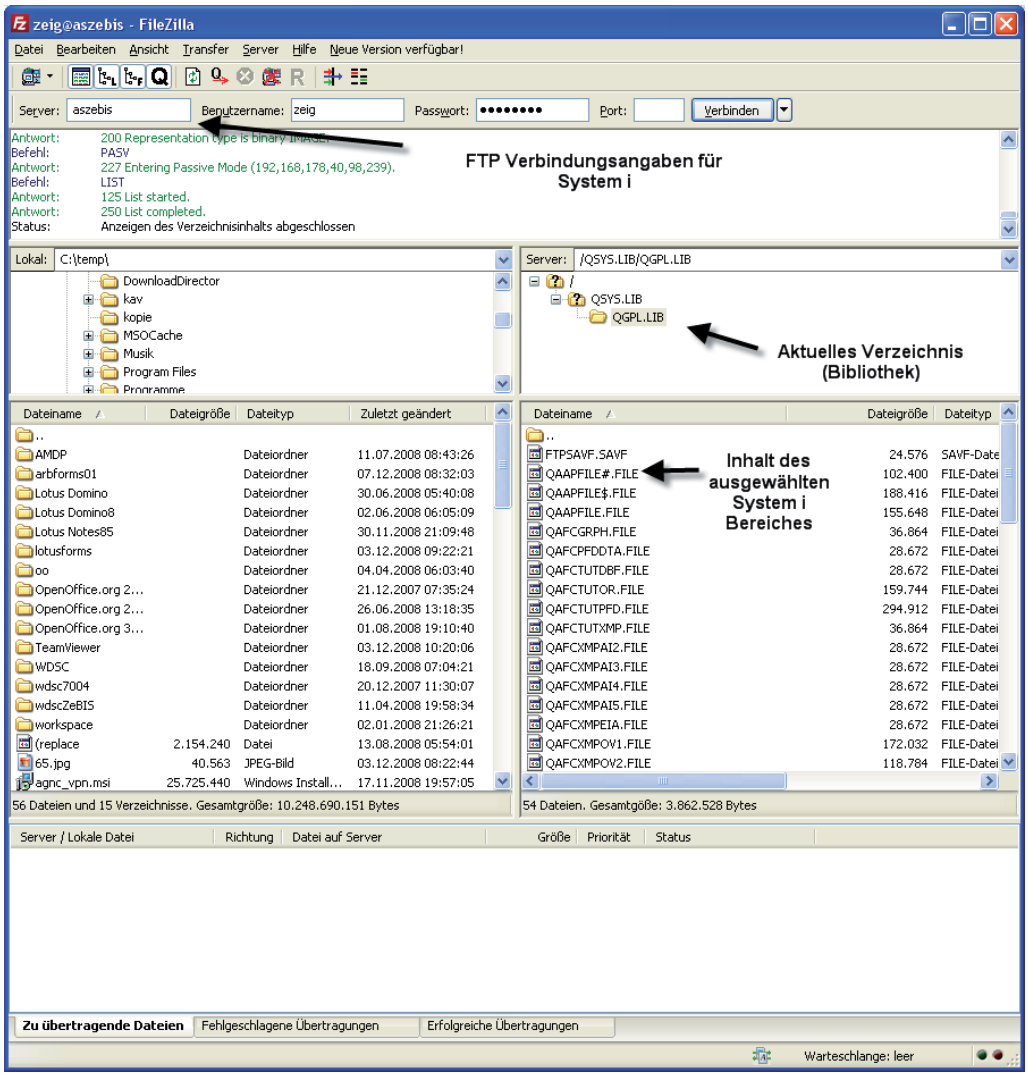
Datenaustausch mit FTP

FTP kann zum bidirektionalen Datenaustausch genutzt werden. Basis dafür bildet das lokale und das ferne System. Es hängt von der jeweiligen Situation ab, ob Daten von dem fernen System geholt werden (Befehl „get“) oder Daten an das ferne System zu übertragen sind (Befehl „put“).

Für den Datenaustausch können wir mit dem System i wahlweise einen der zur Verfügung stehenden 5250-Befehle nutzen oder einen grafischen Client nutzen, die teilweise auch als Freeware oder Vollversionen für PC Systeme angeboten werden. Beispiele für solche grafischen FTP Clients sind WS-FTP oder Filezilla.

Filezille ist zum Beispiel ein kostenfreier FTP Client, der über die folgende Internetadresse heruntergeladen werden kann: http://www.filezilla.de/.

Die folgende Abbildung zeigt Filezilla in Verbindung mit der Anzeige des Inhaltes eines System i:



Filezilla

5.4.6.3
Seite 2

Durch die Eingabe der System i Verbindungsoptionen sehen wir in dem rechten Anzeigenbereich den Inhalt der Bibliothek QGPL. Dieser kann bei Bedarf nun mittels der Funktionen des FTP Clients zum Beispiel in ein Windows Verzeichnis übertragen werden. Alternativ dazu lassen sich auch Übertragungen von dem Windows Rechner auf das System i durchführen.

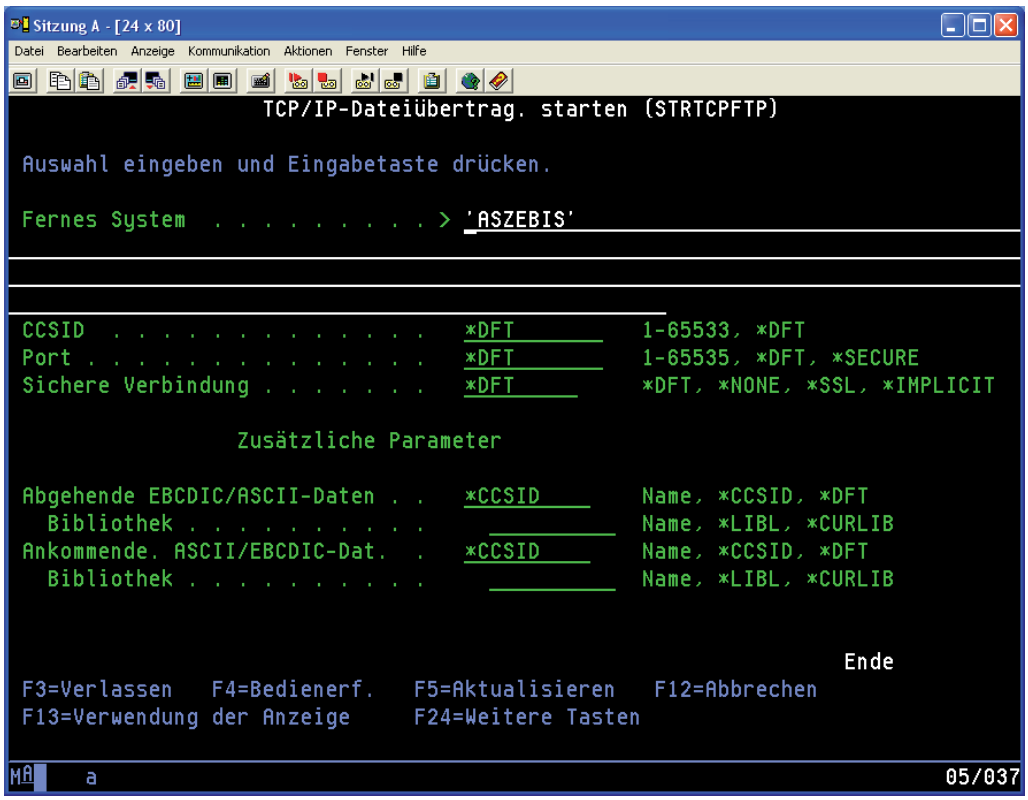
Kommen wir wieder zurück zu den nativen System i FTP Funktionen.

Das Betriebssystem i bietet für die FTP Übertragung zwei Hauptbefehle:

– STRTCPFTP

– FTP

Beide Befehle sind in der Anwendung identisch. Die nachfolgende Abbildung zeigt exemplarisch die Verwendung des Befehls STRTCPFTP.

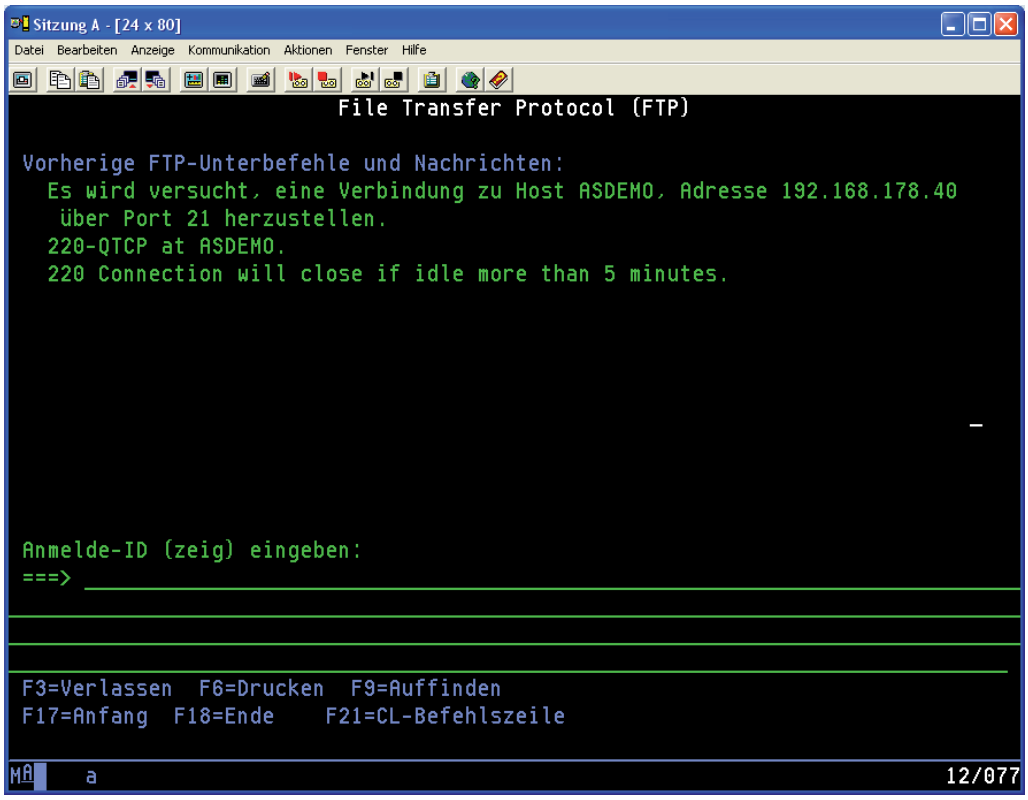


STRTCPFTP

Als erforderlicher Parameter ist hier lediglich der Hostname oder die IP Adresse des fernen Systems anzugeben, mit dem die FTP Verbindung hergestellt werden soll.

Die übrigen Einstellungen werden von den FTP Vorgaben übernommen und können bei Bedarf individuell überschrieben werden.

Nach der Eingabe des Hostnamens gelangen wir in die Hauptanzeige der FTP Verbindung. Dabei handelt es sich um den FTP Client des Systems i.



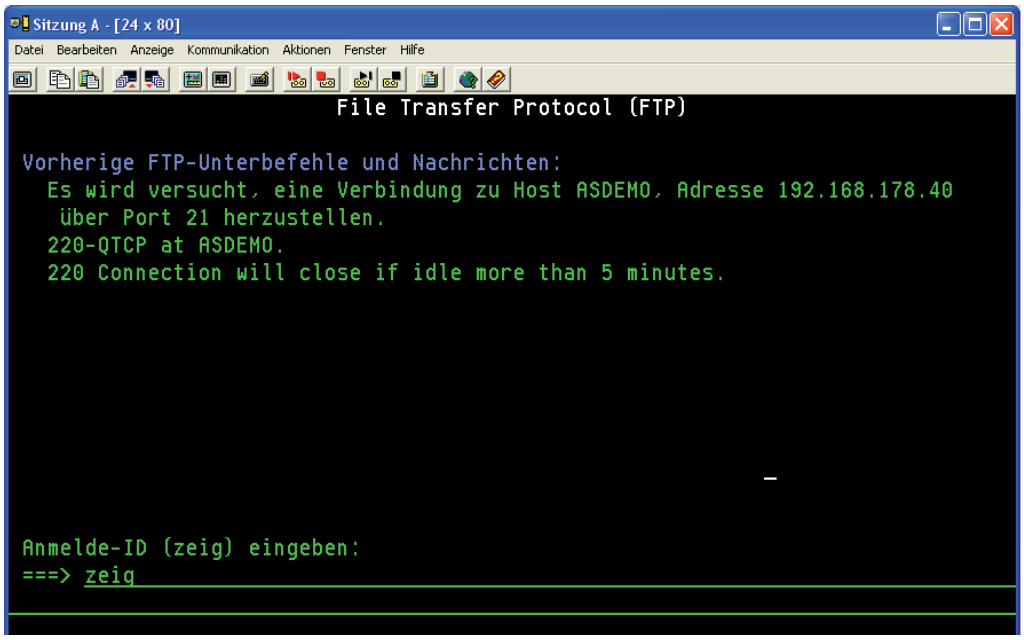
FTP Eingabemaske

Diese FTP Eingabemaske verfügt über eigene Befehlseingabezeilen. Im oberen Bereich finden wir ein Protokoll der aktuellen FTP Sitzung. Der untere Bereich dient der Eingabe von FTP typischen Befehlen.

Der FTP Client liefert zudem Informationen über die anstehenden Pflichteingaben. In der vorhergehenden Abbildung sehen wir zum Beispiel, dass zunächst eine Anmelde-ID angegeben werden muss. Das bedeutet, dass wir zunächst ein gültiges System i Benutzerprofil angeben müssen, damit eine FTP Verbindung mit dem IBM i hergestellt werden kann. Das System hat basierend auf der verwendeten i5-Benutzeranmeldung den i5 Benutzer in Klammern voreingestellt. Dieser kann, muss aber nicht, für die FTP-Übertragung verwendet werden.

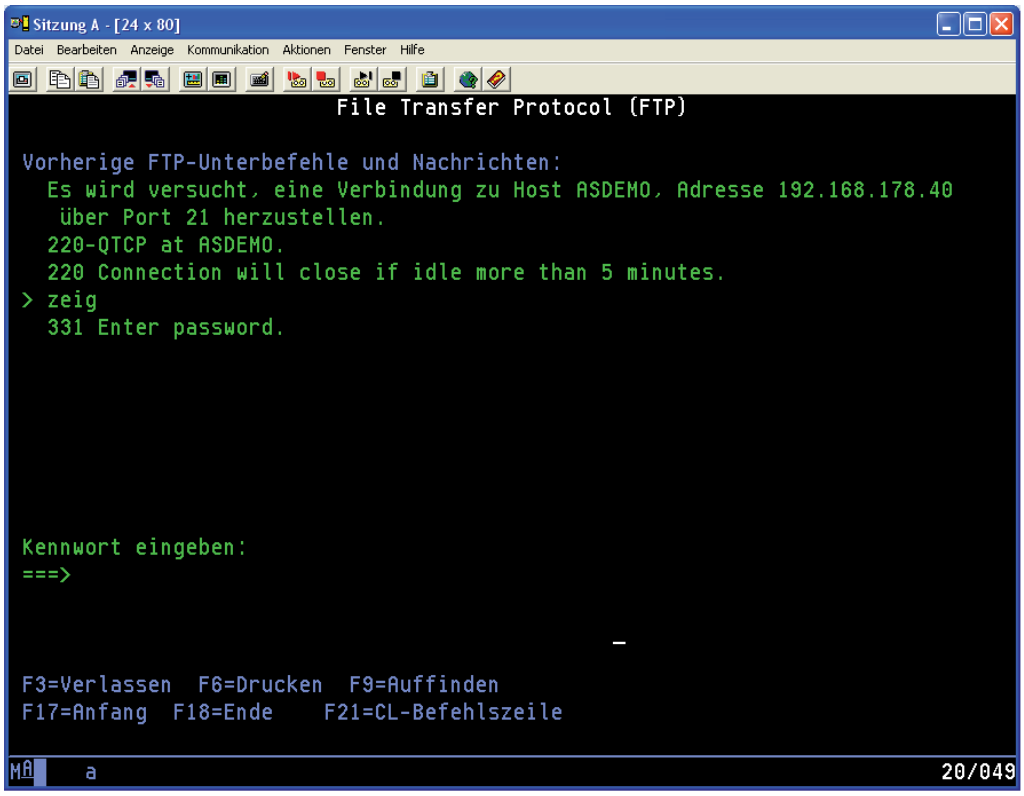
5.4.6.3
Seite 4

Um die FTP Verbindung herzustellen, geben wir also ein gültiges System i Benutzerprofil in der Befehlszeile ein.



Benutzereingabe

Das System prüft, ob der Benutzer existiert und fordert danach zur Eingabe des zugehörigen Kennwortes auf:

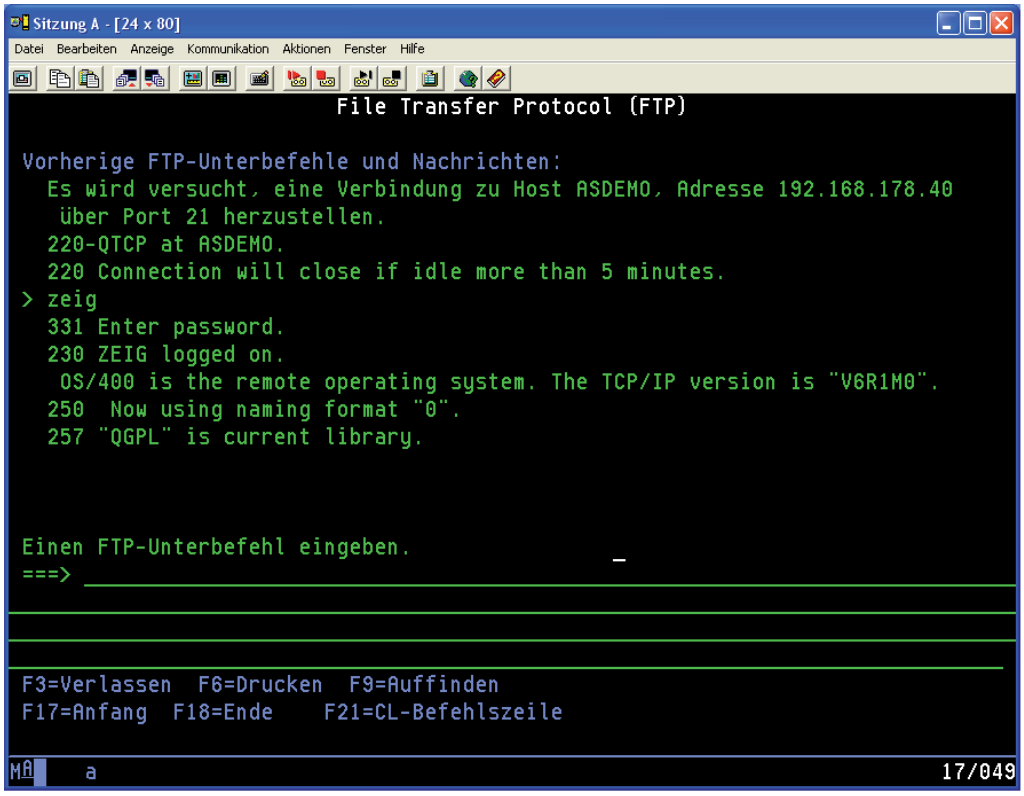


Kennworteingabe

Anmerkung

Sollten Sie eine fehlerhafte Benutzer ID oder ein falsches Kennwort eingegeben haben, dann können Sie diese nach Eingabe des FTP Befehls „User“ erneut eingeben.

Nach der Kennworteingabe und deren erfolgreicher Prüfung erhalten wir den eigentlichen FTP Anmeldebildschirm.



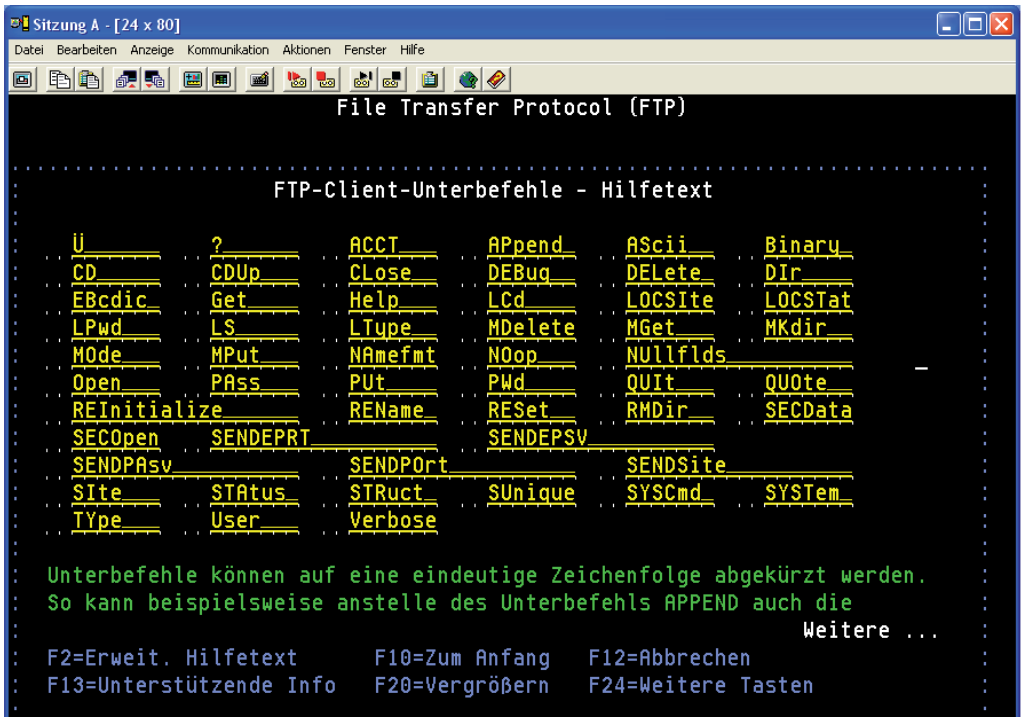
FTP Anmeldung erfolgreich

Basierend auf den FTP Vorgaben nutzen wir für diese FTP Sitzung beispielsweise das Namensformat „0“ und verwenden die aktuelle Bibliothek – in dem Beispiel die QGPL.

In diesem FTP Client können wir nun die gewünschten FTP Befehle eingeben um beispielsweise einen FTP Datenaustausch durchzuführen.

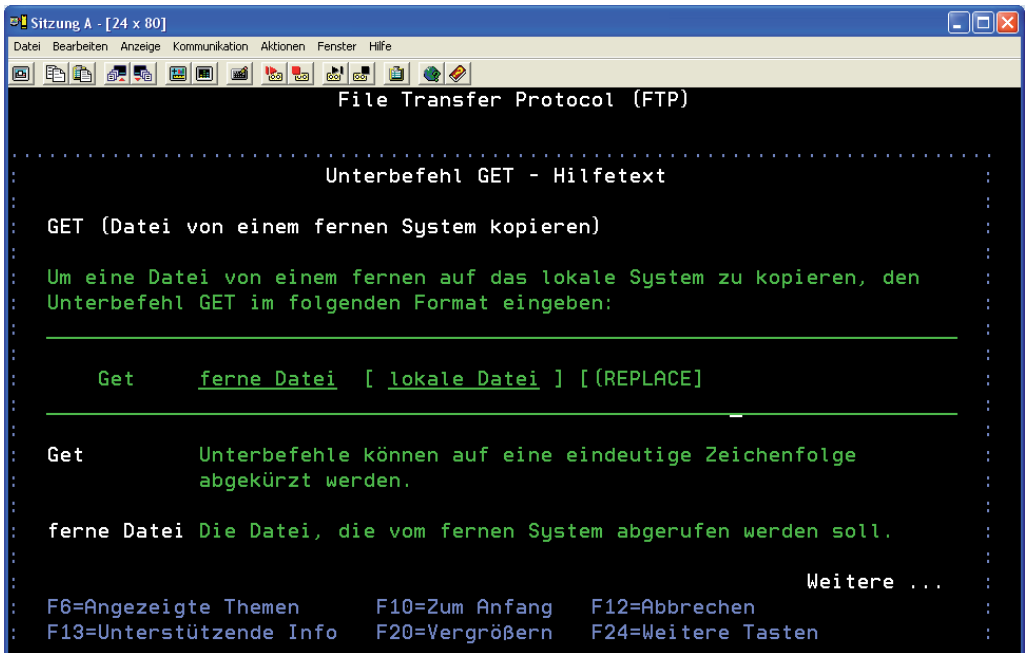
5.4.6.3
Seite 6

Beispiele für die möglichen FTP Befehle lassen sich nach Eingabe des Befehls „help“ einsehen:



help

Um die Bedienerhilfe für einen speziellen Befehl abzurufen, positionieren wir den Cursor auf den gewünschten Befehl und drücken die Eingabetaste. Damit erhalten wir die jeweilige Bedienerführung, wie sie in der folgenden Abbildung zu sehen ist.

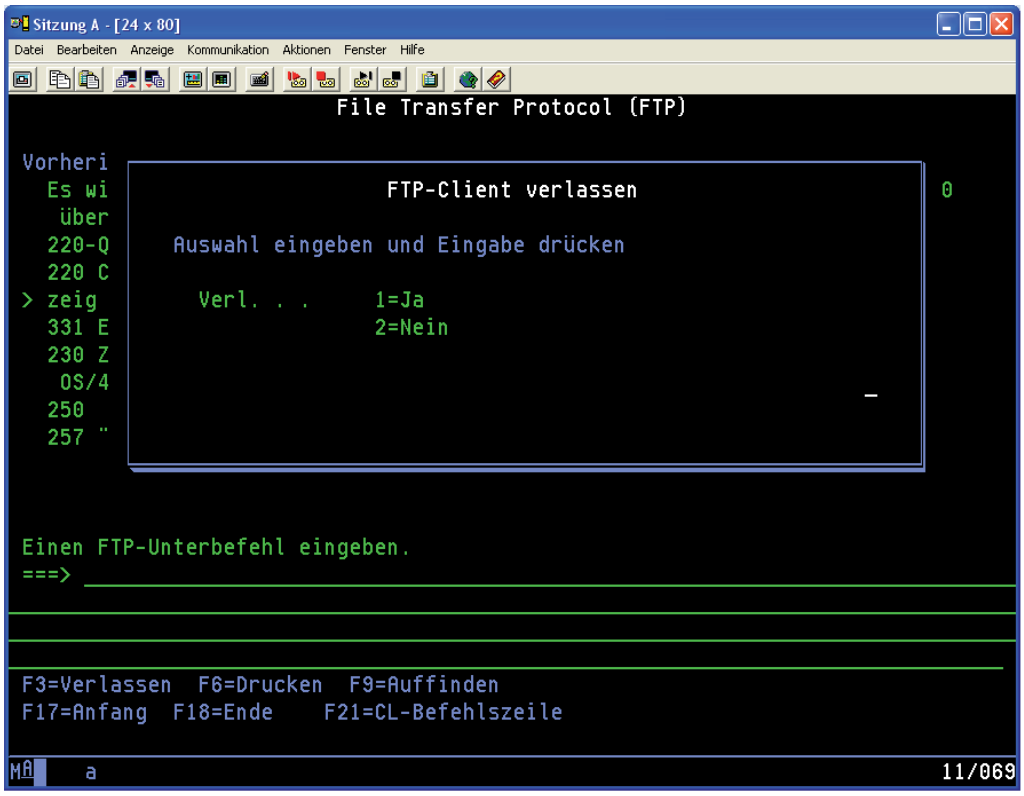


Beispiel für Bedienerhilfe

Achtung!

Sollten Sie bereits mit FTP vertraut sein, dann werden Sie wahrscheinlich den einen oder anderen FTP Befehl in der Auflistung vermissen. Der Grund liegt darin, dass das IBM i nicht alle FTP Befehle unterstützt. Auch sind einige der Befehle für den Einsatz mit dem IBM i optimiert und damit leicht verändert im Vergleich zu deren Einsatz auf anderen Betriebssystemen.

Der i5 FTP Client unterstützt die Befehlstasten, die von der nativen IBM i Umgebung bekannt sind. So kann der FTP Bereich beispielsweise mit der Funktionstaste F3 verlassen werden. Eine zusätzliche Abfrage stellt sicher, dass man nicht irrtümlich den FTP Bereich verlässt:



Verlassen

Alternativ zum Einsatz der Befehlstaste „F3“ kann der FTP Bereich auch mit dem FTP Befehl „Quit“ verlassen werden. In diesem Fall erscheint keine zusätzliche Abfrage, ob der FTP Bereich tatsächlich verlassen werden soll.

Lassen Sie uns nun zwei einfache Beispiele für den Einsatz von FTP anschauen. In einem der Beispiele übertragen wir Daten von dem lokalen IBM i auf ein anderes System, in dem zweiten Beispiel rufen wir Daten von einem fernen System zur Übertragung auf das IBM i ab. In diesen beiden Beispielen werde ich Ihnen auch einige FTP Befehle zeigen, die häufig Anwendung finden.

5.4.6.3

Seite 8

Datenübertragung an fernes System

Die Datenübertragung an ein fernes System basiert auf der Verwendung des Befehls „put“. In dem nachfolgenden Beispiel wollen wir den Inhalt einer Bibliothek „Demo“ auf ein fernes System übertragen. Wie bereits zuvor erwähnt, müssen Bibliotheken oder Objekte für die Übertragung vorbereitet werden. Dazu speichern wir diese in eine Sicherungsdatei (SAVF), die wir zuvor mit dem Befehl CRTSAVF erstellt haben.

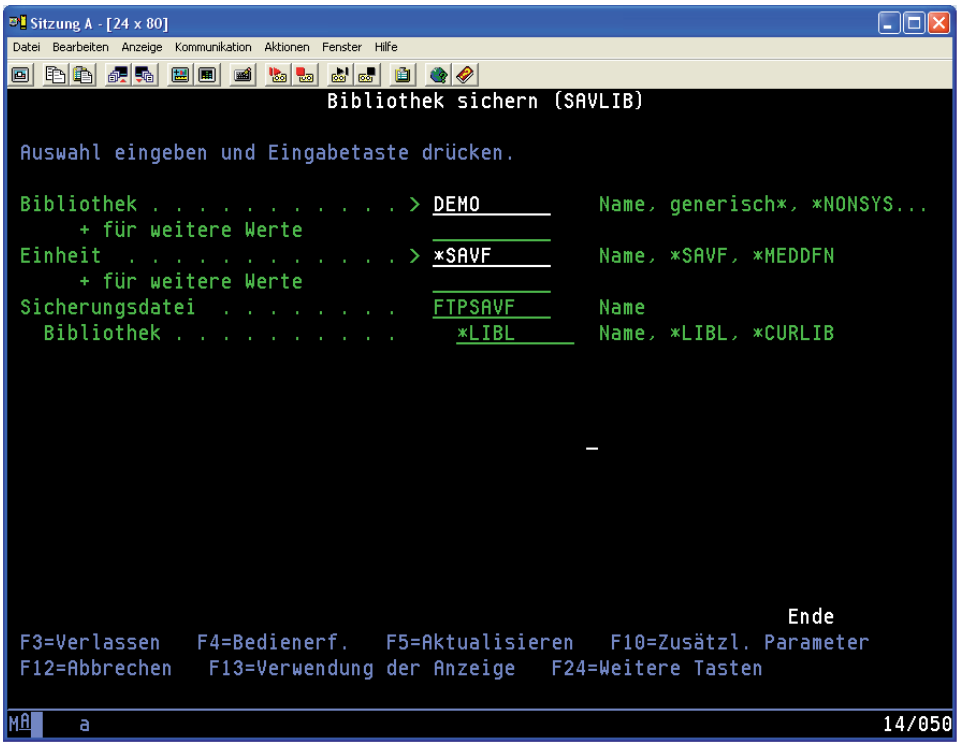
Ausführungsschritte

1. Erstellen Sie eine Sicherungsdatei. Geben Sie dazu den folgenden Befehl ein:

   CRTSAVF *SAVFName* z. B. CRTSAVF FTPSAVF

2. Sichern Sie die zu übertragenden Objekte oder Bibliotheken in die Sicherungsdatei. Dazu verwenden wir einen der typischen Sicherungsbefehle wie z. B. SAVLIB oder SAVOBJ.

   In unserem Beispiel wollen wir den Inhalt der Bibliothek „DEMO“ übertragen. Deshalb verwende ich den Befehl SAVLIB, wie er in der folgenden Abbildung zu finden ist:



SAVLIB

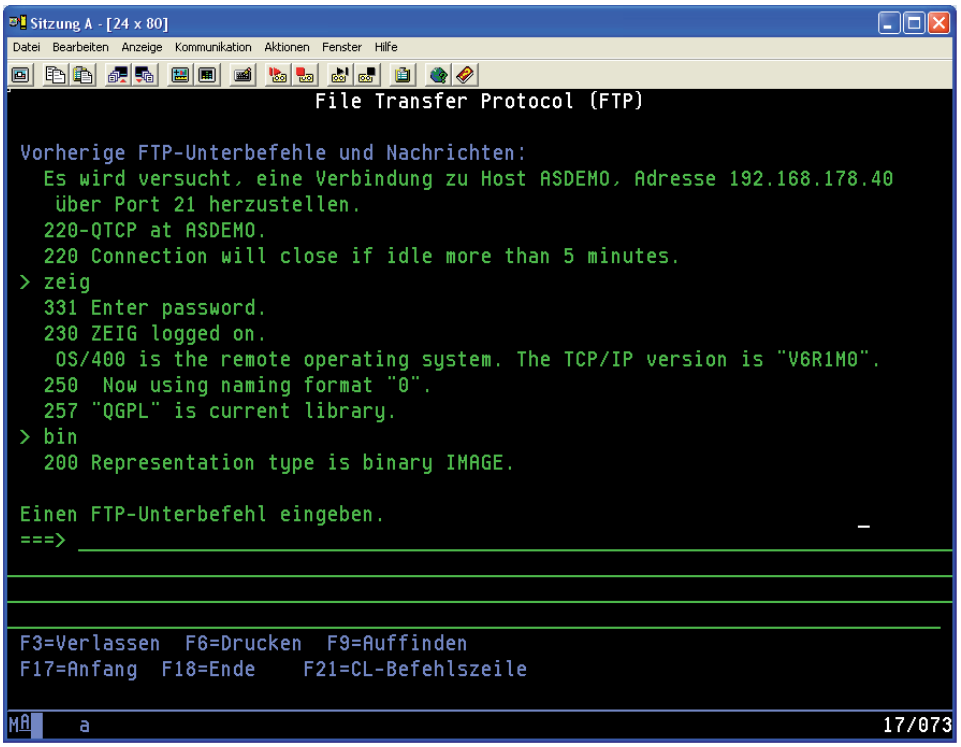
5.4.6.3.1
Seite 2

Damit haben wir die Voraussetzungen für die Übertragung der IBM i Informationen auf ein anderes System geschaffen. Diese Schritte sind lediglich für die Übertragung von Bibliotheksbereichen des IBM i erforderlich. Wenn Sie den Inhalt des IFS übertragen wollen, dann brauchen Sie diesen IBM i spezifischen Zwischenschritt der Erstellung einer Sicherungsdatei nicht ausführen.

3. Starten Sie nun die 5250-FTP-Sitzung durch Eingabe des Befehls „FTP“ gefolgt von dem Hostnamen des fernen Systems (z. B. ftp ASDEMO).

   Authentifizieren Sie sich an dem fernen System mit einer gültigen IBM i Benutzer ID und geben Sie nach Aufforderung das zugehörige Kennwort ein.

   Für die Übertragung der IBM i Daten ist das binäre Format erforderlich. Dieses können wir in FTP mit der Eingabe des Befehls „bin“ initiieren.



Binäres Format

Mit dem Befehl „PUT“ lassen sich nun die IBM i Daten auf das ferne System übertragen. Die Syntax für diesen Befehl ist *„put Bibliothek/Datei“*. In unserem Beispiel befindet sich die zu übertragende Datei FTPSAVF in der Bibliothek QGPL. Da diese die aktuelle Bibliothek ist, braucht die Bibliothek bei dem Befehl „PUT“ nicht angegeben werden. Zum Zwecke der Demonstration gebe ich dennoch diesen qualifizierten Befehl mit der Bibliothek an:

Put QGPL/FTPSAVF

Datenübertragung von einem fernen System

Die Datenübertragung von einem fernen System basiert auf der Verwendung des Befehls „get“. Auch hier gelten die Besonderheiten der Übertragung von Bibliotheksinhalten des IBM i. Wenn man zum Beispiel den Inhalt einer Bibliothek auf das System i übertragen möchte, dann muss zunächst eine leere Sicherungsdatei angelegt werden. In diese kann dann anschließend die Sicherung der Bibliotheken oder Objekte durchgeführt werden. Bei dieser Übertragung ist ebenfalls das binäre Format erforderlich. Die nachfolgenden Schritte zeigen exemplarisch das Abholen von Daten von einem fernen System mittels FTP.

Ausführungsschritte

1. Erstellen Sie eine Sicherungsdatei. Geben Sie dazu den folgenden Befehl ein:

   CRTSAVF *SAVFName* z. B. CRTSAVF FTPSAVF

2. Starten Sie nun die 5250-FTP-Sitzung durch Eingabe des Befehls „FTP“ gefolgt von dem Hostnamen des fernen Systems (z. B. ftp ASDEMO).

3. Authentifizieren Sie sich an dem fernen System mit einer gültigen IBM i Benutzer ID und geben Sie nach Aufforderung das zugehörige Kennwort ein.

4. Für die Übertragung der IBM i Daten ist das binäre Format erforderlich. Dieses können wir in FTP mit der Eingabe des Befehls „bin“ initiieren.

5. Mit dem Befehl „GET“ lassen sich nun die Daten von dem fernen System abrufen. Befinden sich diese Daten in einem speziellen Verzeichnis, dann können wir in dieses beispielsweise mit dem FTP Befehl „CD“ wechseln. Ein Beispiel für den Befehl CD ist „*cd qgpl*“.

6. Um Daten von einem fernen System abzuholen, geben wir den Befehl get wie folgt ein:
   get Bibliotheksname/Dateiname

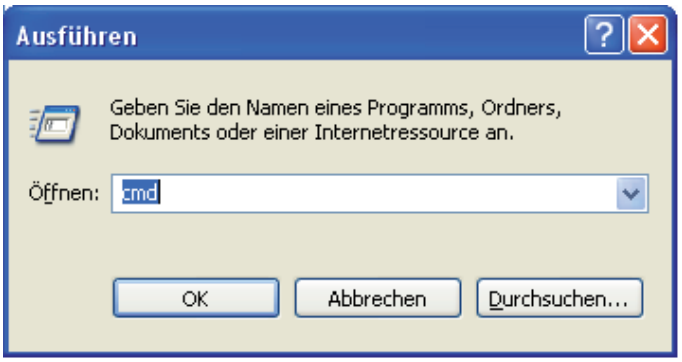
Alternativ zu dem Einsatz des i5 FTP Clients können wir auch den FTP Client eines „fremden“ Betriebssystems zur Datenübertragung verwenden. Ein Beispiel ist die FTP Übertragung von Windows.

5.4.6.3.2
Seite 2

Um zum Beispiel die Daten eines IBM i auf ein Windows System zu übertragen, können wir wie folgt vorgehen:

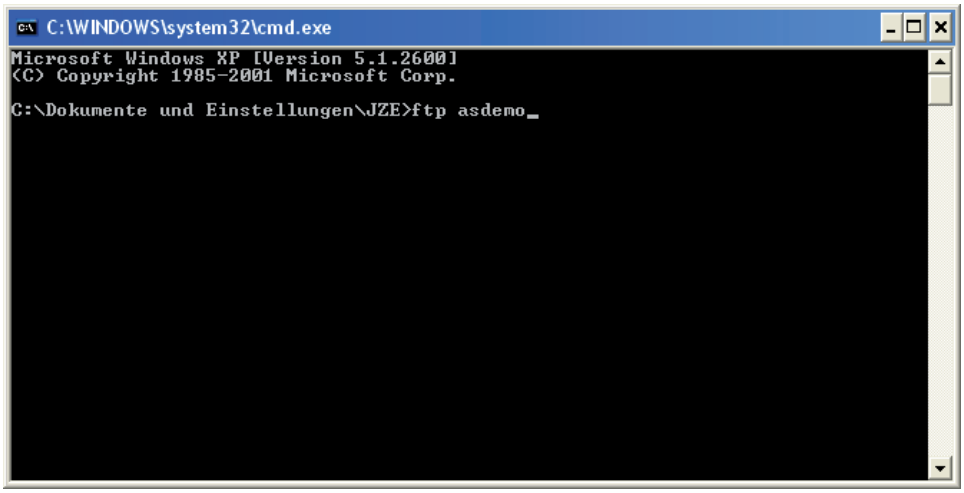
Ausführungsschritte

1. Wählen Sie in Windows die Option „Start / ausführen“.
2. Geben Sie in dem Fenster „Öffnen“ den Befehl „CMD“ ein.



CMD

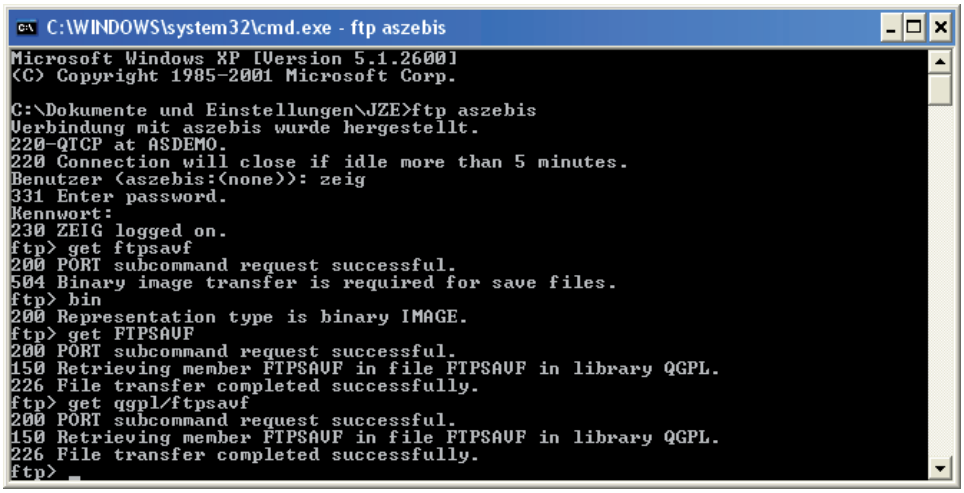
3. In der sich öffnenden DOS-Anzeige geben wir nun den Befehl „FTP“ gefolgt von dem Hostnamen des System i oder dessen IP Adresse an.



FTP Start

4. Folgen Sie den Anweisungen, welche in der Anzeige erscheinen. Geben Sie demnach eine Benutzer ID und anschließend das zugehörige Kennwort ein.

5. Um die Daten eines System i auf ein Windows System zu übertragen, geben wir nun zunächst wieder das binäre Format an. Dazu geben wir in der FTP Anzeige den Befehl „BIN“ ein.



FTP Get

6. Die Datenübertragung starten wir mit dem Befehl „*get Bibliotheksname/Dateiname*“.

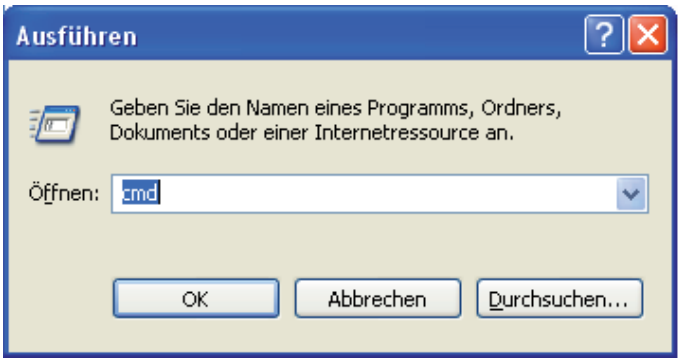
In dem vorhergehenden Beispiel sind wir davon ausgegangen, dass die System i Daten in Form gesicherter Bibliotheken oder Objekte zur Verfügung stehen. Natürlich lassen sich auch die Inhalte des IFS mittels FTP übertragen. Dazu sind beinahe dieselben Schritte erforderlich, wie ich sie im Vorfeld gezeigt habe. Der Unterschied liegt darin, dass wir keine Sicherungsdatei benötigen, um die Dateiinformationen zu übertragen. Auch der binäre Modus ist nicht erforderlich.

5.4.6.3.2
Seite 4

Das nachfolgende Beispiel zeigt exemplarisch das Übertragen eines IFS Inhaltes von dem System i auf einen Windows Rechner:

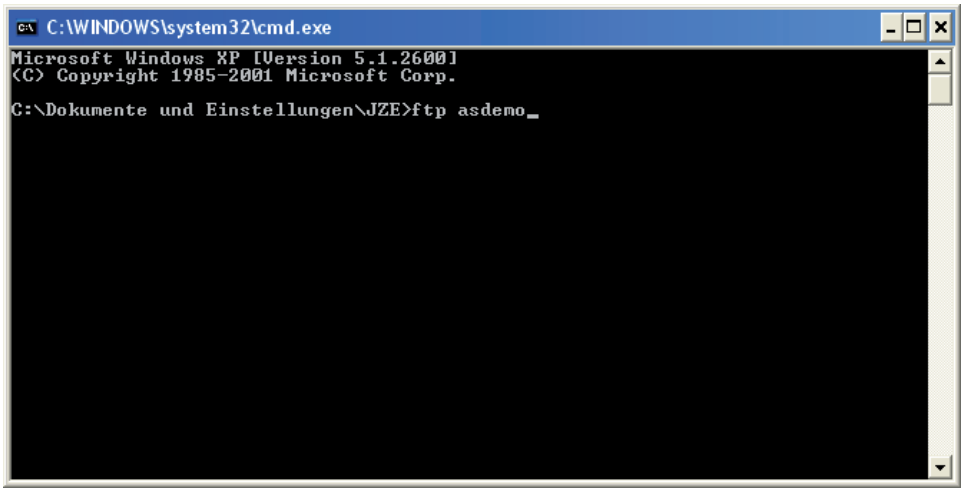
Ausführungsschritte

1. Wählen Sie in Windows die Option „Start / ausführen“.
2. Geben Sie in dem Fenster „Öffnen“ den Befehl „CMD“ ein.



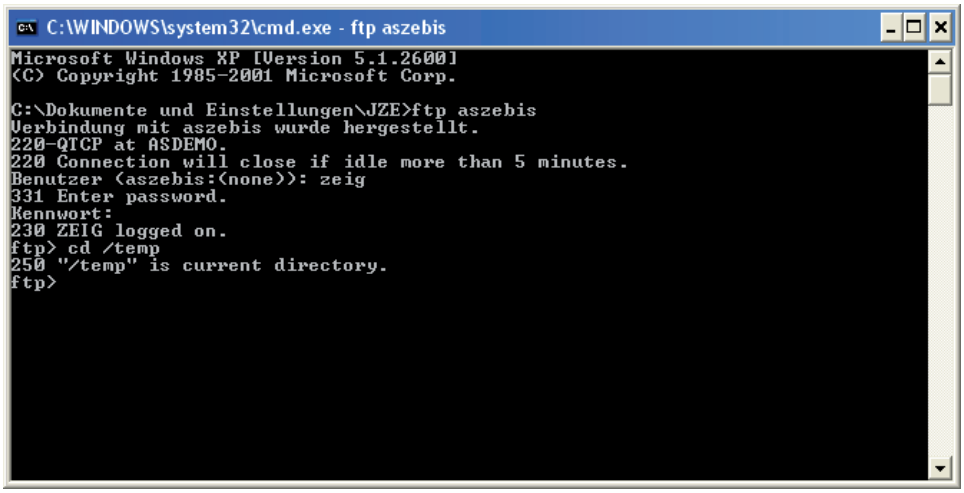
CMD

3. In der sich öffnenden DOS-Anzeige geben wir nun den Befehl „FTP“ gefolgt von dem Hostnamen des System i oder dessen IP Adresse an.



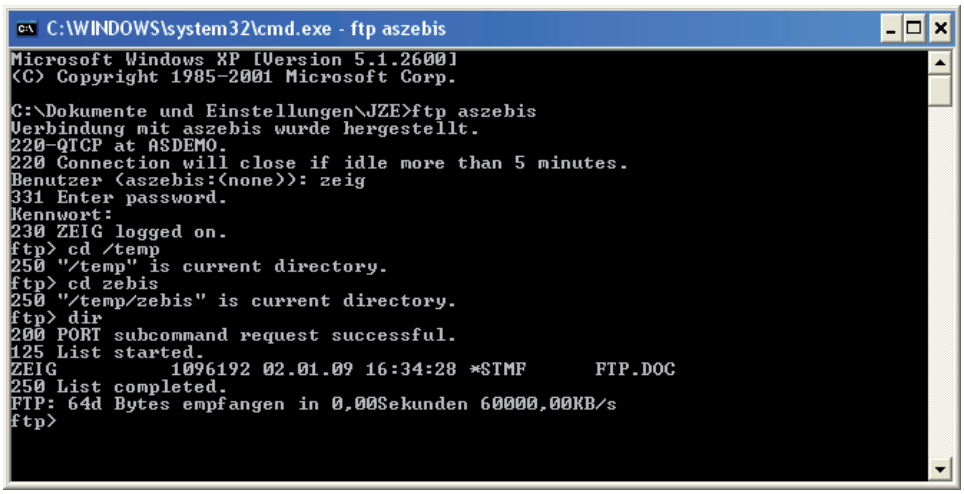
FTP Start

4. Folgen Sie den Anweisungen, welche in der Anzeige erscheinen. Geben Sie demnach eine Benutzer ID und anschließend das zugehörige Kennwort ein.
5. Nun wechseln wir mit dem Befehl „cd“ in das gewünschte Verzeichnis, in dem sich die zu übertragenden Daten befinden. In unserem Beispiel ist dies das Verzeichnis „temp“.



Verzeichniswechsel

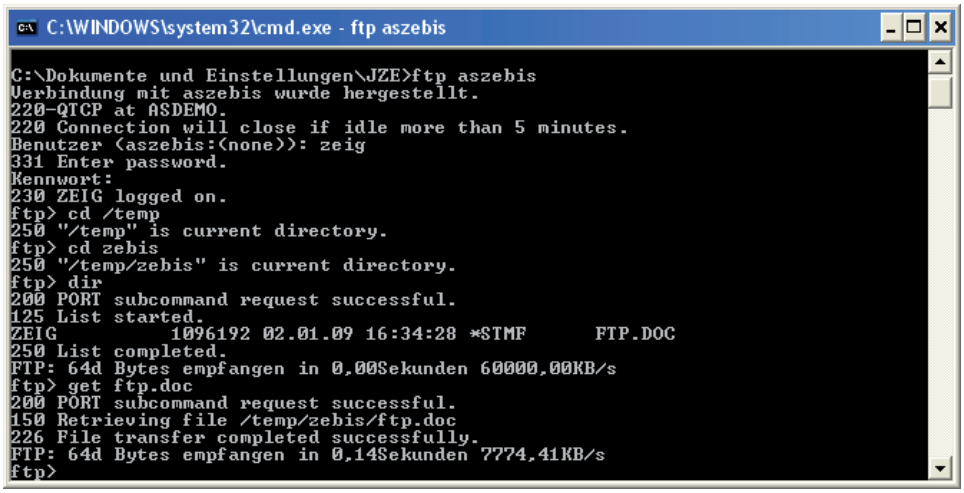
6. Wir können exemplarisch den Inhalt des ausgewählten Verzeichnisses einsehen. Dazu verwenden wir den Befehl „dir“.
7. In unserem Beispiel befinden sich die zu übertragenden Daten in dem Verzeichnis „ZEBIS“, in das ich nun, durch die erneute Eingabe des Befehls „CD“ gefolgt von dem Verzeichnisnamen, wechsle. In diesem Verzeichnis befindet sich ein Word Dokument, das ich nun auf den PC übertragen möchte. Den Inhalt kann man sich erneut mit DIR anzeigen lassen.



Ergebnis von DIR

8. Die Datenübertragung starten wir mit dem bereits bekannten Befehl „Get“.

5.4.6.3.2
Seite 6

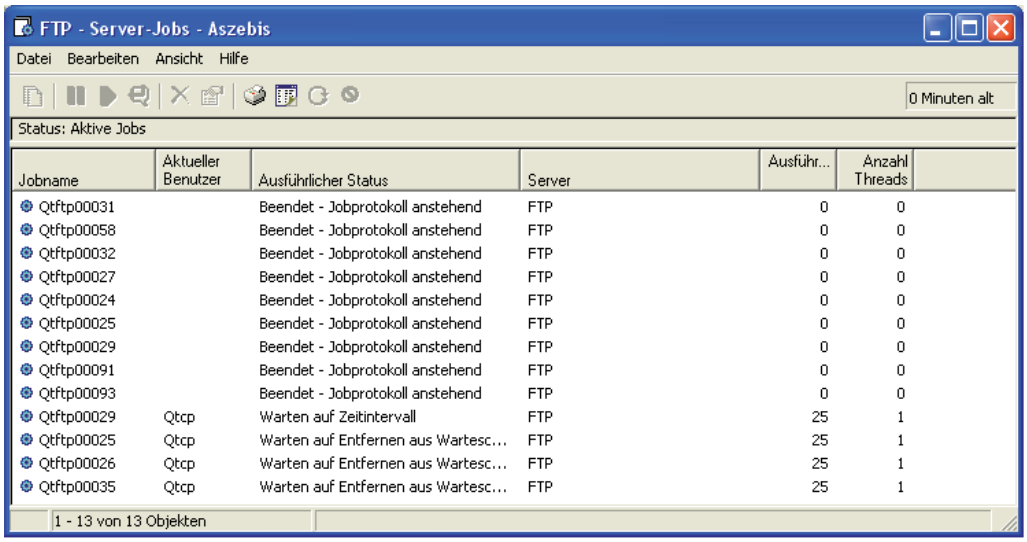


get

9. Damit haben wir PC Daten vom IFS auf einen PC übertragen.

Mit FTP Jobs arbeiten

Nach dem Starten des FTP Server Dienstes werden auf dem System i verschiedene Jobs gestartet, welche für die Ausführung von FTP zuständig sind. Diese Jobs finden wir in dem zugeordneten Subsystem (Standard: QSYSWRK). Diese Jobs können beispielsweise mit dem Befehl WRKACTJOB oder über die iSeries Navigator Funktion „Server-Jobs“ für den FTP Serverdienst eingesehen und verwaltet werden.



Serverjobs

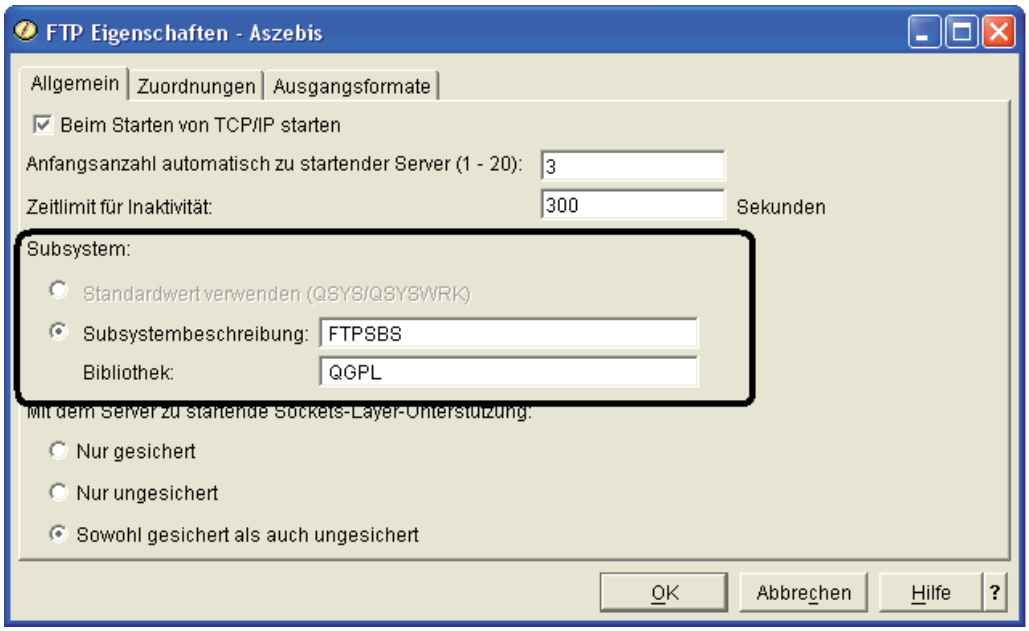
5.4.6.3.3

Seite 2

FTP Performance

Die Geschwindigkeit für FTP Übertragungen hängt von unterschiedlichsten Faktoren ab. Neben der Verbindungsgeschwindigkeit sind auch Firewall oder Virenscanner maßgeblich für die Beeinflussung der Performance verantwortlich. Aber auch das System i bzw. die dort vorherrschenden Einstellungen und Verhältnisse wirken sich auf die Leistungsfähigkeit von FTP aus. Per Standard werden die FTP Jobs in dem Subsystem QSYSWRK ausgeführt. Dieses „Universalsubsystem" dient allerdings auch der Ausführung von Nicht-FTP Jobs. Es kommt auf den individuellen Fall an, ob die Leistungsfähigkeit der QSYSWRK Umgebung ausreicht, um FTP Jobs neben anderen in diesem Subsystem ausgeführten Jobs zufriedenstellend abarbeiten zu können. Wenn die Leistungsfähigkeit nicht ausreicht oder andere Faktoren wie zum Beispiel administrative Gründe für die Verwendung eines anderen Subsystems sprechen, dann kann man den FTP Bereich recht einfach anpassen und ein abweichendes Subsystem angeben, in dem dann die FTP Jobs ausgeführt werden.

Diese Einstellungen lassen sich ebenfalls auf 5250 Ebene oder im iSeries Navigator vornehmen. Die nachfolgende Abbildung zeigt die FTP Eigenschaften im iSeries Navigator. Dort können wir das Subsystem angeben, dass für die FTP Ausführung ab dem nächsten Start des FTP Dienstes verwendet werden soll.



FTP Eigenschaften

Achtung!

Das Subsystem muss vorher definiert worden sein. Details dazu finden Sie in dem Kapitel 2.7.2

5.4.6.3.4

Seite 2

DDM (Distributed Data Management)

Die Kommunikation über Systemgrenzen hinaus ist eine der Standardanforderungen, denen sich moderne Systeme und Anwendungen jeden Tag stellen müssen.

Insbesondere die Anforderungen an die Homogenisierung der Daten im Zuge eines Austauschs zwischen verschiedenen Plattformen (z. B. auch mit unterschiedlichen Zeichensätzen), stellen Entwickler und Planer immer wieder aufs Neue vor Herausforderungen.

IBM hat seit vielen Jahren auch dem Betriebssystem i und dessen Vorgängern Funktionen implementiert, mit denen der System i-Betreiber in der Lage ist, schnell und zielgerichtet auf Informationen fremder Systeme zuzugreifen.

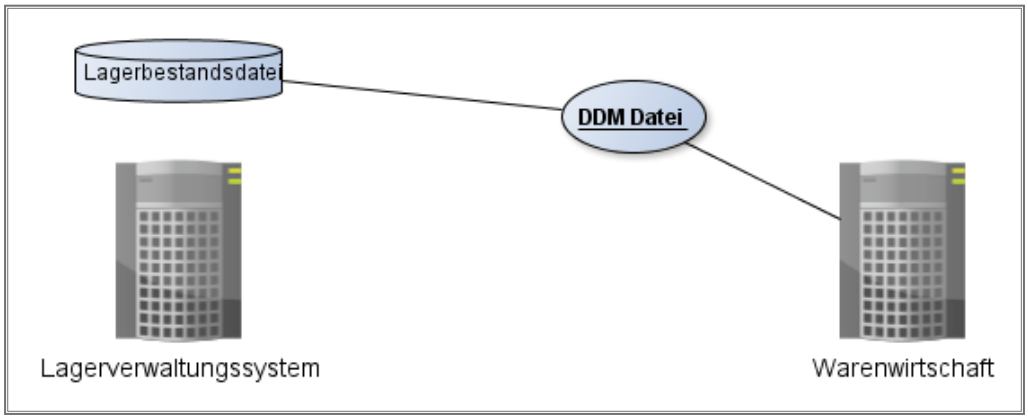
DDM ist die Abkürzung für „Distributed Data Management“ und wird als Bestandteil des Betriebssystems IBM i bereits seit der AS/400 mit jeder Maschine ausgeliefert. Die DDM-Kommunikationsarchitektur kann in unterschiedlicher Art und Weise zum Einsatz kommen, etwa als Grundlage für die Ausführung ferner Befehle oder zur Bereitstellung von Daten auf fernen Systemen, wobei die Daten selbst dort nicht redundant abgelegt sein müssen.

Während in früheren Betriebssystemversionen SNA für die Kommunikation zwischen verschiedenen Systemen eingesetzt wurde, kommt heute dabei meist TCP/IP zum Einsatz. Auch wenn SNA für die Verwendung mit DDM immer noch konfiguriert werden kann, sollten zukunftsorientierte Administratoren doch dem Einsatz von TCP/IP-Komponenten zusammen mit DDM den Vorzug geben – denn der Lebenszyklus von SNA neigt sich allmählich seinem Ende entgegen.

DDM kann in unterschiedlichen Bereichen zum Einsatz kommen:

- Als Grundlage für die Bereitstellung von DB2-Komponenten über die Grenzen der lokalen Maschine hinaus
  - für die Verwendung auf einer anderen i-Maschine
  - für die Verwendung auf einer Nicht-i-Maschine
- Als Grundlage für die Ausführung ferner Befehle
  - z. B. RMTCMD in Windows

Ein Beispiel: Sie haben eine Umgebung, in der auf einer Maschine (oder LPAR) ein Warenwirtschaftssystem läuft. Auf einem anderen Rechner wird das Lagerverwaltungssystem geführt. Um nun auf die Informationen des fernen Rechners (etwa die Lagerbestandsdatei) zugreifen zu können, erstellen wir uns eine Art Link zu der gewünschten Datei. Dieser Link wird als DDM-„Datei“ abgelegt und verweist auf die ferne Datei. Damit kann dieser „Link“ in Programmen so eingesetzt werden, als würde sich die Datei auf dem lokalen System befinden.



Grobskizze DDM

Somit haben wir einen direkten Zugriff auf die Informationen des fernen Systems – denn sobald diese dort fortgeschrieben werden, stehen Sie automatisch auch in der verlinkten DDM-Datei zur Verfügung.

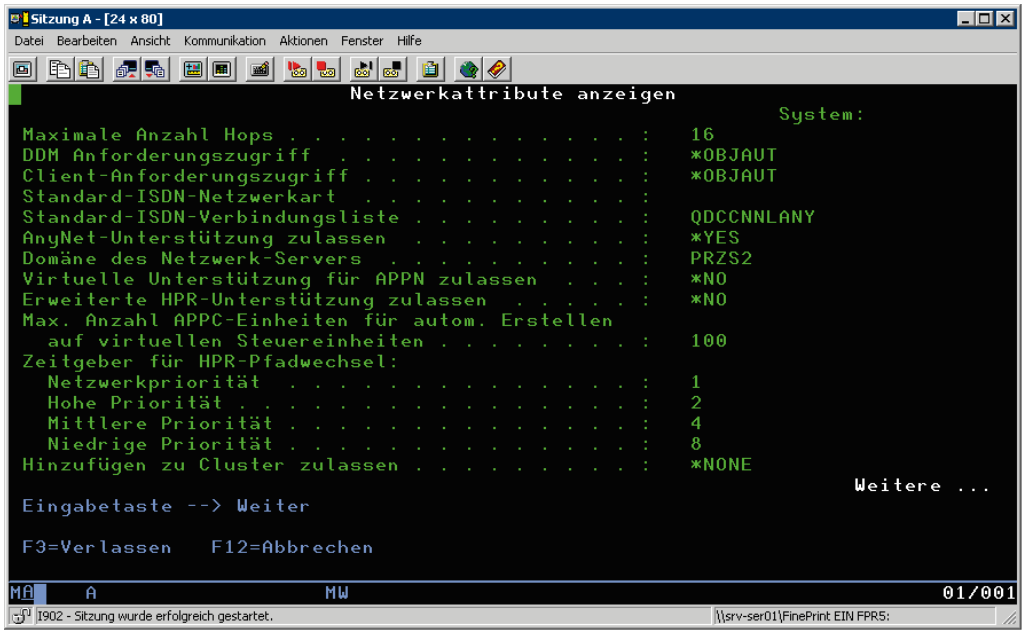
Sicher wird der eine oder andere Leser nun sagen, dass diese Form des Zugriffs auf ferne Informationen nicht zeitgemäß ist – denn moderne SQL-Techniken erlauben es uns schließlich auch, auf ferne SQL-Datenbanken und deren Elemente zugreifen zu können.

Das ist in der Tat eine weitere Option des Zugriffs auf ferne Informationen. Aber wie in vielen Bereichen sind auch hier Vor- und Nachteile der verschiedenen Möglichkeiten gegeneinander abzuwägen.

Ein anderes Beispiel für den Einsatz von DDM ist die Möglichkeit, ferne Befehle von einem Nicht-IBM-i-System absetzen zu können. RMTCMD etwa ist eine Option, die es z. B. in einer Windows-Umgebung erlaubt, einen Befehl auf dem System i absetzen und ausführen zu können. RMTCMD nutzt ebenfalls die DDM-Architektur.

DDM kann wahlweise auf Basis einer APPC-Verbindung oder mittels TCP/IP eingesetzt werden. Als Empfehlung gilt die Verwendung von TCP/IP – denn APPC basiert auf SNA und das ist – wie gesagt – eine „in die Jahre gekommene“ Technologie.

Sicherheit ist natürlich eines der Themen, das auch beim Einsatz von DDM nicht zu sorglos angegangen werden sollte. IBM hat eine Schlüsselstelle im System implementiert, über die allgemein gesteuert werden kann, ob DDM mit dem System eingesetzt werden kann. Diese Schlüsselstelle befindet sich in den Netzwerkattributen und verbirgt sich hinter dem Parameter DDMACC (DDM-Anforderungszugriff). Die folgende Abbildung zeigt den Parameter:



Netzwerkattribute

Eine der Besonderheiten beim Einsatz von DDM liegt darin, dass weder ein Benutzerprofil noch ein Kennwort mitgegeben werden können. DDM arbeitet stets mit dem aktuellen Benutzerprofil – folglich dem Benutzerprofil, das angemeldet ist oder unter dem der Job ausgeführt wird. Demnach ist es eine Anforderung, dass die Benutzerprofile auf den beiden Maschinen gleich zu halten sind. Dies gilt natürlich nur für die Benutzerprofile, die DDM-Komponenten verwenden sollen. Hier hilft der Grundsatz, dass das Benutzerprofil auf beiden Maschinen existent sein soll und die Berechtigungen für den Einsatz von DDM ausgelegt sind.

5.4.7
Seite 4

DDM-Voraussetzungen

Schauen wir uns nun die Voraussetzungen an, die für die Verwendung von DDM Beachtung finden müssen.

DDM und APPC

Sollten Sie sich für den Einsatz von DDM und APPC entscheiden, dann sind die nachfolgenden Voraussetzungen zu erfüllen:

- Mindestens eine Verbindungsleitung, die SDLC verwendet
- APPC/APPN-Unterstützung
  - Lizenzprogramm iSeries Access

DDM und TCP/IP

Bei der Verwendung von DDM über TCP/IP müssen die folgenden Voraussetzungen erfüllt sein:

- Lizenzprogramm 57xx-TC1 muss installiert sein
- Definierte Leitung für den Einsatz von TCP/IP
- Basiseinrichtung von TCP/IP
  - Schnittstellen
  - TCP/IP Route
- DDM-Serverdienst muss gestartet sein

Bis auf den letzten Punkt dürften alle anderen Punkte auf Ihrem System bereits umgesetzt sein – bildet TCP/IP doch die Basis für die Kommunikation in einem Netzwerk!

DDM-Dateianforderungen

Um eine Datei mittels DDM übertragen zu können, müssen bei der Verwendung von APPC die betreffenden Steuereinheiten auf dem System existieren.

34. Ergänzung 3/2013

5.4.7.1

Seite 2

DDM-Zugriff allgemein

Wie bereits zuvor erwähnt, kann über eine zentrale Stelle in den Netzwerkattributen gesteuert werden, dass/ob DDM-Zugriff auf das System erlaubt ist. Der Parameter DDMACC (DDM/DRDA-Anforderungszugriff) steuert die Berechtigungsprüfungen für den DDM-Zugriff. Mögliche Werte sind hier:

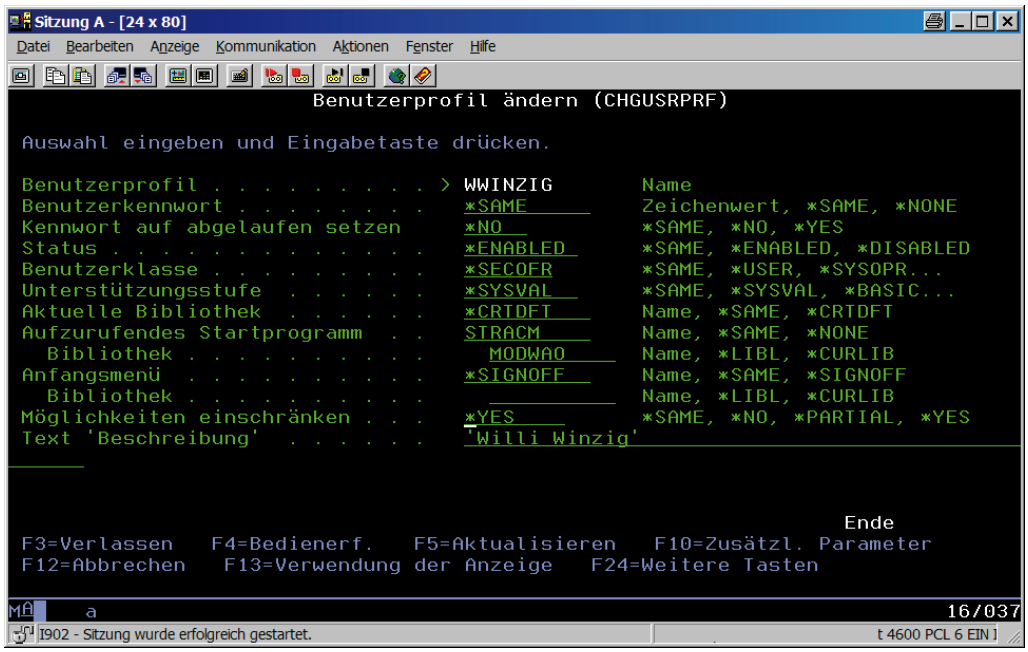
- *REJECT
  - Das System verweigert jeglichen DDM/DRDA-Zugriff, ist aber in der Lage, selbst auf fremde DDM/DRDA-Komponenten zuzugreifen.
- *OBJAUT
  - Wenn das verwendete Benutzerprofil die Berechtigungen für die Datei hat, werden diese entsprechend der Berechtigungsdefinition verwendet.

Ein essentiell wichtiger Punkt beim Einsatz von DDM ist das Thema Sicherheit.

DDM basiert auf dem Berechtigungskonzept von IBM i – zumindest in weiten Bereichen.

Allerdings gibt es eine bedeutende Ausnahme, die DDM als potenzielle Sicherheitslücke erscheinen lässt.

Diese „Sicherheitslücke“ liegt in einer besonderen Sicherheitseinstellung, die es auf dem System i verhindern soll, dass ein Anwender Befehle in einer Befehlszeile eingeben und absetzen kann. Der Parameter LMTCPB(*YES) ist genau dafür zu verwenden und kann auf Ebene des Benutzerprofils definiert werden. Ein Beispiel zeigt die folgende Abbildung:



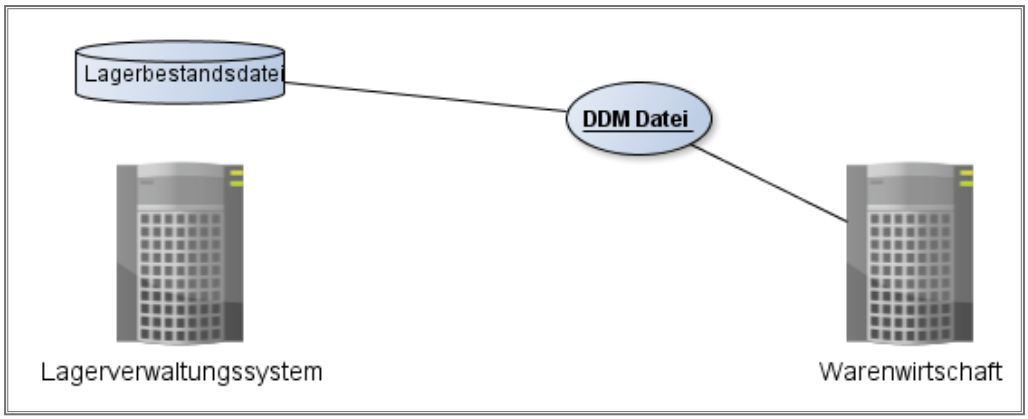
CHGUSRPRF

5.4.7.2
Seite 2

Im Normalfall dient LMTCPB(*YES) dazu, die Eingabemöglichkeiten auf Befehlszeilenebene auf einige wenige „harmlose“ Standardfunktionen zu beschränken. Diese sind z.B.

- Anzeige des eigenen Jobs (DSPJOB)
- Senden von Nachrichten (SNDMSG)

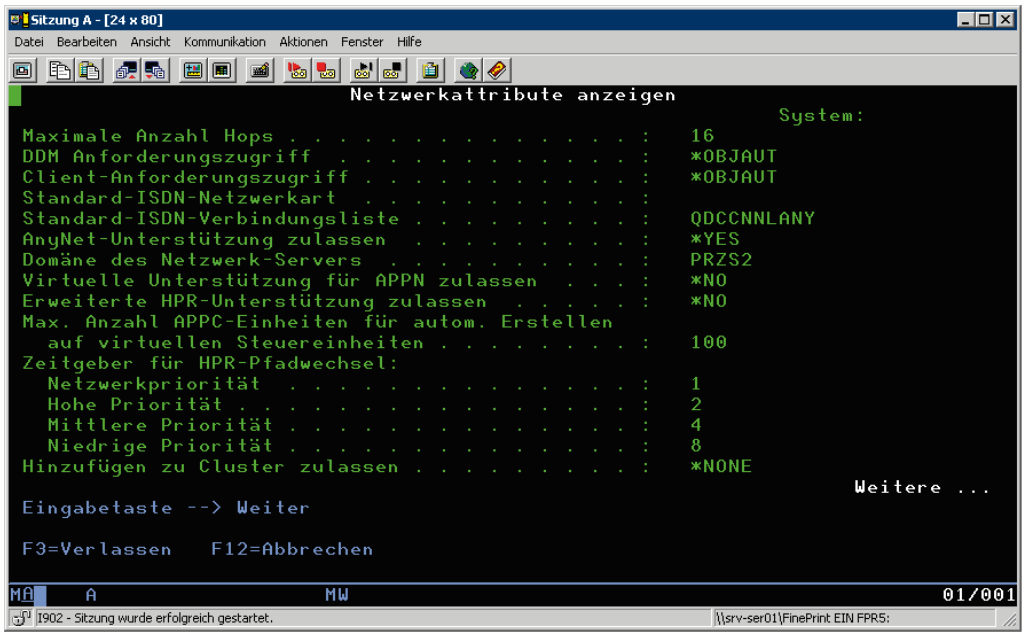
Aber wie verhält sich LMTCPB im Zusammenhang mit DDM?
Die Antwort: „Unerwartet“ – denn sollte man meinen, dass DDM die LMTCPB-Einstellung berücksichtigt, so liegt man schlicht falsch. Das bedeutet, dass ein Anwender, der die LMTCPB(*YES)-Einstellung für sein Benutzerprofil hinterlegt hat, dennoch in der Lage ist, Befehle auszuführen, die er sonst nicht in die Befehlszeile eingeben kann. Hier greift dann zwar das Berechtigungskonzept des Systems IBM i – aber dennoch ist dieser Sicherheitsaspekt unbedingt zu beachten, wenn DDM eingesetzt werden soll!



Aber auch hierfür gibt es eine Lösung. Die Verwendung von speziellen Exit Points kann auch diese Sicherheitslücke schließen. Hier ist dringend anzuraten, ggf. den Einsatz einer Zusatzsoftware in Erwägung zu ziehen, die genau diese Exit-Punkte überwacht und bereitstellt.

DDM-Serverdienst

Die Einstellungen für die Verwendung von DDM über TCP/IP lassen sich mit Hilfe des Befehls CHGDDMTCPA definieren. Die DDM-Kommunikation läuft per Standard über den Port 446. Stellen Sie vor der Konfiguration sicher, dass dieser Port frei ist. Das kann wahlweise mit dem Befehl NETSTAT *CMN oder über den System i Navigator erfolgen.

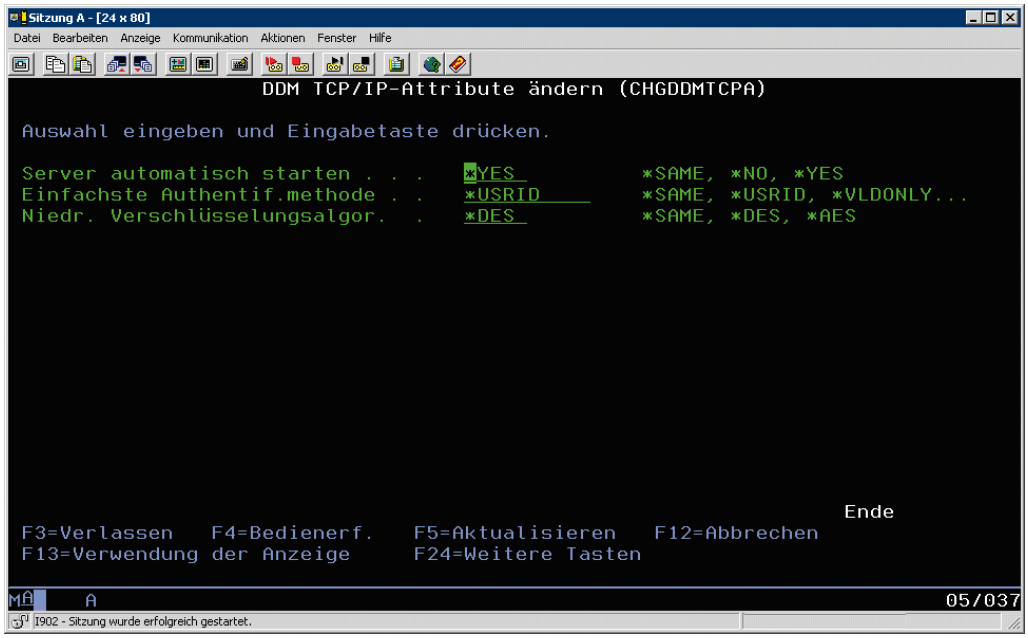


NETSTAT

In der Abbildung oben sehen Sie bereits genutzte DDM Ports.

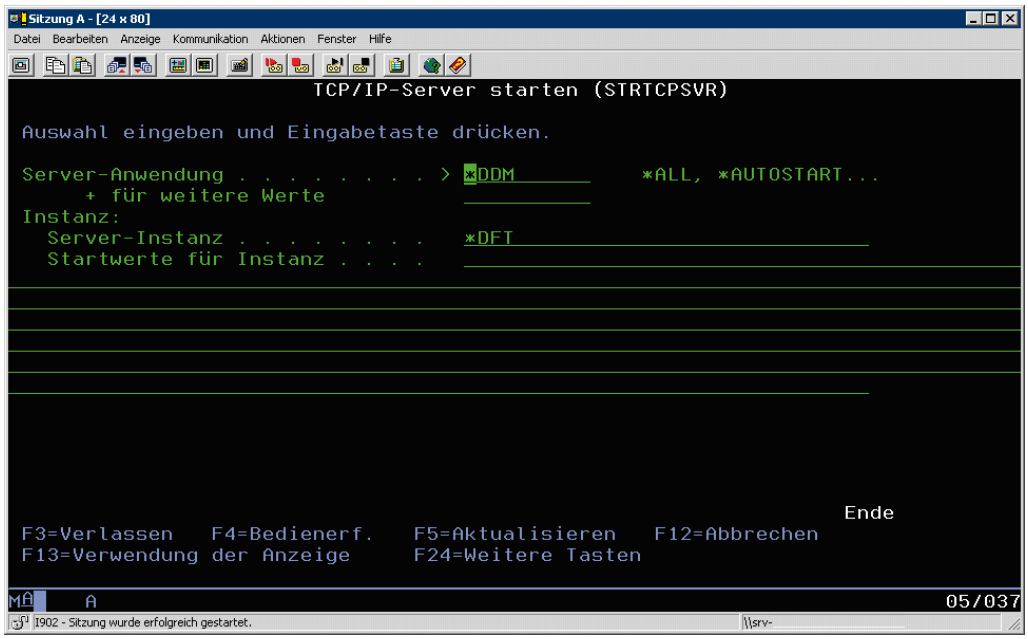
Damit DDM eingesetzt werden kann, muss der TCP/IP-Serverdienst DDM gestartet sein. Das kann wahlweise über die Voreinstellung „Server automatisch starten“ oder manuell über den Befehl STRTCPSVR geschehen.

5.4.7.3
Seite 2



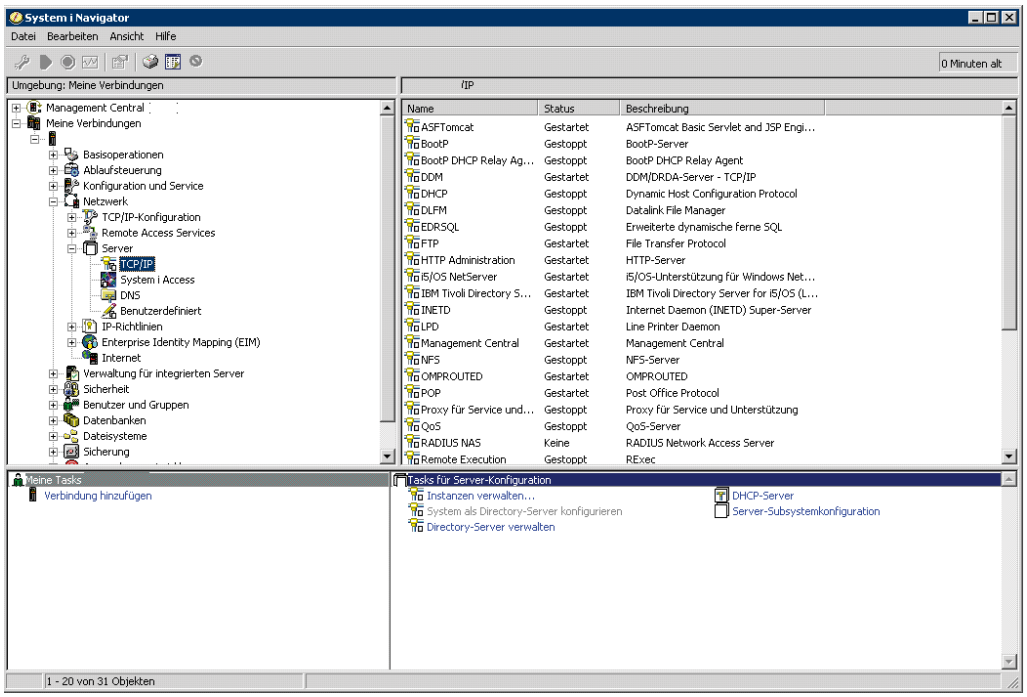
CHGDDMTCPA

Mit dem Befehl STRTCPSVR kann der DDM-Serverdienst gezielt gestartet werden. Verwendet man DDM regelmäßig und lassen es die Sicherheitsrichtlinien zu, dann empfiehlt sich der automatische Start des DDM-Serverdienstes nach dem Start von TCP/IP.



STRTCPSVR

Natürlich lassen sich die TCP/IP-Einstellungen auch über den System i Navigator verwalten, und zwar über „Netzwerk ➜ Server ➜ TCP/IP“:

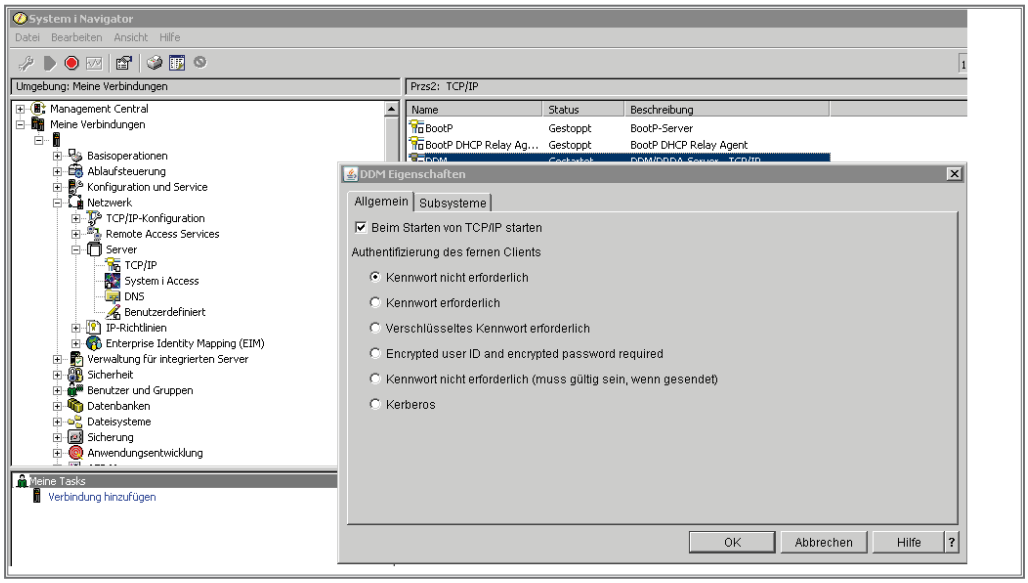


Server im System i Navigator

Bei den verschiedenen TCP/IP-Servern finden wir u. a. auch den Eintrag „DDM/DRDA Server“. Diesen Eintrag benötigen wir für unterschiedliche DDM-Funktionen.

Mit einem Klick auf den Eintrag gelangt man in den Konfigurationsbereich, in dem u. a. festgelegt werden kann, ob der Serverdienst automatisch mit dem Starten von TCP/IP hochgefahren werden soll. Außerdem kann hier die Authentifizierungsmethode hinterlegt werden. Sollte man DDM zusammen mit SSO und EIM nutzen wollen, dann muss hier der Eintrag „Kerberos“ selektiert werden.

5.4.7.3
Seite 4



DDM im System i Navigator

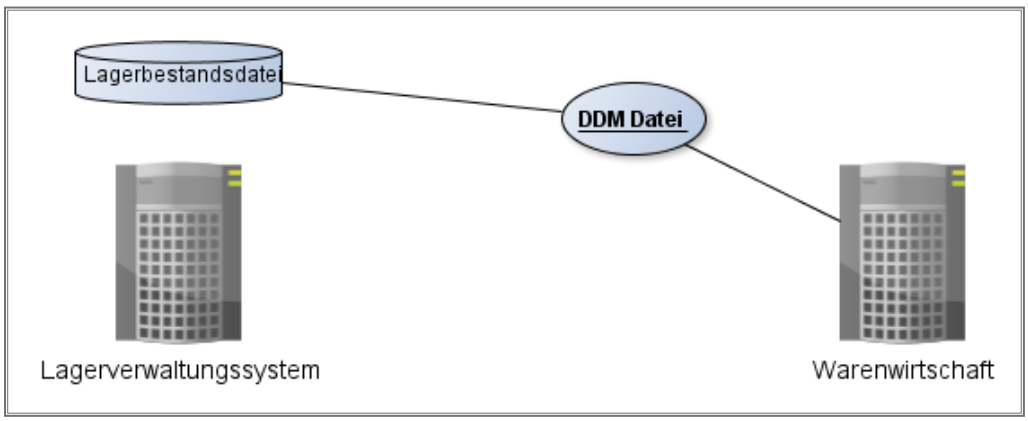
Arbeiten mit DDM-Dateien

DDM-Dateien erlauben den gesteuerten Zugriff auf ferne Daten. Dieser Zugriff erfolgt unter Berücksichtigung der DB2-Definitionen und der zugeordneten Berechtigungen.

Eine DDM-Datei kann folglich von einem zentralen System aus genutzt werden. Nehmen wir als Beispiel eine Zentrale mit verschiedenen Niederlassungen. Während in der Zentrale die deutschen Kunden betreut werden, wickeln in den Auslandsniederlassungen eigene Maschinen die dort anfallenden Prozesse ab. Lagerbestand und Umsatz werden täglich gemeldet. Dieses Melden kann in der Form durchgeführt werden, dass die Informationen vom zentralen System abgerufen werden. Dazu wird auf dem zentralen System eine DDM-Datei erzeugt. Diese DDM-Datei ist letztlich eine Zugriffsdefinition für die Lagerbestandsdatei auf der Maschine im Ausland. Sie ermöglicht das Verarbeiten der Daten – basierend auf den Berechtigungen, die im DB2-Umfeld vergeben werden können.

Programme oder sonstige Anwendungen, wie z. B. Query oder SQL unterscheiden nicht zwischen DDM-Dateien und lokalen Dateien, sofern die DDM-Dateien korrekt definiert wurden.

Auf der anderen Seite kann man natürlich auch Informationen des zentralen Systems der Hauptverwaltung in Deutschland für die Auslandsniederlassungen zur Verfügung stellen. In einem solchen Fall muss in den Auslandsniederlassungen eine DDM-Definition für die Datei auf dem Zentralsystem in Deutschland erstellt werden. Die folgende Grafik verdeutlicht diesen Prozess:



5.4.7.4

Seite 2

Erstellen einer DDM-Datei

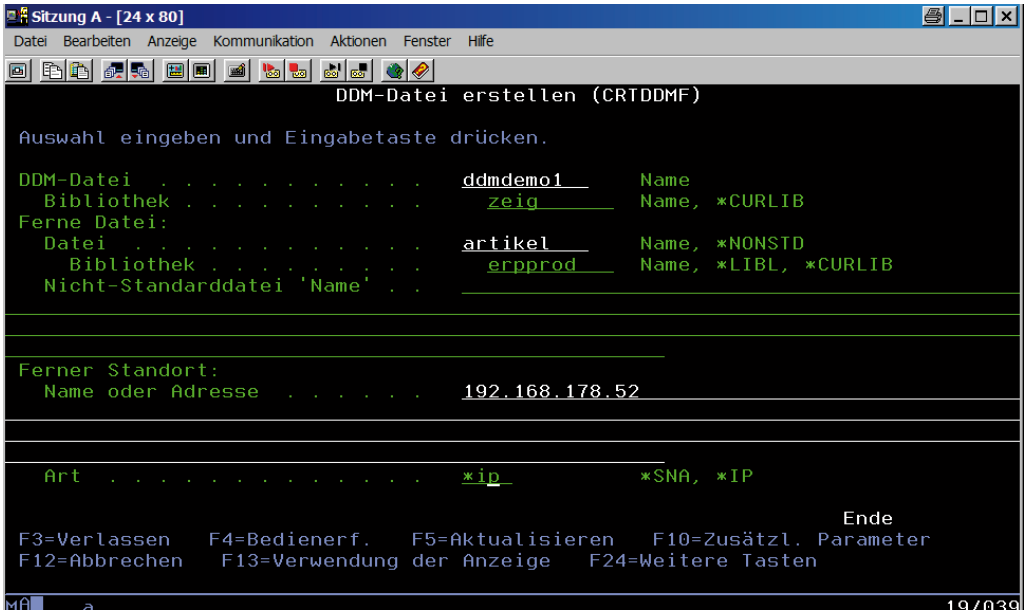
Nachdem die Voraussetzungen für den Einsatz von DDM geschaffen sind, können wir uns nun dem Bereitstellen von Dateien mit DDM zuwenden.

Das Erstellen einer DDM-Datei ist denkbar einfach – zumindest dann, wenn die Voraussetzungen erfüllt sind.

Für das Erstellen einer DDM-Datei verwendet man den Befehl CRTDDMF. Der Befehl verfügt über verschiedene Parameter, die anzugeben sind, damit die korrekten Informationen mit der DDM-Datei verlinkt werden.

Ein Beispiel für das Absetzen des CRTDDM-Befehls:

```
CRTDDMF FILE(ZEIG/DDMDEMO1) RMTFILE(ARTIKEL)
RMTLOCNAME(,192.168.178.52‘ *IP) TEXT(,DDM BEISPIEL ARTIKEL‘)
```



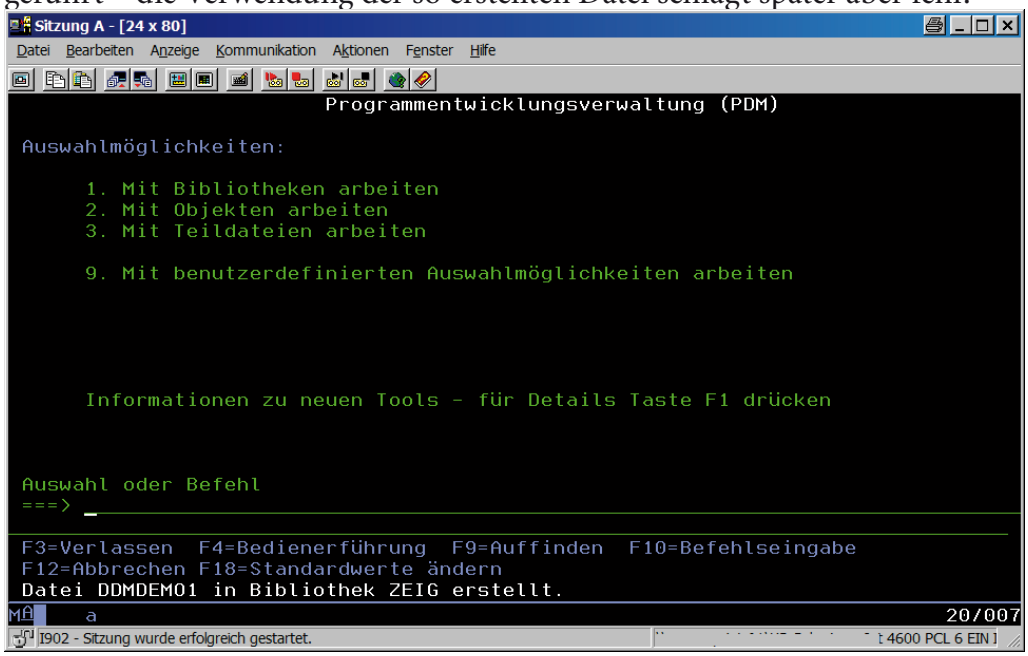
CRTDDMF

Alternativ zur IP-Adresse kann man auch den Systemnamen für den Parameter RMTLOCNAME verwenden.

Bei der Ausführung des Befehls CRTDDMF wird eine neue DDM-Datei erzeugt und in der angegebenen Bibliothek DDMLIB gespeichert. Während der Ausführung des Befehls CRTDDMF wird nicht auf die ferne Datei ARTIKEL zugegriffen.

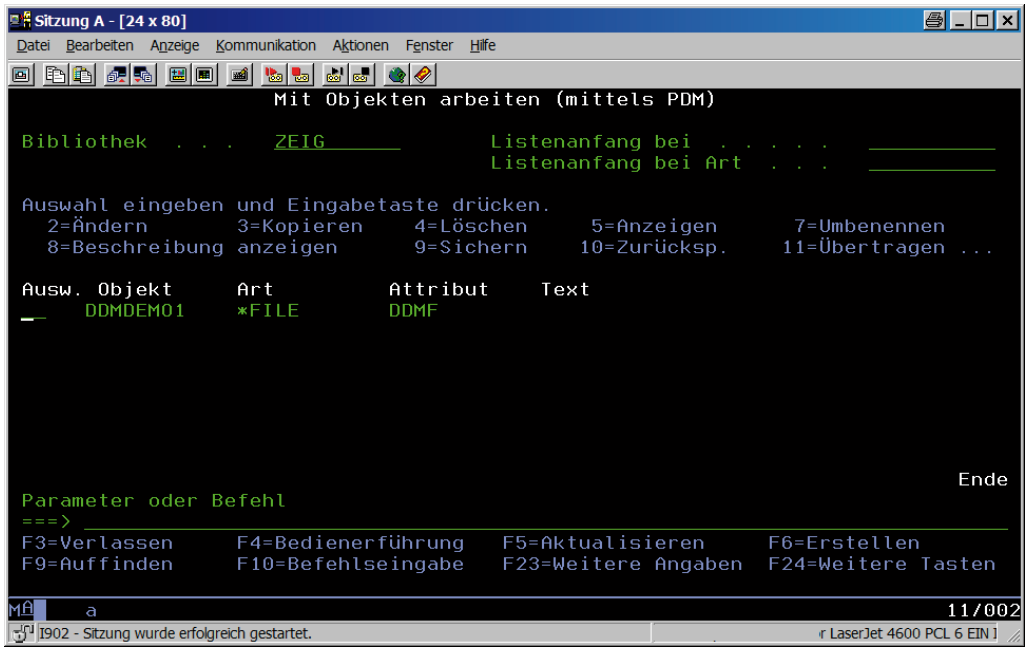
5.4.7.4.1
Seite 2

Bedauerlicherweise wird bei der Ausführung des Befehls CRTDDMF keine Prüfung durchgeführt, ob die ferne Datei tatsächlich in der angegebenen Bibliothek existiert. Stellen Sie deshalb sicher, dass die gemachten Angaben korrekt sind – anderenfalls wird zwar der Befehl CRTDDMF erfolgreich ausgeführt – die Verwendung der so erstellten Datei schlägt später aber fehl!



Nachricht nach Erstellen einer DDM-Datei

Nach dem Erstellen der DDM-Datei wird sie in der Liste der Objekte anzeigt. Auch diese Anzeige suggeriert das erfolgreiche Erstellen der DDM-Datei.



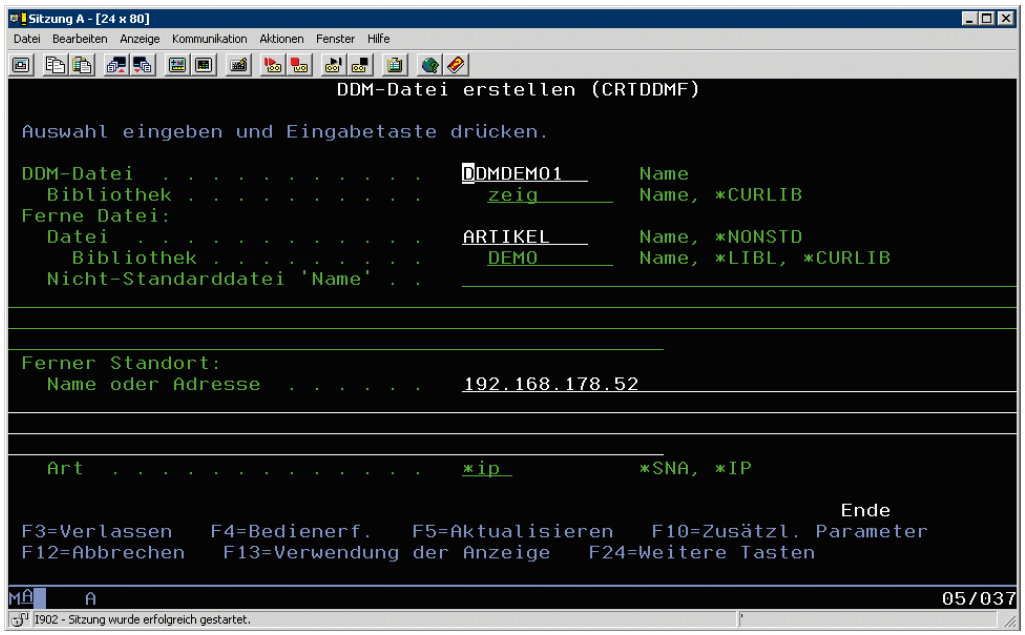
DDM-Datei

Aber auch dieses Objekt ist kein Garant dafür, dass die DDM-Datei eingesetzt werden kann.

Existiert das ferne Objekt nicht, so kann die DDM-Datei nicht wie gewünscht genutzt werden!

Deshalb ist es wichtig, nach dem Erstellen einer DDM-Datei deren Funktion zu testen. Ist ein Zugriff auf den Inhalt der fernen Datei (z. B. DSPPFM) möglich, dann war das Erstellen der DDM-Datei tatsächlich erfolgreich!

Die folgende Abbildung zeigt exemplarisch die Verwendung des Befehls CRTDDMF:



CRTDDMF

Ein weiterer Parameter, der zusammen mit CRTDDMF eingesetzt werden kann, regelt die gemeinsame Nutzung der DDM-Inhalte – unabhängig von möglichen Berechtigungen auf dem Ursprungssystem. Da diese Einstellung aus Sicherheitsgründen nicht zu empfehlen ist, habe ich sie in unserem Beispiel auch nicht verwendet. Sollten Sie nach reiflicher Überlegung zu dem Schluss kommen, dass die Informationen der DDM-Datei von anderen Anwendern genutzt werden dürfen, dann geben Sie im Parameter SHARE ein *YES ein.

Als Empfehlung gilt jedoch, diesen Parameter mit *NO zu verwenden, um ungebetene Zugriffe zu verhindern.

Der Parameter ACCMTH definiert die Zugriffsmethode. Eine Definition, die nur dann erforderlich ist (und auch nur dann unterstützt wird), wenn es sich bei dem fernen System um ein System i oder ein Vorgängermodell (bis hin zur /38) handelt.

5.4.7.4.1
Seite 4

Nachdem wir eine DDM-Datei erzeugt haben, lässt sie sich in weiten Bereich so einsetzen, als handele es sich um eine lokale Datei auf einem System IBM i.

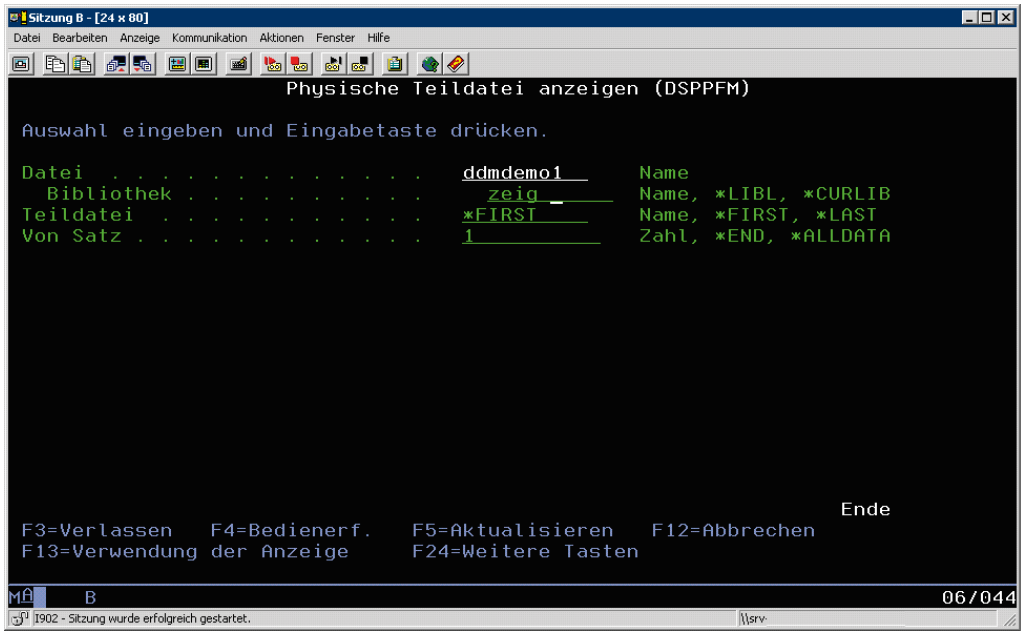
Nachfolgend zeige ich Ihnen einige der wesentlichen Funktionen, die bei der Arbeit mit einer DDM-Datei genutzt werden können. Diese Beispiele zeigen die Verwendung von DDM-Dateien nur exemplarisch und beziehen sich nicht auf alle Befehle oder Bereiche, die zusammen mit einer DDM-Datei genutzt werden können!

Wenn man eine DDM-Datei nutzen will, dann muss man sich zunächst einmal Gedanken darüber machen, welche Systeme bei ihrer Verwendung zum Einsatz kommen. Idealerweise handelt es sich bei den beiden Systemen um IBM i (oder Vorgängermodelle). Damit unterliegen die Anwendungen den IBM i-Standards in Bezug auf Sicherheit, Zugriffsrechte und Zeichenumsetzungen. In den nachfolgenden Beispielen gehen wir also davon aus, dass die beiden Systeme, die in der DDM-Umgebung zum Einsatz kommen, IBM i-Systeme sind.

Anzeigen des Inhalts einer fernen Datei

Je nach Anwendung und Anforderung kann es notwendig sein, den Inhalt einer fernen Datei (DDM-Datei) einzusehen. Dazu können wahlweise Programmfunktionen (z. B. CHAIN, READ in RPG) oder auch Standardfunktionen von IBM i verwendet werden, mit denen ein Datenzugriff möglich ist. Beispielsweise lässt sich der Inhalt der DDM-Datei mit dem Befehl DSPPFM einsehen:

```
DSPPFM FILE(ZEIG/DDMDEMO1)
```



DSPPFM

5.4.7.4.2

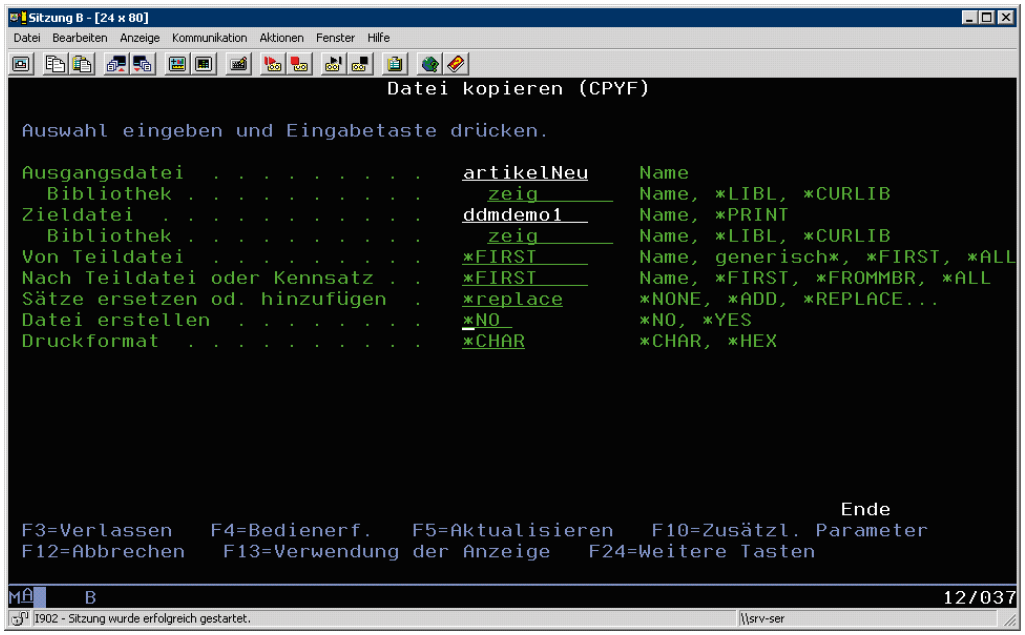
Seite 2

Kopieren einer lokalen Datei in eine ferne Datei

Eine lokale Datei kann nicht nur zum Verarbeiten der Inhalte einer fernen Datei verwendet werden, sondern auch um Daten von einem System auf ein anderes System zu übertragen. Zum Beispiel kann man den Inhalt einer lokalen Datei durchaus auch in die DDM-Datei einstellen und somit auf dem fernen System verfügbar machen.

Ein Beispiel für das Kopieren von Daten in eine DDM-Datei ist der Befehl CPYF:

```
CPYF FROMFILE(ZEIG/ARTIKELNEU) TOFILE(ZEIG/DDMDEMO1) MBROPT(*REPLACE)
```



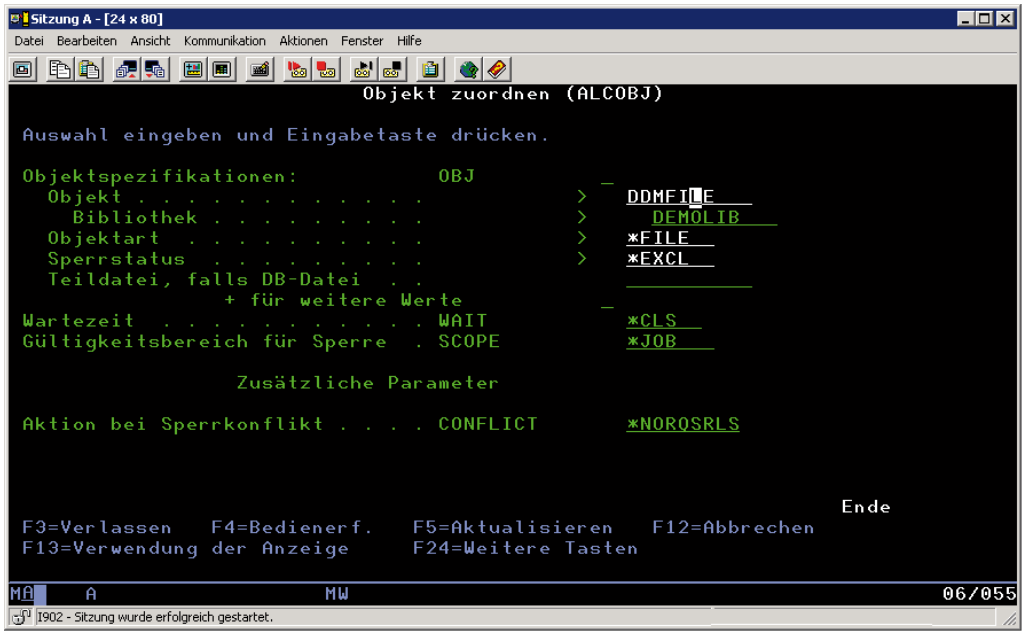
CPYF

Mit einer DDM-Datei können nahezu alle Datenbankaktionen durchgeführt werden, die wir auch mit einer lokalen Datei vornehmen können. Beispielsweise kann mit einer DDM-Datei der Befehl ALLOCATE genutzt werden, um einen Zugriff exklusiv durchzuführen.

```
ALCOBJ OBJ((DEMOLIB/DDMFILE *FILE *EXCL))
```

5.4.7.4.3

Seite 2



ALCOBJ

Mit diesem Befehl wird die DDM-Datei DDMFILE auf dem Quellsystem exklusiv für die weitere Verarbeitung zugeordnet.

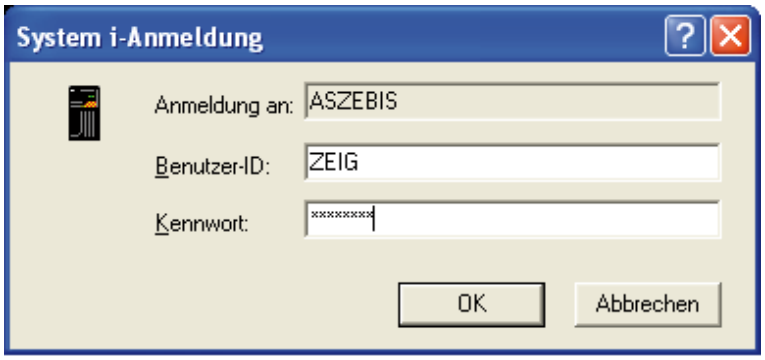
RMTCMD

Neben der Verwendung von DDM für den Zugriff auf Dateninhalte auf einem fernen System kann DDM auch dazu verwendet werden, ferne Befehle abzusetzen.

RMTCMD ist das Werkzeug, mit dem wir z. B. unter Windows einen IBM i-Befehl absetzen können. Voraussetzungen sind hier die korrekte DDM-Konfiguration sowie die Notwendigkeit, dass der Benutzer auf dem System i über die erforderlichen Berechtigungen für die Ausführung des Befehls verfügt. Auf PC-Seite sind die für DDM notwendigen Komponenten in System i Access enthalten und müssen auf dem PC installiert sein, auf dem der ferne Befehl ausgeführt werden soll.

Mittels RMTCMD sind wir in der Lage, eine Vielzahl der IBM i-Befehle von einem fremden System absetzen zu können. Dabei unterliegt die Syntax natürlich den Vorgaben des Systems IBM i.

Bei Eingabe des Befehls RMTCMD in einem DOS-Fenster wird geprüft, ob eine Verbindung zu dem in System i Access definierten System möglich ist. Die Verbindungsprüfung gibt zudem auch das Anmeldefenster aus, über das wir uns mittels eines Benutzerprofils sowie des dazugehörigen Kennworts zunächst für das System i authentifizieren müssen.



Login

Schauen wir uns den Einsatz von RMTCMD anhand eines einfachen Beispiels an:

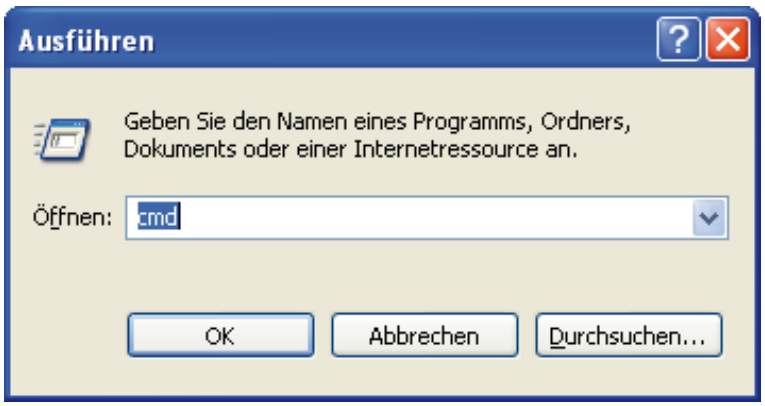
Wie bereits erwähnt, muss auf dem PC-System i Access (oder eine entsprechende Vorgängerversion) installiert sein.

Einen Befehl RMTCMD können wir z. B. über die DOS-Eingabe absetzen. Dazu gehen wir wie folgt vor:

1. Öffnen der DOS-Eingabe.
   - Auswahl „Start“ in Windows
   - Auswahl „Ausführen“

5.4.7.5
Seite 2

Eingabe von „cmd“ im Fensterbereich „Öffnen“.



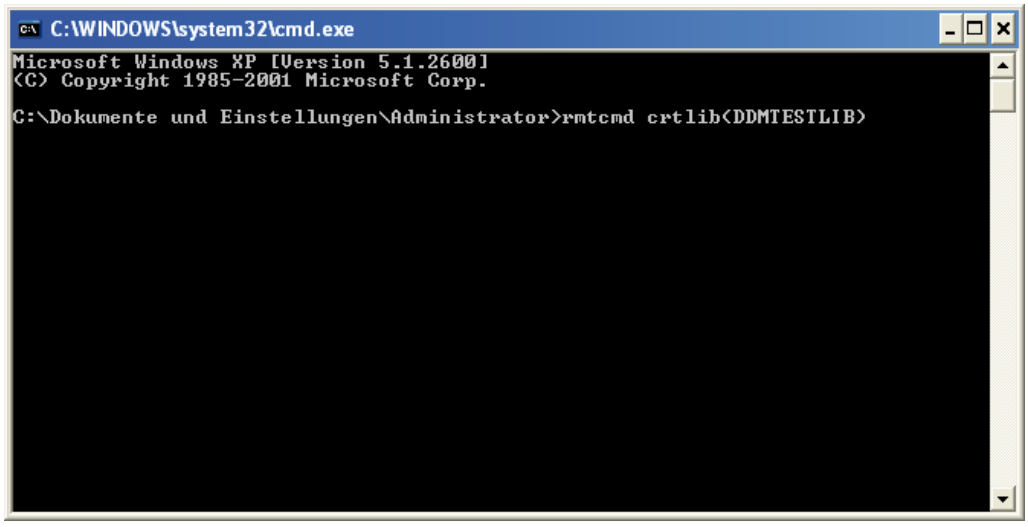
Ausführen

Anschließend öffnet sich das bekannte DOS-Eingabefenster.

In dem DOS-Eingabefenster können wir den Befehl RMTCMD – also den fernen Befehl des Systems IBM i – nun direkt eingeben. Nehmen wir hier als Beispiel den Befehl für das Erstellen einer Bibliothek: CRTLIB.

Der Befehl CRTLIB benötigt als Mindestangabe den Namen der zu erstellenden Bibliothek.

Die folgende Abbildung zeigt ein Beispiel für die Verwendung des Befehls RMTCMD:



RMTCMD

Sobald der Befehl abgeschickt wird, erscheint in Abhängigkeit der System i Access-Einstellungen das Login-Fenster oder der Befehl wird (nach vorheriger erfolgreicher Anmeldung am System) sofort an das System i übergeben.

6 Kapitel

Drucken

6

Seite 2

Drucken im Netzwerk

Die Zeiten, als das System lediglich IBM bzw. kompatible Drucker ansprechen konnte, sind glücklicherweise schon seit einiger Zeit vorbei. Dass aber das System i Druckdienste in einem heterogenen Netzwerk wahrnehmen kann und sich gerade durch ihre Solidität und Standfestigkeit in diesem Umfeld positiv absetzt, ist weitgehend unbekannt.

Bedingt durch die Heterogenität des Umfelds – zur Verwendung anstehende Drucker, Datenströme, Protokolle, Dienste – ergibt sich eine gewisse Unübersichtlichkeit bei der Realisation von Konfigurationsaufgaben.

Deshalb versuche ich in einem ersten Schritt, die an einer Konfiguration beteiligten Komponenten zu systematisieren. Dabei möchte ich folgende Gruppen voneinander absetzen:

- Welche Netzwerkdienste stellen Druckfunktionen bereit?
- Welche Datenströme kommen im i5/OS zur Anwendung?
- Welche Druckprotokolle können in und mit i5/OS verwendet werden?
- Welche Funktionen stellt i5/OS für heterogenes Netzwerkdrucken bereit?

10. Ergänzung 3/2007

6.1 Seite 2

Druckdienste für Netzwerkdrucken

LPR / LPD

LPR oder Line printer requester stellt einen TCP/IP-basierten Druckdienst dar, der sowohl im i5/OS als auch in anderen Betriebssystemen zur Verfügung steht. LPR (Line printer requester) ist ein Client, der auf einen Druckserver LPD (Line printer daemon) zugreift. Wenn also von einem Windows-PC auf das System i gespoolt werden soll, so benötigt man unter Windows einen LPR (Client) und im i5/OS einen LPD (Server). Das LPR/LPD-Protokoll wurde zu einer Zeit konstruiert, als direkt mit dem Rechner verbundene, zeilen-orientierte Drucker-Ströme von ASCII-Daten zu Papier brachten (z.B. mittels „Hammerschlägen" auf Buchstabenformen und Farbbändern als Schwärzungsmedium), das sie als zusammengefaltete Endlos-Zickzack-Schlange aus einem Karton zogen.

Positive Merkmale

- LPR und LPD ist für die meisten Betriebssysteme und / oder Drucker verfügbar.

Negative Merkmale

- Es unterstützt keine bi-direktionale Kommunikation.
- Es erlaubt keine Verschlüsselung von Druckdaten.
- Es erfordert keine Authentifizierung der Anwender.

Implementation

- i5/OS
  - Als Remote QUTQ
    (CRTOUTQ – Ausgabewarteschlange erstellen)
  - Als Druckeinheit (LAN-Device)
    (CRTDEVPRT – Drucker Einheitenbeschreibung erstellen)
  - Als CL-Befehl LPR/SNDTCPSPLF
    (C)
- Windows
  - Als Befehl LPR
  - Als Direkter TCP/IP-Port
  - Als Unix
- UNIX / Linux
  - Als Befehl
  - Als Dienst

6.1.1 Seite 2

IPP

Das IPP-Protokoll (Internet Print Protocol) ist quasi der Nachfolger von LPR und LPD. IPP beseitigt alle negativen Merkmale eines LPR/LPD-basierten Druckprozesses. Ist das IPP eine völlige Neukonstruktion? Nein. Mit dem IPP wird das Rad nicht neu erfunden; das IPP setzt auf ein seit Jahren bewährtes und robustes Internet-Protokoll auf: auf HTTP 1.1. Das Internet Printing Protocol ist als Datenübertragungsmethode eine Erweiterung von HTTP 1.1., jedoch angepasst an die speziellen Bedürfnisse von Druckdaten (mit den entsprechenden bi-direktionalen Fähigkeiten: Status-Abfragen und -Meldungen etc.).

Merkmale: Es erbt alle guten Eigenschaften von HTTP 1.1

- HTTP funktioniert rund um die Welt, mit und ohne Firewalls, in LAN und WAN; es ist ein bewährtes, stabiles, universell gesprochenes Protokoll und ein offener Standard, der auch später noch leicht erweitert werden kann.
- Aufgrund seiner Verwandtschaft kann das IPP alle an http „andockbaren" anderen Internet-Standards ebenfalls bei minimalen Anpassungen mitverwenden:
  - SSL3 und TLS für Verschlüsselungszwecke;
  - Die Methoden Basic und *Digest Authentification* (auch über *Certificates*) zur Authentifizierung von Benutzern;
- Zwecks Zugangskontrolle können alle auch bei Web-Servern üblichen Methoden angewendet werden:
  - nach *Absende-IP, -Subnet, -Host-Name* oder *-Domäne* differenziert;
- Mittels Verzeichnis-Diensten wie *LDAP (Lightweight Directory Access Protocol)* und *SLP (Service Location Protocol)* können Informationen über Drucker, ihre Eigenschaften, Seitenpreise, Nutzberechtigungen, Passwörter usw. im Netzwerk veröffentlicht und/oder überprüft werden.

Implementation

- i5/OS
  - IPP-Client: als Druckeinheit (LAN-Device) (CRTDEVPRT – Drucker Einheitenbeschreibung erstellen)
  - IPP-Server: als Tomcat Application Server
- Windows
  - IPP-Client: als http-port
  - IPP-Server: als Dienst des IIS (Internet Information Server)

6.1.2 Seite 2

- UNIX/Linux
  - IPP-Client: als Ipp-Port und http-port
  - IPP-Server: im Rahmen von CUPS (Common Unix Printing System)

System i-Netserver

Der System i-Netserver ist eine Komponente des i5/OS, der eine System i bzw. einen iSeries-Server in die Lage versetzt, ohne eine Middleware wie Client Access Dienste des i5/OS in das Microsoft Netzwerk zu stellen.

Ab i5/OS V5R1 können zwei Systeme System i-Ressourcen direkt von einem Microsoft Client aus in Anspruch genommen werden.

- Das Integrated File System (IFS)
- Die Drucker eines iSeries-Server

Allgemein können wir formulieren, dass der Netserver des i5/OS einem System i-Server im Microsoft Netzwerk zu einem File-, Druck- und Anmeldeserver des Microsoft Netzwerks macht.

Dabei wird die Berechtigungssteuerung aller Zugriffe auf die iSeries-Ressourcen über die Objekt und IFS-Rechte des i5/OS wahrgenommen.

Wie funktioniert der Netserver? Eigentlich müssten wir davon ausgehen, dass zwei Systeme auf denen TCP/IP implementiert ist, auch direkt auf der Dienste-Ebene miteinander „reden“ können. Wie wir aus eigener Erfahrung wissen, ist dieses jedoch nicht der Fall. Denn Sie können mit Sicherheit nicht ohne irgendeine zusätzliche Software vom System i auf den Verzeichnisbaum eines Windows-Systems zugreifen. Auch der Zugriff von einem PC auf einen System i-Drucker war bisher nur über eine Middleware wie Client Access möglich.

Implementation

- i5/OS
  - Als TCP/IP-Server-Job
- Windows
  - Mittels des cifs/SMB-Protokolls auf jedem Windows-System verfügbar
- UNIX / Linux
  - Als Samba SMB-Client

10. Ergänzung 3/2007

6.1.3 Seite 2

iSeries Access für Windows

Die Funktion des emulierten Druckens gehört zum Funktionsumfang des iSeries Access für Windows innerhalb des Programms PC5250. Emuliertes Drucken beschreibt ein Verfahren, bei dem ein lokaler PC-Drucker als System i-Systemdrucker verwendet wird.

Definieren wir also einen PC-Drucker als emulierten Drucker für das System i, so wird auf dem System i eine Ausgabewarteschlange für diesen Drucker eingerichtet, so dass ein i5/OS-Prozess diesen Drucker verwenden kann.

Wesentlich für eine erfolgreiche Installation ist der Sachverhalt, dass an dieser Stelle zwei Druckprozesse miteinander kommunizieren müssen. Einerseits der System i-Prozess, der das gesamte Spooling steuert und leitet, andererseits der Windows-Druckprozess, der dann das physische Drucken übernimmt. Daraus ergibt sich die Voraussetzung, dass auf dem Windows-PC bereits ein Drucker installiert wurde, der dann mit dem System i-Druckprozess verbunden werden muss.

- i5/OS
  - Client: als native virtuelle Druckeinheit (*VRT)
- Windows
  - Server: im Rahmen der 5250-Emulation des iSeries Access für Windows

Die Funktion des emulierten Druckens ist „alter“ Dienst, der mehrere eher unangenehme Eigenschaften hat:

- iSeries Access Installation auf Windows-PC notwendig
- Permanent verfügbare und konsistente Prozesse sind schwer zu realisieren
- Verwendet 5250-Protokoll (proprietär!)

Tipp

Nutzen Sie diese Funktion nur dann, wenn alle anderen Möglichkeiten ausgeschöpft sind und kein funktionsfähiges Ergebnis hervorgebracht wurde.

6.1.4 Seite 2

Datenströme

Produziert ein System i Druckausgaben, so kann über die Anwahl des Ausgabedatenstroms, die Qualität der Ausgabe definiert werden. Welchen Datenstrom eine Ausgabe letztendlich erzeugt, wird in der Druckdatei, die ein Druckprozess verwendet, definiert.

```
                    Druckerdatei ändern (CHGPRTF)

 Auswahl eingeben und Eingabetaste drücken.

 Datei  . . . . . . . . . . . . . . FILE       >   QSYSPRT
   Bibliothek . . . . . . . . . .                    *LIBL
 Einheit:                          DEV
   Drucker  . . . . . . . . . . .                    *JOB
 Druckereinheitentyp  . . . . . . DEVTYPE           *SCS
 Seitengröße:                      PAGESIZE
   Länge--Zeilen pro Seite  . . .                    72
   Breite--Stellen pro Zeile  . .                    132
   Maßangabe  . . . . . . . . . .                    *ROWCOL
 Zeilen pro Zoll  . . . . . . . . LPI              6
 Zeichen pro Zoll . . . . . . . . CPI              10
 Überlaufzeile  . . . . . . . . . OVRFLW           66
 Satzformat-Aktualitätsprüfung  . LVLCHK           *NO
 Text 'Beschreibung'  . . . . . . TEXT             'System non-described printer
file'
```

Datenstrom einstellen

Möglich sind hier folgende Werte:

- *SCS
- *IPDS
- *LINE
- *AFPDSLINE
- *USERASCII
- *AFPDS

Von diesen möglichen Werten werde ich jedoch nur die im Folgenden aufgeführten beschreiben.

6.2 Seite 2

SCS

Der SCS (SNA character string) ist ein einfacher Datenstrom, der nur Zeichendaten an einen Drucker übermittelt. Der Datenstrom sendet satzweise Daten, denen ein 1 Byte-großer hexadezimaler Kontrollcode vorangestellt ist, mittels derer der Drucker Steuerungsinformationen erhält.

Hier einige der Kontrollcodes:

Hex	Description
03	ASCII transparent data
05	Horizontal tab
0B	Vertical tab
0C	Forms feed
0D	Carriage return
1A	Unit backspace
15	New line
16	Backspace
35	Transparent
2843	Set attribute
2BC1	Set horizontal format
2BC2	Set vertical format
2BC6	Set line density
2BD2	Set presentation page size
2BD4	Begin underscore
2BFE	Load alternate character

SCS – Datenstrom: Kontrollcodes

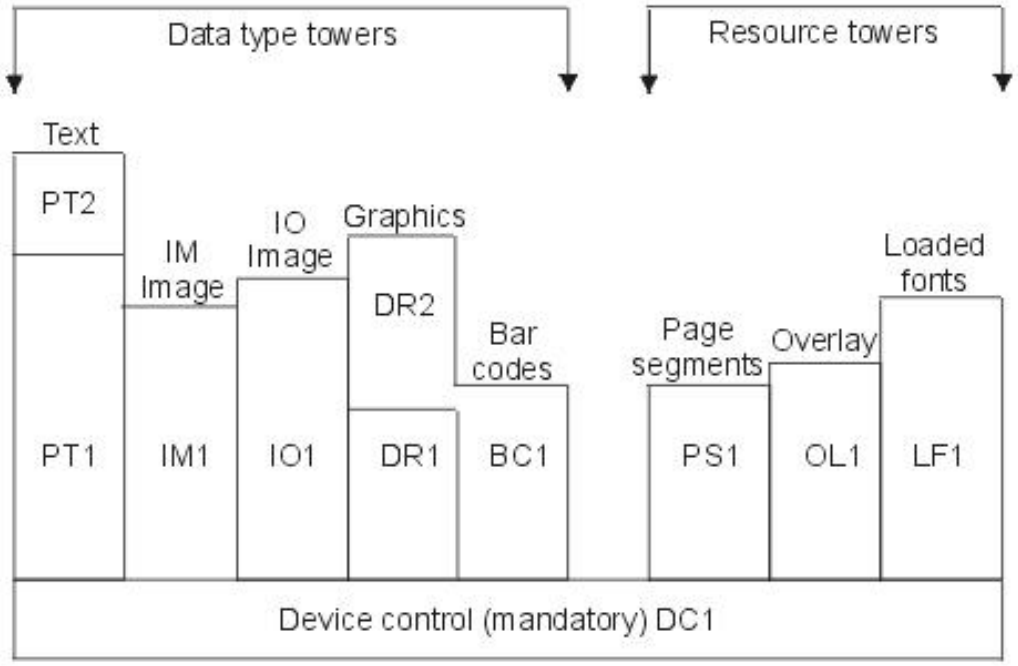
(Quelle: IBM iSeries Information Center, Version 5 Release 4:
http://publib.boulder.ibm.com/infocenter/iseries/v5r4/index.jsp)

6.2.1 Seite 2

IPDS

IPDS™ (Intelligent Printer Data Stream™) sieht IBM als seinen Host-zu-Drucker-Datenstrom. Der Datenstrom kann eine unbegrenzte Anzahl unterschiedlicher Datentypen transportieren: quantitativ hochwertige Textinformationen, Rasterbilder, Vektorgrafiken und Barcodes.

Der IPDS-Datenstrom besteht aus einer Reihe von Funktionsbereichen, die jeweils über eigene Befehlssätze verfügen.



IPDS-Datenstrom: Architektur

(Quelle: IBM iSeries Information Center, Version 5 Release 4: http://publib.boulder.ibm.com/infocenter/iseries/v5r4/index.jsp)

Damit ein Drucker IPDS-Fähigkeit für sich beanspruchen kann, muss er über folgende Funktionen verfügen:

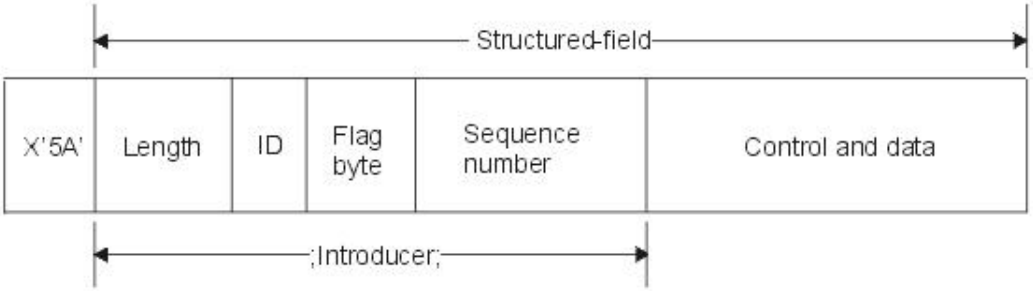
- Es müssen alle Funktionen der Device Control Schicht (DC1) unterstützt werden.
- Es muss wenigstens ein Subset des Data Type Towers implementiert sein.
- Er muss alle Befehle einer implementierten Funktion bzw. eines implementierten Subsets unterstützen.

Fazit: Arbeiten Sie mit IPDS-Druckern, so ist IPDS nicht gleich IPDS, je nachdem welchen Druckertyp von welchem Hersteller Sie verwenden.

6.2.2 Seite 2

AFPDS

Der AFPDS (Advanced Function Presentation Data Stream) Datenstrom ist die Anwendungsschnittstelle zum AFP (Advanced Function Presentation™). Der AFPDS ist ein strukturierter Datenstrom. Strukturierte Felder werden verwendet, um Seitenformat-Informationen, Zeilenformat-Informationen und eine Mischung aus Seiten- und Zeilenformat-Informationen zu transportieren.



AFPDS – Datenstrom: Architektur

(Quelle: IBM iSeries Information Center, Version 5 Release 4: http://publib.boulder.ibm.com/infocenter/iseries/v5r4/index.jsp)

Ein strukturiertes Feld besteht aus einer sich selbst identifizierenden Zeichenfolge, die Daten bzw. Parameterwerte beinhaltet. Es wird eingeleitet durch einen Introducer, der aus Längeninformation, ID, einem Flag und einer Folgenummer besteht. Daraufhin folgt ein Parameterbereich, der Steuerungsinformationen bzw. die zu druckenden Daten enthält.

6.2.3 Seite 2

USERASCII

ASCII-Datenströme werden insofern vom System i unterstützt, als dass die Netzwerk-Druckfunktionen SCS- und AFPDS-Datenströme in ASCII-Datenströme wie PCL, PDF, Postscript übersetzen können.

ASCII-Datenströme werden durch den Begriff USERASCII dargestellt. Spoolt i5/OS einen Datenstrom der Gruppe USERASCII, so wird er lediglich durch den Spoolprozess hindurchgeleitet. Er ist für das i5/OS nicht einsehbar, so dass nicht alle Drucksteuerungsbefehle des i5/OS eingesetzt werden können (DSPSPLF, CPYSPLF, etc.).

6.2.4 Seite 2

Druckerprotokolle

In Abhängigkeit vom verwendeten Drucker, werden zur internen Steuerung des Druckers Protokolle verwendet, die als proprietär, quasi-druckerübergreifend bzw. druckerübergreifend anzusehen sind. Im Rahmen des Buches werde ich mich derzeit auf folgende zwei Protokolle beschränken:

- PJL
- SNMP

6.3 Seite 2

PJL

PJL (Printer Job Language) ist von HP designtes quasi Drucker-übergreifendes Steuerungsprotokoll. Mittlerweile ist es in den meisten Druckern implementiert, die als HP-Clones anzusehen sind (Lexmark, Kyocera, etc, pp.).

Arbeiten Sie auf dem System i mit Ausgabewarteschlangen (OUTQ), die Druckausgaben sowohl vom nativen i5/OS-Bereich als auch von Windows-, Linux-, Unix-Bereich verarbeiten, so zeigt PCL unter Umständen sein hässliches Gesicht.

Erlangt z.B. anfänglich ein nativer i5/OS-Druckprozess eine so genannte Portsperre zum Drucken, so wird die Portsperre erst dann aufgehoben, wenn die Ausgabewarteschlange komplett geleert – also alle Ausgaben gedruckt – wurden. Dieser Sachverhalt kann zu dem Problem führen, dass eine quasi parallel zum i5/OS-Durckprozess begonnene Windows-Ausgabe mit der Meldung abgewiesen wird, dass für den Windows-Druck keine Portsperre erhalten werden kann.

Umgekehrt (Windows zuerst, dann i5/OS) kann natürlich das gleiche Problem entstehen.

6.3.1 Seite 2

SNMP

SNMP (Simple Network Management Protocol) ist ein Drucker-übergreifendes Steuerungsprotokoll. Analog zu PJL arbeitet SNMP.

Unterschiede ergeben sich im Bereich der Portsperren. SNMP bewirkt, dass ein Ausgabeprozess eine Portsperre erhält, die nach Beendigung des Prozesses aufgegeben wird.

SNMP ist als Steuerungsprotokoll immer dann zu bevorzugen, wenn unterschiedliche Systeme ihre Druckausgaben über den i5/OS-Spoolprozess abwickeln.

6.3.2 Seite 2

Achtung!

Welches Drucksteuerungsprotokoll Sie im Einzelfall verwenden können, hängt von Ihrem Drucker bzw. Ihren Druckern ab. Das beste Protokoll für die angestrebte Lösung steht möglicherweise nicht bereit, weil Ihr Drucker das jeweilige Protokoll nicht unterstützt. Sollten Sie also vor einer Kaufentscheidung stehen, so sollten Sie diesen Punkt berücksichtigen.

6.3.3 Seite 2

Der LAN-Drucker – CRTDEVPRT(1)

Remote OUTQ ist ein sehr flexibles Mittel, um Drucker unterschiedlichster Hersteller, Printserver und Betriebsysteme sowie deren Drucker anzusprechen. Eine Remote-Ausgabewarteschlange hat aber auch Defizite:

- keine Fachsteuerung bei der Ausgabe
- kein Wiederaufsetzen innerhalb der Spool-Datei

Diese Defizite können durch eine andere Art von Netzwerkdrucker-Konfiguration auf der iSeries ausgeglichen werden. Seit OS/400 V4R4 ist es möglich, eine so genannte LAN-Device aufzubauen, die als eine Art virtueller Drucker anzusehen ist, der auf einen Netzwerkdrucker zeigt.

Die LAN Device unterstützt die Funktionen, die der Remote OUTQ fehlen:

- Fachsteuerung
- Kopien drucken
- Wiederaufsetzen bei abgebrochenen Druckvorgängen etc.

Da stellt sich doch die Frage, warum man überhaupt noch mit Remote-Ausgabewarteschlangen arbeiten soll? Der Grund ist folgender: LAN-Devices lassen das Drucken nur über Geräte zu, die technisch mit bestimmten Merkmalen ausgestattet sind.

Die Drucker müssen folgende Merkmale aufweisen:

- PCL5/6 fähig
- PJL-Monitor muss unterstützt werden (bis OS/400 V4R4).
- Alternativ ab OS/400 V4R5 muss SNMP-Protokoll unterstützt werden.
- Drucker müssen über eine TCP/IP-Portsteuerung verfügen.

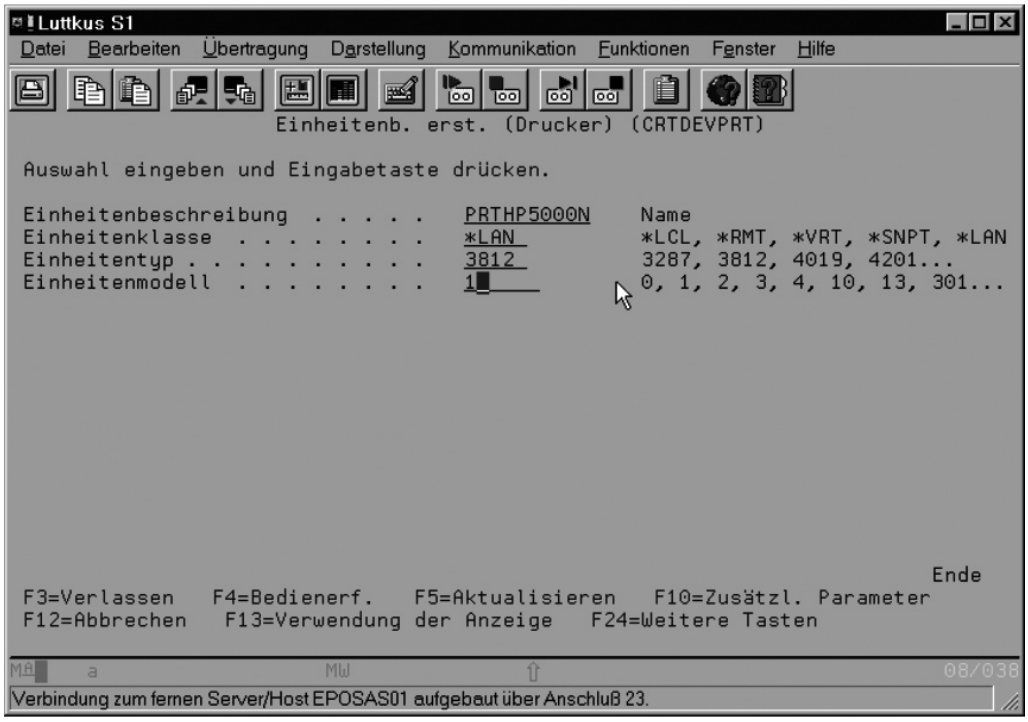
Wenn Sie Drucker haben, die diese Merkmale nicht aufweisen, so können Sie unter TCP/IP keine LAN-Device verwenden.

In diesem Beispiel konfiguriere ich eine LAN-Device für einen HP5000N, der die geforderten Eckwerte aufweist.

Die Konfiguration beginnt mit der Eingabe des Befehls CRTDEVPRT. Einige der Parameter sind identisch mit denen der Remote-Ausgabewarteschlange, deshalb werde ich nur noch die Parameter erläutern, die für eine LAN-Device signifikant sind.

Grundwerk 10/2004

6.4
Seite 2



CRTDEBPRT 1

Folgende Parameter müssen gefüllt werden:

Einheitenbeschreibung PRTHP5000N

Dieser Name ist individuell zu ergeben.

Einheitenklasse . . . . . *LAN

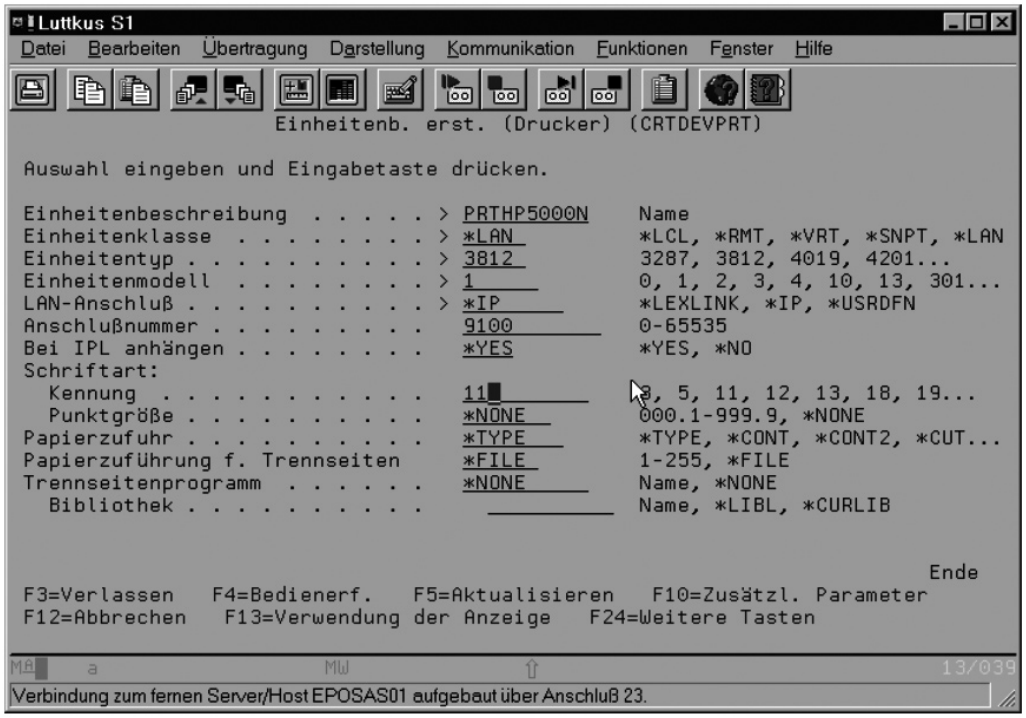
Dieser Parameter sorgt für die Erstellung der LAN-Device

Einheitentyp 3812

Dieser Druckertyp ist zwingend notwendig.

Einheitenmodell 1

Auch das Modell muss so angegeben werden.



CRTDEVPRT 2

Notwendige Parameter:

LAN-Anschluss . . . . . *IP

Seit OS400 V4R4 kann das TCP/IP-Protokoll angegeben werden.

Anschlussnummer 9100

Die TCP/IP-Portnummer ist nicht standardisiert und muss für den jeweiligen Drucker ermittelt werden. Viele Hersteller, die Drucker mit PCL-Emulationen anbieten, unterstützen mittlerweile den Port 9100. Am Ende dieses Kapitels habe ich Ihnen eine Liste der bekannten Druckerport-Nummern beigefügt.

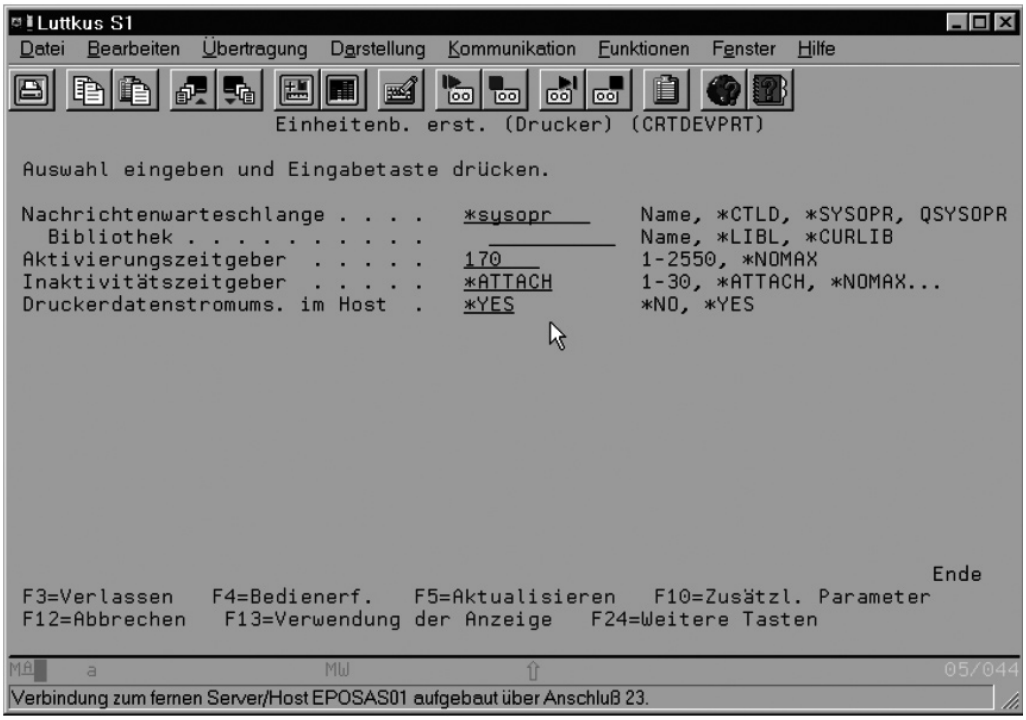
Schriftart

Kennung 11

Die Standardschriftart, die verwendet wird, falls aus der Spool keine Schriftart angewählt wird.

Punktgröße . . . . . *NONE

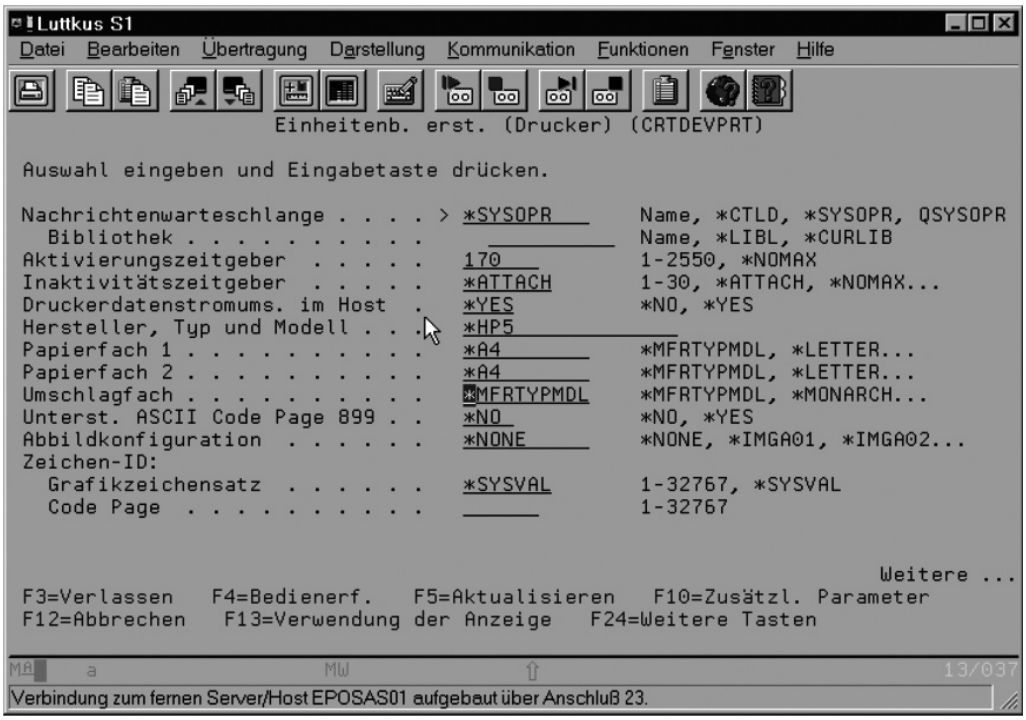
6.4
Seite 4



CRTDEVPRT 3

Nicht notwendig, aber sinnvoll:

Nachrichtenwarteschlange . . . . . *sysopr

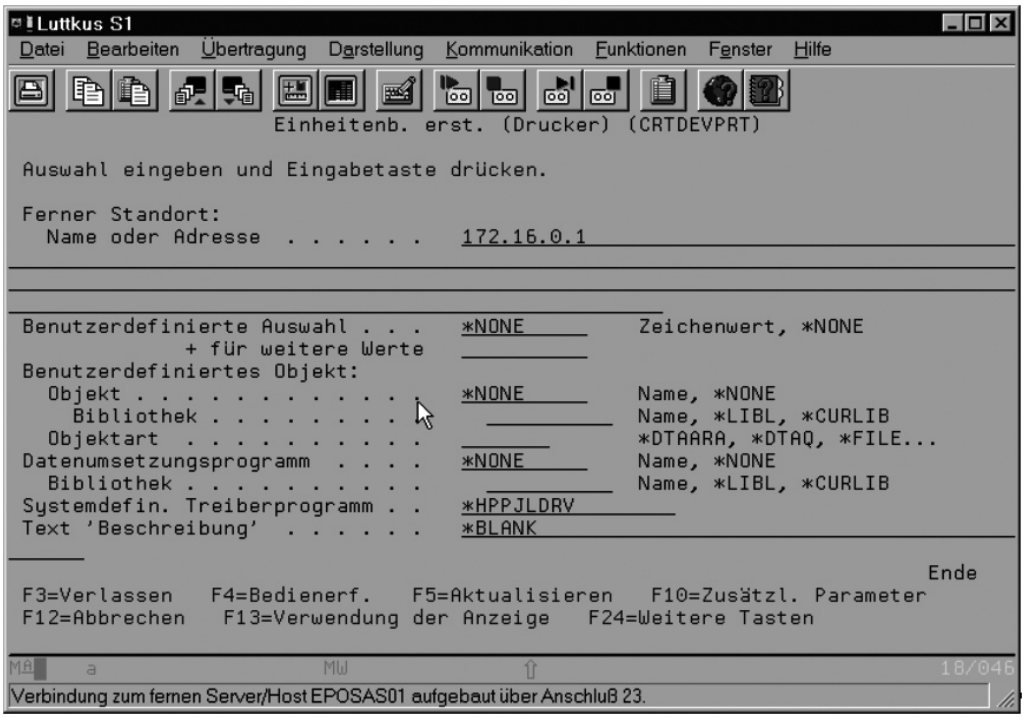


CRTDEVPRT 4

Wichtige Parameter:

Druckerdatenstromums. im Host	*YES
Hersteller, Typ und Modell	*HP5
Papierfach 1	*A4
Papierfach 2	*A4
Umschlagfach	*MFRTYPMDL

6.4
Seite 6



CRTDEVPRT 5

Wichtige und notwendige Parameter:

Ferner Standort:

Name oder Adresse 172.16.0.1

Systemdefin. Treiberprogramm . . . . . *HPPJLDRV

Die Liste der bekannten TCP/IP-Druckports:

Network Interface or System Type	PJL Port Number
Axis LAN Adapters (external) print servers	Note: The Reverse Telnet port(s) must be set to some value since by default they are disabled. It is recommended that you set RTEL_PR1 = 9100 for parallel port 1 and RTEL_PR2 = 9101 for parallel port 2, but any values can be used.
Castelle LANPress 2+1MP print servers	Parallel Port 1 = 3010 Parallel Port 2 = 3020 Serial Port = No PJL Support
Dell printers	9100
Epson Interface Card	Unknown at this Time
Extended Systems print servers	Unknown at this Time Note: According to Extended Systems, their print servers support PJL, but the port numbers are not known at this time.
Extend Net print servers	9100
Gestetner printers	same as Osicom JetXPrint (internal): 9100 Note: The Base Port Number in the NIC card may need to be set to 9099. See the section entitled Information Concerning Osicom Print Servers and Ricoh, Savin, Lanier and Gestetner Printers below.
HP JetDirect (Internal) print servers	9100
HP JetDirect (External) print servers (3 ports - 1 I/P address)	9100 for Parallel Port 1 9101 for Parallel Port 2 9102 for Parallel Port 3
HP LaserJet 4 or later (with internal HP JetDirect print servers)	9100
HP WP110 Wireless print servers (External)	No PJL Support
HP Wireless print servers (External)	Unknown at this Time
IBM 3116 (with internal Lexmark Marknet print servers)	9100
IBM 3130 printers IBM 3160 printers IBM Infoprint 60 printers	No PJL Support
IBM 6400 Network Connection	No PJL Support
IBM Infoprint Color 8 printers (IBM 4308)	9100

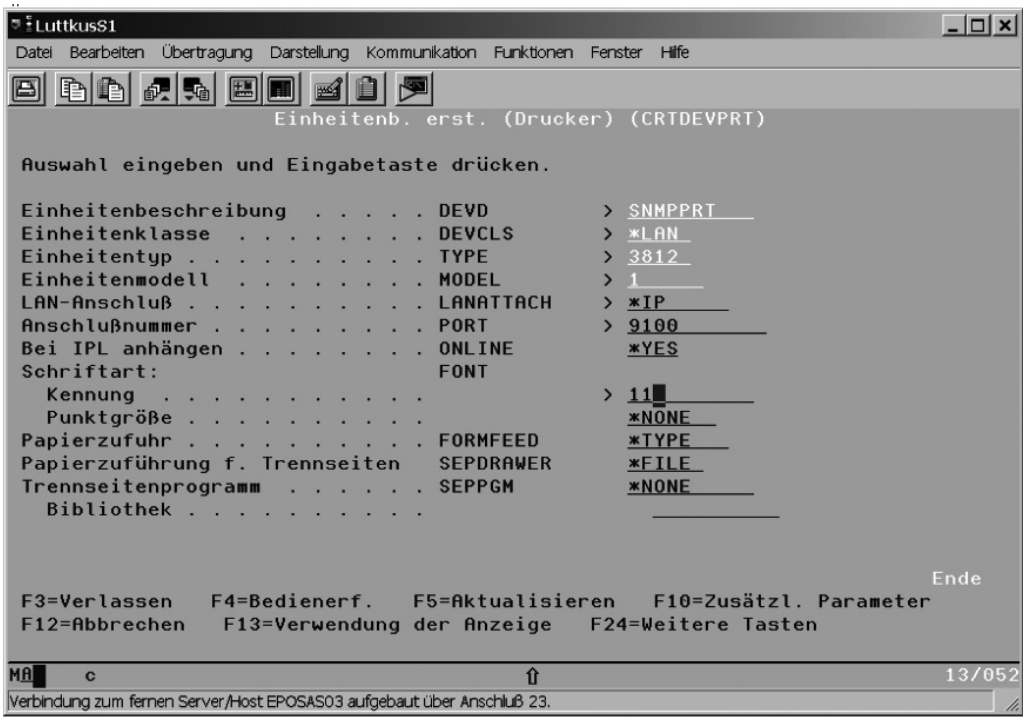
IBM Infoprint 20, 32 or 40 printers (IBM 4320, 4332 or 4340)	2501
IBM Infoprint 21 printers (IBM 4322)	No PJL Support
IBM Infoprint 1100 Series: IBM Infoprint 1116 printers IBM Infoprint 1120 printers IBM Infoprint 1125 printers IBM Infoprint 1130 printers IBM Infoprint 1140 printers IBM Infoprint 1145 printers	9100
IBM Infoprint 1200 Series: IBM Infoprint 1222 printers IBM Infoprint 1226 printers	9100
IBM Infoprint 1300 Series: IBM Infoprint 1312 printers IBM Infoprint 1332 printers IBM Infoprint 1352 printers IBM Infoprint 1372 printers	9100
IBM Infoprint Color Series: IBM Infoprint Color 1220 printers IBM Infoprint Color 1228 printers IBM Infoprint Color 1357 printers	9100
IBM Integrated Network Option print servers (IBM 4039, 4049, 3112, 3116)	9100
IBM iSeries 400 (printer attached to another iSeries 400)	No PJL Support
IBM Network Print Server (NPS) print servers	2501
IBM Network Printer 12, 17 and 24 (IBM 4312, 4317, and 4324)	2501
IBM RS/6000 (printer attached to RS/6000 system)	No PJL Support
i-data EasyCom, LinkCom or PlusCom (external) print servers	9100 for Parallel Port 1 9101 for Parallel Port 2
i-data LinkCom or PlusCom (internal) print servers	9100
Intel NetportExpress 10/100 (single-port) print servers	9100 Note: See the section entitled Information Concerning Intel Netport Print Servers below.

Intel NetportExpress 10/100 (3-port) print servers	9100 for Parallel Port 1 9101 for Parallel Port 2 9102 for Parallel Port 3 Note: See the section entitled Information Concerning Intel Netport Print Servers below.
Intel NetportExpress PRO and PRO/100 print servers	No PJL Support
Intel NetportExpress XL and EL print servers	No PJL Support
JETNet print servers	Unknown at this Time
Kyocera IB20 (internal) print servers	9100
Lanier printers	same as Osicom JetXPrint (internal): 9100 Note: The Base Port Number in the NIC card may need to be set to 9099. See the section entitled Information Concerning Osicom Print Servers and Ricoh, Savin, Lanier and Gestetner Printers below.
Lantronix print servers	30xx, xx represents the port number. These servers can have 1 to 32 ports (for example, port 1 is 3001)
Lexmark MarkNet XL (Internal) print servers	9100
Lexmark MarkNet XLe (External) print servers Lexmark MarkNet Pro (External) print servers Lexmark Marknet X2000 (External) print servers	Parallel Port 1 = 9100 Parallel Port 2 = 9101 Serial Port = No PJL Support
Lexmark Optra (with internal Lexmark Marknet print server)	9100
Linksys print servers	No PJL Support
Minolta DiAlta printers	10001
Mita 3010 printers	10001
Net Print NIC card (internal) print servers	same as Osicom JetXPrint (internal): 9100 Note: The Base Port Number in the NIC card may need to be set to 9099. See the section entitled Information Concerning Osicom Print Servers and Ricoh, Savin, Lanier and Gestetner Printers below.

Osicom JetXPrint NIC card (internal) print servers	9100 Note: The Base Port Number in the NIC card may need to be set to 9099. See the section entitled Information Concerning Osicom Print Servers and Ricoh, Savin, Lanier and Gestetner Printers below.
Osicom NetPrint print server (external) print servers	9100 for Parallel Port 1 9101 for Parallel Port 2 9102 for Parallel Port 3 Note: The Base Port Number in the NIC card may need to be set to 9099. See the section entitled Information Concerning Osicom Print Servers and Ricoh, Savin, Lanier and Gestetner Printers below.
Ricoh printers	same as Osicom JetXPrint (internal): 9100 Note: The Base Port Number in the NIC card may need to be set to 9099. See the section entitled Information Concerning Osicom Print Servers and Ricoh, Savin, Lanier and Gestetner Printers below.
Savin printers	same as Osicom JetXPrint (internal): 9100 Note: The Base Port Number in the NIC card may need to be set to 9099. See the section entitled Information Concerning Osicom Print Servers and Ricoh, Savin, Lanier and Gestetner Printers below.
Toshiba eStudio printers and digital copiers	9100 or possibly 9101
TRENDnet TE310 (external) print servers	No PJL Support and No Bi-Directional Support
Unix Systems (Printer Attached to Unix System)	No PJL Support
Windows 95, 98 or NT (Printer Attached to a PC)	No PJL Support
Xerox Document Centre printers	9100
Xerox DocuPrint printers (if Set Up to Support PCL/PJL, HP Mode)	2000 or 9100

Seit OS/400 V4R5 können LAN-Devices für Drucker konfiguriert werden, die nicht die PJL-Treiber unterstützen. In diesem Fall müssen die Drucker alternativ das TCP/IP-Diensteprotokoll SNMP unterstützen.

Die Konfiguration einer PJL LAN Device ist weitgehend identisch mit der einer SNMP LAN Device. Deshalb gebe ich Ihnen hier nur eine kurze Auflistung der entsprechenden Parameter an.



CRTDEVPRT 6

Nicht notwendig, aber sinnvoll:

Nachrichtenwarteschlange . . . . . *sysopr

Default ist notwendig:

Druckerdatenstromums. im Host . . . . . *YES

LAN-Anschluss . . . . . > *IP

Anschlussnummer 9100

Grundwerk 10/2004

Liste der SNMP Ports:

Network Interface or System Type	SNMP Port Number
Axis LAN Adapters (external) print servers	9900 Note: See the section entitled Information Concerning IBM NPS and Axis Print Servers below.
Canon ImageRunner Printers with internal Fiery LAN Adapter	same as EFI Fiery print server: 9100
Castelle LANPress 2+1MP print servers	Unknown at this Time
EFI Fiery print servers	9100
Epson Multiprotocol 10/100 Ethernet, Type B print servers	9100 Note: See the section entitled Information Concerning Epson Print Servers below.
Extended Systems print servers	No SNMP Support
HP JetDirect Card (Internal) print servers	9100
HP JetDirect Server (External) print servers (3 ports - 1 I/P address)	9100 for Parallel Port 1 9101 for Parallel Port 2 9102 for Parallel Port 3
HP LaserJet with internal HP JetDirect print server	9100
HP WP110 Wireless print servers (External)	9100
HP Wireless print server (External)	Unknown at this Time
IBM 3116 with internal Lexmark Marknet print server	Unknown at this Time
IBM 3130 printers IBM 3160 printers IBM Infoprint 60 printers	No SNMP Support
IBM 4247 with IBM NPS or Axis print server IBM 6400 with IBM NPS or Axis print server	9900 Note: See the section entitled Information Concerning IBM NPS and Axis Print Servers below.
IBM 6400 with Enhanced Ethernet Adapter printer server	9100 (or possibly 5001) Note: Check the ETHERNET PARAMETERS MENU in the printer hardware and use the value specified in the ASCII DATA PORT NUMBER parameter. If this is set to 0, then set it to 9100 and save the Custom Set.
IBM iSeries 400 (printer attached to another iSeries 400)	No SNMP Support
IBM Infoprint Color 8 (IBM 4308) printers	9100

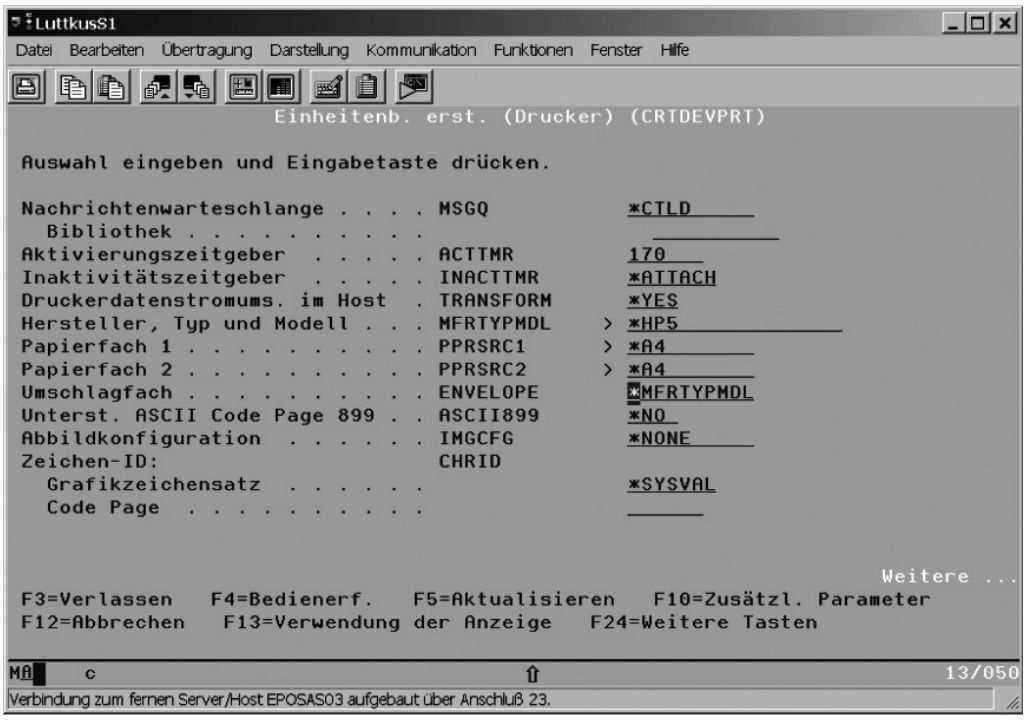
IBM Infoprint 12 (IBM 4912) printers	9100
IBM Infoprint 21 (IBM 4322) printers	2501 (or possibly 9100)
IBM Infoprint 70 (IBM 2770) printers	2501
IBM Infoprint 20 (IBM 4320) printers IBM Infoprint 32 (IBM 4332) printers IBM Infoprint 40 (IBM 4340) printers	2501
IBM Infoprint 1100 Series: IBM Infoprint 1116 printers IBM Infoprint 1120 printers IBM Infoprint 1125 printers IBM Infoprint 1130 printers IBM Infoprint 1140 printers IBM Infoprint 1145 printers	9100
IBM Infoprint 1200 Series: IBM Infoprint 1222 printers IBM Infoprint 1226 printers	9100
IBM Infoprint 1300 Series: IBM Infoprint 1312 printers IBM Infoprint 1332 printers IBM Infoprint 1352 printers IBM Infoprint 1372 printers	9100
IBM Infoprint Color Series: IBM Infoprint Color 1220 printers IBM Infoprint Color 1228 printers IBM Infoprint Color 1357 printers	9100
IBM 4039 printers with IBM Integrated Network Option IBM 4049 printers with IBM Integrated Network Option IBM 3112 printers with IBM Integrated Network Option IBM 3116 with IBM Integrated Network Option	Unknown at this Time
IBM Network Print Server (NPS) printer server	9900 Note: See the section entitled Information Concerning IBM NPS and Axis Print Servers below.
IBM Network Printer 12 (IBM 4312) printers IBM Network Printer 17 (IBM 4317) printers IBM Network Printer 24 (IBM 4324) printers	2501
IBM RS/6000 (printer attached to RS/6000 system)	No SNMP Support

i-data PlusCom or LinkCom (internal) print servers	Unknown at this Time
i-data EasyCom, PlusCom or LinkCom (external) print servers	Unknown at this Time
I-O 5430e print servers	No SNMP Support
Intel NetportExpress 10/100 (single-port) print servers	Unknown at this Time
Intel NetportExpress 10/100 (3-port) print servers	Unknown at this Time
Intel NetportExpress PRO and PRO/100 print servers	Unknown at this Time
Intel NetportExpress XL and EL print servers	Unknown at this Time
JETNet print servers	Unknown at this Time
Kyocera IB20 (internal) print servers	9100
Lanier copier/printers	9100
Lantronix print servers	Unknown at this Time
Lexmark MarkNet XL (Internal)	9100
Lexmark MarkNet XLe or PRO (External)	9100 for Parallel Port 1 No SNMP Support for Parallel Ports 2 and 3
Lexmark Optra with internal Lexmark Marknet print server	9100
Linksys print servers	Unknown at this Time
Minolta DiAlta printers	10001
Osicom JetXPrint (Internal) print servers	Unknown at this Time
Osicom NetPrint (External) print servers	Unknown at this Time
Ricoh Internal NIC Card (Osicom JetXPrint)	9100
Toshiba e-Studio digital copiers	9100
Unix Systems (Printer Attached to Unix System)	No SNMP Support
Windows 95, 98 or NT (Printer Attached to a PC)	No SNMP Support
Xerox Document Centre 2xx printers	No SNMP Support
Xerox Document Centre 3xx printers	No SNMP Support
Xerox Document Centre 4xx printers	9100: Note: See the section entitled Information Concerning Xerox Document Centre 4xx Printers below.
Xerox DocuPrint printers (if Set Up to Support PCL/PJL, HP Mode)	Unknown at this Time

Schriftart:

Kennung 11

Punktgröße . . . . . *NONE



CRTDEVPRT 7

Druckerdatenstromums. im Host . . . . . *YES

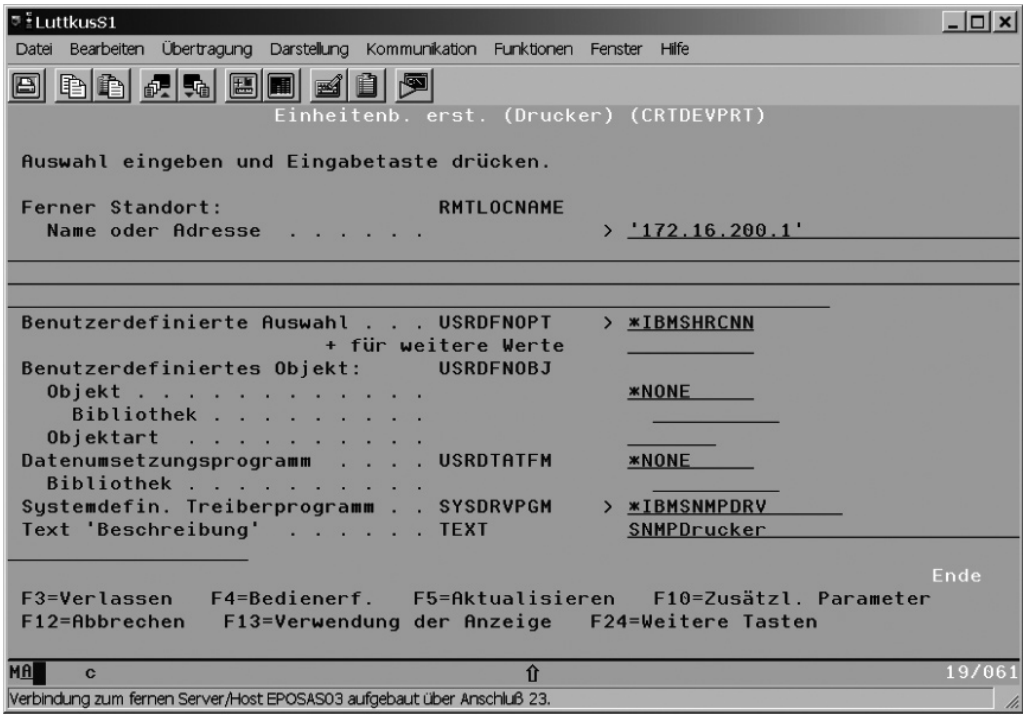
Hersteller, Typ und Modell *HP5

Papierfach 1 . . . . . *A4

Papierfach 2 . . . . . *A4

Umschlagfach . . . . . *MFRTYPMDL

6.4
Seite 16



CRTDEVPRT 8

Ferner Standort:

Name oder Adresse	172.16.200.1
Benutzerdefinierte Auswahl	*IBMSHRCNN
Systemdefin. Treiberprogramm	*IBMSNMPDRV

Netzwerk-Druckprotokolle

LPR / LPD

Im Hinblick auf das soeben in Abschnitt 6.3 formulierte, haben die beteiligten Netzwerkprotokolle bzw. -Dienste erhebliche Auswirkungen. Arbeiten Sie mit LPR / LPD, so können Sie folgende Druckerprotokolle nutzen:

- PJL
- SNMP

Bedenken Sie hierbei die Einschränkungen dieser Protokolle (siehe 6.1.1). Ob Sie überhaupt eine Wahl haben, hängt dann von Ihrem verwendeten Drucker ab (siehe 6.3.3).

6.4.1 Seite 2

IPP

Das IPP-Protokoll stellt Funktionen bereit, die sowohl dem Bereich des Netzwerkprotokolls als auch dem Bereich des Druckerprotokolls zugeordnet werden müssen.

IPP übernimmt einerseits den Transport des Datenstroms vom Client zum Server, andererseits die Ansteuerung des Druckers – Aufgaben, die sonst PJL und SNMP zuzuordnen sind.

Bezüglich der Portsperren verhält sich IPP wie SNMP.

In die Zukunft geblickt ist anzunehmen, dass IPP das Protokoll ist, welches in der Lage ist, die anderen in Abschnitt 6.1 genannten zu ersetzen.

10. Ergänzung 3/2007

6.4.5 Seite 2

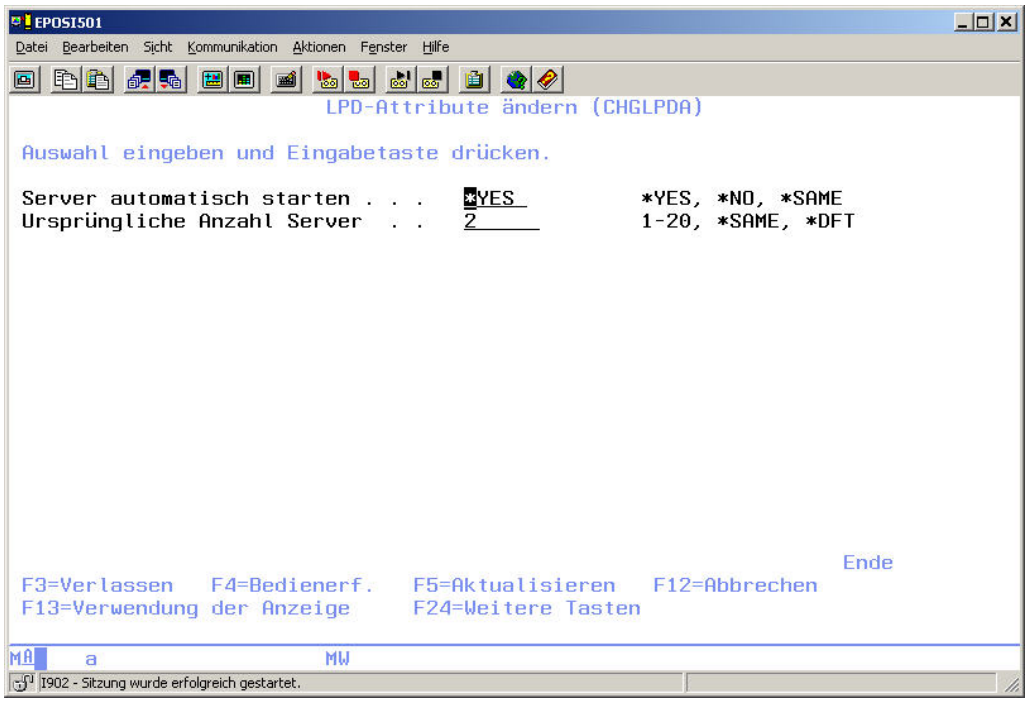
Druckobjekte im i5/OS

Druckdienste werden im i5/OS als Objekte / Jobs dargestellt. Die Frage die sich dabei stellt ist, welches Objekt repräsentiert welchen Dienst?

6.5 Seite 2

Der System i LPD-Server

Der LPD-Server ist im i5/OS als TCP/IP-Netzwerkdienst implementiert. Die Eigenschaften des LPD werden mit dem Befehl CHGLPDA – LPD Attribute ändern – eingestellt.



CHGLPDA – LDP-Attribute ändern

Dieser Dienst ist für mich „nicht konfigurierbar". Wenn Sie sich die Anzahl und die Qualität der Parameter in o.a. Hardcopy ansehen, verstehen Sie was ich meine.

Ich möchte hier lediglich den Parameter „Ursprüngliche Anzahl Server" erläutern.

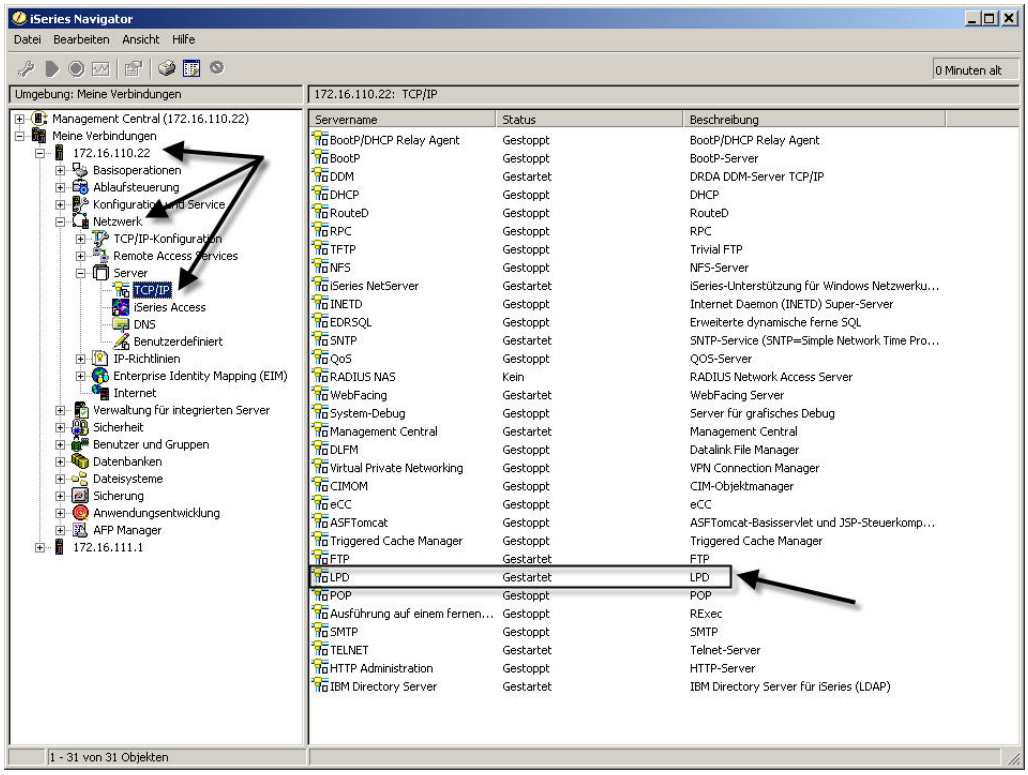
Für jeden Clientprozess, der die iSeries als Printserver verwendet, wird ein so genannter Verbindungsjob benötigt. Dieser Parameter besagt, dass zwei dieser Verbindungsjobs vorgestartet werden und auf die Verbindungsaufnahme eines Druckclients warten. Dadurch wird der Zeitaufwand, der für eine Verbindungsaufnahme notwenig ist, minimiert.

Tipp

Sollte der LPD-Server unter Last geraten, so sollten Sie die Anzahl der Serverjobs erhöhen. Der Wertebereich ist zwischen 1 und 20 anzugeben. Der Wert 1 ist für mich nicht Praxis-relevant, da dann immer nur ein Client zur Zeit abgearbeitet wird.

6.5.1
Seite 2

Im iSeries Navigator sehen Sie den Status des LPD-Servers im Pfad Verbindung → Netzwerk → Server → TCP/IP.

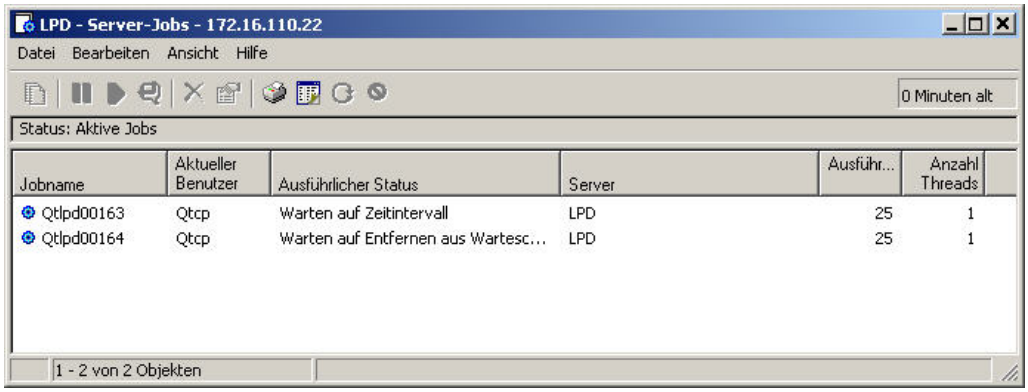


Der LPD-Dienst

Mittels Rechtsclick auf LPD erhalten Sie das Kontextmenü.

LPD-Kontextmenü

Wenn Sie auf „Server-Jobs“ klicken, so sehen Sie die beiden vorgestarteten Verbindungsjobs, die im Subsystem QSYSWRK ausgeführt werden.



Die LPD-Verbindungsjobs

Der LPD wird im Rahmen der Systemdefaults mit dem Start der TCP/IP-Schnittstelle hochgefahren.

Ist der Dienst gestartet, so sind grundsätzlich alle Drucker der iSeries mit Hilfe eines LPRs ansprechbar. Diese Aussage ist noch dahingehend zu relativieren, dass sich nicht nur physikalische Drucker ansprechen lassen, sondern auch Ausgabewarteschlangen. Folgende Druckobjekte werden durch den iSeries LPD verfügbar gemacht:

- Systemdrucker der iSeries
- Ausgabewarteschlange ohne Drucker
- Remote Ausgabewarteschlangen für das Netzwerkdrucken
- LAN-Druckeinheiten für das Netzwerkdrucken

Achtung! Sollte der LPD-Server trotz eingetragenem Autostart nicht automatisch starten, oder sollte nach einem Start nur ein Serverjob zu sehen sein, obwohl Sie zwei angegeben haben, so müssen Sie wissen, dass der LPD-Serverjob unter der Berechtigung des Benutzers QTMPLPD ausgeführt wird. Ist dieser Benutzer deaktiviert (*DISABLED), so kommt es zu den dargestellten Fehlern.

Der System i-LPR-Client

LPR ist im i5/OS mehrfach implementiert:

- Als Remote OUTQ
- Als Befehl LPR/SNDTCPSPLF
- Als LAN-Device

6.5.1 Seite 4

Remote-Ausgabewarteschlangen konfigurieren

Lassen Sie uns folgendes Beispiel durchkonfigurieren:

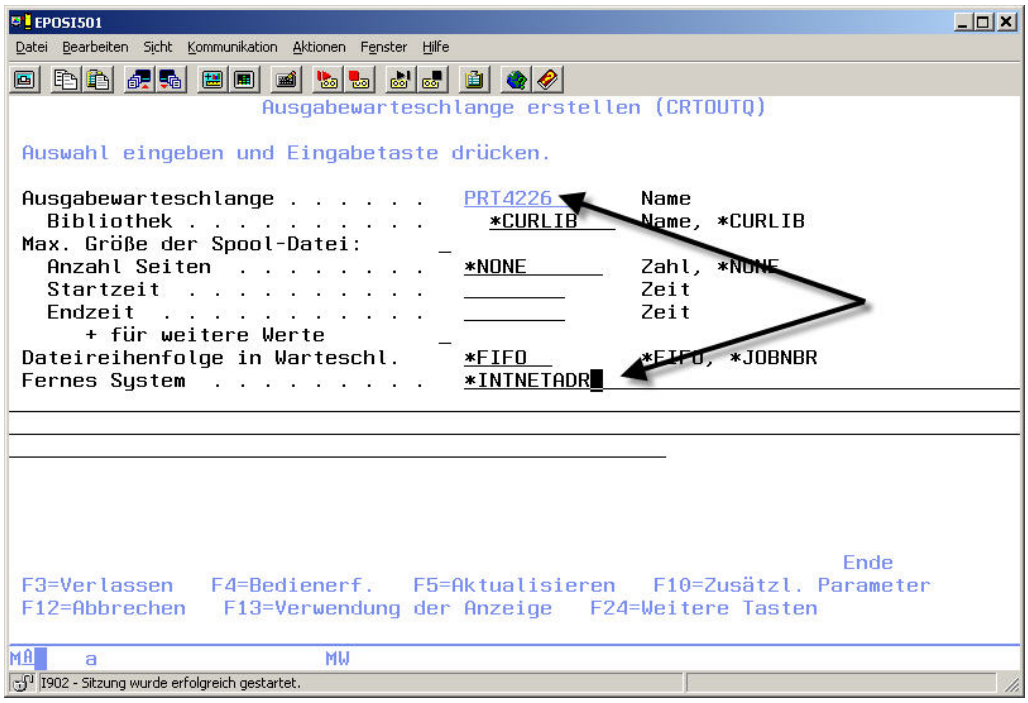
Wir haben einen IBM 4226-Drucker, der mit einer parallelen Schnittstelle ausgestattet ist.

Dieser Drucker soll direkt über das Netzwerk angeschlossen werden.

Dafür benötigen wir einen Printserver. Der Printserver ist ein HP Jetdirect – ein Gerät, das drei LPD-Ports hat.

Um diese Aufgabe zu realisieren, erstellen wir eine Remote Ausgabewarteschlange, mit deren Hilfe wir die IP-Adresse des Printservers adressieren können. Der Printserver leitet die Ausgabe dann an den Drucker weiter.

Ich gebe deshalb den Befehl „CRTOUTQ Ausgabewarteschlange erstellen“ ein.

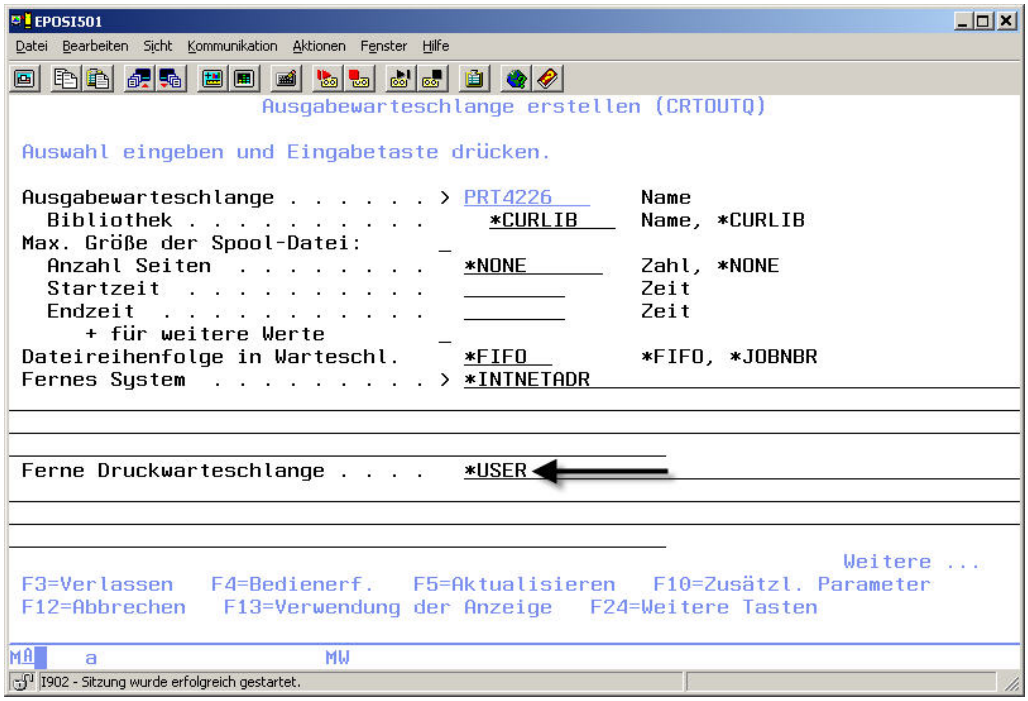


CRTOUTQ-Parameter

- OUTQ: Der Name der zu erstellenden Ausgabewarteschlange ist frei wählbar: hier PRT42226.
- RMTSYS: Sonderwert *INTNETADR.. Dieses Parameterwerk besagt, dass die Remote-Ausgabewarteschlange den Printserver und damit den Drucker mittels einer IP-Adresse ansprechen wird. Tragen Sie hier nicht den Hostnamen oder die IP-Adresse des Printservers ein!

6.5.1.1
Seite 2

Ich drücke auf die Datenfreigabetaste und dann erscheint der Parameter „Ferne Druckwarteschlange“.



Weitere Parameter 1

CRTOUTQ - Parameter

- RMTPRTQ: Name der entfernten Druckwarteschlange, die den Drucker des entfernten Systems ansteuert.
- Der hier einzutragende Wert ist abhängig vom angesprochenen Zielsystem.
- iSeries unter i5/OS: Name der lokalen Ausgabewarteschlange des entfernten Systems.
- Windows Netzwerk: Name der Print Queue des Windows Systems Achtung: Groß- und Kleinschreibung beachten!
- Printserver: Die Namensgebung ist abhängig vom Hersteller des Printservers. Für ein HP Jetdirect-Gerät sind die Namen RAW1, RAW2 und RAW3 für die jeweiligen LPT-Ports vorgesehen.

Im Folgenden finden Sie eine Liste von Remote Print Queue Werten für die unterschiedlichen Printserver-Hersteller:

Network Interface or System Type	Remote Printer Queue (RMTPRTQ) Value
Argecy ACC 6xe Coax/Twinax to Ethernet print server	‚LBLPRT1'.
Avery Dennison Printers	‚lp'
Axis LAN Adapters (external print servers)	‚prl'
Axis 1620 for Canon PCL printers	‚prl'

Network Interface or System Type	Remote Printer Queue (RMTPRTQ) Value
Axis 5400+	‚pr1'
Axis 5600+ Axis 5800+ Mobile Axis 5900	‚pr1'
Axis 540+ Axis 560 Axis 5810 Print Plus	‚pr1'
Axis Office Basic	‚LPT1'
Axis PrintPoint 140 BJC	Unknown at this Time
Barr Systems	‚BARRTCP1'
Canon GP Series copier/printers	‚print'
Canon imageRUNNER printers with internal Fiery LAN Adapter	‚print' or ‚PRINT'
Canon imageRUNNER printers with imagePASS-M1 print controller	‚pt1' or possibly ‚lpt1'
Canon imageCLASS printers	‚XJDIRECT'
Castelle LANpress	‚L1' for parallel port 1 ‚L2' for parallel port 2
Castelle LANpress 3	‚P1' for parallel port 1 ‚P2' for parallel port 2 ‚P3' for parallel port 3
Cisco PPSX1 Print Server	‚L1'
CompUSA Parallel 1 Port Print Server	‚LPT1'
D-Link DP-101 D-Link DP-300 D-Link DP-301 Plus	‚LPT1_TEXT'
D-Link DP-311P	Unknown at this Time
D-Link DP-G321 wireless print server	‚PS-xxxxxx-U1' (for USB Port 1) ‚PS-xxxxxx-U2' (for USB Port 2) ‚PS-xxxxxx-P3' (for LPT Port) where ‚xxxxxx' represents the last 6 digits of the print server MAC address.
Datamax printers	‚text'
Decision Data	‚d1prn'
Dell Laser printers	‚/prt0'
Digi Portserver	No LPD Support
Digi Portserver II	‚raw_1' for parallel port 1 ‚raw_2' for parallel port 2 ... ‚raw_8' for parallel port 8 ‚raw_9' for parallel port 9 Parallel ports 10 - 16 are unknown at this time (possibly either ‚raw_10 or ‚raw_A' through ‚raw_16' or ‚raw_F').
Edimax Print Servers	‚lpt1' or ‚LPT1'
EFI Fiery Print Servers	‚print' or ‚PRINT'
Emulex Print Servers	PASS or TEXT Can be set to accept any value if the Emulex adapter is set to Passthru.
Epson Interface Card	‚LPB' ‚LPBF'(for formfeed after splf)
Extend Net 1005	‚L1'

6.5.1.1
Seite 4

Network Interface or System Type	Remote Printer Queue (RMTPRTQ) Value
Extended Systems	‚portLF1' or ‚anydata1' for parallel port 1 ‚portLF2' or ‚anydata2' for parallel port 2 ‚portLF3' or ‚anydata3' for parallel port 3 Note: If ‚portLF1', ‚portLF2' or ‚portLF3' causes a blank page before and after each spooled file, try using ‚anydata1', ‚anydata2' or ‚anydata3' instead.
GENICOM 5000 series with PS1000 series NIC	‚PASSTHRU'
GENICOM 5000 series with EasyCom Express NIC	‚LPDPRT1'
GENICOM 5000 Series With Emulex PS1000 Network Interface Card (Internal)	‚GENICOM' (all uppercase)
GENICOM printers with Microplex M205, M206 or M305 NIC	‚d1prn'
GENICOM printers with Microplex M202 or M202+ external print server	‚d1prn1' for parallel port 1 ‚d2prn2' for parallel port 2 ‚d3com1' for serial port 1 ‚d4com2' for serial port 2
GENICOM printers with Printronix PrintNet	‚d1prn'
GENICOM printers with MPS100	‚MPS_nnnnnn_P1' where „nnnnnn" represents the last six digits of the MAC address
Gestetner printers	‚PORT1'
Hawking Technology print server (parallel models)	‚PS_Port-1', ‚LP1' or ‚lpt1'
Hawking Technology print server (USB models)	‚PS_Port-1', ‚LP1' or ‚lpt1'
Hitachi DDS 32/40 printers	‚pcl5'
Hitachi DDS 50/62 printers with internal Fiery LAN Adapter	‚print' or ‚PRINT'
HP DeskJet 5850 wireless printer HP DeskJet 5850w wireless printer	No LPR Support
HP JetDirect Card (Internal)	‚raw' (formatted output) ‚text' (unformatted text) Note: Some newer HP JetDirect cards, like the 170X and 300X, may instead use ‚RAW' for formatted output or ‚TEXT' for unformatted text.
HP JetDirect Server (External) (3 Ports - 1 I/P Address)	‚raw1' for parallel port 1 (formatted output) ‚text1' for parallel port 1 (unformatted text) ‚raw2' for parallel port 2 (formatted output) ‚text2' for parallel port 2 (unformatted text) ‚raw3' for parallel port 3 (formatted output) ‚text3' for parallel port 3 (unformatted text) Note: Some newer HP JetDirect servers may instead use ‚RAW1', ‚RAW2' or ‚RAW3' for formatted output or ‚TEXT1', ‚TEXT2' or ‚TEXT3' for unformatted text.
HP LaserJet printers (with internal HP JetDirect print server)	‚raw' (formatted output) ‚text' (unformatted text) Note: Some newer HP JetDirect cards, like the 170X and 300X, may instead use ‚RAW' for formatted output or ‚TEXT' for unformatted text.
HP LineJet Printer with internal HP LineJet print server	‚d1prn'
HP WP110 Wireless print servers (External)	‚raw' (formatted output) ‚text' (unformatted text)
HP Wireless print server (External)	‚raw' (formatted output) ‚text' (unformatted text)
IBM 3130	‚afccu2'
IBM 4247	‚pr1'
IBM 4400 Thermal printerwith IBM Network print server (NPS) or Axis LAN adapter	‚pr1'

Network Interface or System Type	Remote Printer Queue (RMTPRTQ) Value
IBM 4400 Thermal printer with Network Connection (Internal)	‚d1prn' for single port
IBM 6400 Network Connection (Internal)	‚d1prn' for single port
IBM 6400 Network Connection (External)	‚pr1' for parallel port 1 ‚pr2' for parallel port 2 .... ‚pr8' for parallel port 8
IBM i5/OS or OS/400 with CNNTYPE(*IP) and DESTTYPE(*OS400)	output queue name (for example. ‚PRT01') output queue library / output queue name (for example, ‚QUSRSYS/PRT01')
IBM i5/OS or OS/400 with CNNTYPE(*SNA) and DESTTYPE(*OS400)	output queue name (for example. ‚PRT01') output queue library / output queue name (for example, ‚QUSRSYS/PRT01') *USER *SYSTEM
IBM Infoprint 20 (4320) IBM Infoprint 32 (4332) IBM Infoprint 40 (4340) IBM Infoprint 70 (2770)	‚PASS' (formatted output) ‚TEXT' (unformatted text)
IBM Infoprint 60 (3160) IBM Infoprint 62 (4370)	‚afccu2'
IBM Infoprint 1100 Series: IBM Infoprint 1116 IBM Infoprint 1120 IBM Infoprint 1125 IBM Infoprint 1130 IBM Infoprint 1140 IBM Infoprint 1145 with internal MarkNet NIC card	‚/prt0'
IBM Infoprint 1200 Series: IBM Infoprint 1222 IBM Infoprint 1226 with internal MarkNet NIC card	‚/prt0'
IBM Infoprint 1300 Series: IBM Infoprint 1312 IBM Infoprint 1332 IBM Infoprint 1352 IBM Infoprint 1372 with internal MarkNet NIC card	‚/prt0'
IBM Infoprint 1400 Series: IBM Infoprint 1410 MFP IBM Infoprint 1412 IBM Infoprint 1422 with internal MarkNet NIC card	‚/prt0'
IBM Infoprint 1500 Series: IBM Infoprint 1532 Express IBM Infoprint 1552 IBM Infoprint 1572 IBM Infoprint 1585 with internal MarkNet NIC card	‚/prt0'
IBM Infoprint 1500 MFP Series: IBM Infoprint 1540 MFP IBM Infoprint 1560 MFP IBM Infoprint 1570 MFP IBM Infoprint 1580 MFP with internal MarkNet NIC card	‚/prt0'
IBM Infoprint Color Series: IBM Infoprint Color 1220 IBM Infoprint Color 1228 IBM Infoprint Color 1334 IBM Infoprint Color 1357 IBM Infoprint Color 1454 IBM Infoprint Color 1464 IBM Infoprint Color 1567 with internal MarkNet NIC card	‚/prt0'

6.5.1.1
Seite 6

Network Interface or System Type	Remote Printer Queue (RMTPRTQ) Value
IBM Infoprint 6500 with internal Ethernet print server	‚pr1'
IBM Integrated Network Option (IBM 4039, 4049, 3112, 3116)	‚/prt0'
IBM Network Print Server (NPS)	‚pr1' for parallel port 1 ‚pr2' for parallel port 2 .... ‚pr8' for parallel port 8
IBM Network Printer 12 (4312) IBM Network Printer 17 (4317) IBM Network Printer 24 (4324)	‚PASS' (formatted output) ‚TEXT' (unformatted text) Note: ‚TEXT' appears to work best for the IBM Network Printer 20
IBM Network Station	‚PARALLEL1' for the parallel port ‚SERIAL1' for the serial port
IBM RS/6000 (Printer Attached to RS/6000 System)	printer queue name (case sensitive)
IBM S/390 Mainframe with CNNTYPE(*IP) and DESTTYPE(*OTHER)	printer queue name (for example. ‚PRT01')
IBM S/390 Mainframe with CNNTYPE(*SNA) and DESTTYPE(*S390)	printer queue name (for example. ‚PRT01') *USER *SYSTEM
i-Data EasyCom, PlusCom, LinkCom print servers	‚LPDPRT1' or ‚LPDPRT2'
I-O LAN RPC print server	‚tcp1' or ‚tcp1_nt_nb_nff' where the „_nt_nb_nff" indicates no trailer page, no banner page and no extra Form Feed (FF) Note: See the section entitled Information Concerning Print Servers Designed for Twinax SCS Printers below.
I-O Multiprotocol print server (Single-Port)	‚LPT' or ‚LPT_nt_nb_nff' where the „_nt_nb_nff" indicates no trailer page, no banner page and no extra Form Feed (FF)
I-O Multiprotocol print server (Multi-Port)	‚LPT1' or ‚LPT1_nt_nb_nff' for parallel port 1 ‚LPT2' or ‚LPT2_nt_nb_nff' for parallel port 2 ‚COM1' or ‚COM1_nt_nb_nff' for serial port 1 where the „_nt_nb_nff" indicates no trailer page, no banner page and no extra Form Feed (FF)
Imagistics printers	‚PORT1'
	‚print'
Infotec printers	‚PORT1'
Intel InBusiness Print Station	No LPR Support
Intel Netport XL	‚TEXT1' for parallel port 1 ‚TEXT2' for parallel port 2 ‚PASSTHRU' to disable blank page
Intel Netport Pro	‚LPTx_PASSTHRU' (formatted output) ‚LPTx_TEXT' (unformatted text) where x = port (The PASSTHRU options will disable banner or trailer pages)
Intel NetportExpress XL and Pro	‚LPT1_PASSTHRU' for parallel port 1 (formatted output) ‚LPT1_TEXT' for parallel port 1 (text) ‚LPT2_PASSTHRU' for parallel port 2 (formatted output) ‚LPT2_TEXT' for parallel port 2 (text) ‚PASSTHRU' for serial port (formatted output) ‚TEXT' for serial port (text) ‚COM1_PASSTHRU' for serial port if ‚PASSTHRU' does not work. (The PASSTHRU options will disable banner or trailer pages)
Intermec printers (Internal)	‚pr1' for parallel port 1 ‚pr2' for parallel port 2 .... ‚pr8' for parallel port 8

Network Interface or System Type	Remote Printer Queue (RMTPRTQ) Value
JETNet Card(?)	‚PASS'
Konica printers Konica-Minolta printers with optional IP-XXX print controller	‚lptprt1'
Konica printers Konica-Minolta printers (certain other models)	‚LPDPRT1' ‚print' or ‚PRINT' ‚PORT1_PASTHRU'
Lanier printers	‚PORT1'
Lantronix print Servers	‚DSS_TEXT' for single parallel port (unformatted text) ‚EPS_nnnnnn_P1' for parallel port 1 ‚EPS_nnnnnn_P2' for parallel port 2 where „nnnnnn" represents the alpha-numeric name stamped on the device. Note: This is the default value for the Lantronix print server, but it can be changed by telnetting to the print server.
Lantronix LPS	‚LPS_nnnnnn_TEXT' for simple ASCII text or dot matrix printers. ‚LPS_nnnnnn_PCL' for HP LaserJet compatible printers ‚LPS_nnnnnn_PS' for PostScript printers where ‚nnnnnn' represents the alpha-numeric name stamped on the printer server. Note: This is the default value for the Lantronix print server, but it can be changed by telnetting to the print server.
Lantronix MPS1-T and MPS1-2	‚MPS_nnnnnn_TEXT' for text and Zebra bar code printers (unformatted text) ‚MPS_nnnnnn_PCL' for HP LaserJet and compatible printers (formatted output) where „nnnnnn" represents the alpha-numeric name stamped on the device. Note: This is the default value for the Lantronix print server, but it can be changed by telnetting to the print server.
Lexmark MarkNet XL (Internal)	‚/prt0'
Lexmark MarkNet XLe (External) Lexmark MarkNet Pro (External) Lexmark N4000e (External)	‚/prt1' for parallel port 1 ‚/prt2' for parallel port 2 .... ‚/prt8' for parallel port 8 ‚/prt9' or ‚/ser' for serial port
Lexmark MarkNet Pro3 (External) Lexmark MarkNet 2000 (External)	‚/prt0' or ‚/par0' for parallel port 1 ‚/prt1' or ‚/par1' for parallel port 2 ‚/prt9' or ‚/ser' for serial port
Lexmark Optra with internal MarkNet NIC card	‚/prt0'
LINKSYS print servers (Internal) LINKSYS Pocket print server (External)	‚L1'
LINKSYS print servers (Multi-Port External)	‚L1' for parallel port 1 ‚L2' for parallel port 2 ‚L3' for parallel port 3
Mannesman-Tally	‚PR2'
Meto LIS-1830e Continuous Form Laser Imaging System Meto LIS-1865e Continuous Form Laser Imaging System	‚LIS_METO' or ‚lis_meto' (not case sensitive)
Microplex M205 (Internal) Microplex M206 (Internal)	‚d1prn'
Microplex M202 (External) Microplex M202+ (External)	‚d1prn1' for parallel port 1 ‚d2prn2' for parallel port 2 ‚d3com1' for serial port 1 ‚d4com2' for serial port 2
MiLAN Technology FastPort Print Servers	‚FPxxxxxx' where ‚xxxxxx' represents the last 6 digits of the print server MAC address.

6.5.1.1
Seite 8

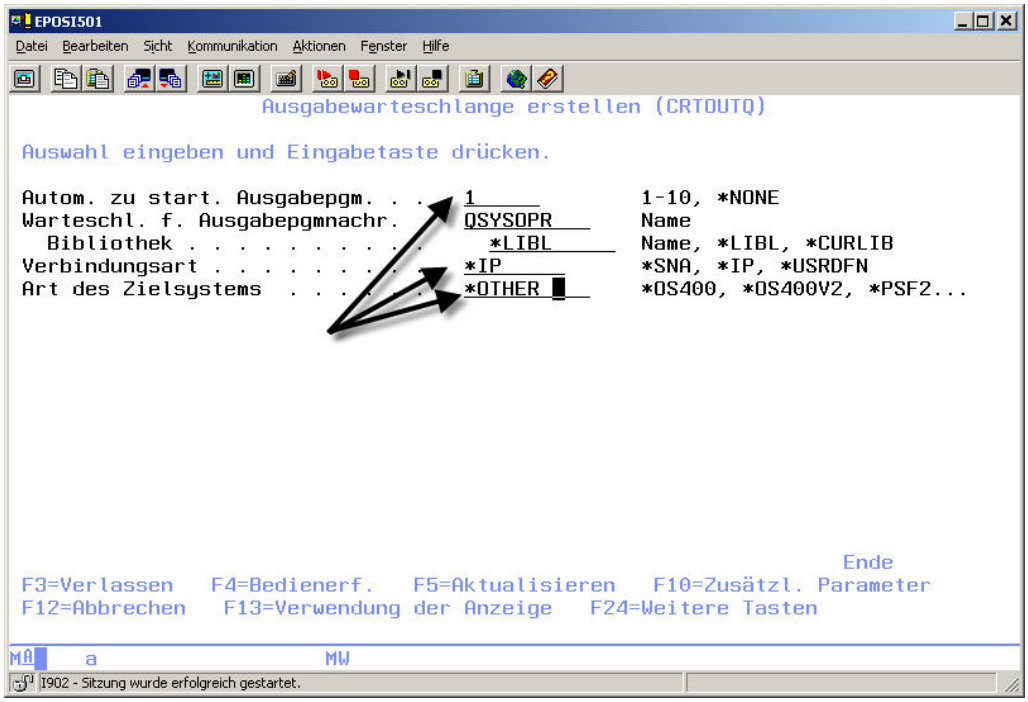
Network Interface or System Type	Remote Printer Queue (RMTPRTQ) Value
Minolta Di6200 print server	‚print' or ‚PRINT'
Minolta PageWorks 25	‚PORT1_PASTHRU'
MINOLTA-QMS CrownNet print servers	‚raw'
Mita	‚LPAF'
Monarch Thermal Label printers (Internal)	printer queue name specified in the printer or print server hardware (case sensitive)
Monarch 9820 printer Monarch 9850 printer Monarch 9855 printer Monarch 9860 printer	‚TEXT_P1' for plain text (unformatted output) ‚BINARY_P1' for binary files (formatted output)
Monarch PaxarNet 7410 print server	‚TEXT_P1' for plain text (unformatted output) ‚BINARY_P1' for binary files (formatted output)
Nashuatec printers	‚PORT1'
Netgear print servers	‚L1' for parallel port 1 ‚L2' for parallel port 2
Net Print (Internal)	‚PORT1'
Océ printers with external Fiery LAN Adapter	‚print' or ‚PRINT'
Océ 3155/3165 Digital Copier	printer queue name configured through OS/2 Warp on attached Dell PC (case sensitive)
OKI print servers Okidata print servers	‚lp'
Osicom JetXPrint (Internal)	‚PORT1'
Osicom NetPrint (External)	‚PORT1' for parallel port 1 ‚PORT2' for parallel port 2
Panasonic DP-3510 printer (Internal)	‚lp'
Printek (Internal)	‚pr1'
Printronix Printers with Printronix PrintNet adapters	‚d1prn' for single port ‚d1prn' for parallel port 1 ‚d2prn' for parallel port 2 ‚d3prn' for parallel port 3 ‚d4prn' for parallel port 4
Rex-Rotary printers	‚PORT1'
Ricoh printers	‚PORT1'
Ricoh EB70, EB105, EB105E Printers	‚print'
Samsung printers	‚PORT1'
Sato printers	*NONE Note: Presumably, Sato printers with internal Sato print servers will work with any RMTPRTQ value.
Savin printers	‚PORT1'
Sharp printers	‚PORT1'
Star Micronics print servers	Unknown at this Time Note: The proper value is unknown, but it appears that Star Micronics print servers will work with almost any value.
StarTech print servers	‚lpt1' for parallel port 1 ‚lpt2' for parallel port 2 ‚lpt3' for parallel port 3
Symbol CB1000 Wireless print servers	‚raw'
TallyGENICOM	Depends on the printer model. See the entries for GENICOM and Mannesmann-Tally printers above.

Network Interface or System Type	Remote Printer Queue (RMTPRTQ) Value
Tektronix Phaser 550 printer	‚PS', ‚HPGL', ‚PCL', or ‚AUTO' The PCL or AUTO should work in most cases.
Toshiba DC printers and digital copiers	‚raw'
Toshiba eStudio printers	‚direct' or ‚dssc'
TransAct Ethernet Adapter	‚PORT1'
TRENDnet print servers	‚lpt1'
Troy	‚raw'
Typhoon	‚pcl5'
Unisys WinPath 3000 Series Terminals	printer name (case sensitive) specified in the LPD Config on the terminal.
Unix (Printer Attached to Unix System)	printer queue name (case sensitive)
Windows 95, Windows 98 and Windows ME	LPD queue name (case sensitive)
Windows NT Windows 2000 Windows 2003 Server Windows XP Professional	printer share name (case sensitive)
WYSE 3000 Series Terminals	printer name (case sensitive) specified in the LPD Config on the terminal
XCD	‚BINARY_P1' for parallel port 1 (formatted output) ‚TEXT_P1' for parallel port 1 (unformatted text) ‚BINARY_P2' for parallel port 2 (formatted output) ‚TEXT_P2' for parallel port 2 (unformatted text) ‚BINARY_S1' for serial port 1 (formatted output) ‚TEXT_S1' for serial port 1 (unformatted text)
Xerox 4517	‚PASSTHRU'
Xerox DC230 printer	‚lp'
Xerox N40	‚lp'
Zebra Printers with ZebraNet Print Server (MPS1-T and MPS1-2)	‚MPS_nnnnnn_TEXT' for text and Zebra bar code printers (unformatted text) where „nnnnnn" represents the alpha-numeric name stamped on the device. Note: This is the default value for the ZebraNet print server; however, it can be changed by telnetting to the print server.
Zebra Printers with ZebraNet Print Server II (External)	‚portLF1' for parallel port 1 ‚portLF2' for parallel port 2 ‚portLF3' for parallel port 3
Zebra Printers with ZebraNet Print Server II (Internal)	‚portLF1'
Zonet ZPS3601 external print server Zonet ZPS3603 external print server Zonet ZPS3611 external print server	‚LPT1' for parallel port 1 ‚LPT2' for parallel port 2 ‚LPT3' for parallel port 3
Zonet ZPS1000 external print server Zonet ZPS2120 external print server Zonet ZPW1000 external printer server	‚LP1' for parallel port 1 ‚LP2' for parallel port 2 ‚LP3' for parallel port 3

Quelle: IBM Software Knowledgebase URL: http://www-912.ibm.com/s_dir/slkbase.nsf/1ac66549a21402188625680b0002037e/b49aa3cf3ce97d0d862565c2007d4393?OpenDocument&Highlight=2,Remote,Printer,Queue,Value,RMTPRTQ

Weitere Parameter sind zu versorgen.

6.5.1.1
Seite 10



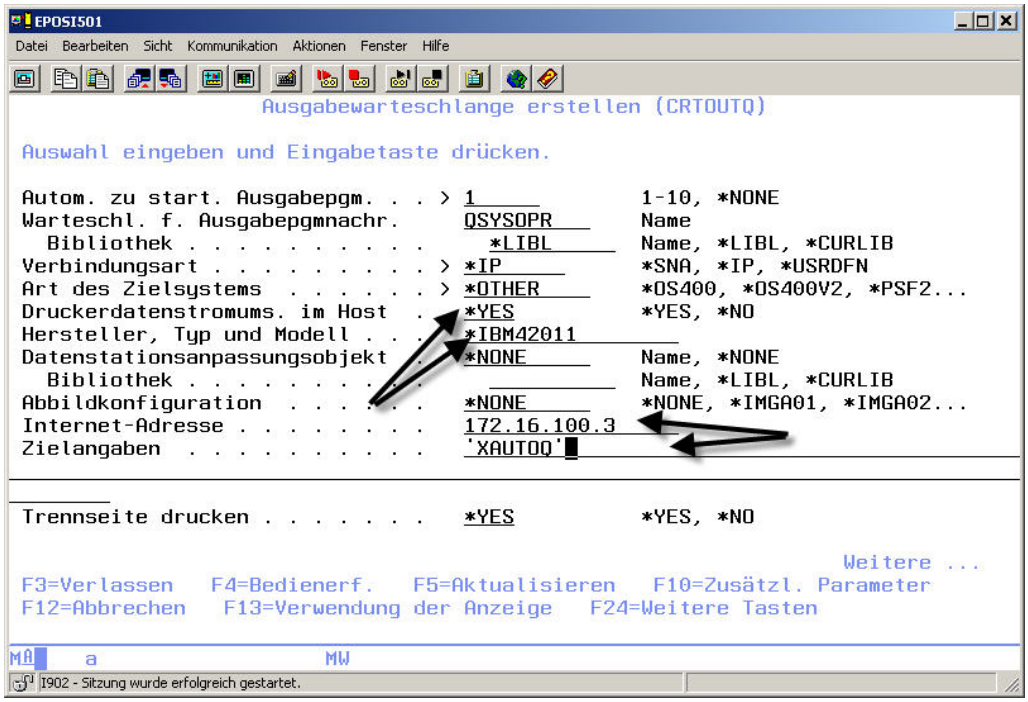
Weitere Parameter 2

CRTOUTQ-Parameter

- AUTOSTRWTR: Anzahl der zu startenden Remote Writer Programme, die für das Versenden der Druckausgabe an das entfernte System zuständig sind.
- CNNTYPE: Transportprotokoll für die Verbindung

  *IP
- DESTTYPE: Art des Zielsystems

  *OTHER



Weitere Parameter 3

CRTOUTQ-Parameter

- TRANSFORM: Im Falle unseres IBM 4226-Druckers muss die iSeries die Konvertierung des Datenstroms von EPSCDIC nach ASCII realisieren. Benutzt der Zieldrucker ASCII, so ist dieser Parameter zwingend auf *YES zu setzen.
- MFRTYPMOD: Den passenden Hersteller auswählen.
  Achtung: Auf PTFs achten!
- INTNETADR: IP-Adresse des empfangenden Systems eintragen:
  - Schnittstelle des Printservers.
- DESTOPT: Zielangaben. Hier kann eine Zeichenkette eingetragen werden, die bei der Auswertung des Initialisierungsstrings für den Druckvorgang als Befehl interpretiert wird.
  - XAUTOQ
    Wiederaufnahme des Drucks bei Abbruch.
  - XAIX
    Kopien während des Druckens erzeugen.

6.5.1.1
Seite 12

Folgende EBCDIC zu ASCII-Konvertierungstreiber werden von der iSeries im Release V5R4 und aktuellen Gruppen-PTF SF99347 zur Verfügung gestellt:

*IBM2380
*IBM2381
*IBM2390
*IBM2391
*IBM3112
*IBM3116
*IBM3130
*IBM3812
*IBM3816
*IBM3912HP
*IBM3916HP
*IBM39302
*IBM39303
*IBM42021
*IBM42022
*IBM42023
*IBM42071
*IBM42072
*IBM42081
*IBM42082
*IBM4212
*IBM4216
*IBM4226
*IBM4230
*IBM4232
*IBM4244ASF
*IBM47121
*IBM47122
*IBM47221
*IBM47222
*IBM4770
*IBM4912
*IBM5152
*IBM5201
*IBM5202
*IBM5204
*IBM5216
*IBM4019
*IBM4019HP
*IBM4029
*IBM4029HP
*IBM4037
*IBM4039HP
*IBM4070
*IBM4070EP
*IBM4072
*IBM4076
*IBM42011
*IBM42012
*IBM42013
*IBM4244DUAL
*IBM4244MAN
*IBM4247ASF
*IBM4247DUAL
*IBM4247MAN
*IBM4308
*IBM4312
*IBM4317
*IBM4320
*IBM4322
*IBM4324
*IBM4332
*IBM4340
*IBM6400EP
*IBM6404
*IBM6404EP
*IBM6408
*IBM6408EP
*IBM6412
*IBM6412EP
*IBMPAGES
*IBMPAGESNPB
*IBMPAGES300
*IBMPAGES300NPB

*IBM5575
*IBM6400
*INFOPRINT20
*INFOPRINT21
*INFOPRINT32
*INFOPRINT40
*INFOPRINT70
*INFOPRINT85
*INFOPRINT105
*INFOPRINT1116
*INFOPRINT1120
*INFOPRINT1125
*INFOPRINT1130
*INFOPRINT1140
*INFOPRINT1145
*INFOPRINT1412
*INFOPRINT1422
*INFOPRINT1500
*INFOPRINT2000
*INFOPRINT2085
*INFOPRINT2105
*INFOPRINT2705
*INFOPRINT2706
*INFOPRINT2761
*INFOPRINT2775
*INFOPRINT2785
*INFOPRINT2790
*INFOPRINT6500
*EPFX870
*EPFX1170
*EPLQ510
*EPLQ570
*EPLQ860
*EPLQ870
*EPLQ1070
*EPLQ1170
*EPLQ2550
*EPLX810
*EPSQ870
*INFOPRINT8C
*INFOPRINT12
*INFOPRINT1220C
*INFOPRINT1222
*INFOPRINT1226
*INFOPRINT1228C
*INFOPRINT1312
*INFOPRINT1332
*INFOPRINT1334C
*INFOPRINT1352
*INFOPRINT1354C
*INFOPRINT1357C
*INFOPRINT1372
*INFOPRINT1400C
*INFOPRINT1410
*CANLIPS3
*CANLIPS3NPB
*CPQPM15
*CPQPM20
*EPAP2250
*EPAP3250
*EPAP5000
*EPAP5500
*EPDFX5000
*EPDFX8000
*EPEPL7000
*EPEPL8000
*EPFX850
*HPII
*HPIID
*HPIIP
*HPIII
*HPIIID
*HPIIIP
*HPIIISI
*HP4
*HP5
*HP5SI
*HP6

6.5.1.1
Seite 14

*EPSQ1170
*ESCPDBCS
*HP500
*HP520
*HP540
*HP550C
*HP560C
*HP1100
*HP1200C
*HP1600C
*HP4000
*HP5000
*HP8000
*HPCOLORLJ
*HPDBCS
*LEXMARKE322
*LEXMARKE323
*LEXMARKE330
*LEXMARKT
*LEXMARKT420
*LEXMARKT520
*LEXMARKT522
*LEXMARKT620
*LEXMARKT622
*LEXMARKT630
*LEXMARKT632
*LEXMARKT634
*LEXMARKW
*NECP5200
*NECP5300
*NECP6200
*NECP6300
*NECPCPR201
*NONE
*OKI184IBM
*OKI320IBM
*OKI321IBM
*OKI390IBM
*OKI391IBM
*HP310
*HP320
*HPPAINT
*LEX2380
*LEX2381
*LEX2390
*LEX2391
*LEX4227
*LEXMARKC
*LEXMARKC510
*LEXMARKC750
*LEXMARKC752
*LEXMARKC910
*LEXMARKC912
*LEXMARKE
*LEXMARKW812
*LEXMARKW820
*LEXMARKX422
*LEXOPTRA
*LEXOPTRAC
*LEXOPTRAN
*LEXOPTRAS
*LEXOPTRASC
*LEXOPTRAT
*LEXOPTRAW
*NECP2
*NECP2200
*NECP2200XE
*OKi591IBM
*OKI400
*OKI800
*OKI810
*OKI820
*OKI3410
*PAN1123EP
*PAN1124EP
*PAN1124IEP
*PAN1180EP
*PAN1180IEP

*OKI393IBM
*OKi590IBM
*PAN1654EP
*PAN1695EP
*PAN2123EP
*PAN2124EP
*PAN2180EP
*PAN2624EP
*PAN4410HP
*PAN4420HP
*PAN4430HP
*PAN4450IHP
*PAN4451HP
*PANASONIC2310
*PANASONIC3010
*RICOH2045
*RICOHAP400
*RICOHAP600N
*RICOHAP900
*RICOHAP3200
*RICOHAP4510
*RICOHCL2000
*RICOHCL3100
*RICOHCL4000
*RICOHCL5000
*RICOHCL7000
*RICOHCL7100
*WORKIO_BL
*WSCSTCONT80
*WSCSTCONT132
*WSCSTEXECUTIVE
*WSCSTLEDGER
*WSCSTLEGAL
*WSCSTLETTER
*WSCSTNONE
*WSCST
*PAN1191EP
*PAN1624EP
*PANASONIC3510
*PANASONIC3520
*PANASONIC4510
*PANASONIC4520
*PANASONIC6010
*PANASONIC6020
*RICOH1515
*RICOH2015
*RICOH2018
*RICOH2022
*RICOH2027
*RICOH2032
*RICOH2035
*WORKIO_BM
*WORKIO_CR
*XRX4215MRP
*XRX4219MRP
*XRX4220MRP
*XRX4230MRP
*XRX4235
*XRX4700II
*WSCSTA3
*WSCSTA4
*WSCSTA5
*WSCSTB4
*WSCSTB5

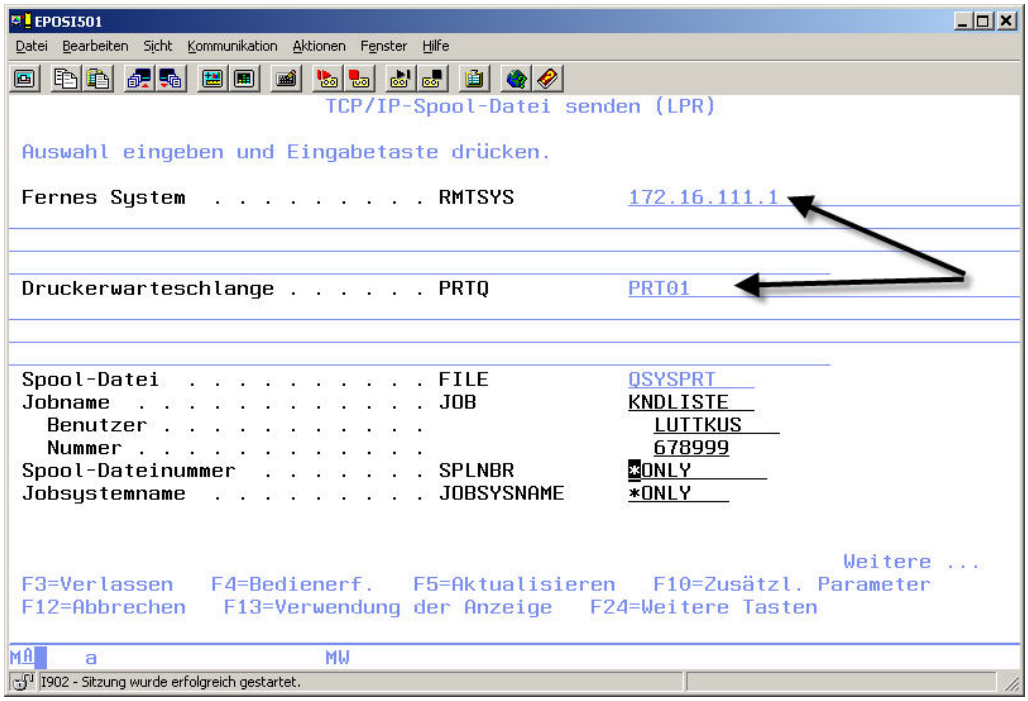
6.5.1.1
Seite 16

Die iSeries Befehle LPR und SNDTCPSPLF

Die Befehle LPR und SNDTCPSPLF sind Synonyme und dienen dem spontanen Drucken auf einem Netzwerkdrucker bzw. dem programmgesteuerten Drucken auf einem Netzwerkdrucker. D.h., Sie können die Befehle direkt von der Befehlszeile eines i5/OS-Dialogs oder z.B. aus einem CL-Programm heraus nutzen.

Die Parameter dieser Befehle entsprechen den Parametern des CRTOUTQs für eine Remote-OUTQ.

Zur Abwechslung setze ich in dem gezeigten Befehl voraus, dass Sie von System i zu System i drucken wollen.



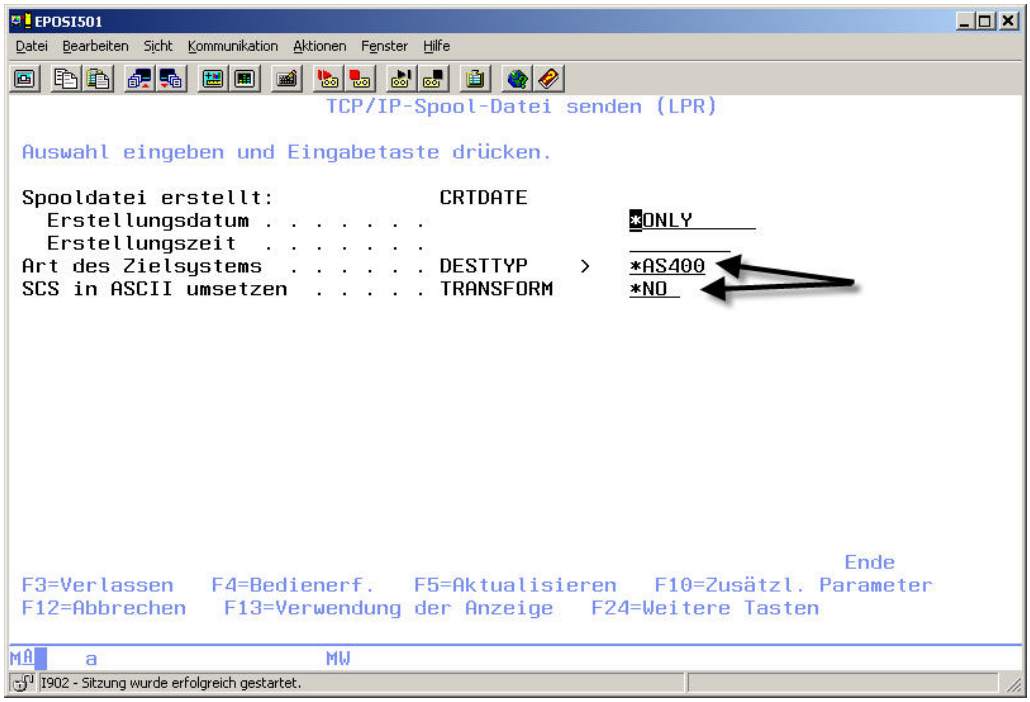
Der Befehl LPR / SNDTCPSPLF

In diesem Beispiel wird spontan auf einer entfernten AS400 mittels TCP/IP gedruckt.

LPR-Parameter:

- RMTSYS: Hostname der entfernten Schnittstelle
- PRTQ: Name der entfernten Druckerwarteschlange. Wird nicht auf eine entfernte iSeries gedruckt, so gelten hier die gleichen Namensregelungen wie beim CRTOUTQ-Befehl.
- FILE: Name der Spooldatei

6.5.1.2
Seite 2



LPR/SNDTCPSPLF – weitere Parameter

Weitere LPR/SNDTCPSPLF-Parameter

- DESTTYP: Art des Zielsystems
- TRANSFORM: SCS in ASCII umsetzen

Der LAN-Drucker – CRTDEVPRT(1)

Die Remote-OUTQ ist ein sehr flexibles Mittel, um Drucker unterschiedlichster Hersteller, Printserver und Betriebssysteme und deren Drucker anzusprechen. Eine Remote-Ausgabewarteschlange hat aber auch Defizite.

- Keine Fachsteuerung bei der Ausgabe.
- Kein Wiederaufsetzen innerhalb der Spooldatei.

Diese Defizite können durch eine andere Art von Netzwerkdruckerkonfiguration auf der iSeries ausgeglichen werden. Seit OS/400 V4R4 ist es möglich eine so genannte LAN-Device aufzubauen, die als eine Art virtueller Drucker anzusehen ist, der auf einen Netzwerkdrucker verweist.

Die LAN-Device unterstützt die Funktionen, die der Remote-OUTQ fehlen:

- Fachsteuerung
- Kopien drucken
- Wiederaufsetzen bei abgebrochenen Druckvorgängen etc.

Da stellt sich doch die Frage, warum man überhaupt noch mit remoten Ausgabewarteschlangen arbeiten soll? Der Grund ist folgender: LAN-Devices lassen das Drucken nur über Geräte zu, die technisch mit bestimmten Merkmalen ausgestattet sind.

- Die Drucker müssen folgende Merkmale aufweisen:
- PCL5/6 fähig.
- PJL-Monitor muss unterstützt werden (bis OS/400 V4R4).
- Alternativ ab OS/400 V4R5 SNMP-Protokoll muss unterstützt werden. Drucker müssen über eine TCP/IP-Portsteuerung verfügen.

Haben Sie Drucker / Endgeräte, die diese Merkmale nicht aufweisen, so können Sie unter TCP/IP keine LAN-Device verwenden.

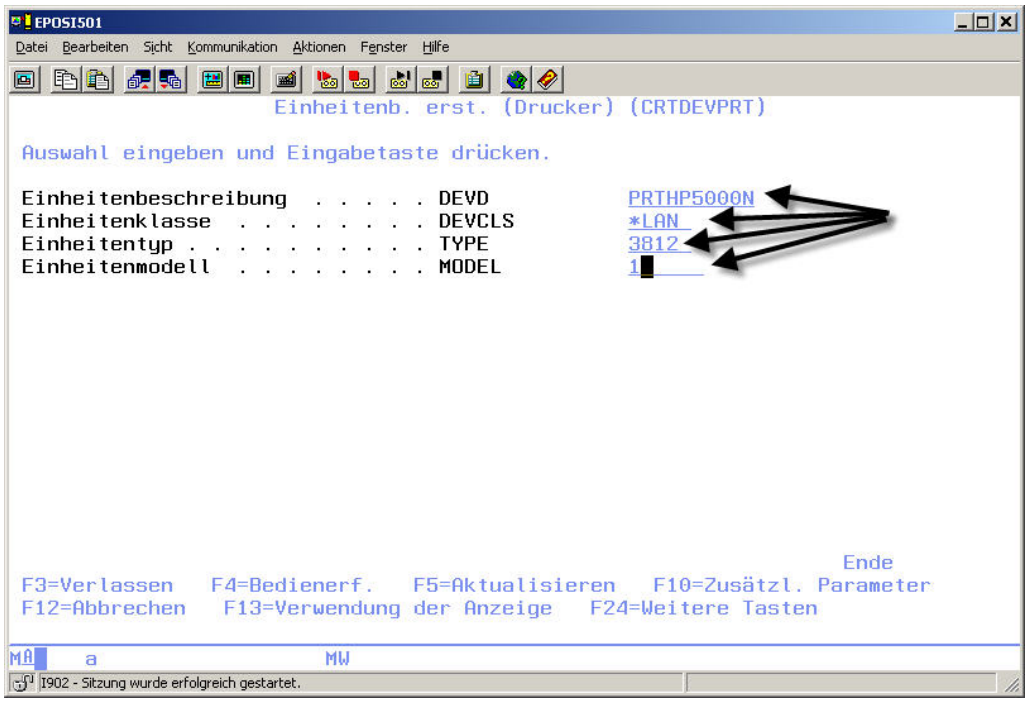
In diesem Beispiel konfiguriere ich eine LAN-Device für einen HP5000N, der die geforderten Eckwerte aufweist.

6.5.1.3

Seite 2

LPR – LAN Device

Die Konfiguration beginnt mit der Eingabe des Befehls CRTDEVPRT. Einige der Parameter sind identisch mit denen der Remote-Ausgabewarteschlange, deshalb werde ich nur noch die Parameter erläutern, die für eine LAN-Device signifikant sind.



CRTDEVPRT 1

Folgende Parameter müssen gefüllt werden:

Einheitenbeschreibung PRTHP5000N

Dieser Name ist individuell zu vergeben.

Einheitenklasse . . . . . *LAN

Dieser Parameter sorgt für die Erstellung der LAN-Device.

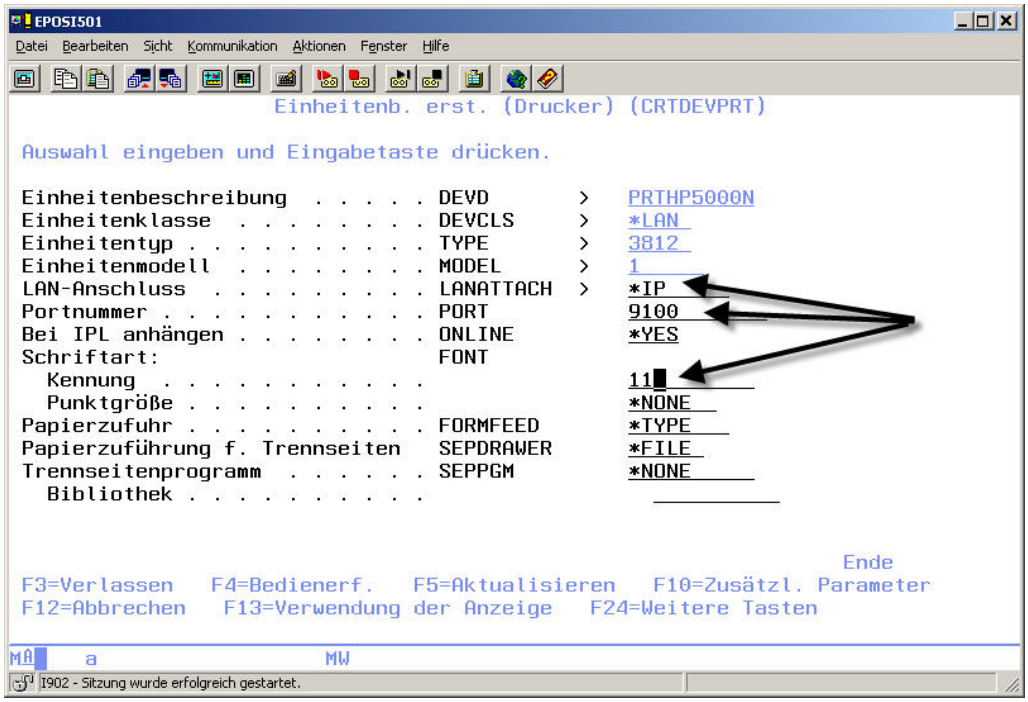
Einheitentyp 3812

Dieser Druckertyp ist zwingend notwendig.

Einheitenmodell 1

Auch das Modell muss so angegeben werden.

6.5.1.3.1
Seite 2



CRTDEVPRT 2

Notwendige Parameter:

LAN-Anschluss . . . . . *IP

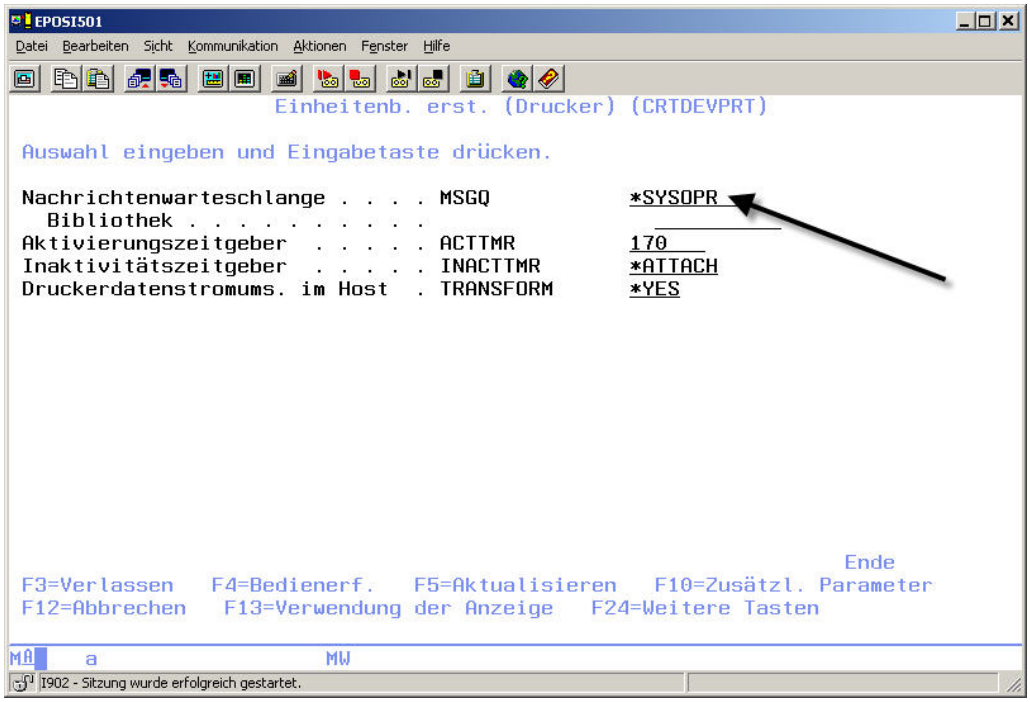
Seit OS/400 V4R4 kann das TCP/IP-Protokoll angegeben werden.

Anschlußnummer 9100

Die TCP/IP-Portnummer ist nicht standardisiert und muss für den jeweiligen Drucker ermittelt werden. Viele Hersteller, die Drucker mit PCL-Emulationen anbieten, unterstützen mittlerweile den Port 9100. Am Ende dieses Kapitels habe ich Ihnen eine Liste der bekannten Druckerport-Nummern beigefügt.

Schriftart: Kennung 11

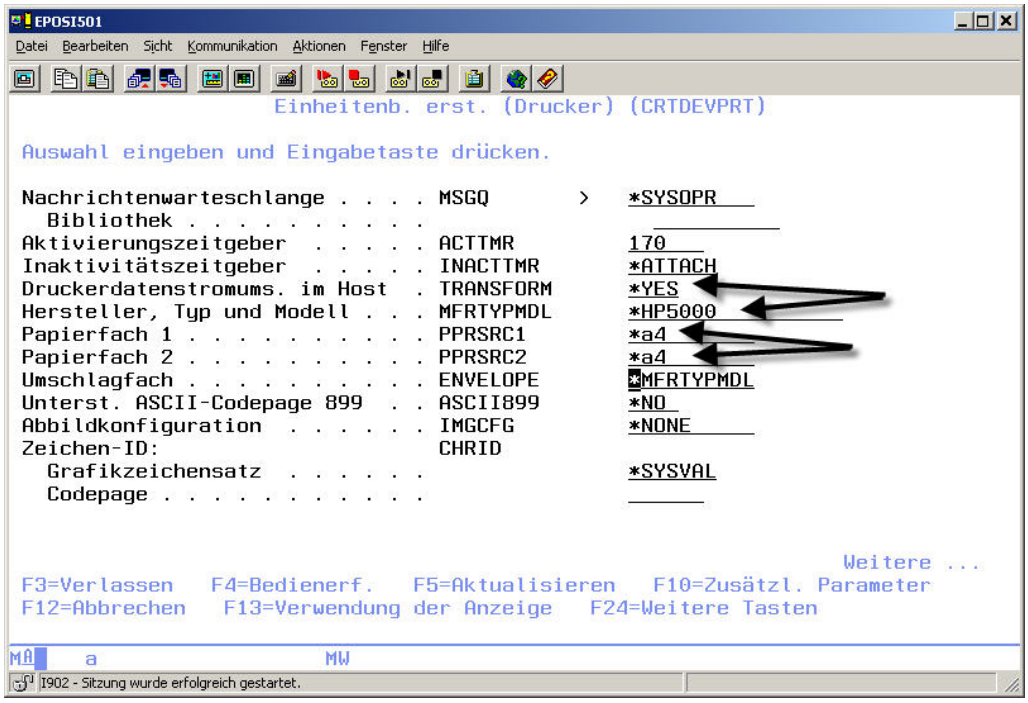
Die Standardschriftart, die verwendet wird, falls aus der Spool keine Schriftart angewählt wird.



CRTDEVPRT 3

Nicht notwendig, aber sinnvoll:

Nachrichtenwarteschlange . . . . *SYSOPR



CRTDEVPRT 4

6.5.1.3.1
Seite 4

Wichtige Parameter:

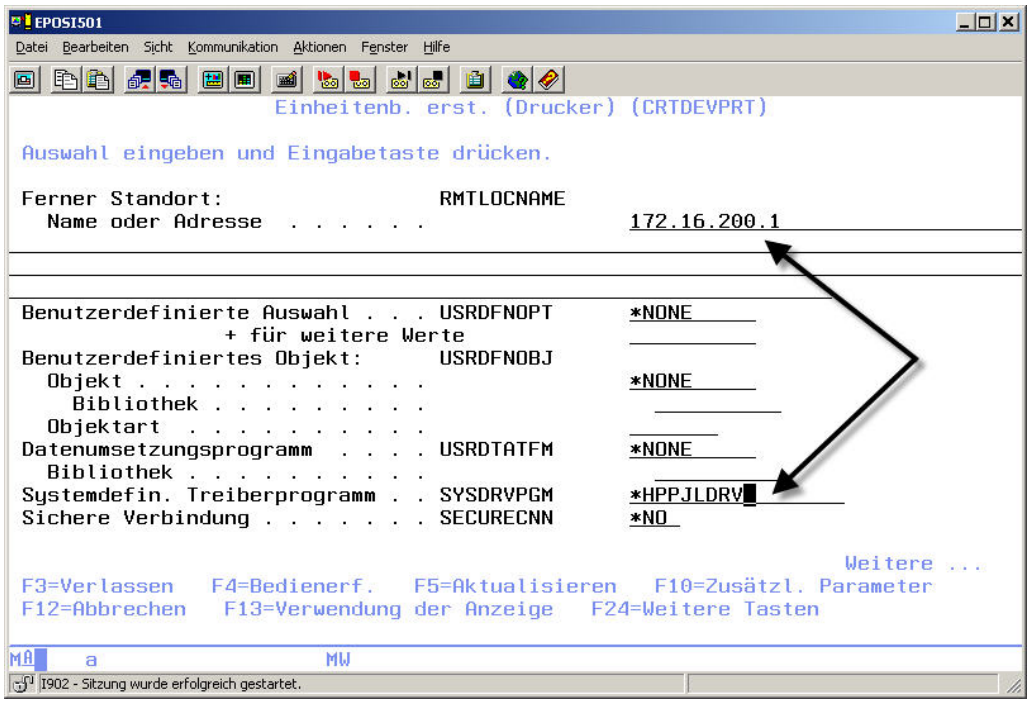
Druckerdatenstromums. im Host . *YES

Hersteller, Typ und Modell . . . *HP5000

Papierfach 1 . . . . . *A4

Papierfach 2 . . . . . *A4

Umschlagfach . . . . . *MFRTYPMDL



CRTDEVPRT 5

Wichtige und notwendige Parameter:

Ferner Standort:

Name oder Adresse 172.16.200.1

Systemdefiniertes Treiberprogramm . . *HPPJLDRV

Die Liste der bekannten TCP/IP-Druckports

Network Interface or System Type	PJL Port Number
Argecy ACC 6xe Coax/Twinax to Ethernet print server	No PJL Support
Avery Dennison Printers	No PJL Support
Axis LAN Adapters (external print servers)	Note: The Reverse Telnet port(s) must be set to some value since by default they are disabled. It is recommended that you set RTEL_PR1 = 9100 for parallel port 1 and RTEL_ PR2 = 9101 for parallel port 2, but any values can be used.
Axis 1620 for Canon PCL Printers	Unknown at this Time
Axis 5400+	Note: The Reverse Telnet port(s) must be set to some value since by default they are disabled. It is recommended that you set RTEL_PR1 = 9100 for parallel port 1 and RTEL_ PR2 = 9101 for parallel port 2, but any values can be used.
Axis 5600+ Axis 5800+ Mobile Axis 5900	Note: The Reverse Telnet port(s) must be set to some value since by default they are disabled. It is recommended that you set RTEL_PR1 = 9100 for parallel port 1 and RTEL_ PR2 = 9101 for parallel port 2, but any values can be used.
Axis 540+ Axis 560 Axis 5810 Print Plus	Note: The Reverse Telnet port(s) must be set to some value since by default they are disabled. It is recommended that you set RTEL_PR1 = 9100 for parallel port 1 and RTEL_ PR2 = 9101 for parallel port 2, but any values can be used.
Axis OfficeBasic	Unknown at this Time
Axis PrintPoint 140 BJC	Unknown at this Time
Barr Systems	Unknown at this Time
Brother printers	9100
Castelle LANPress 2+1MP print servers	Parallel Port 1 = 3010 Parallel Port 2 = 3020 Serial Port = No PJL Support
D-Link DP101 D-Link DP300	No PJL Support
Dell printers	9100
Epson Interface Card	Unknown at this Time
Extended Systems print servers	Unknown at this Time Note: According to Extended Systems, their print servers support PJL, but the port numbers are not known at this time.
Extend Net print servers	9100
Gestetner printers	9100 Note: The Base Port Number in the NIC card may need to be set to 9099. See the section entitled Information Concerning Osicom Print Servers and Ricoh, Savin, Lanier, Gestetner, Nashuatec, Rex-Rotary and Infotec Printers below.
Hawking Technology print server (parallel models)	No PJL Support
Hawking Technology print server (USB models)	No PJL Support
HP JetDirect (Internal) print servers	9100
HP JetDirect (External) print servers (3 ports - 1 I/P address)	9100 for Parallel Port 1 9101 for Parallel Port 2 9102 for Parallel Port 3
HP LaserJet printers (with internal HP JetDirect print server)	9100
HP WP110 Wireless print servers (External)	No PJL Support
HP Wireless print servers (External)	Unknown at this Time

6.5.1.3.1
Seite 6

Network Interface or System Type	PJL Port Number
i-data EasyCom, LinkCom or PlusCom (external) print servers	9100 for Parallel Port 1 9101 for Parallel Port 2
i-data LinkCom or PlusCom (internal) print servers	9100
IBM 3116 (with internal Lexmark Marknet print servers)	9100
IBM 3130 printers IBM 3160 printers IBM Infoprint 60 printers	No PJL Support
IBM 6400 Network Connection	No PJL Support
IBM Infoprint Color 8 printers (IBM 4308)	9100
IBM Infoprint 12	No PJL Support
IBM Infoprint 20, 32 or 40 printers (IBM 4320, 4332 or 4340)	2501
IBM Infoprint 21 printers (IBM 4322)	No PJL Support
IBM Infoprint 1100 Series: IBM Infoprint 1116 printers IBM Infoprint 1120 printers IBM Infoprint 1125 printers IBM Infoprint 1130 printers IBM Infoprint 1140 printers IBM Infoprint 1145 printers	9100
IBM Infoprint 1200 Series: IBM Infoprint 1222 printers IBM Infoprint 1226 printers	9100
IBM Infoprint 1300 Series: IBM Infoprint 1312 printers IBM Infoprint 1332 printers IBM Infoprint 1352 printers IBM Infoprint 1372 printers	9100
IBM Infoprint 1400 Series: IBM Infoprint 1410 MFP IBM Infoprint 1412 IBM Infoprint 1422	9100
IBM Infoprint 1500 Series: IBM Infoprint 1532 Express IBM Infoprint 1552 IBM Infoprint 1572 IBM Infoprint 1585	9100
IBM Infoprint 1500 MFP Series: IBM Infoprint 1540 MFP IBM Infoprint 1560 MFP IBM Infoprint 1570 MFP IBM Infoprint 1580 MFP (with internal MarkNet NIC card)	9100
IBM Infoprint Color Series: IBM Infoprint Color 1220 IBM Infoprint Color 1228 IBM Infoprint Color 1334 IBM Infoprint Color 1357 IBM Infoprint Color 1454 IBM Infoprint Color 1464 IBM Infoprint Color 1567	9100
IBM Infoprint 6500 (with internal Ethernet print server)	No PJL Support
IBM Integrated Network Option print servers (IBM 4039, 4049, 3112, 3116)	9100
IBM iSeries 400 (printer attached to another iSeries 400)	No PJL Support
IBM Network Print Server (NPS) print servers	2501
IBM Network Printer 12, 17 and 24 (IBM 4312, 4317, and 4324)	2501

Network Interface or System Type	PJL Port Number
IBM RS/6000 (printer attached to RS/6000 system)	No PJL Support
Imagistics printers	9100
Infotec printers	9100 Note: The Base Port Number in the NIC card may need to be set to 9099.
Intel InBusiness Print Station print server	No PJL Support
Intel NetportExpress 10/100 (single-port) print servers	9100
Intel NetportExpress 10/100 (3-port) print servers	9100 for Parallel Port 1 9101 for Parallel Port 2 9102 for Parallel Port 3
Intel NetportExpress PRO and PRO/100 print servers	No PJL Support
Intel NetportExpress XL and EL print servers	No PJL Support
JETNet print servers	Unknown at this Time
Konica and Konica Minolta printers (with optional IP-511 print controller)	9100 Note: The IP-511 print controller must have firmware version 6.2.0m or higher and the NIC port must be set to 9100.
Konica and Konica Minolta printers (with optional IP-511A print controller) IP602, 26.4.0h	9100 Note: The IP-511A print controller must have firmware version 26.3.0e or higher and the NIC port must be set to 9100.
Konica and Konica Minolta printers (with optional IP-602 print controller)	9100 Note: The IP-602 print controller must have firmware version 26.4.0h or higher and the NIC port must be set to 9100.
Kyocera IB20 (internal) print servers	9100
Lanier printers	9100 Note: The Base Port Number in the NIC card may need to be set to 9099.
Lantronix print servers	30xx, xx represents the port number. These servers can have 1 to 32 ports (for example, port 1 is 3001)
Lexmark MarkNet XL (Internal) print servers	9100
Lexmark MarkNet XLe (External) print servers Lexmark MarkNet Pro (External) print servers Lexmark Marknet X2000 (External) print servers	Parallel Port 1 = 9100 Parallel Port 2 = 9101 Serial Port = No PJL Support
Lexmark Optra (with internal Lexmark Marknet print server)	9100
Linksys print servers	No PJL Support
Meto Printers	Unknown at this Time
Minolta DiAlta printers	10001
Mita 3010 printers	10001
Nashuatec printers	9100 Note: The Base Port Number in the NIC card may need to be set to 9099.
NetGear print servers	Unknown at this Time
Net Print NIC card (internal) print servers	9100 Note: The Base Port Number in the NIC card may need to be set to 9099.
Osicom JetXPrint NIC card (internal) print servers	9100 Note: The Base Port Number in the NIC card may need to be set to 9099.
Osicom NetPrint print server (external) print servers	9100 for Parallel Port 1 9101 for Parallel Port 2 9102 for Parallel Port 3 Note: The Base Port Number in the NIC card may need to be set to 9099.

6.5.1.3.1
Seite 8

Network Interface or System Type	PJL Port Number
Printronix Printers with Printronix PrintNet adapters	No PJL Support
Rex-Rotary printers	9100 Note: The Base Port Number in the NIC card may need to be set to 9099.
Ricoh printers	9100 Note: The Base Port Number in the NIC card may need to be set to 9099.
Samsung printers	No PJL Support
Savin printers	9100 Note: The Base Port Number in the NIC card may need to be set to 9099
Sharp printers and copiers	9100
Toshiba eStudio printers and digital copiers	9100 or possibly 9101
TransAct Ethernet Adapter	Unknown at this Time
TrendNet print servers	No PJL Support
Unix Systems (Printer Attached to Unix System)	No PJL Support
Windows 95, 98 or NT (Printer Attached to a PC)	No PJL Support
Xerox Document Centre printers	9100
Xerox DocuPrint printers (if Set Up to Support PCL/PJL, HP Mode)	2000 or 9100
Zonet ZPS 3603 external print server	No PJL Support

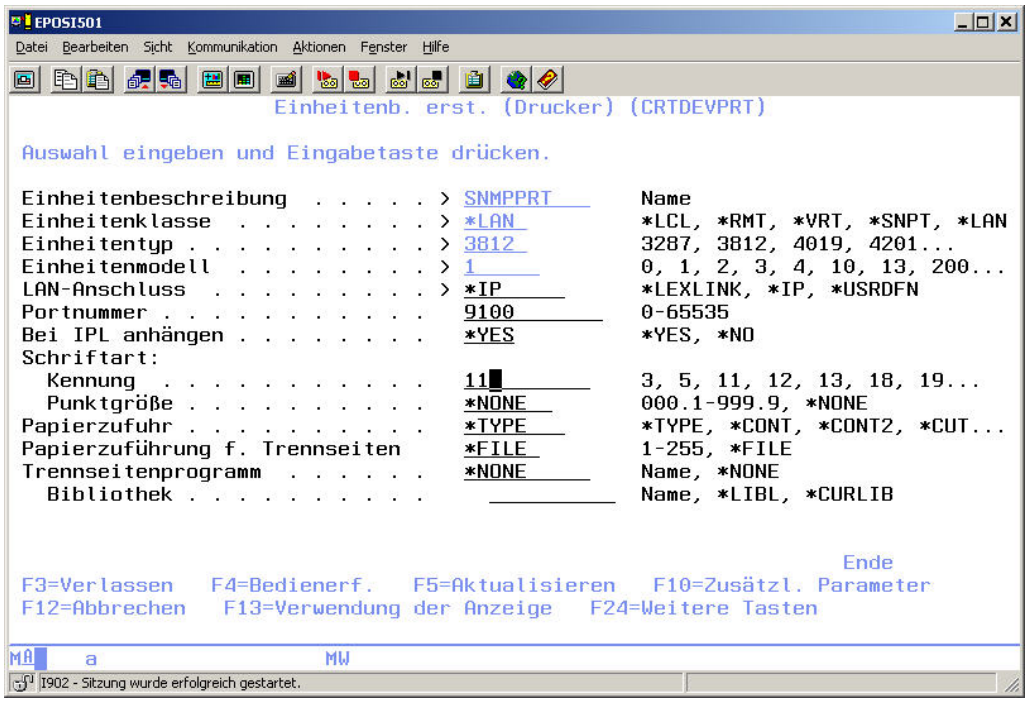
Quelle: IBM Software Knowledgebase URL: http://www-912.ibm.com/s_dir/slkbase.nsf/1ac66549a21402188625680b0002037e/ab93a7bf123bf3db8625668a0056a853?OpenDocument&ExpandSection=3&Highlight=2,PJL,Port,Number#_Section3

SNMP – LAN Device

6.5.1.3.2
Seite 1

Seit OS/400 V4R5 können LAN-Devices für Drucker konfiguriert werden, die nicht die PJL-Treiber unterstützen. In diesem Fall müssen die Drucker alternativ das TCP/IP-Diensteprotokoll SNMP unterstützen.

Die Konfiguration eine PJL-LAN-Device ist weitgehend identisch mit der einer SNMP-LAN-Device. Deshalb gebe ich Ihnen hier nur eine kurze Auflistung der entsprechenden Parameter an.



CRTDEVPRT 6

Diese Parameter werden so gesetzt wie bei der LPR-LAN-Device.

LAN-Anschluss . . . . . > *IP

Anschlussnummer 9100

6.5.1.3.2
Seite 2

Liste der SNMP-Ports

Network Interface or System Type	SNMP Port Number
Argecy ACC 6xe Coax/Twinax to Ethernet print server	No SNMP Support
Avery Dennison Printers	No SNMP Support
Axis LAN Adapters (external print servers)	9900 Note: See the section entitled Information Concerning IBM NPS and Axis Print Servers below.
Axis 1620 for Canon PCL Printers	Unknown at this time
Axis 5400+	9900 Note: See the section entitled Information Concerning IBM NPS and Axis Print Servers below.
Axis 5600+ Axis 5800+ Mobile Axis 5900	9900 Note: See the section entitled Information Concerning IBM NPS and Axis Print Servers below.
Axis 540+ Axis 560 Axis 5810 Print Plus	9900 Note: See the section entitled Information Concerning IBM NPS and Axis Print Servers below.
Axis OfficeBasic	Unknown at this time
Axis PrintPoint 140 BJC	9900 Note: See the section entitled Information Concerning IBM NPS and Axis Print Servers below.
Barr Systems	Unknown at this time
Brother printers	9100
Canon ImageRunner Printers with internal Fiery LAN Adapter	9100
Castelle LANPress 2+1MP print servers	Unknown at this time
D-Link DP101 D-Link DP300 D-Link DP301 Plus	No SNMP Support
Dell printers	9100
EFI Fiery print servers	9100
Epson Multiprotocol 10/100 Ethernet, Type B print servers	9100 Note: See the section entitled Information Concerning Epson Print Servers below.
Extended Systems print servers	No SNMP Support
Gestetner printers	9100
Hawking Technology print server (parallel models)	Unknown at this time (try 9100)
Hawking Technology print server (USB models)	No SNMP Support
HP JetDirect Card (Internal) print servers	9100
HP JetDirect Server (External) print servers (3 ports – 1 I/P address)	9100 for Parallel Port 1 9101 for Parallel Port 2 9102 for Parallel Port 3
HP LaserJet printers (with internal HP JetDirect print server)	9100
HP LineJet with internal print server	9100
HP WP110 Wireless print servers (External)	9100
HP Wireless print server (External)	Unknown at this time
i-data PlusCom or LinkCom (internal) print servers	Unknown at this time
i-data EasyCom, PlusCom or LinkCom (external) print servers	Unknown at this time
I-O 5430e print servers	No SNMP Support
IBM 3116 with internal Lexmark Marknet print server	Unknown at this time

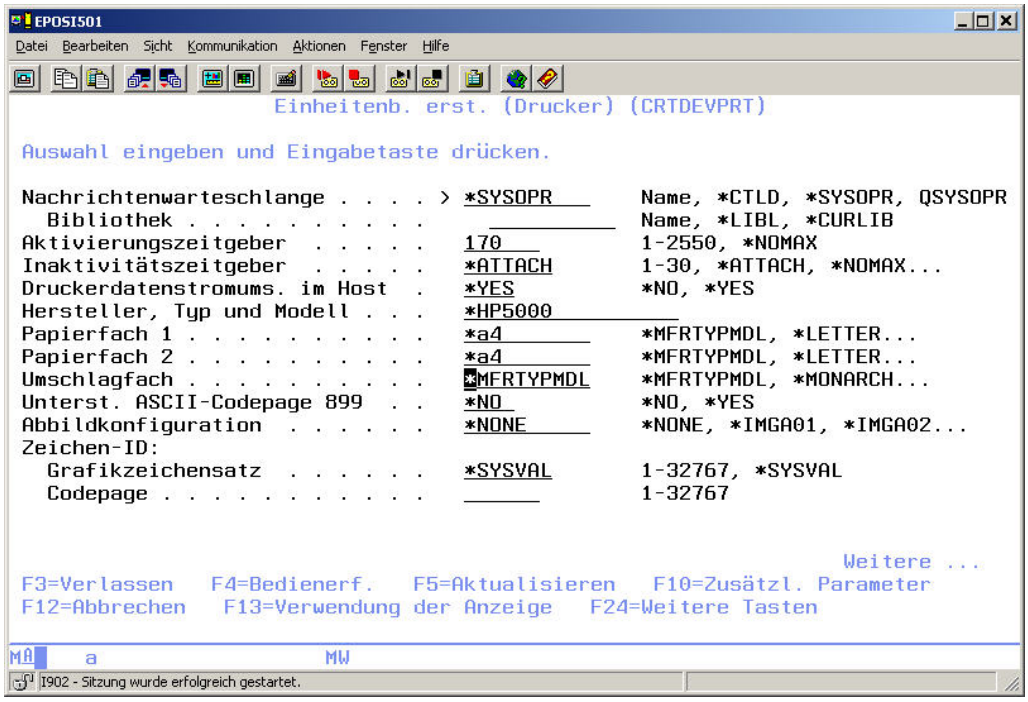
Network Interface or System Type	SNMP Port Number
IBM 3130 printers IBM 3160 printers IBM Infoprint 60 printers	No SNMP Support
IBM 4247 (with IBM NPS or Axis print server)	9900 (or possibly 9100) Note: See the section entitled Information Concerning IBM NPS and Axis Print Servers below.
IBM 4247 (with internal Ethernet print server)	9100
IBM 6400 (with IBM NPS or Axis print server)	9900 (or possibly 9100) Note: See the section entitled Information Concerning IBM NPS and Axis Print Servers below.
IBM 6400 (with Enhanced Ethernet Adapter print server)	9100 (or possibly 5001) Note: Check the ETHERNET PARAMETERS MENU in the printer hardware and use the value specified in the ASCII DATA PORT NUMBER parameter. If this is set to 0, then set it to 9100 and save the Custom Set.
IBM iSeries 400 (printer attached to another iSeries 400)	No SNMP Support
IBM® Infoprint® Color 8 (IBM 4308) printers	9100
IBM Infoprint 12 (IBM 4912) printers	9100
IBM Infoprint 21 (IBM 4322) printers	2501 (or possibly 9100)
IBM Infoprint 70 (IBM 2770) printers	2501
IBM Infoprint 20 (IBM 4320) printers IBM Infoprint 32 (IBM 4332) printers IBM Infoprint 40 (IBM 4340) printers	2501
IBM Infoprint 1100 Series: IBM Infoprint 1116 printers IBM Infoprint 1120 printers IBM Infoprint 1125 printers IBM Infoprint 1130 printers IBM Infoprint 1140 printers IBM Infoprint 1145 printers	9100
IBM Infoprint 1200 Series: IBM Infoprint 1222 printers IBM Infoprint 1226 printers	9100
IBM Infoprint 1300 Series: IBM Infoprint 1312 printers IBM Infoprint 1332 printers IBM Infoprint 1352 printers IBM Infoprint 1372 printers	9100
IBM Infoprint 1400 Series: IBM Infoprint 1410 MFP IBM Infoprint 1412 IBM Infoprint 1422	9100
IBM Infoprint 1500 Series: IBM Infoprint 1532 Express IBM Infoprint 1552 IBM Infoprint 1572 IBM Infoprint 1585	9100
IBM Infoprint 1500 MFP Series: IBM Infoprint 1540 MFP IBM Infoprint 1560 MFP IBM Infoprint 1570 MFP IBM Infoprint 1580 MFP (with internal MarkNet NIC card)	9100
IBM Infoprint Color Series: IBM Infoprint Color 1220 IBM Infoprint Color 1228 IBM Infoprint Color 1334 IBM Infoprint Color 1357 IBM Infoprint Color 1454 IBM Infoprint Color 1464 IBM Infoprint Color 1567	9100

6.5.1.3.2
Seite 4

Network Interface or System Type	SNMP Port Number
IBM Infoprint 6500 (with internal Ethernet print server)	9100
IBM 4039 printers with IBM Integrated Network Option IBM 4049 printers with IBM Integrated Network Option IBM 3112 printers with IBM Integrated Network Option IBM 3116 with IBM Integrated Network Option	Unknown at this time
IBM Network Print Server (NPS) printer server	9900 Note: See the section entitled Information Concerning IBM NPS and Axis Print Servers below.
IBM Network Printer 12 (IBM 4312) printers IBM Network Printer 17 (IBM 4317) printers IBM Network Printer 24 (IBM 4324) printers	2501
IBM RS/6000 (printer attached to RS/6000 system)	No SNMP Support
Imagistics printers	9100
Infotec printers	9100
Intel InBusiness Print Station print server	No SNMP Support
Intel NetportExpress 10/100 (single-port) print servers	Unknown at this time
Intel NetportExpress 10/100 (3-port) print servers	Unknown at this time
Intel NetportExpress PRO and PRO/100 print servers	Unknown at this time
Intel NetportExpress XL and EL print servers	Unknown at this time
JETNet print servers	Unknown at this time
Konica and Konica Minolta printers (with optional IP-421, IP-422, IP-423, IP-511, IP-511A or IP-602 print controller)	10001
Konica and Konica Minolta printers (with optional IP-432 print controller)	9100
Konica Minolta bizhub printers	9100
Konica Minolta DiAlta printers	10001
Kyocera IB20 (internal) print servers	9100
Lanier copier/printers	9100 or 10001
Lantronix print servers	Unknown at this time
Lexmark MarkNet XL (Internal)	9100
Lexmark MarkNet XLe or PRO (External)	9100 for Parallel Port 1 No SNMP Support for Parallel Ports 2 and 3
Lexmark Optra with internal Lexmark Marknet print server	9100
Linksys print servers	Unknown at this time
Meto Printers	Unknown at this time
Minolta DiAlta printers	10001
Nashuatec printers	9100
NetGear print servers	Unknown at this time
Osicom JetXPrint (Internal) print servers	Unknown at this time
Osicom NetPrint (External) print servers	Unknown at this time
Panasonic WORKiO Printers	9100
Printronix Printers with Printronix PrintNet adapters	9100
Rex-Rotary printers	9100
Ricoh printers	9100
Samsung printers	9100
Savin printers	9100
Sharp printers	9100
Toshiba e-Studio digital copiers	9100

Network Interface or System Type	SNMP Port Number
TransAct Ethernet Adapter	Unknown at this time
TrendNet print servers	No SNMP Support
Unix Systems (Printer Attached to Unix System)	No SNMP Support
Windows 95, 98 or NT (Printer Attached to a PC)	No SNMP Support
Xerox Document Centre 2xx printers	No SNMP Support
Xerox Document Centre 3xx printers	No SNMP Support
Xerox Document Centre 4xx printers	9100 Note: See the section entitled Information Concerning Xerox Document Centre 4xx Printers below.
Xerox DocuPrint printers (if Set Up to Support PCL/PJL, HP Mode)	Unknown at this time
Zonet ZPS 3603 external print server	No SNMP Support

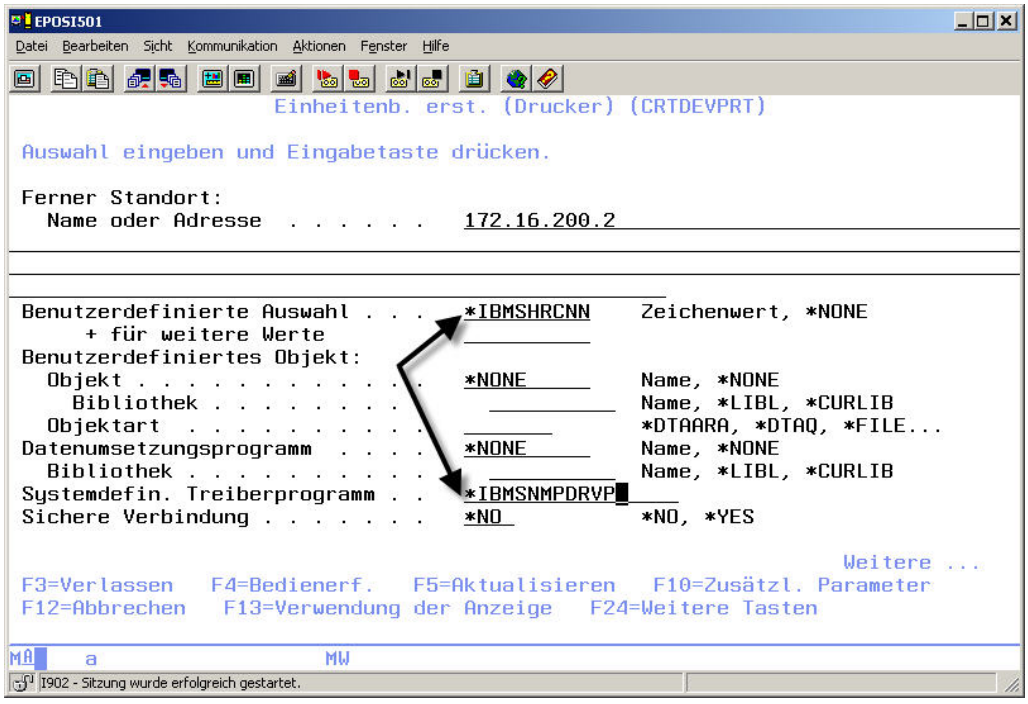
Quelle: IBM Software Knowledgebase URL: http://www-912.ibm.com/s_dir/slkbase.nsf/1ac66549a21402188625680b0002037e/94e73364f979e388862569030071c398?OpenDocument&ExpandSection=3&Highlight=2,SNMP,Port,Number#_Section3



CRTDEVPRT 7

Auch hier werden die Parameter so gesetzt, wie bei der LPR-LAN-Device.

6.5.1.3.2
Seite 6



CRTDEVPRT 8

SNMP-Parameter:

Benutzerdefinierte Auswahl . . . *IBMSHRCNN

Systemdefiniertes Treiberprogramm . . *HPPJLDRV

Bei allen bisher gezeigten Konfigurationen handelt es sich um globale Konfigurationsgänge, die nicht auf bestimmte individuelle Umstände eingehen. Sie sollten lediglich die Vorgehensweise bei der Konfiguration von LPR/LPD-basierten Druckprozessen im i5/OS verdeutlichen.

Differenzierter werden einzelne Konfigurationen im Abschnitt 6.6 Fallstudien durchgeführt.

Fallstudien – Drucken

In den vorherigen Kapiteln wurden die auf der iSeries zur Verfügung stehenden Werkzeuge grundsätzlich erläutert, die bei der Erstellung von netzwerkbezogenen Druckverfahren angewendet werden können. In diesem Kapitel werde ich eine ganze Reihe von Fallstudien entwickeln, die die beschriebenen iSeries Werkzeuge zur Anwendung bringen.

Folgende Szenarien sind dabei vorläufig geplant:

- iSeries druckt auf einem an einen Windows-Arbeitsplatz angeschlossenen Drucker.
- iSeries druckt auf einem LINUX-System.
- Ein Windows Client benutzt eine iSeries als Spooler, um auf einem Netzwerkdrucker zu drucken.
- Ein Windows Client benutzt eine iSeries als Spooler, um auf einem Print-Server zu drucken.
- Eine System-iSeries druckt auf einer zweiten System-iSeries.

Sollte „Ihre“ Konfiguration nicht dabei sein, so lade ich Sie dazu ein, mir weitere Vorschläge für dieses Kapitel zu liefern.

6.6 Seite 2

Fallstudie: iSeries druckt auf einem an einen Windows-Arbeitsplatz angeschlossenen Drucker

In dieser Fallstudie geht es darum, dass iSeries Programme direkt auf einem Drucker drucken sollen, der an einen Arbeitsplatz angeschlossen ist. Für diese Problemstellung gibt es zwei Ansätze:

1.) Auf dem Arbeitsplatz ist iSeries Access installiert. Deshalb wird die iSeries Access Funktion „Emuliertes Drucken“ verwendet.

2.) Auf dem Arbeitsplatz ist kein iSeries Access installiert. Deshalb werden die TCP/IP-Dienste LPR (Line Printer Requester) und LPD (Line Printer Daemon) verwendet.

5. Ergänzung 12/2005

6.6.1 Seite 2

iSeries Access – Emuliertes Drucken

Die Funktion des emulierten Druckens gehört zum Funktionsumfang des Programms PC5250. Emuliertes Drucken beschreibt ein Verfahren, bei dem ein lokaler PC-Drucker als iSeries-Systemdrucker verwendet wird. Definieren wir also einen PC-Drucker als emulierten Drucker für die iSeries, so wird auf der iSeries eine Ausgabewarteschlange für diesen Drucker eingerichtet, so dass ein iSeries -Prozess diesen Drucker verwenden kann.

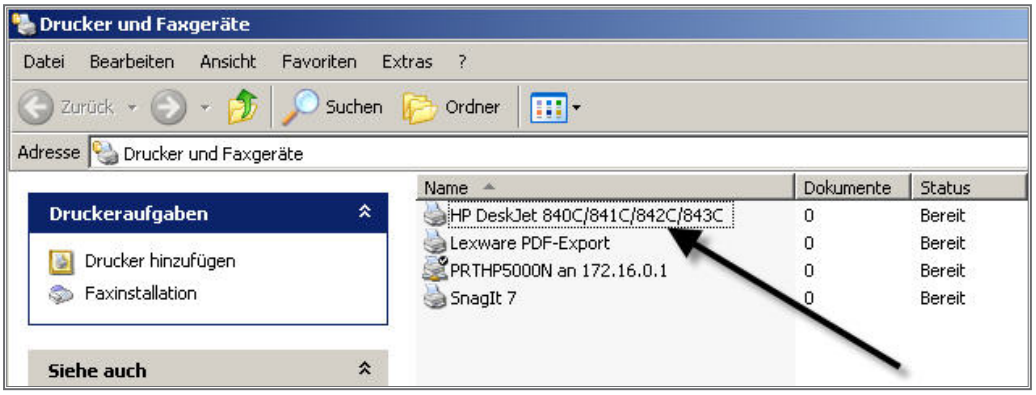
Wesentlich für eine erfolgreiche Installation ist der Sachverhalt, dass an dieser Stelle zwei Druckprozesse miteinander kommunizieren müssen. Einerseits der iSeries-Prozess, der das gesamte Spooling steuert und leitet, andererseits der Windows-Druckprozess, der dann das physische Drucken übernimmt.

Daraus ergibt sich die Voraussetzung, dass auf dem Windows-PC bereits ein Drucker – entweder direkt oder als Netzwerkdrucker – installiert wurde, der dann mit dem iSeries-Druckprozess verbunden werden muss.

Die Konfiguration des emulierten Druckers entspricht vom allgemeinen Vorgehen her der Konfiguration eines Bildschirms.

Im ersten Konfigurationsschritt sehen wir uns den Windows-Arbeitsplatz an, um den als emulierten Drucker zu konfigurierenden lokalen Drucker zu identifizieren. Dazu wechsele ich in die Windows-Drucker anzeige:

Start ➜ Einstellungen ➜ Systemsteuerung ➜ Drucker und Faxgeräte



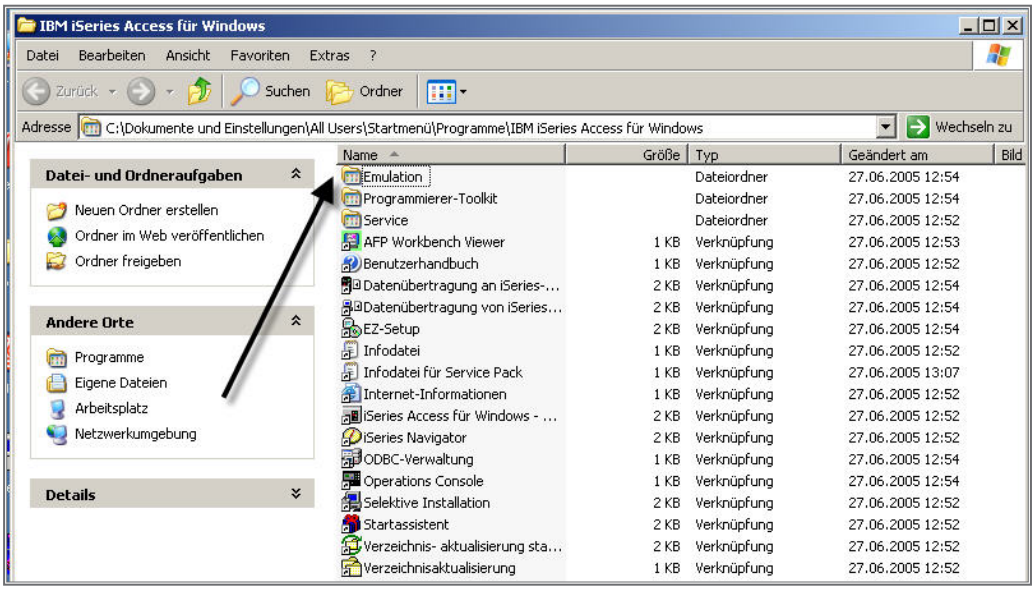
Drucker auswählen

Ich wähle aus den in Windows zur Verfügung stehenden Druckern das Gerät „HP Deskjet 840C“ aus.

Ich beabsichtige also, die iSeries-Ausgaben lokal auf meinen HP Deskjet-Drucker umzuleiten.

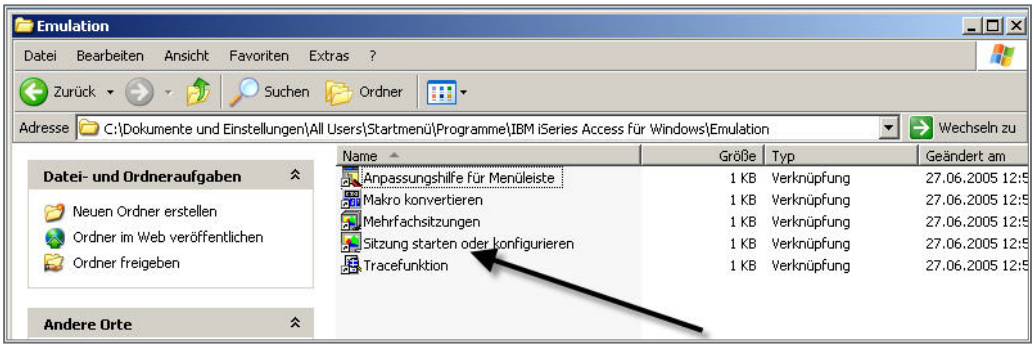
Im nächsten Arbeitsschritt wird im iSeries Access der emulierte Drucker definiert. Dazu wechsele ich in die IBM iSeries Access für Windows-Programmgruppe.

6.6.1.1
Seite 2

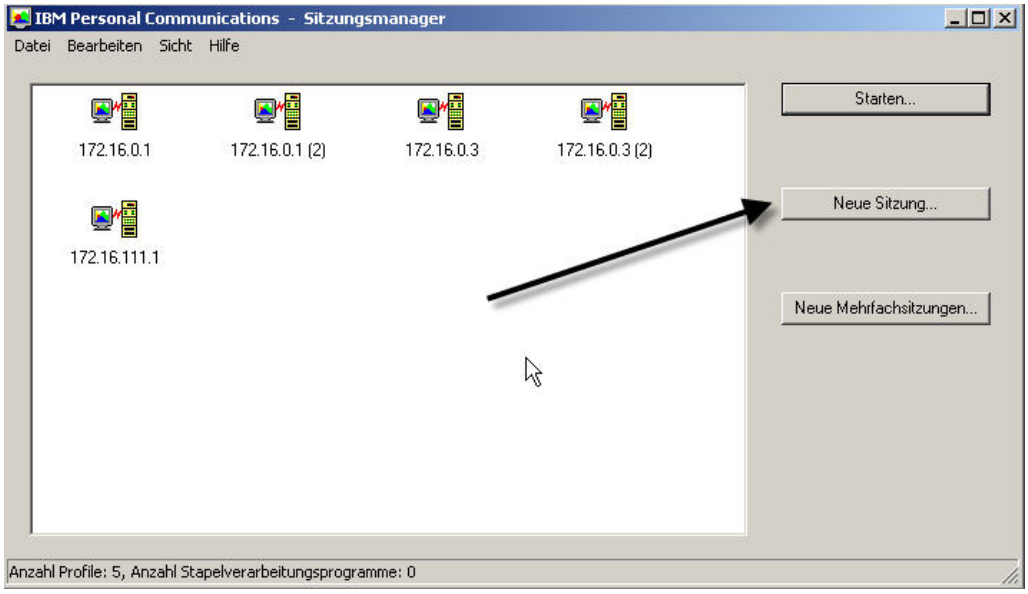


Ins Verzeichnis Emulation wechseln

Danach wechsele ich in das Verzeichnis „Emulation“.

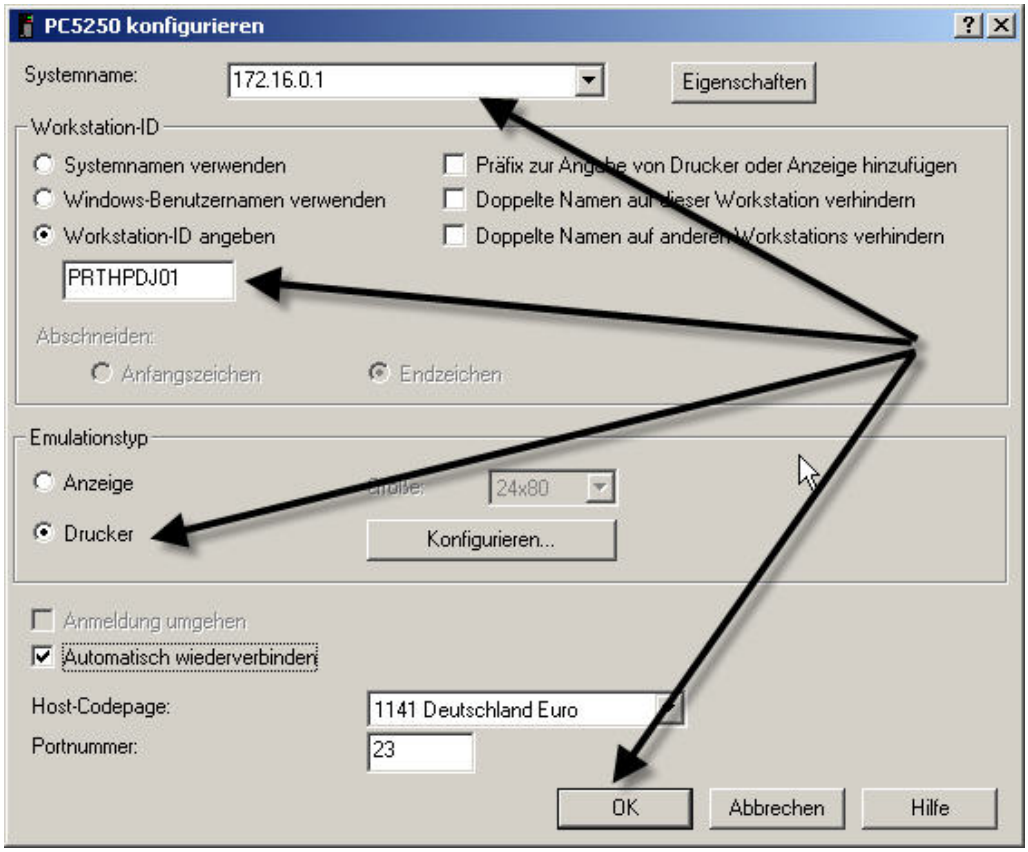


Emulierte Drucker: Beginn der Konfiguration



Neue Sitzung erstellen

Sie beginnt damit, dass ich eine neue Sitzung starte. Im folgenden Konfigurationsdialog werden dann die Verbindungsparameter zu konfigurieren sein.



Verbindungs- und Geräteparameter

Folgende Parameter sind zu versorgen:

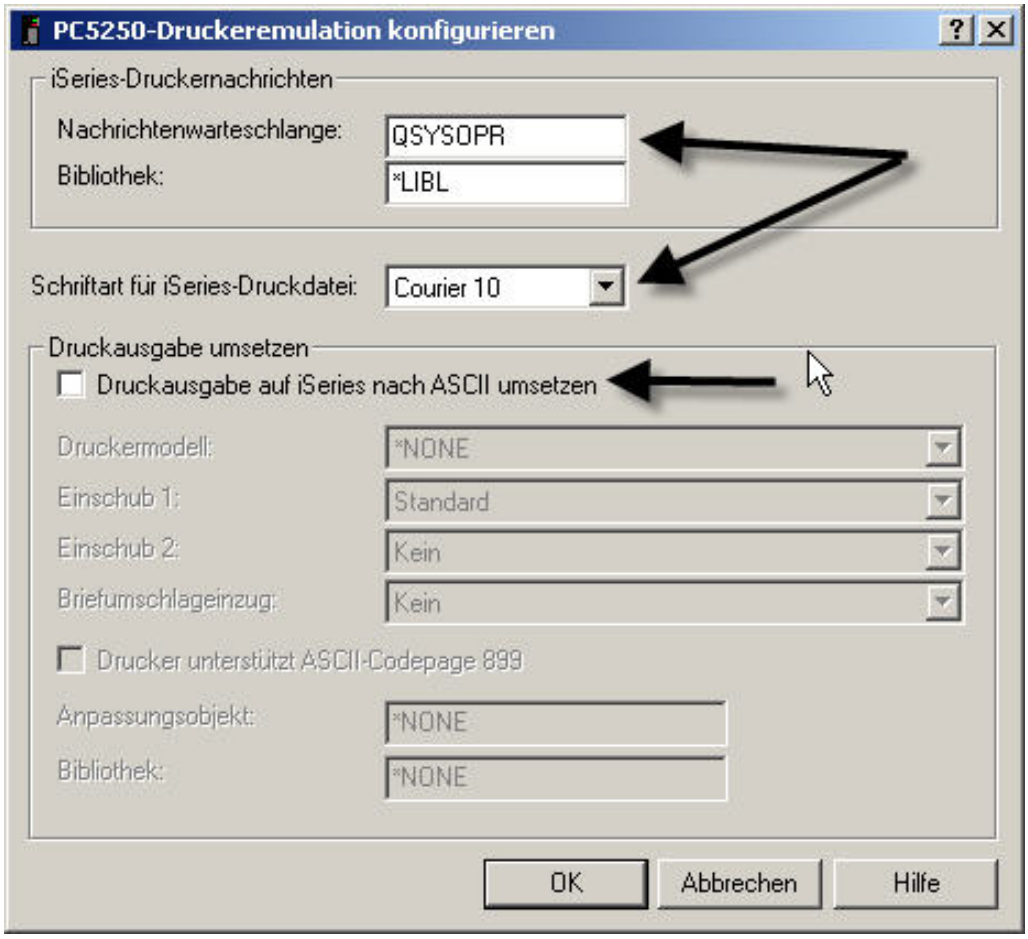
Systemname: Hier trage ich die IP-Adresse der iSeries ein, deren Ausgaben an diese Druckersitzung geleitet werden sollen.

Workstation-ID angeben: Hier trage ich den Namen der auf der iSeries zu erstellenden Ausgabewarteschlange/Druckers ein.

Drucker: Diesen Parameter markiere ich, weil ich einen emulierten Drucker erstellen will.

Betätigen Sie dann den Knopf „Konfigurieren...“ .

6.6.1.1
Seite 4



iSeries-Druckangaben einstellen

An dieser Stelle können Sie folgende Werte einstellen:

iSeries-Druckernachrichten
An welche Nachrichtenwarteschlange sollen Nachrichten, die von diesem Drucker abgehen, gesendet werden? Defaulteinstellung ist hier die Nachrichtenwarteschlange des QSYSOPR der iSeries. Da es sich hier um einen Arbeitsplatz-bezogenen Drucker handeln kann, sollten Sie überlegen, ob Sie an dieser Stelle nicht die Nachrichtenwarteschlange dieses Bildschirms eintragen.

Schriftart für iSeries-Druckdatei
Das ist die Standardschriftart, die für diesen Druckprozess bei iSeries eingesetzt wird.

Druckausgabe umsetzen
Sollten Sie diese Option markieren, dann übernimmt die iSeries weiterführende Steuerungsfunktionen.

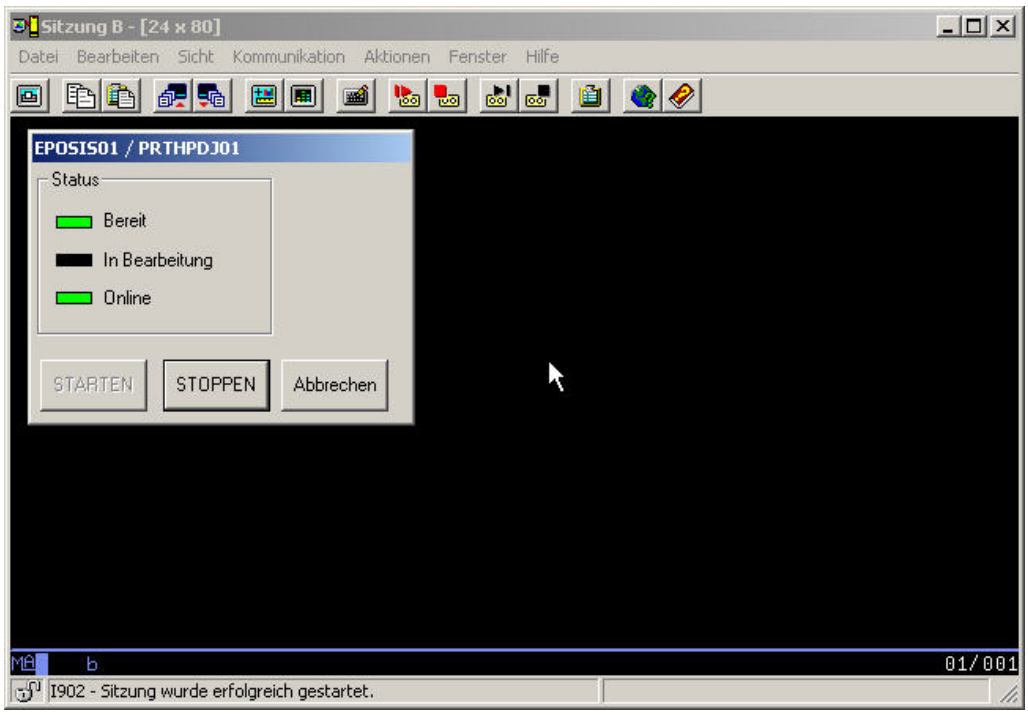
In diesem Bereich stellen Sie ein, wo die Zeichenumsetzung stattfinden soll; auf dem PC oder auf der iSeries. Markieren Sie die Option „Druckausgabe im System iSeries umsetzen“, so übernimmt die iSeries für diesen Drucker die gesamte Zeichenumsetzung von EBCDIC nach ASCII. Aber Vorsicht, sollten

Sie diese Option großflächig, also nicht nur bei einem Drucker, einstellen, so wird die Performance der iSeries stark davon beeinflusst. In einem Testlauf sollten Sie vorher die Auswirkung auf den restlichen Systembetrieb überprüfen.

Ansonsten lässt sich an dieser Stelle einstellen, welcher iSeries-Druckertyp emuliert wird und welche Einzüge an diesem Drucker zur Verfügung stehen.

Bestätigen Sie die hier getätigten Einstellungen mit „OK“ und beenden Sie den Gesamtdialog mit „OK“.

Nach einiger Zeit erscheint dann folgendes Fenster:

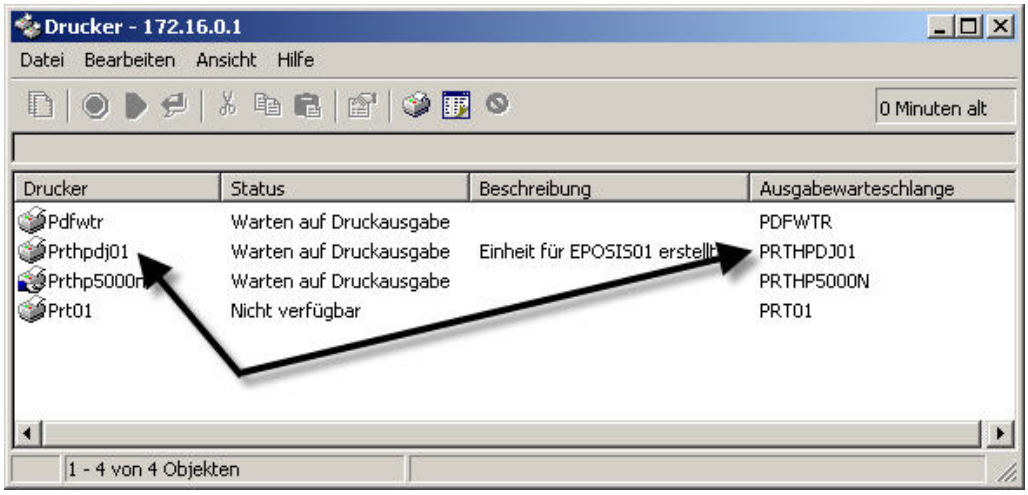


Das Bedienungsfeld des emulierten Druckers

Sie sehen hier das Bedienungsfeld des emulierten Druckers, das im Wesentlichen so aufgebaut ist, wie das Bedienungspanel eines Druckers. Sie können hier den Druck starten bzw. anhalten und sehen auch, ob gerade ein Druckjob bearbeitet wird.

Für diesen emulierten Drucker ist auf der iSeries eine Druckereinheit und Ausgabewarteschlange erzeugt worden.

6.6.1.1
Seite 6



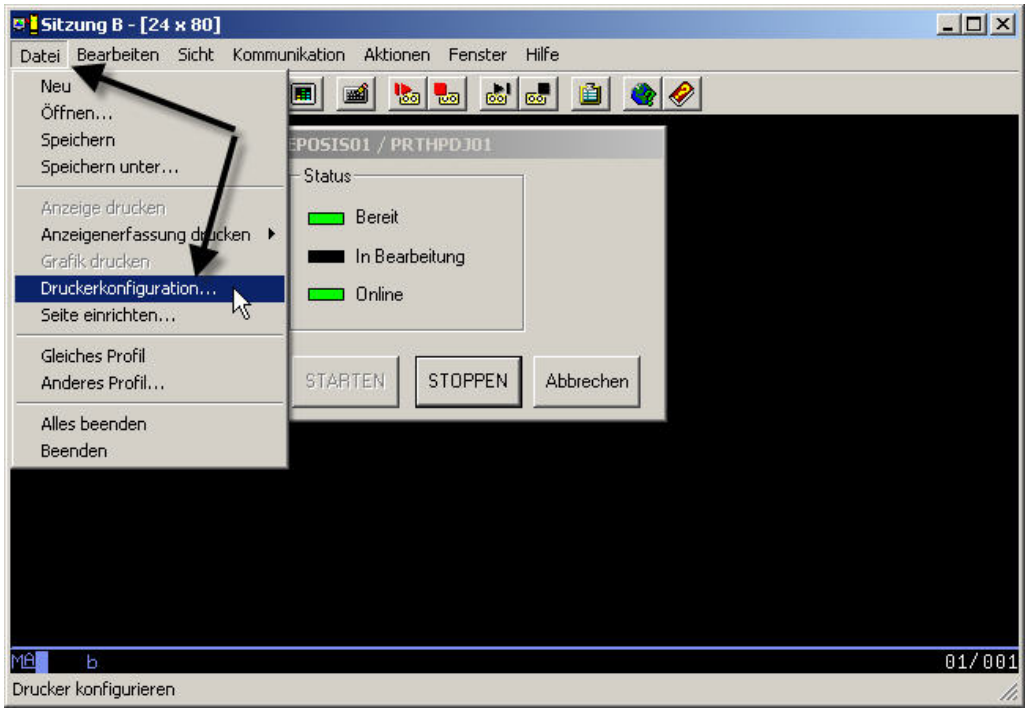
Anzeige der iSeries Drucker

In dem oben gezeigten Bild sehen Sie die auf der iSeries neu erstellten Druckobjekte.

Damit ist die Konfiguration von iSeries beendet.

Der nächste Schritt ist wichtig, denn jetzt muss die Verbindung zum Windows-Druckprozess hergestellt werden.

Hinter dem Bedienfeld des Druckers sehen Sie einen leeren iSeries-Bildschirm. Die in diesem Bereich dargestellten Menüs dienen der Konfiguration des emulierten Druckers. Sie gehen in das Menü „Datei“ und wählen den Menüpunkt „Druckerkonfiguration“.



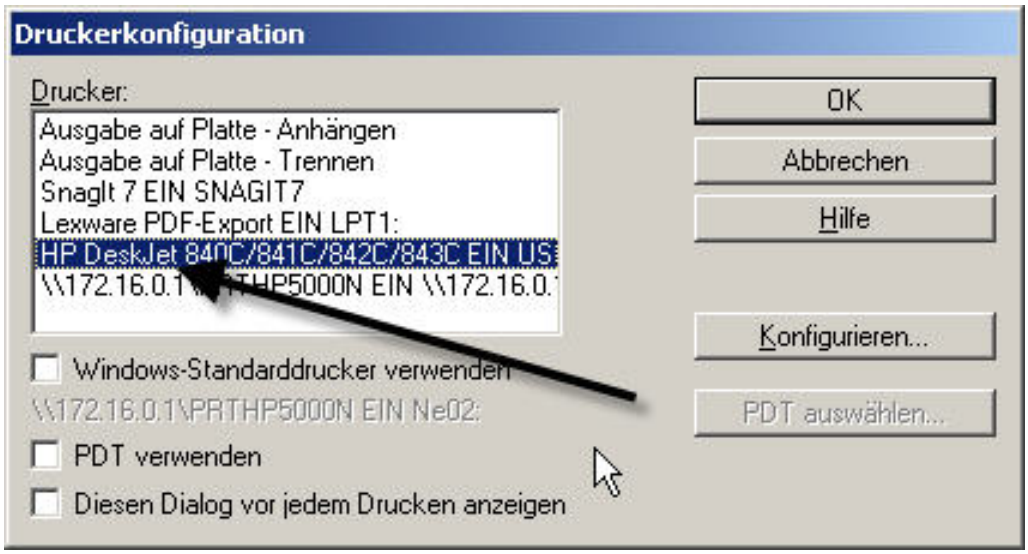
Verbindung zum Windows-Drucker definieren

Die Verbindung lässt sich auf zwei Arten konfigurieren.

1.) Wählen Sie im folgenden Dialog den Windows-Drucker aus. In diesem Dialogfenster werden alle mit diesem Windows-PC verbundenen Drucker angezeigt:

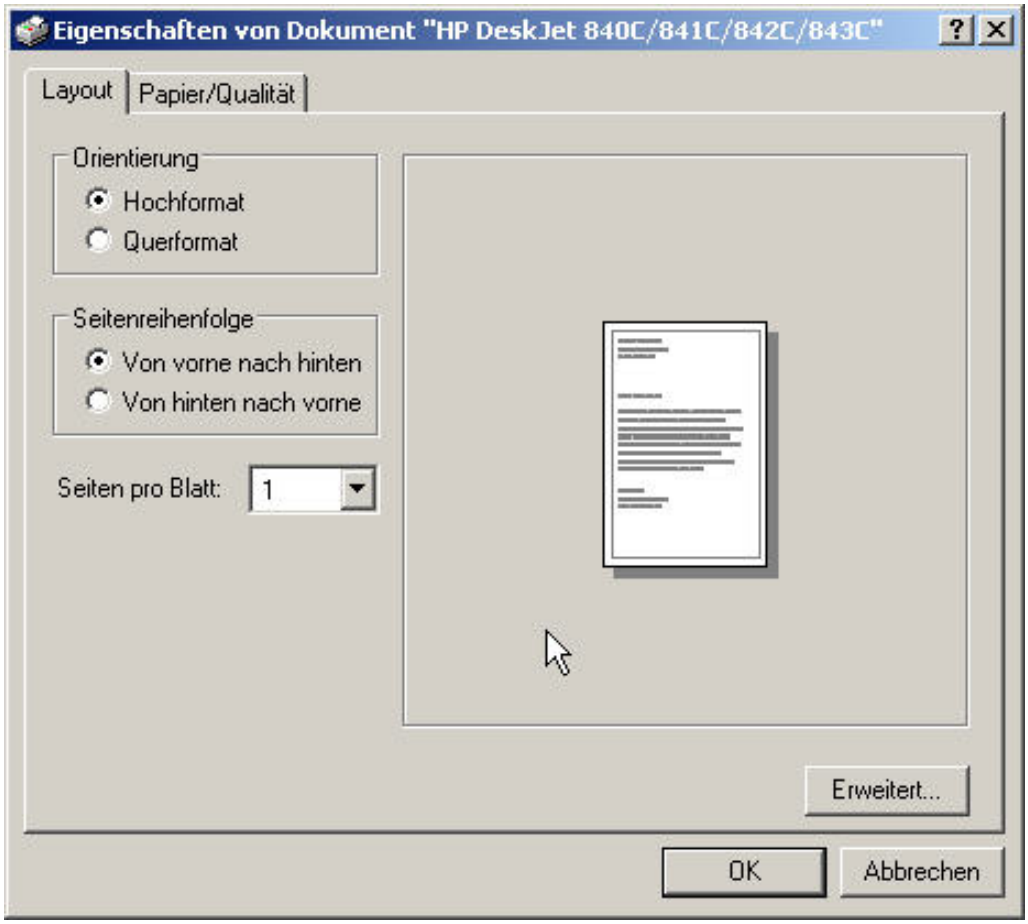
- Lokale Drucker: erkennbar an der Zuordnung zur Hardwareschnittstelle – LPT1, LPT2 etc.
- Netzwerkdrucker: erkennbar an der Netzwerkressourcenzuordnung mittels des UNC-Namens – \\System\Ressource

In meinem Fall sollte der lokale HP DeskJet 840C angeschlossen werden.



Windows-Drucker zuordnen

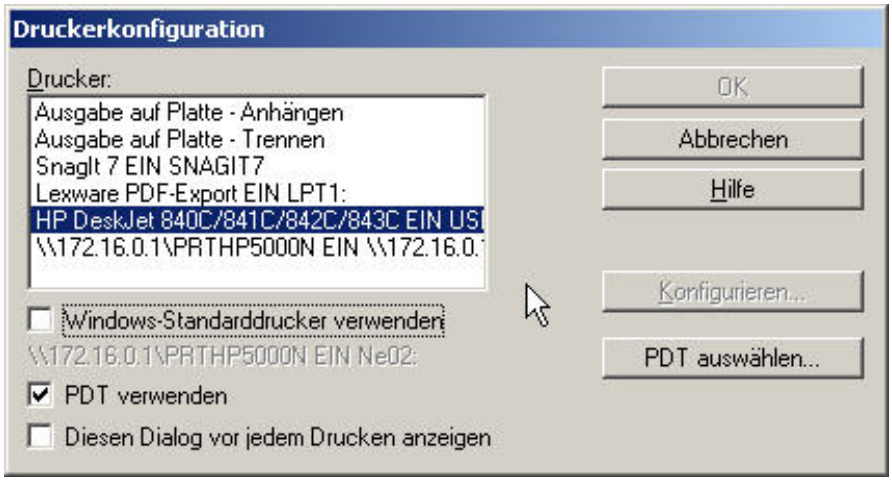
6.6.1.1
Seite 8



HP Deskjet konfigurieren

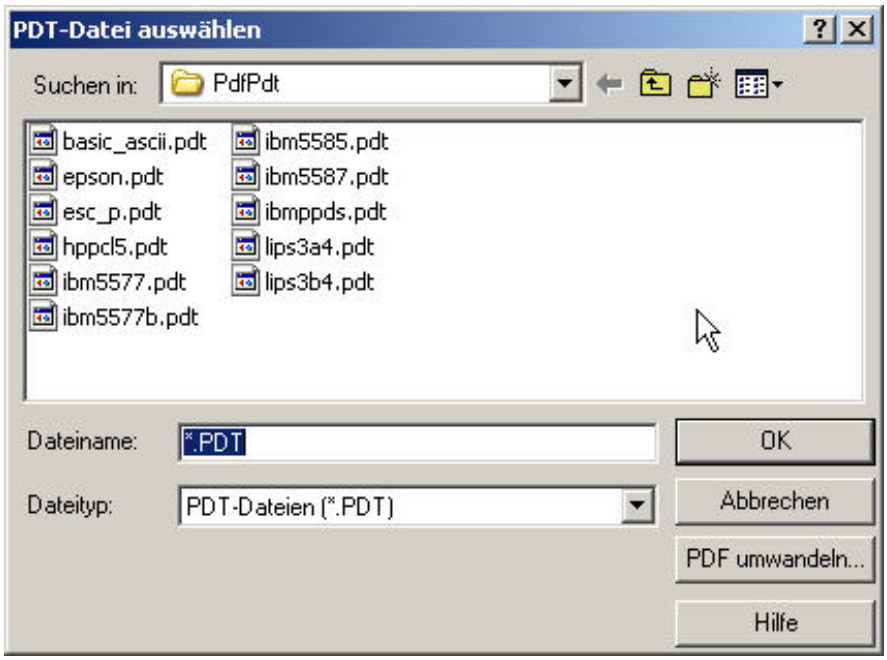
Mit Hilfe des Button „Konfigurieren“ lassen sich dann die Grundeinstellungen dieses lokalen Druckers einstellen. Der hier gezeigte Dialog ist typisch für den genannten HP-Drucker. Verwenden Sie den Drucker eines anderen Herstellers, so wird unter Umständen ein anderes Dialogfenster angezeigt.

2.) Wählen Sie im Dialog „Druckerkonfiguration“ den HP-Deskjet aus.



Mit Printer Definition Tables (PDT) arbeiten 1

Aktivieren Sie die Option „PDT verwenden“.



Anzeige der Printer Definition Tables

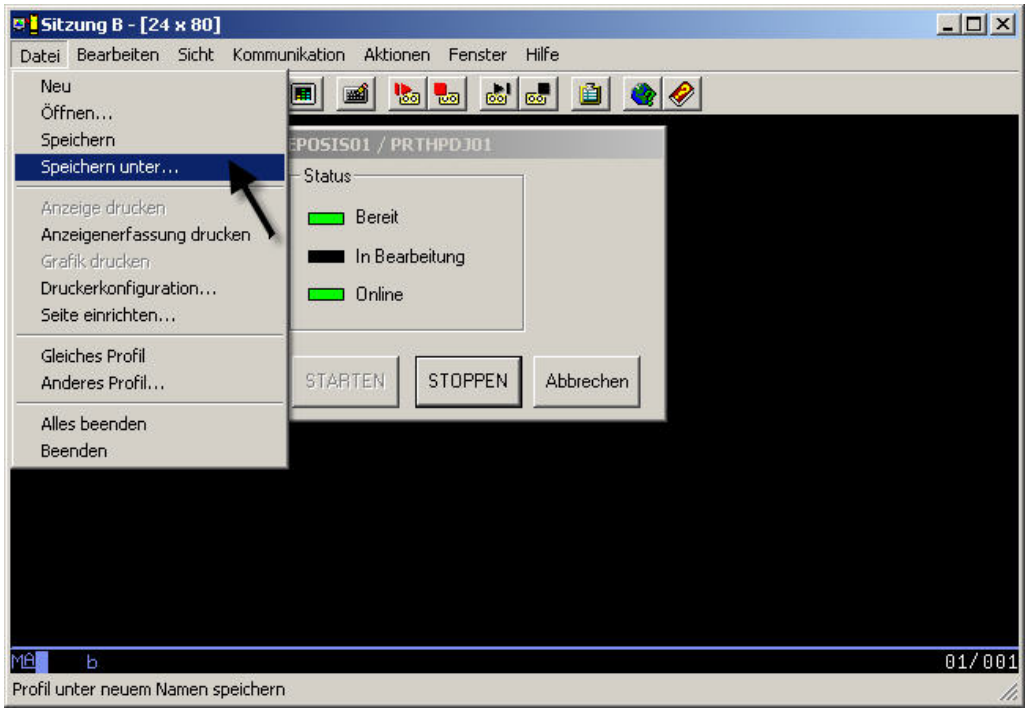
Was verbirgt sich hinter dieser Auswahl? Mit dem iSeries Access werden eine Reihe von so genannten Printer Definition Tables (PDT) ausgeliefert, die iSeries Access-spezifische Druckerdefinitionstabellen enthalten. Diese Tabellen steuern unter Umständen einen emulierten Drucker effizienter als es die normale lokale Druckerkonfiguration ermöglicht.

Betätigen Sie danach den Knopf „PTD auswählen...“ und gehen Sie in dem oben angezeigten Dialog weiter. In dem Fenster „PDT-Datei auswählen“ suchen Sie jetzt eine geeignete Printer Definition Table aus.

Bestätigen Sie die Auswahl mit „OK“.

6.6.1.1
Seite 10

Unabhängig davon, welchen Weg Sie gewählt haben sollten Sie jetzt, wie schon bei den emulierten Bildschirmen, die Konfiguration in einem Emulationsprofil sichern.



Die Druckeremulation sichern

Dazu verwenden Sie den von der Bildschirmemulation bekannten „Speichern als...“-Dialog.

Haben Sie für den Drucker ein Symbol erzeugt, so können Sie dieses auf dem Desktop oder in einem Ordner des Windows-PC ablegen.

Windows Desktop mit Druckeremulations-Symbol

Sie sehen an dieser Stelle das für die Druckeremulation auf dem Windows-Desktop abgelegte Icon.

Danach sollten Sie einen Probedruck vornehmen, um die Installation/Konfiguration zu überprüfen.

Fazit: Die soeben erstellte Konfiguration ist lange Jahre eine Standardlösung für die oben beschriebene Problemstellung gewesen und wird sicherlich auch heute noch gut funktionieren. Jedoch muss ich anmerken, dass die Lösung mehrere Merkmale hat, die es zu bedenken gilt.

Auf dem Windows-PC, auf dem die Funktion ausgeführt werden soll muss iSeries Access für Windows installiert sein. Verwenden Sie andere Emulationsprodukte wird eine Funktion „emuliertes Drucken“ nicht notwendigerweise zur Verfügung gestellt.

Die Funktion kann nur ausgeführt werden, wenn der Windows-PC und eine Arbeitssitzung gestartet ist. Soll ein Windows-Server diese Funktion als Print-Server ausführen, so muss sichergestellt sein, dass der Windows-PC gestartet wurde und weiterhin muss garantiert sein, dass eine Anmeldung an diesem Server erfolgt sein muss.

Ein positives Merkmal ist, dass diese Lösung der iSeries die Konvertierungslast von EBCDIC in der Client-spezifischen Zeichensatz- und Druckersprache abnehmen kann, sofern Sie eine Client-seitige Datenstromumsetzung eingestellt haben.

6.6.1.1
Seite 12

Drucken mit LPD und LPR

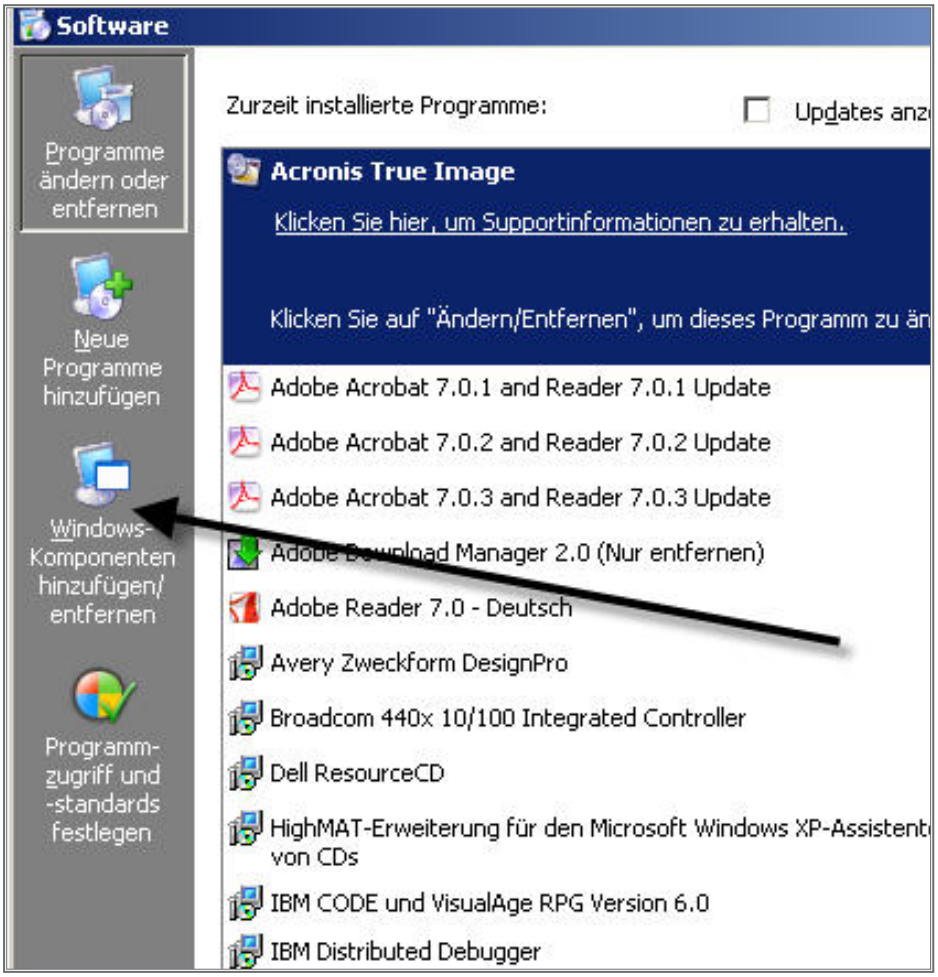
Eine weitere Lösung für das oben dargestellte Problem liegt in der Anwendung eines Lineprinter Daemon (LPD) unter Windows und eines Line Printer Requesters (LPR) unter i5/OS. Windows wird als TCP/IP-basierter Print-Server konfiguriert, während die iSeries als TCP/IP-basierter Druck-Client konfiguriert wird.

Die Windows-Konfiguration

Wurde ein Windows-PC mit einer Standardinstallation von Windows versehen, so beinhaltet diese Installation keinen LPD. Der Windows LPD-Dienst ist jedoch auf der Windows Installations-CD vorhanden, so dass Sie diese für die Umsetzung dieser Lösung benötigen.

Um den Windows-basierten LPD-Dienst zu installieren, gehen Sie in den Windows Softwareinstallationsdialog:

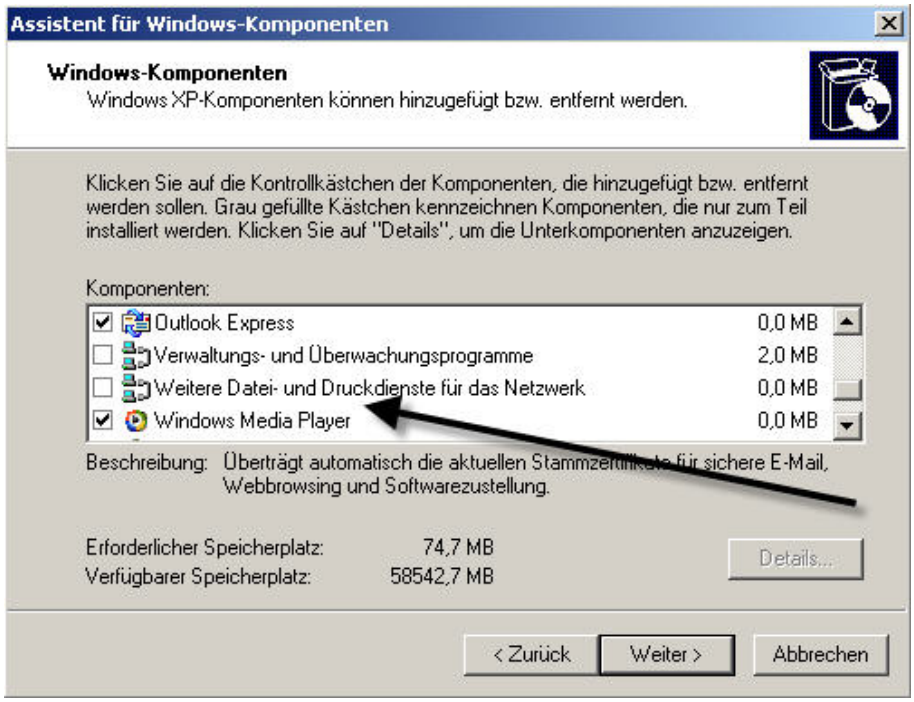
Start ➜ Einstellungen ➜ Systemsteuerung ➜ Software ➜ Windows Komponenten hinzufügen/entfernen



Windows-Komponenten hinzufügen

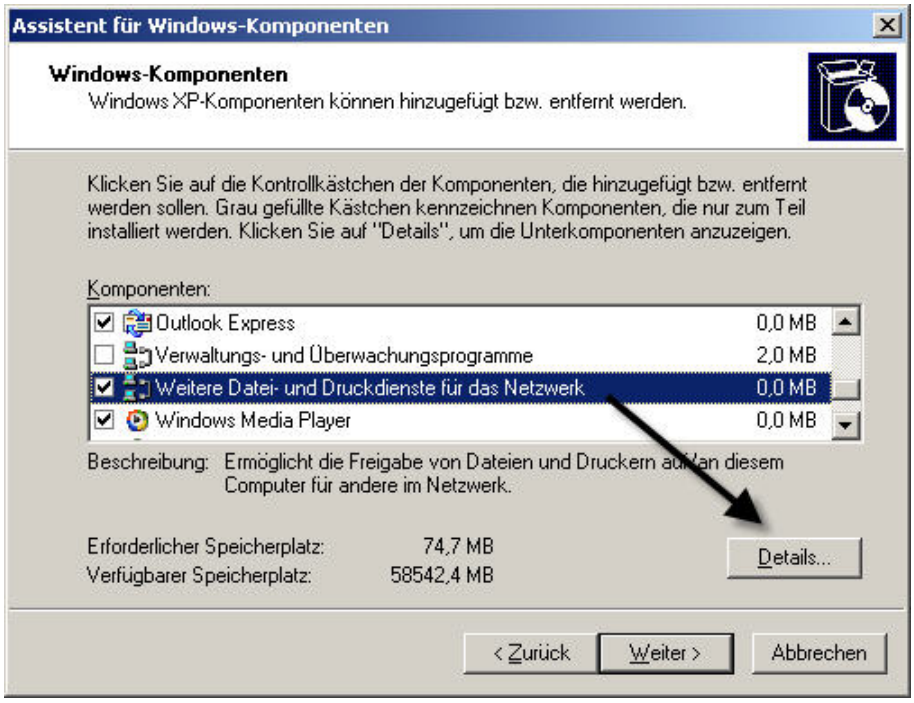
6.6.1.2
Seite 2

Öffnen Sie „Windows-Komponenten hinzufügen/entfernen“.



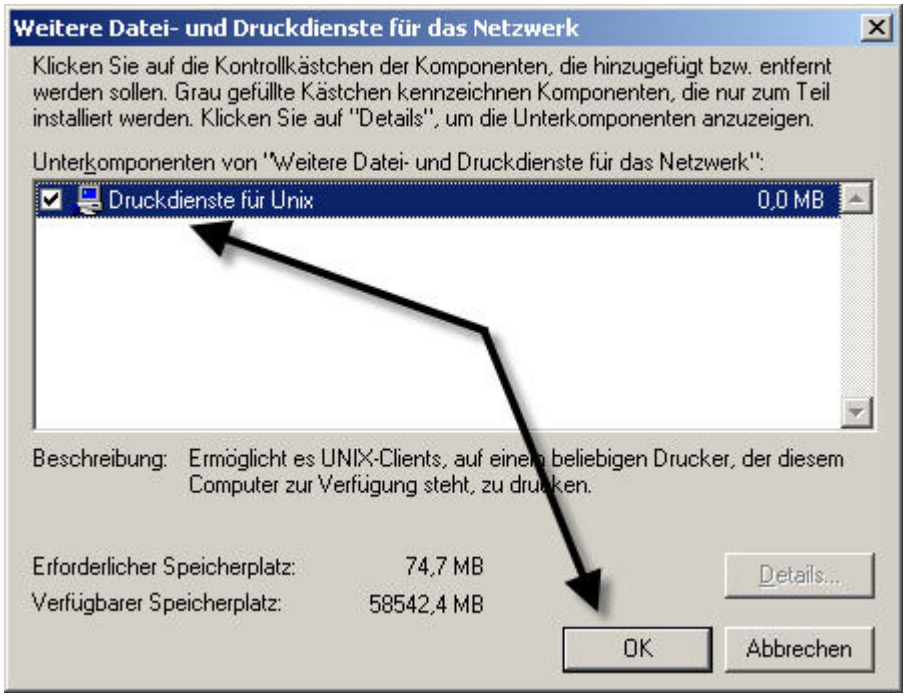
Weitere Datei- und Druckdienste….

Markieren Sie den Punkt „Weitere Datei- und Druckdienste für das Netzwerk“ und drücken Sie „Details“.



Weitere Datei- und Druckdienste…. Details

Jetzt öffnet sich folgender Dialog.



Druckdienste für Unix

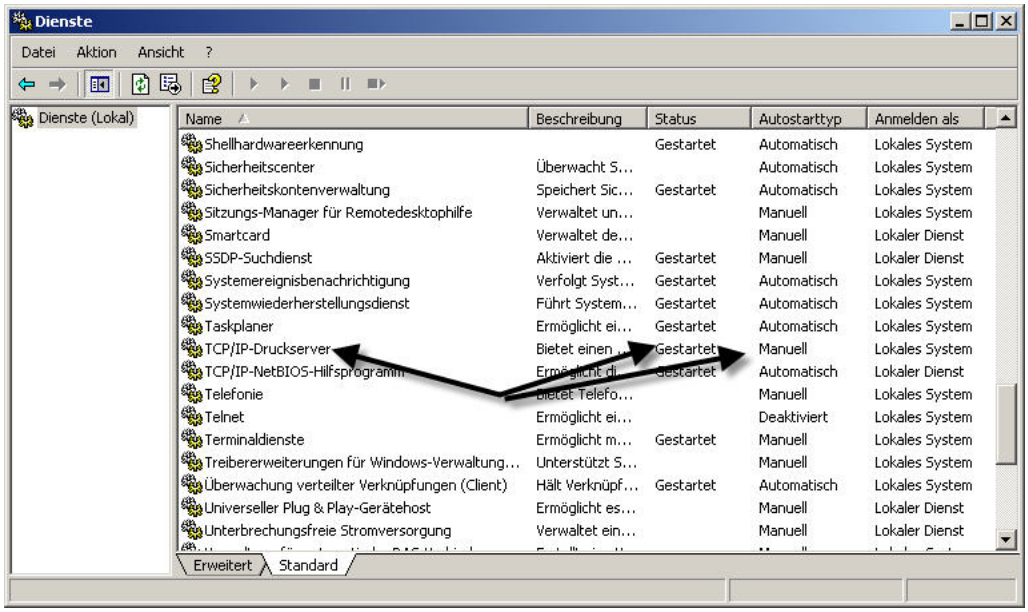
Wählen Sie die Druckdienste für Unix aus und bestätigen Sie den Dialog mit „OK“.

Nach der erfolgreichen Installation finden Sie in den Windows-Diensten einen LPD.

Öffnen Sie dazu die Dienste-Verwaltung des Windows:

Start ➜ Einstellungen ➜ Systemsteuerung ➜ Software ➜ Verwaltung ➜ Dienste

6.6.1.2
Seite 4

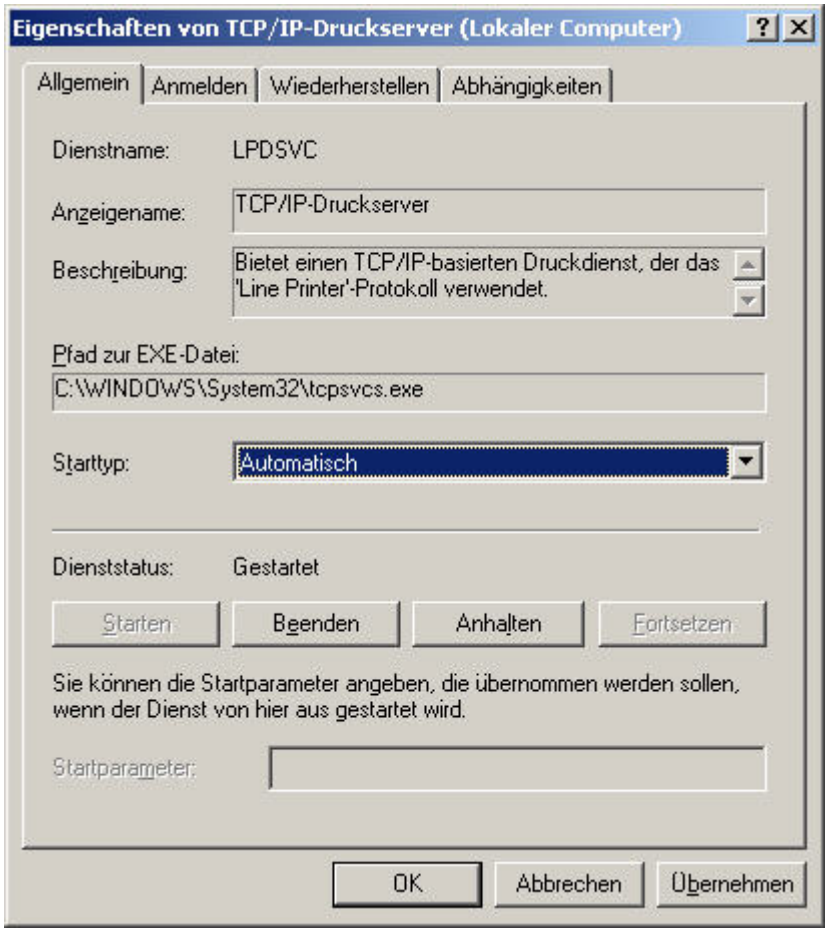


TCP/IP-Druckserver der Windows LPD

Der Windows LPD wird in der Dienste-Sicht als TCP/IP-Druckserver aufgeführt. Wie Sie sehen, wurde der Dienst nach der Installation automatisch gestartet. Beachten Sie jedoch die Splate-Autostarttyp. Hier erkennen Sie, dass der Windows TCP/IP-Druckserver beim erneuten Start des Betriebssystems nicht automatisch mitgestartet wird.

Damit dies geschieht, muss die Startart auf automatisch gesetzt werden.

Mit der rechten Maustaste fordern Sie das Kontextmenü für diesen Dienst an und bearbeiten Sie sodann die Eigenschaften des Druckserverdienstes.



TCP/IP-Druckserver-Eigenschaften

Ändern Sie in diesem Dialog den Starttyp auf „Automatisch“ und bestätigen Sie den Dialog mit „OK“.

Damit ist die Windows-Konfiguration abgeschlossen.

Die iSeries-Konfiguration

Auf der iSeries muss jetzt eine Remote-Ausgabewarteschlange erstellt werden, die auf einen Windows-Drucker zeigt.

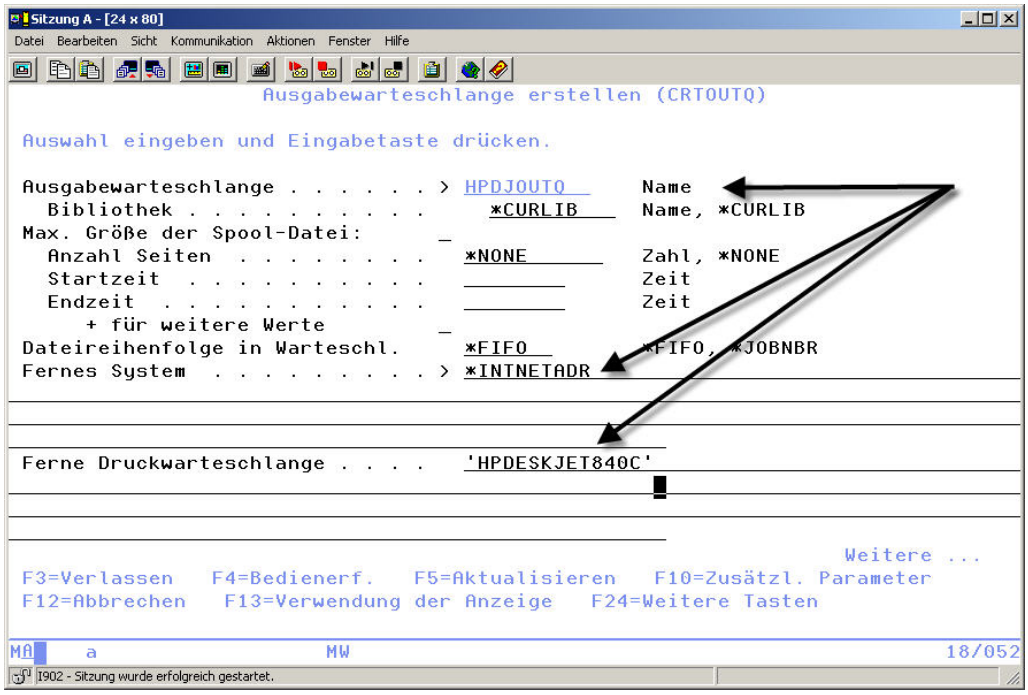
Dazu benötige ich die IP-Adresse des Windows Client, über den die iSeries drucken soll, sowie den Namen des Druckers.

Der Name des zu verwendenden Druckers ist in meinem Fall HPDESKJET840C.

Die IP-Adresse des Windows-PC ist 172.16.157.11

Ich rufe in einer iSeries-Bildschirmsitzung den Befehl CRTOUTQ auf und drücke die F4-Taste.

6.6.1.2
Seite 6



CRTOUTQ – Parameter 1

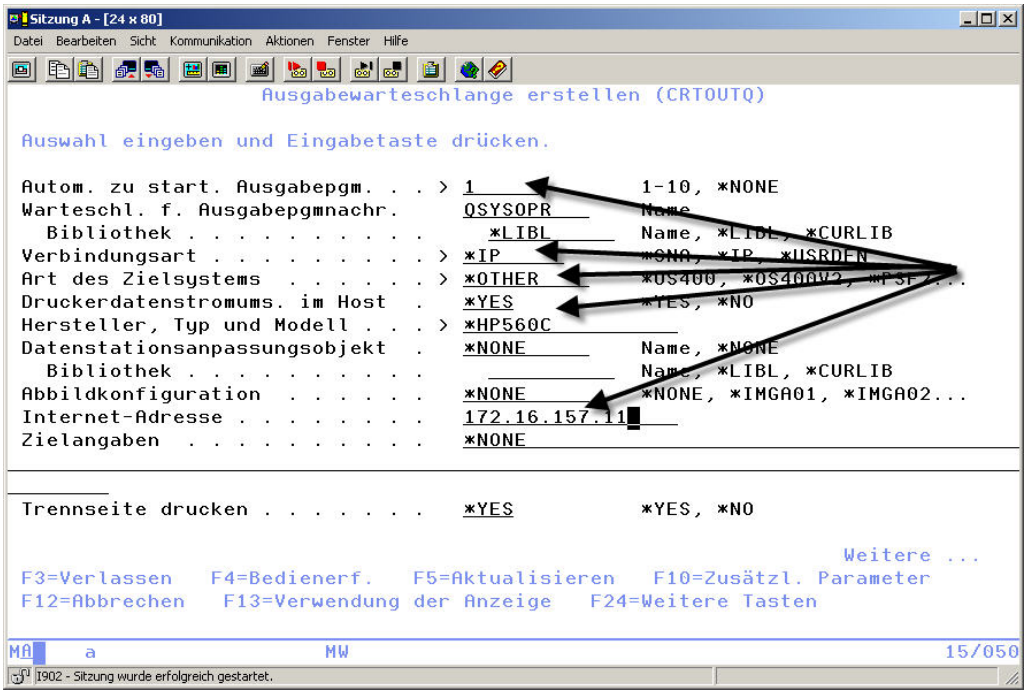
Folgende Parameter sind zu versorgen:

Ausgabewarteschlange: HPDJOUTQ, der Name der zu erstellenden Ausgabewarteschlange.

Fernes System: *INTNETADR, Sie beabsichtigen den fernen Druckprozess IP-basiert zu adressieren.

Ferne Druckwarteschlange: HPDESKJET840C, der Name des Windows-Druckers. Achtung! Sollte der Name des Windows-Druckers Kleinbuchstaben enthalten, so sollten Sie den Namen des Windows-Druckers wie gezeigt, in Hochkommata eingeben. Es ist wichtig, dass der hier eingegebene Name absolut mit dem Namen in der Windows-Umgebung übereinstimmt.

Auf der nächsten Seite sind folgende Parameter zu versorgen:



CRTOUTQ – Parameter 2

Autom.zu start. Ausgabeprogramm: 1

Verbindungsart: *IP

Art des Zielsystems: *OTHER

Druckerdatenstrumums. In Host: *YES – Die iSeries soll die Konvertierung in den Drucker-spezifischen Datenstrom realisieren.

Hersteller, Typ und Modell: *HP560C – Da es für den HP DeskJet 840C keinen spezifischen Treiber gibt, versuche ich einmal einen ähnlichen HP-Treiber. Sollte dieser nicht funktionieren, so werde ich weitere Treiber von Druckern probieren, die der HP Deskjet emulieren kann.

Internet-Adresse: 172.16.157.11

Fazit: Die soeben erstellte Konfiguration weist folgende Merkmale auf:

Es muss unter Windows keine spezielle Emulationssoftware installiert werden, lediglich der LPD-Serverdienst muss unter Windows nachinstalliert werden.

Dieser Serverdienst steht auch dann zur Verfügung, wenn noch keine Anmeldung am Windows erfolgt ist.

Der PC führt keine Datenstromumsetzungen durch.

6.6.1.2
Seite 8

7 Kapitel

Client-Connectivity / Netzwerk

7

Seite 2

Client Connectivity – IBM iSeries Access

Hinter dem Produktnamen iSeries Access verbirgt sich eine ganze Reihe von einzelnen Komponenten, die in diesem Kapitel erläutert werden sollen.

Folgende Komponenten sind derzeit Bestandteil des iSeries Access:

- iSeries Access Familie: 5722-XW1 mit den Komponenten *Base und Product enablement Support (1)
- iSeries Access für Windows: 5722-XE1 mit der Komponente *BASE
- iSeries Access für Web: 5722-XH2 mit der Komponente *BASE
- iSeries Access for Wireless: 5722-XP1 mit der Komponente *BASE

Weitere Produkte, die ebenfalls zur iSeries Access-Lizenz gehören, jedoch nicht als iSeries Lizenzprogramme zur Installation gehören, sind:

- HATS LE: Host Access Transformation Server Limited Edition
- IBM Host Publisher

Grundwerk 10/2004

7

Seite 4

IBM System i Access für Windows

System i Access für Windows ist der Nachfolger von iSeries Access. Von Version V5R4 auf Version V6R1 sind – oberflächlich betrachtet – nur wenige Funktionserweiterungen vorgenommen worden. Jedoch hat es unter der Oberfläche eine Reihe von Erweiterungen und Verbesserungen gegeben, die dieses Produkt über ähnliche Produkte anderer Anbieter deutlich hinausragen lässt.

Vor der Nutzung der Funktionen steht jedoch eine erfolgreiche Installation. Die Installation umfasst ggf.

- die Installation der System-i-Access-Schnittstelle (V5R4:5722XW1, V6R1:5722XW1, V7R1:5771XW1),
- die Installation des System i Access für Windows (V5R4:5722XE1, V6R1:5722XE1, V7R1:5771XE1) und
- die Installation der Client-Funktionen auf einem Windows PC.

23. Ergänzung 6/2010

7.1
Seite 2

Die Installation der System-i-Access-Lizenzprogramme

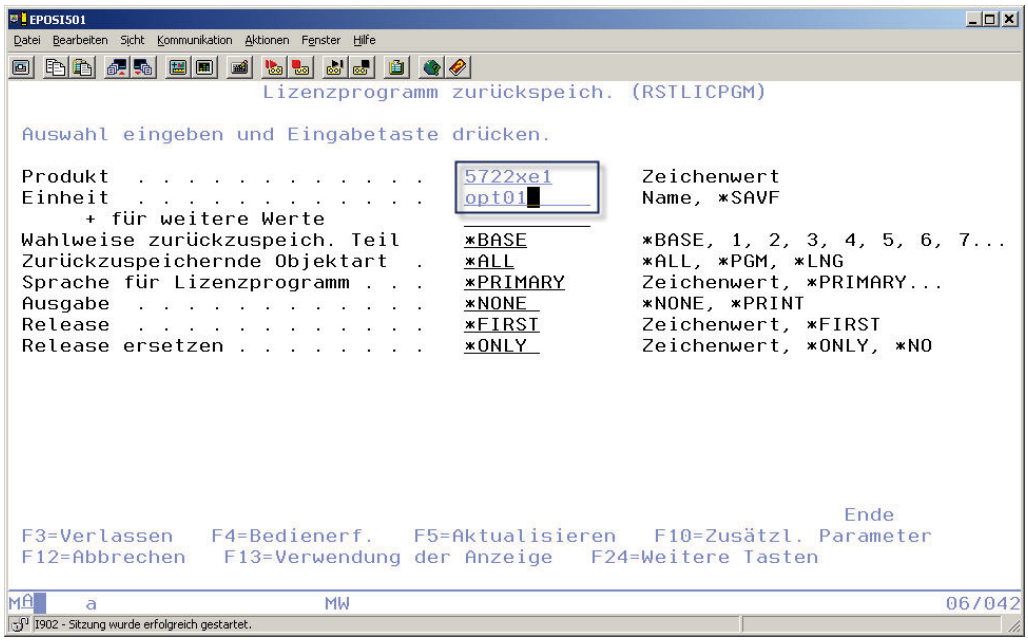
Bevor Sie mit der Installation der System-i-Access-Programme beginnen, sollten Sie sicherstellen, auf welchem PTF-Stand sich Ihr System befindet. Erfragen Sie dazu den aktuellen CUM-PTF-Stand beim IBM Software Service oder im Web unter folgender URL:

http://www-947.ibm.com/systems/support/i/fixes/index.html

und vergleichen Sie die genannte PTF-Nummer mit der PTF-Nummer des zuletzt von Ihnen eingespielten PTF-Mediums. Verwenden Sie hierzu den Befehl

```
WRKPTFGRP
```

Nachdem dies erledigt ist, beginnt die eigentliche Installation. Mit dem Befehl RSTLICPGM installieren Sie das Lizenzprogramm V5R4:5722XE1 oder V6R1:5722XE1 bzw. V7R1:5771XE1. Dieses Lizenzprogramm enthält alle PC-Dateien, die für die Installation eines Clients (!) notwendig sind.



RSTLICPGM – System i Access für Windows installieren

7.1.1
Seite 2

Die Installationseinheit wurde hier mit OPT01 angegeben, da die Installation vom CD-ROM-Laufwerk aus erfolgt. Überprüfen Sie den Ausgang der Installation mit dem Befehl

```
V5R4: CHKPRDOPT PRDID(5722XE1)
```

oder

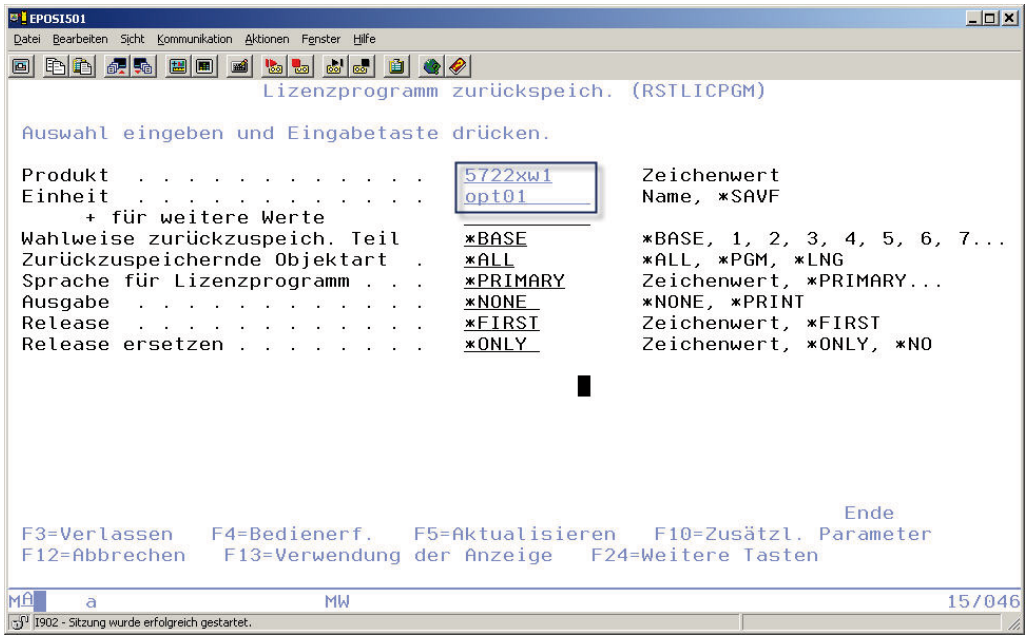
```
V6R1: CHKPRDOPT PRDID(5761XE1)
```

bzw.

```
V7R1: CHKPRDOPT PRDID(5771XE1)
```

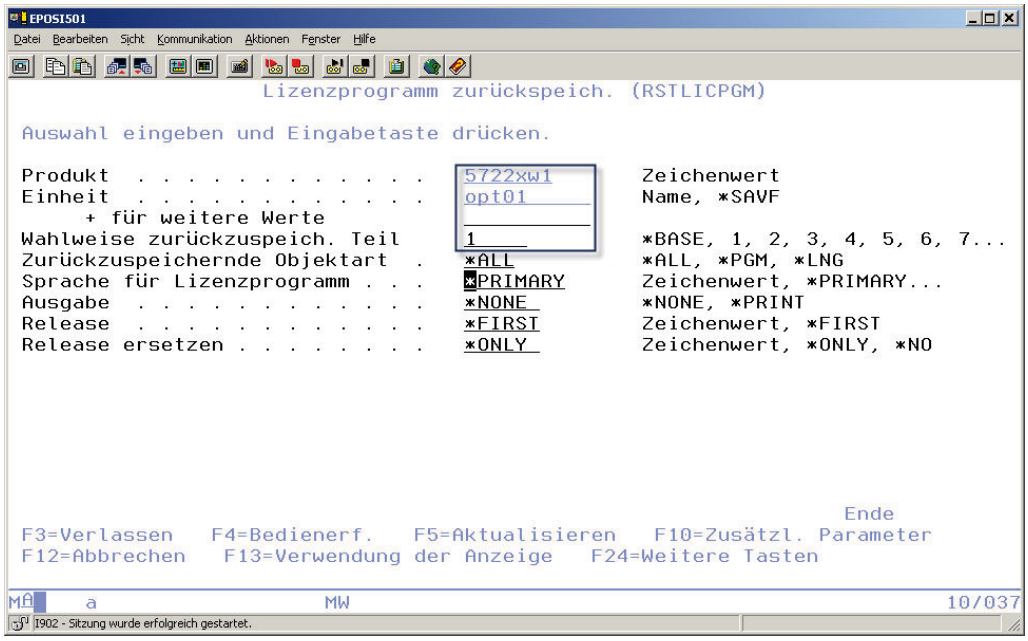
Nach der Ausführung des Befehls sollten Sie in der Fußzeile Ihres System-i-Bildschirms die Meldung „Mit CHKPRDOPT keine Fehler festgestellt“ sehen.

Vergewissern Sie sich, dass Sie die für diesen Client erforderliche und aktuelle (!) System-i-Schnittstelle V5R4:5722XW1 oder V6R1:5722XW1 bzw. V7R1:5771XW1 installiert haben. Sollte das nicht der Fall sein, so müssen Sie dieses Lizenzprogramm ebenfalls installieren. Verwenden Sie hierzu den Lizenzprogramm-Installationsdialog der System i oder den Befehl RSTLICPGM (Lizenzprogramm zurückspeichern).



*System-i-Access-Schnittstelle installieren: *BASE*

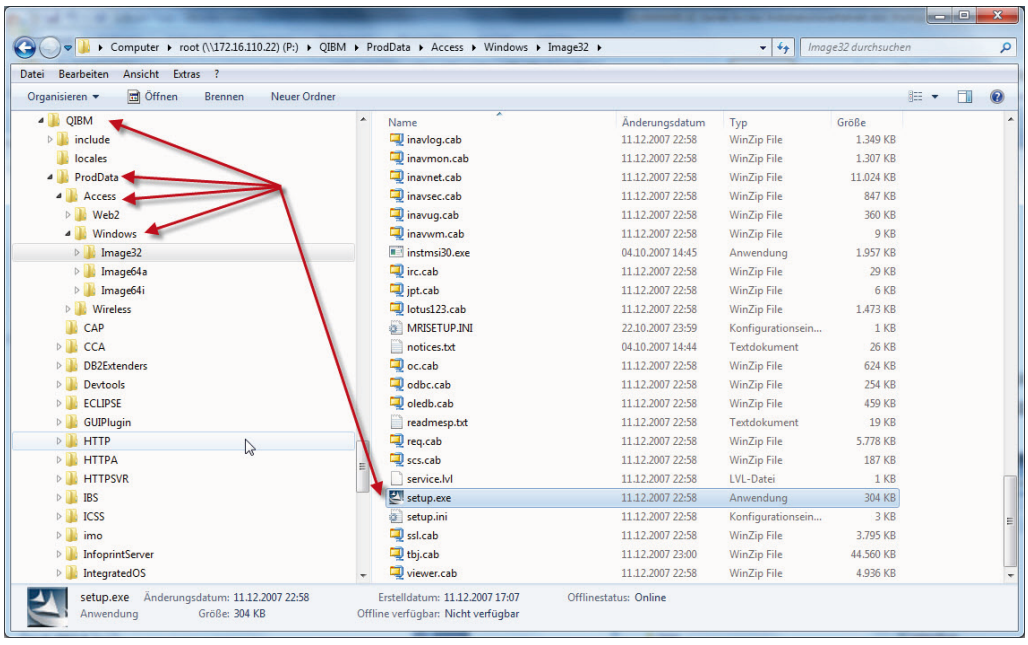
Verfahren Sie gleichermaßen mit der Komponente 1 von V5R4:5722XW1, V6R1:5722XW1 bzw. V7R1:5771XW1.



System-i-Access-Schnittstelle installieren: Komponente 1

Vergewissern Sie sich auch hier mit dem i-Befehl CHKPRDOPT, ob die Installation wirklich fehlerfrei realisiert wurde.

Die Installation des System i Access für Windows erfolgt in den folgenden Pfad des Integrated File Systems: \QIBM\ProdData\Access\Windows



Das Installationsverzeichnis

7.1.1
Seite 4

Im Unterverzeichnis „Windows“ befinden sich die Installationsverzeichnisse. Darunter finden Sie folgende Verzeichnisse:

- Image32,
- Image64a und
- Image64i.

Diese Verzeichnisse enthalten Installationsimages des System i Access für Windows für die unterschiedlichen 32-bit und 64-bit Intel- und AMD-Prozessoren.

Tipp: Haben Sie Ihre System i Access Installations-DVDs verlegt, so kopieren Sie den Inhalt des System-i-Verzeichnispfads QIBM/ProdData/CA400/ / Installxxx in ein Verzeichnis Ihres PCs oder Fileservers und schon haben Sie wieder ein Installationsimage des System i Access.

Oder rufen Sie einfach Setup.exe direkt aus dem beschriebenen Pfad auf.

System i Access für Windows auf dem PC installieren

In der Folge werde ich die Installation und Konfiguration des IBM System i Access auf einer Windows XP Workstation beschreiben. Das bedeutet jedoch nicht, dass der System i Access nur unter diesem PC-Betriebssystem verfügbar ist. Er kann auf folgenden PC-Betriebssystemen zur Ausführung kommen:

- V5R4: Microsoft Windows 2000, XP, Windows 2003 und Vista
- V6R1: Microsoft Windows XP, Windows 2003, Vista und Windows 7

23. Ergänzung 6/2010

7.2
Seite 2

Vor der Installation

Bevor Sie mit der Installation beginnen, sollten Sie sicherstellen, dass für die Installation von System i Access für Windows genügend Plattenspeicher zur Verfügung steht.

Der benötigte Plattenspeicher hängt von der gewählten Installationsmethode ab:

- Komplettinstallation: ca. 256 MB
- PC5250-Benutzer: ca. 27 MB
- Angepasst: variiert je nach installierten Funktionen

Im Folgenden sehen Sie eine Tabelle, die die technischen Voraussetzungen für die Installation von System i Access für Windows beschreibt:

Betriebssystem	System i Access für Windows	System i Access für Windows mit System i Navigator
»Windows 2000«	Pentium 133 MHz und mindestens 64 MB	Pentium 850 MHz (Minimum) und mindestens 256 MB (512 MB empfohlen)
Windows XP Professional	Pentium 233 MHz und mindestens 128 MB	Pentium 850 MHz und mindestens 256 MB (512 MB empfohlen)
Windows XP Professional, x64 Edition	Jede durch Windows XP Professional, x64 Edition, unterstützte Zentraleinheit und mindestens 256 MB Arbeitsspeicher	1 GHz und 512 MB Arbeitsspeicher
Windows XP Tablet	300 MHz und 256 MB Arbeitsspeicher	1 GHz und 512 MB Arbeitsspeicher
»Windows Vista Business, Enterprise oder Ultimate (32-Bit-Editionen)«	Zentraleinheit mit 1 GHz und 1 GB Arbeitsspeicher	Zentraleinheit mit 1 GHz und 1 GB Arbeitsspeicher
»Windows Vista Business, Enterprise oder Ultimate (64-Bit-Editionen)«	Zentraleinheit mit 1 GHz und 1 GB Arbeitsspeicher	Zentraleinheit mit 1 GHz und 1 GB Arbeitsspeicher (Itanium-Prozessor wird nicht unterstützt)
Windows Server 2003 (32-Bit-PC)	Pentium 133 MHz und mindestens 128 MB	Pentium 850 MHz und mindestens 256 MB (512 MB empfohlen)
Windows Server 2003 (x64-Editionen)	Pentium 733 MHz und mindestens 192 MB	Pentium 850 MHz und mindestens 256 MB (512 MB empfohlen)

Installationsvoraussetzungen (Quelle: IBM i Information Center)

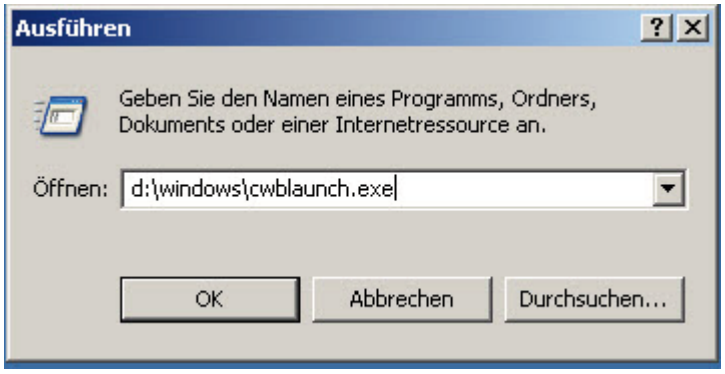
Sehen Sie sich bitte diese Tabelle an. Sie werden erkennen, dass ein „Standard"-Rechner durchaus ausreicht, um System i Access für Windows zu installieren.

7.2.1
Seite 2

Die IBM System i Access PC-Installation

Die Installation des System i Access auf Ihrem PC ist ein einfacher, sich selbst steuernder Windows-Dialog. Für diese Beispielinstallation habe ich eine virtuelle Maschine mit einem 3.2 GHz Intel-Prozessor, 1024 MB Hauptspeicher, einer 30 GB Festplatte, einem DVD-Laufwerk und Windows XP als Betriebssystem ausgewählt.

- Legen Sie die CD in das entsprechende Laufwerk, z. B. Laufwerk D:
- Über die Windows-Startleiste gehen Sie dann in die Menüoption „Ausführen“ .
- In das daraufhin erscheinende Pop-Up-Fenster geben Sie dann ein: D:\Windows\cwblaunch.exe
- Wichtig: Das Programm CWBLAUCH.EXE analysiert die technischen Voraussetzungen und wählt aus den verschiedenen Installationsimages (32-bit bzw. 64-bit) das für Ihren PC passende Image aus.



Start der Installation des IBM System i Access für Windows

Abweichend von allen früheren Versionen des System i Access wird jetzt der Standard Windows MSI-Installer aufgerufen, der den gesamten Installationsprozess steuert.

Sollten Sie bereits die Vorgängerversion des System i Access installiert haben, so erscheint in einem Popup-Dialog die Anfrage, ob Sie beabsichtigen, die alte Version des System i Access zu aktualisieren. Bejahen Sie diese Frage ggf. In diesem Fall wird die alte Version des System i Access für Windows durch den neuen System i Access ersetzt. Es werden jedoch alle Konfigurationswerte, die aus der älteren Version übernommen werden können, im neuen Client beibehalten.

7.2.2 Seite 2

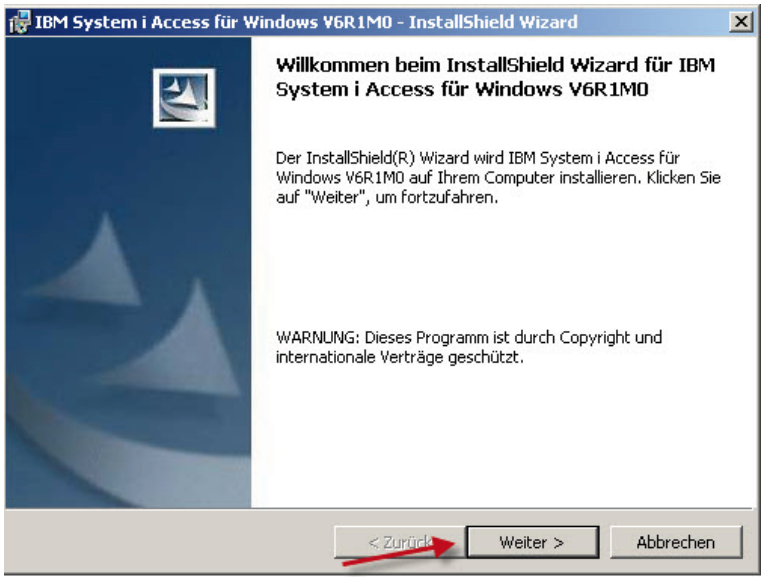
Die Installation beginnt mit der Abfrage der Sprache, die bei den Installationsdialogen verwendet werden soll.

Sprache auswählen

Bemerkenswert ist an dieser Stelle, dass auf den Installations-DVDs sämtliche Sprachversionen des System i Access ausgeliefert werden, so dass Sie unabhängig von der verwendeten Primärsprache und den installierten Sekundärsprachen Ihres Systems jede ausgelieferte Sprachversion auf einem Client installieren können.

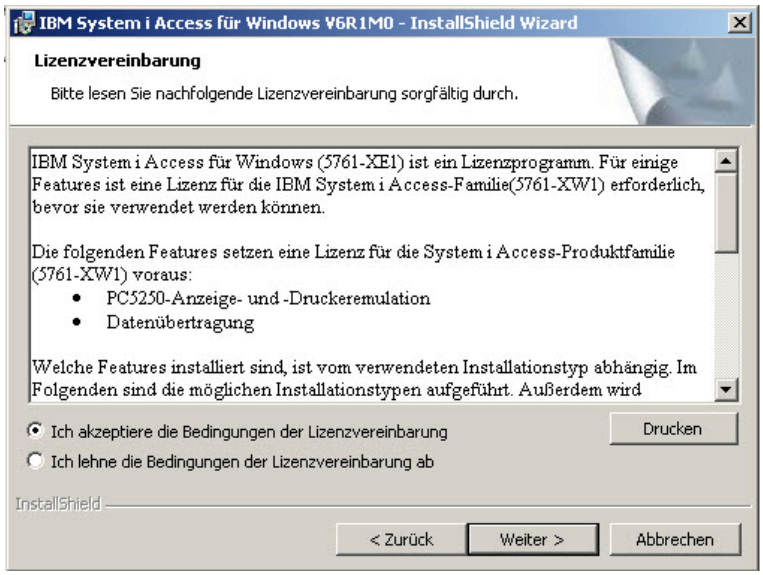
Beginnen Sie die Installation mit „OK“.

Es erscheint der Begrüßungsbildschirm, den Sie mit dem Button „Weiter“ bestätigen.



Willkommen

Bevor die eigentliche Installation beginnt, erscheint der Lizenzrechtbildschirm.

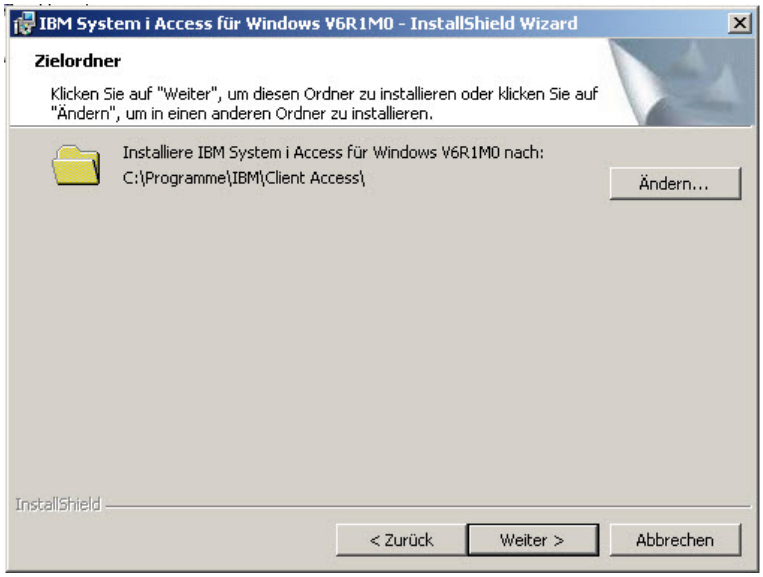


Lizenzinformationen

Lesen Sie die Lizensierungsangaben durch und akzeptieren Sie diese.

Jetzt wird der Pfad abgefragt, in den System i Access für Windows installiert werden soll.

Als Standard wird der IBM System i Access in das Verzeichnis d:\programme\ibm\System i Access installiert.



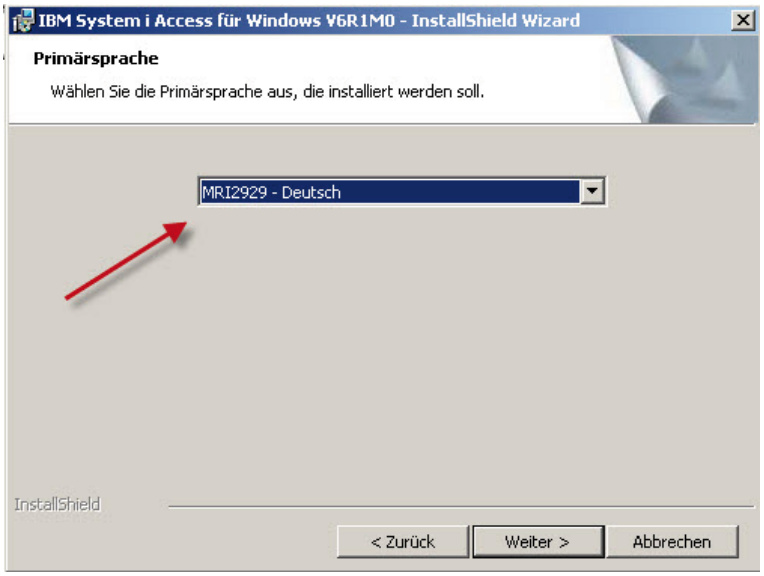
Zielverzeichnis auswählen

7.2.2
Seite 4

Diese Grundeinstellung kann von Ihnen jedoch überschrieben werden, indem Sie den Button „Durchsuchen“ drücken und dann selbst ein Installationsverzeichnis benennen.

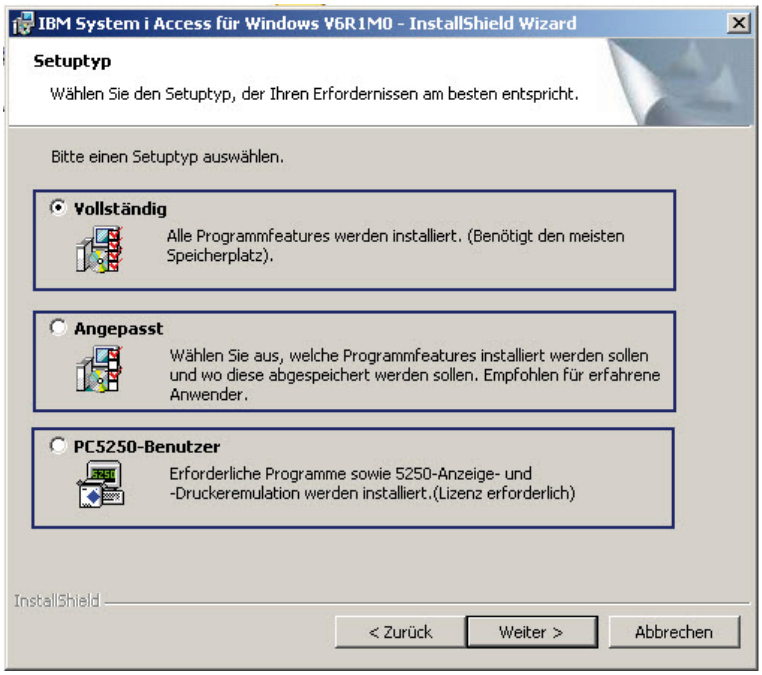
Ich habe es beim Standard belassen.

Danach wählen Sie die Primärsprache des zu installierenden Clients aus.



Primärsprache auswählen

Der nächste Bildschirm fragt Sie nach der Installationsmethode. Die verfügbaren Methoden unterscheiden sich im Umfang der zu installierenden Funktionsmodule.

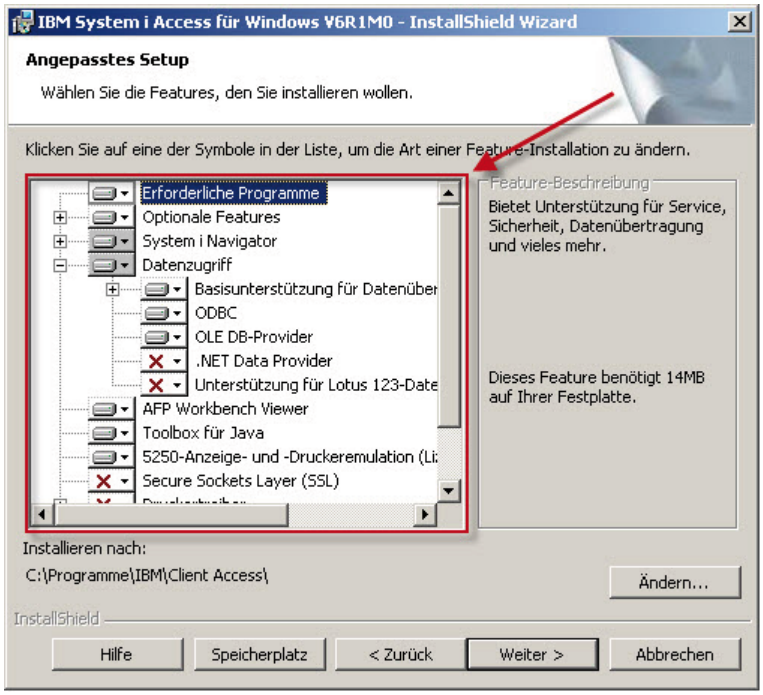


Die Installationsmethoden

PC5250-Benutzer: Installiert die Mindestunterstützung, die für die Druckeremulation und für PC5250 benötigt wird. Bei dieser Option werden die folgenden Komponenten installiert:

- Erforderliche Programme für System i Access
- 5250-Anzeige- und Druckeremulation
- Basisunterstützung für PC5250 (Standardmäßig wird die Version verwendet, die der Zeichenumsetztabelle auf Ihrem PC entspricht. Für jede Sprache gibt es eine Zeichenumsetztabelle.)
- PDF/PDT-Dateien für die Standardversion von PC5250 (eine pro Sprache)
- PC5250-Schriftarten

Angepasst: Sie können die Funktionsmodule, die installiert werden sollen, selbst auswählen.



Komponentenauswahl

7.2.2
Seite 6

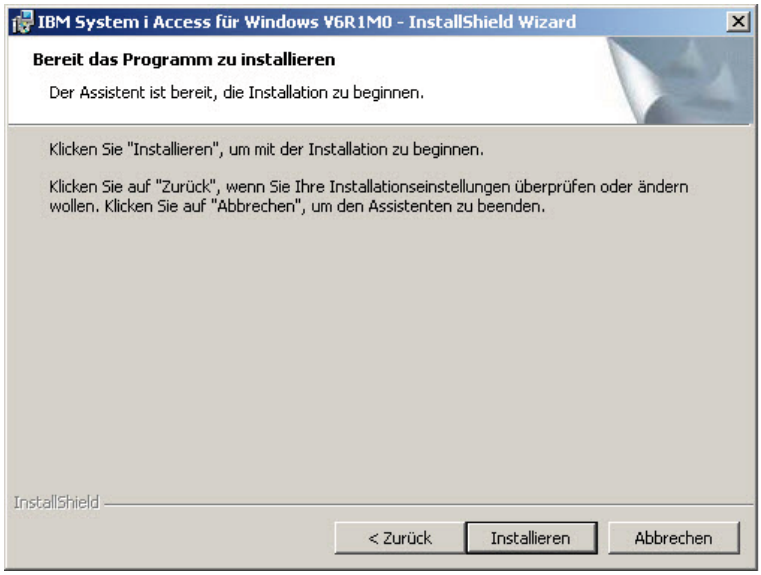
Vollständig: Installiert alle Komponenten von System i Access. Bei einer Installation des Typs „Vollständig“ werden die folgenden Komponenten installiert:

- Erforderliche Programme für System i Access
- Optionale Features
- Verzeichnisaktualisierung
- Eingehender ferner Befehl
- Online-Benutzerhandbuch für System i Access
- Datenzugriff
  - Datenübertragung
  - Basisunterstützung für Datenübertragung
  - Excel-Add-In für Datenübertragung
  - WK4-Dateiunterstützung für Datenübertragung
  - ODBC
  - OLE DB-Unterstützung
  - ADO .Net Datenprovider
- Secure Socket Layer(SSL)
- System i System i Navigator
- Basisunterstützung für System i Navigator
- Basisoperationen
- Jobverwaltung
- Systemkonfiguration
- Netzwerk
- Sicherheit
- Benutzer und Gruppen
- Datenbank
- Dateisysteme
- Multimedia
- Sicherung
- Anwendungsentwicklung
- Management Central
- Anwendungsverwaltung
- AFP Workbench Viewer
- Druckertreiber
  - AFP-Druckertreiber
  - SCS-Druckertreiber

- Operations Console
- System i Access Toolkit
- Visual Basic-Assistenten
- Header, Bibliotheken und Dokumentation
- System i Java Toolbox
- 5250-Anzeige- und Druckeremulation
- Basisunterstützung für PC5250 (Standardmäßig wird die Version verwendet, die der Zeichenumsetztabelle auf dem PC des Benutzers entspricht. Für jede Sprache gibt es eine Zeichenumsetztabelle.)
- PDF/PDT-Dateien für die Standardversion von PC5250 (eine pro Sprache)
- PC5250-Schriftarten (nur für Sprachen mit Einzelbytezeichensatz)

Ich wähle hier die vollständige Installation, so dass alle oben angezeigten Komponenten auf meinen Windows PC installiert werden.

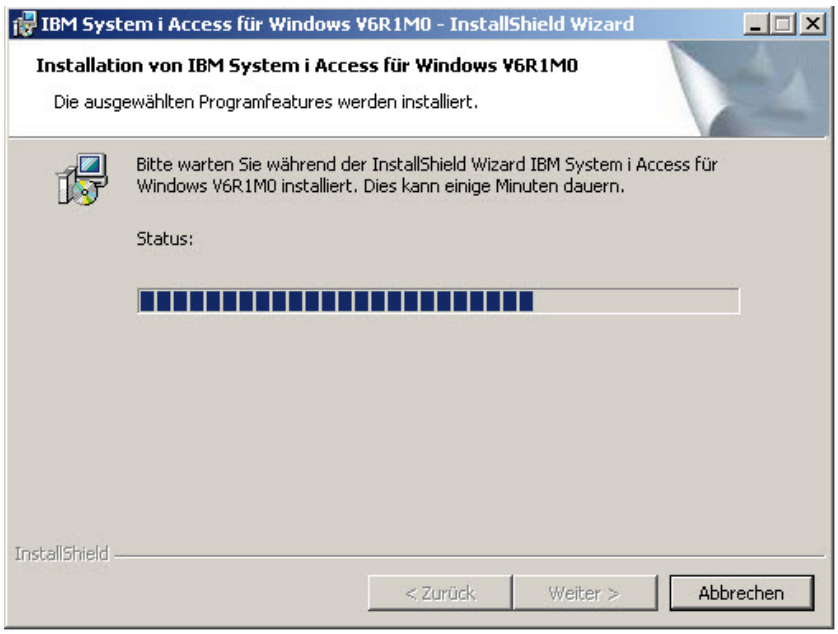
Mit der Bestätigung des folgenden Dialogs beginnt die Installation.



Beginn der Installation

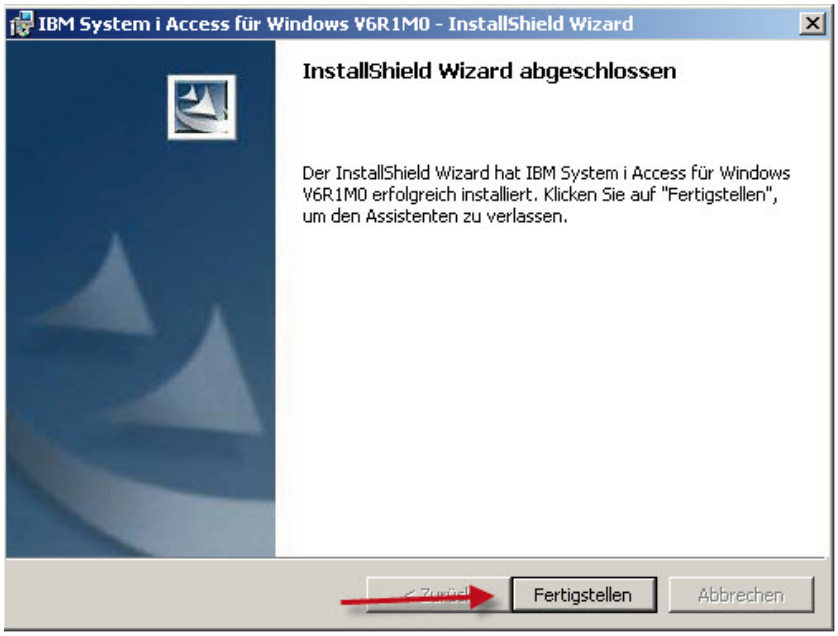
7.2.2
Seite 8

Danach erscheint eine Fortschrittsanzeige.



Ausgewählte Komponenten überprüfen

Nach einigen Minuten (ca. fünf) ist die Installation beendet.

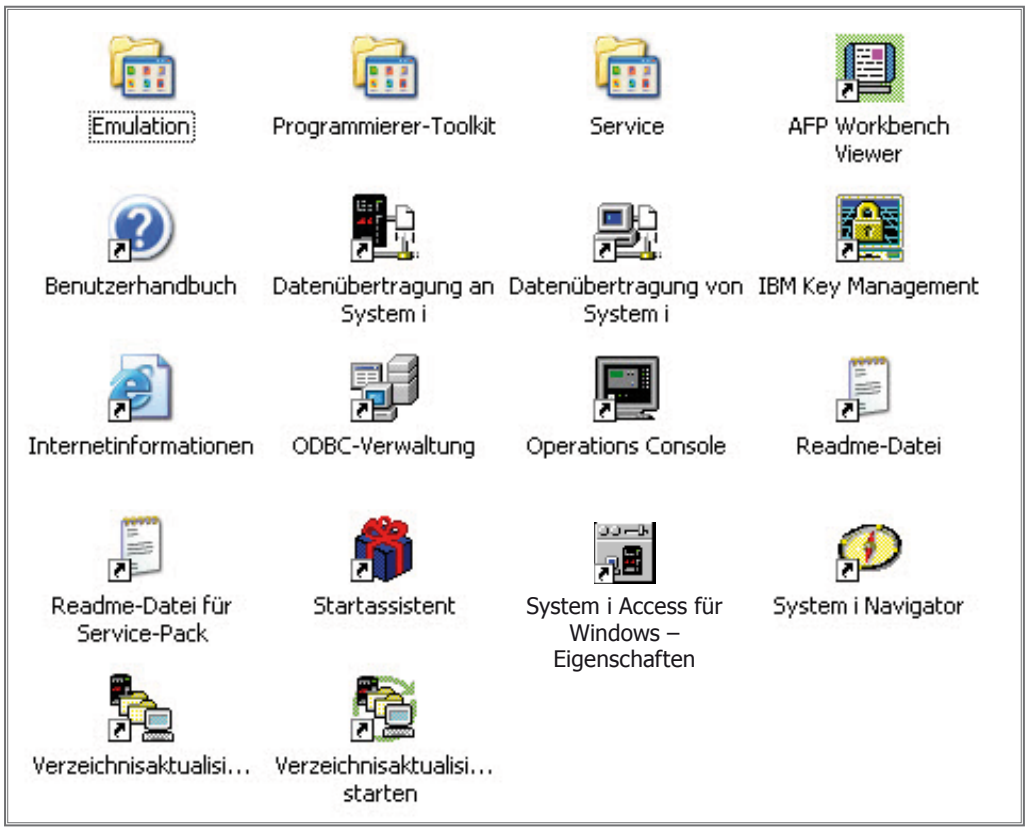


Ende der Installation

Zu guter Letzt werden Sie aufgefordert, Ihren PC nach Abschluss aller Kopiervorgänge neu zu starten.

Den Windows Client neu starten

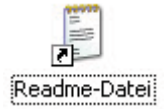
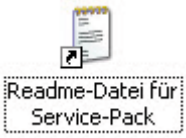
Nach dem Neustart des Systems können Sie vom Desktop des Windows PCs aus die Programmgruppe System i Access öffnen.



Die Programmgruppe IBM System i Access

7.2.2
Seite 10

Hier sehen Sie die System-i-Access-Programmgruppe nach der vollständigen Installation. Folgende Funktionen verbergen sich hinter den einzelnen Symbolen:

 Readme-Datei	Eine Textdatei, die Installationshinweise beschreibt. Unbedingt lesen!
 Readme-Datei für Service-Pack	Eine Textdatei, die Hinweise zur Service Pack-Installation enthält. Unbedingt lesen!
Emulation	Ordner der Bildschirmemulation PC5250
Service	Ordner für Diagnosetools und Servicepack-Installation
AFP Workbench Viewer	AFP Workbench Viewer. Ein Programm, um AFPDS-Ausgaben lokal anzuzeigen und um APF-Objekte zu erstellen.
Operations Console	Bildschirmemulation für den Aufbau einer System-i-Systemkonsole. Besitzen Sie eine System i ohne Twinax-Datenstationseinheit bzw. ohne HMC, so kann mit Hilfe dieser Funktion eine PC-basierte Systemkonsole erzeugt werden.
System i Navigator	Aufruf des System i Navigators. Dieses Icon sollten Sie zusätzlich auch auf dem Windows Desktop finden.
System i Access für Windows – Eigenschaften	Die Grundeinstellungen des System i Access bearbeiten.
Datenübertragung an System i	Filetransfer zur System i
Datenübertragung von System i	Filetransfer von der System i

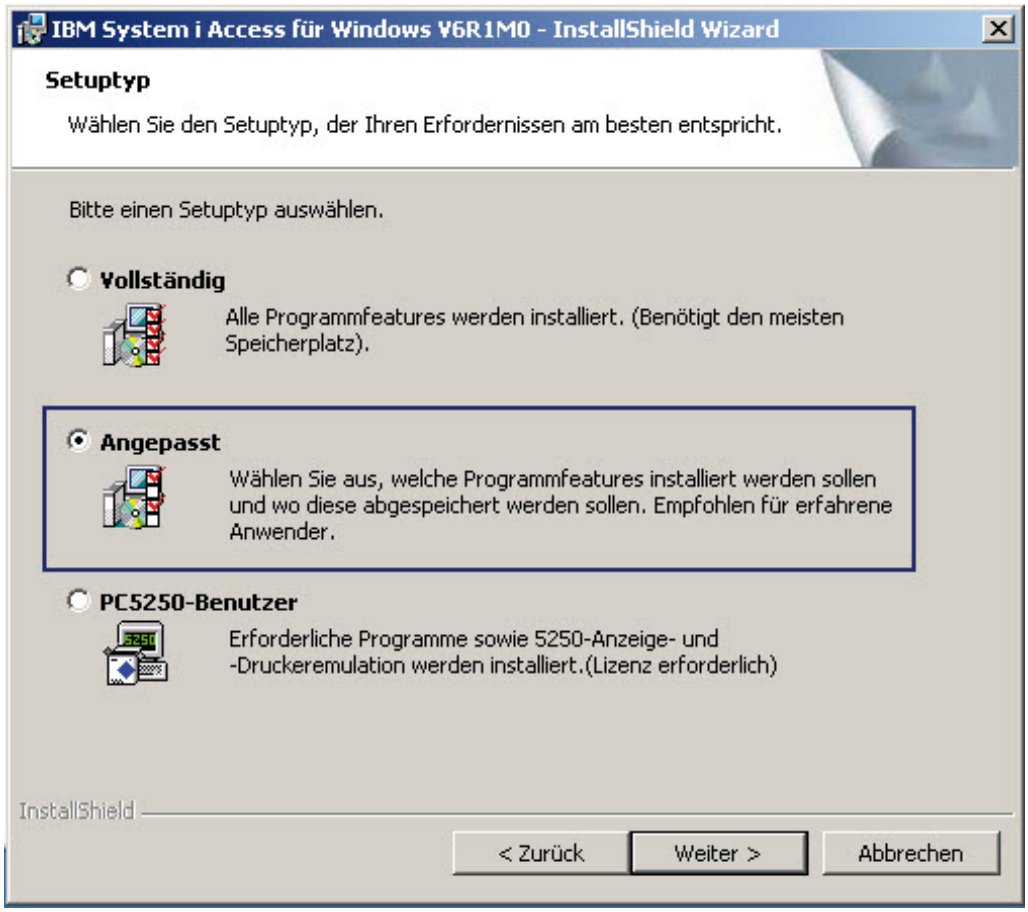
Programmierer-Toolkit	Programmierertools und Headerdateien für Programmierer
Benutzerhandbuch	Das Online-Benutzerhandbuch. Deutlich gegenüber seinen Vorgängern verbessert.
IBM Key Management	Schlüsselverwaltung, falls Sie sichere Verbindungen (SSL) aufbauen wollen.
ODBC-Verwaltung	Der ODBC-Administrator. Konfiguration von ODBC-Datenquellen für den Online-Zugriff auf DB2/400.
Selektive Installation	Nachinstallation von System-i-Access-Komponenten, sofern Sie nicht die vollständige Installation gewählt haben.
Verzeichnisaktualisi... starten	System-i-Access-Aktenkofferfunktion ausführen
Verzeichnisaktualisi...	System-i-Access-Aktenkofferfunktion konfigurieren
Internetinformationen	Wichtige Links ins Web. Z. B. Zugriff auf die Servicepack Homepage
Startassistent	Startassistent, enthält Links zum Online-Handbuch und zum i Information Center.

7.2.2
Seite 12

Die selektive Installation

Die angepasste Installation des System i Access für Windows ermöglicht Ihnen eine sehr feinfühlige Abstimmung der System-i-Access-Funktionen auf die Bedürfnisse eines Benutzers bzw. seines Arbeitsplatzes.

Der Installationsdialog des System i Access für Windows ermöglicht Ihnen die Angabe eines Installationstyps:



Auswahl des Installationstyps

Die hier sichtbaren Buttons stellen die derzeit interaktiv abrufbaren Installationstypen dar.

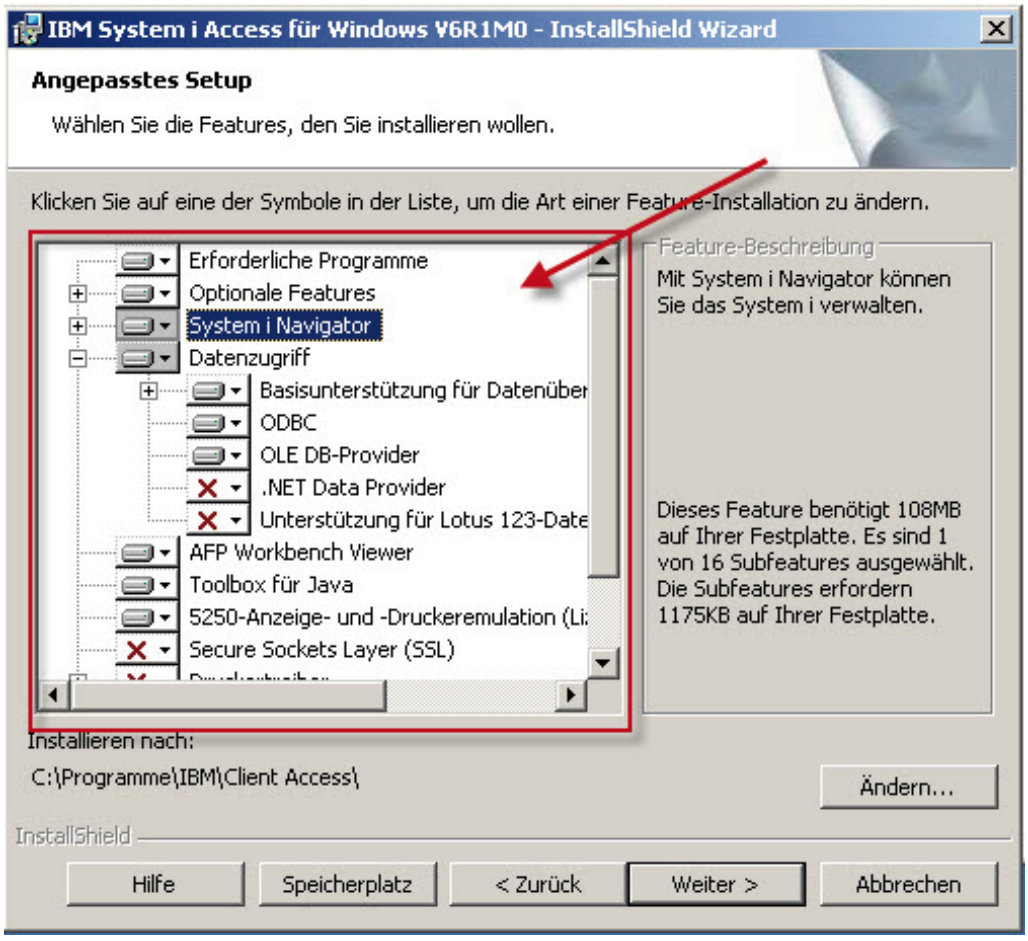
- Vollständig
- PC5250-Benutzer
- Angepasst

Diese Bezeichnungen sprechen für sich selbst und bedürfen meines Erachtens keiner Erläuterung. Da wir in diesem Abschnitt die angepasste Installation besprechen, werde ich jetzt den entsprechenden Dialog anfordern.

7.2.3 Seite 2

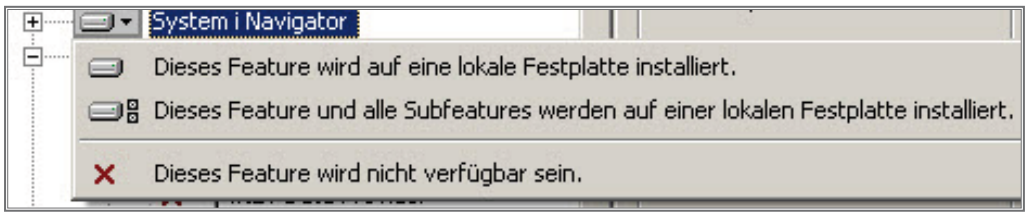
Dieser Dialog folgt mittlerweile den allgemeinen Anforderungen von Installationsdialogen. Programmfunktionen werden in Haupt- und Sub-Komponenten unterteilt und als Verzeichnisbaum dargestellt.

Haupt- und Subkomponenten können in die Installation durch Aktivieren des Submenüeintrags einbezogen bzw. durch Nicht-Setzen ausgeschlossen werden. Wird nur ein Teil der Subkomponenten für eine Hauptkomponente angewählt, so wird das Häkchen für die Hauptkomponente grau hinterlegt. Einfach formuliert: Ist das Häkchen für eine Hauptkomponente grau hinterlegt, so sind nicht alle Subkomponenten dieser Hauptkomponente ausgewählt worden.



Komponentenauswahl 1

Haupt- und Subkomponenten können in die Installation durch Aktivieren eines Submenüeintrags einbezogen bzw. ausgeschlossen werden.



Komponentenauswahl 2

Drei Verarbeitungsmöglichkeiten stehen Ihnen zur Verfügung:

- Dieses Feature …: Nur die angewählte Funktion wird installiert.
- Dieses Feature und alle Subfeatures…: Alle Unterfunktionen einer Hauptfunktion werden installiert.
- Dieses Feature wird nicht …: Die angewählte Funktion wird nicht installiert.

Wird nur ein Teil der Subkomponenten für eine Hauptkomponente angewählt, so ist die Hauptkomponente grau hinterlegt.

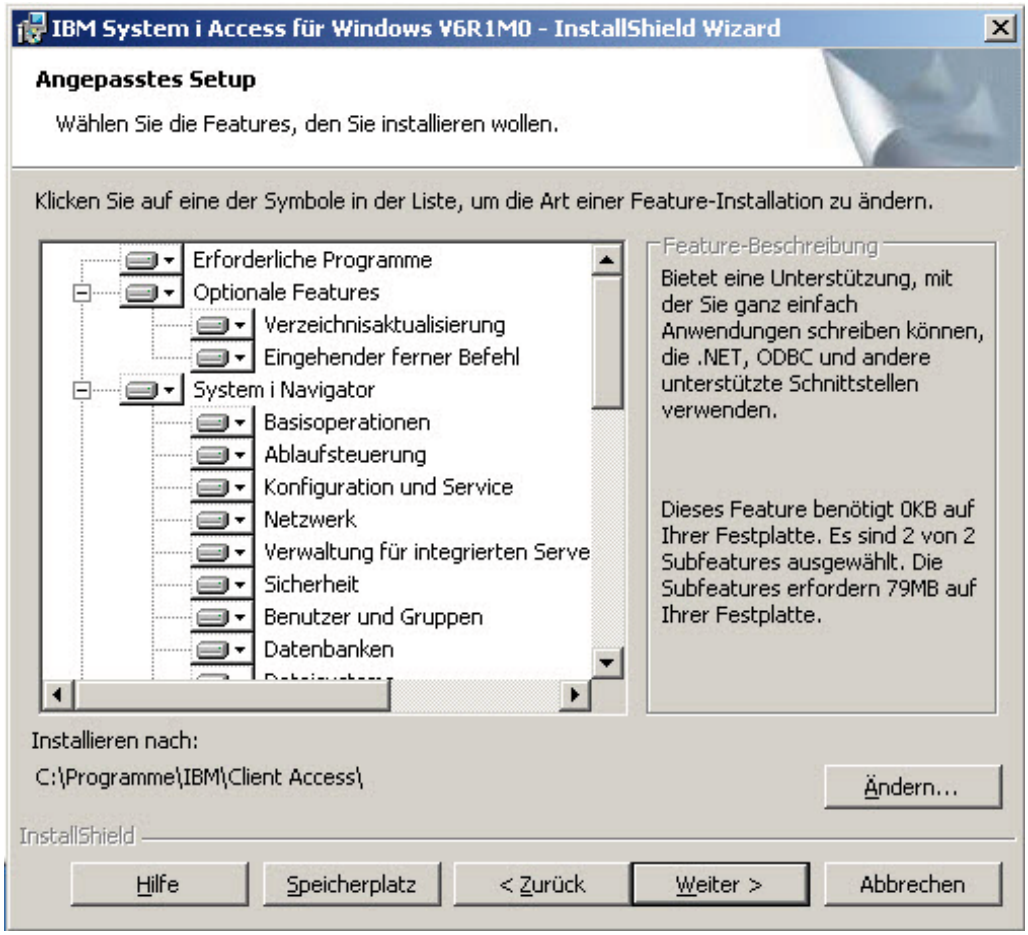
Sie sehen, dass in diesen Dialogen bereits einige Haupt- und Subkomponenten ausgewählt wurden. Die hier gezeigten Auswahlen stellen eine Standard-Installation dar.

Sie sehen hier ebenfalls, dass nur Teile der Hauptkomponente System i Navigator für die Installation markiert wurden. Deshalb ist das Verzeichnis für den System i Navigator grau hinterlegt.

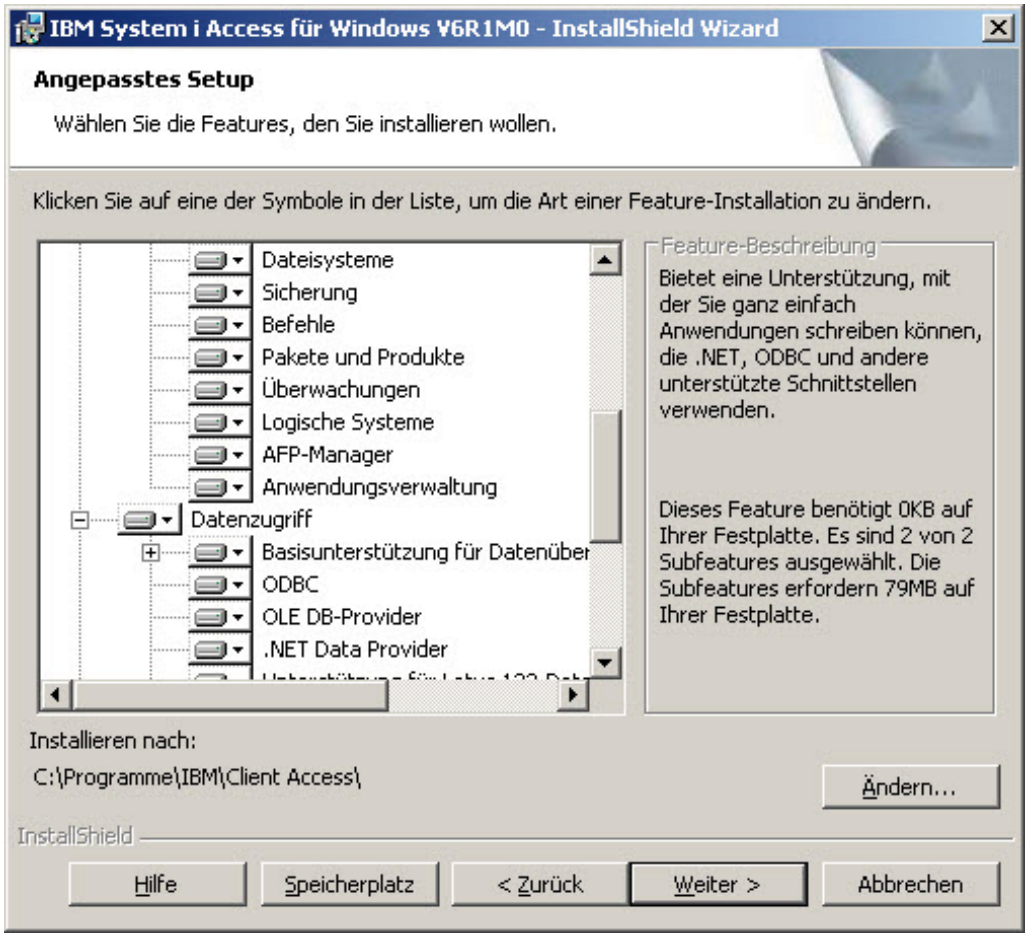
Weiterhin wird die Objektgröße jeder Haupt- bzw. Subkomponente angezeigt und darüber hinaus der Gesamtspeicherbedarf der gewählten Installation sowie der noch verfügbare Speicher auf der Platte des PCs.

Der Vollständigkeit halber habe ich Ihnen die Anzeige aller weiteren anwählbaren Komponenten beigefügt.

7.2.3
Seite 4

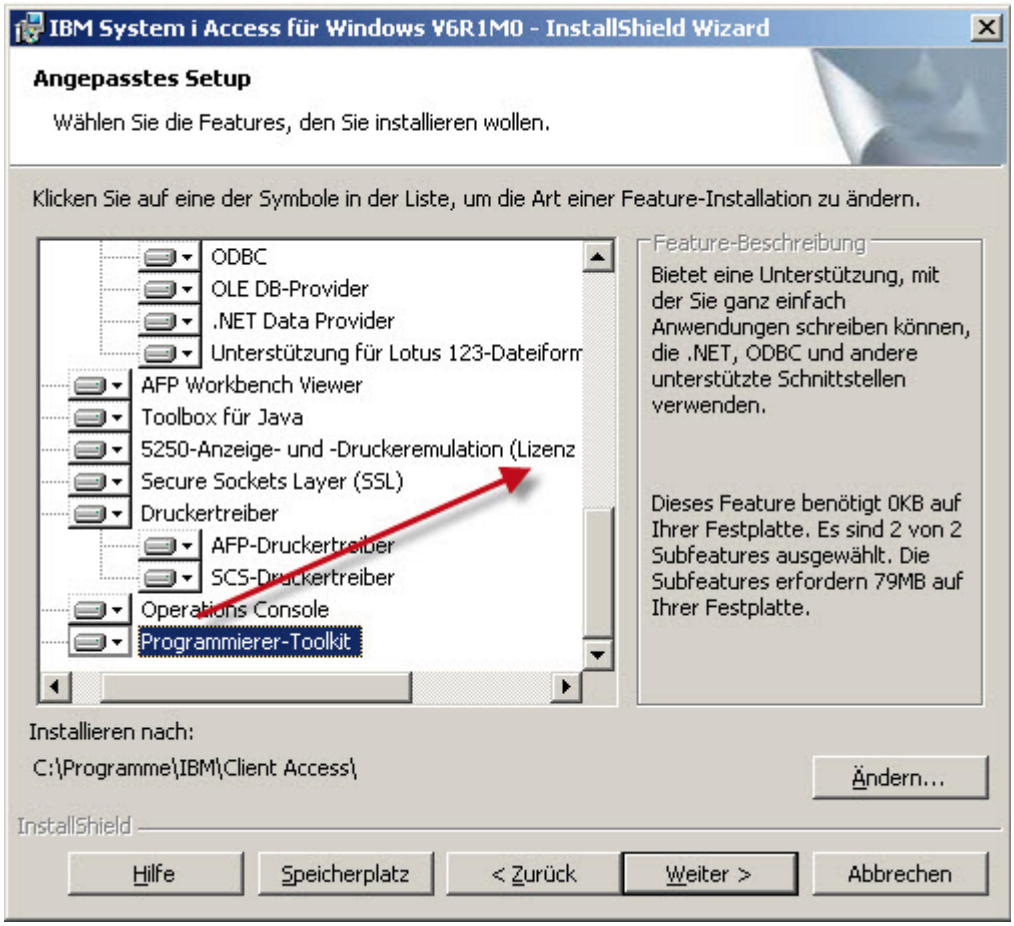


Komponentenauswahl 3



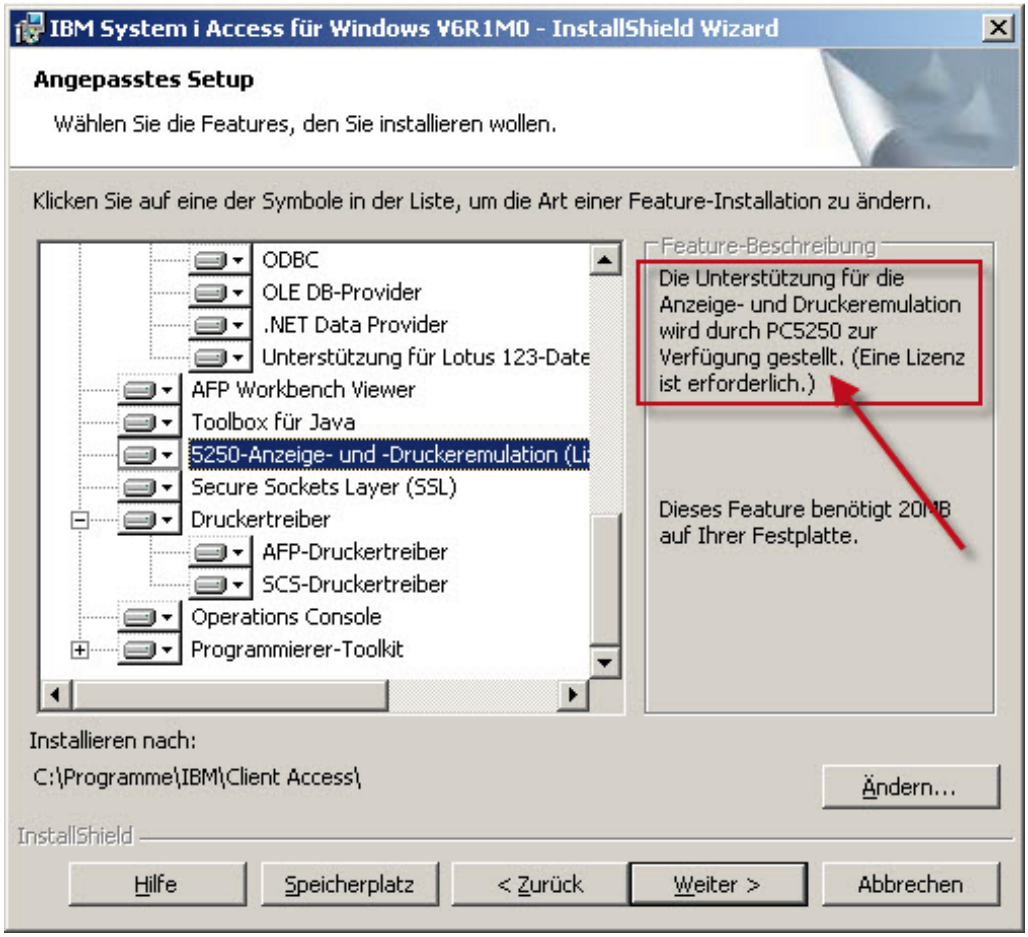
Komponentenauswahl 4

7.2.3
Seite 6



Komponentenauswahl 5

Beachten Sie in den Dialogen die Klammer „Eine Lizenz ist erforderlich“. Hier markierte Komponenten bedürfen des Erwerbs einer entsprechenden Softwarelizenz.



Komponentenauswahl 6

7.2.3
Seite 8

Fallstudie

Zur Verdeutlichung des soeben Dargestellten werden wir jetzt am Beispiel einer Fallstudie die einzelnen Arbeitsschritte durchgehen, um eine angepasste Installation durchzuführen. Folgendes Szenario soll gelten.

Für einen Arbeitsplatz sollen folgende Funktionen des System i Access aufgebaut werden:

1. Möglichkeit der Verbindungsaufnahme
2. Bildschirmemulation mit emulierten Druckern
   Für die Druckfunktionen kommen IBM-Drucker zum Einsatz, die Advanced Function Printing (AFP) unterstützen.
3. Dateitransfer
   Die Anwender nutzen Microsoft Excel und wollen Dateitransfers direkt aus Excel heraus aufrufen.
4. ODBC
   Direktzugriffe auf System-i-Dateien
5. System i Navigator – Basisfunktionen
   - Mit Nachrichten arbeiten
   - Mit Druckausgaben arbeiten
   - Mit Druckern arbeiten

Ich starte die Installation, indem ich von der Installations-CD das Programm SETUP.EXE im Verzeichnis „Express“ aufrufe.

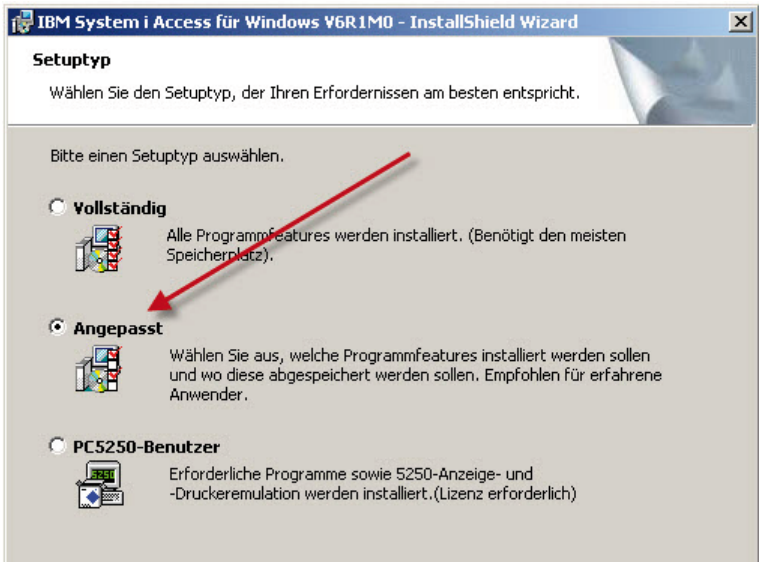
Zunächst bestimme ich den Installationspfad für meinen System i Access für Windows:

System i Access für Windows – Installationsverzeichnis

Da auf meinem PC alle Programme auf dem Laufwerk C: gespeichert werden, übernehme ich den Standardpfad.

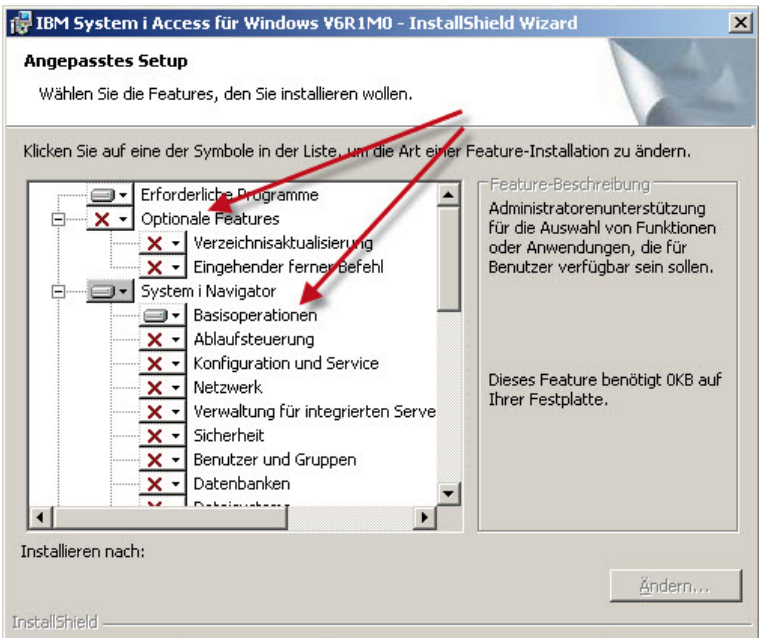
7.2.3.1
Seite 2

An geeigneter Stelle wähle ich den Installationstyp „Angepasst“ aus.



Auswahl des Installationstyps

Jetzt erfolgt die Auswahl der einzelnen Komponenten:

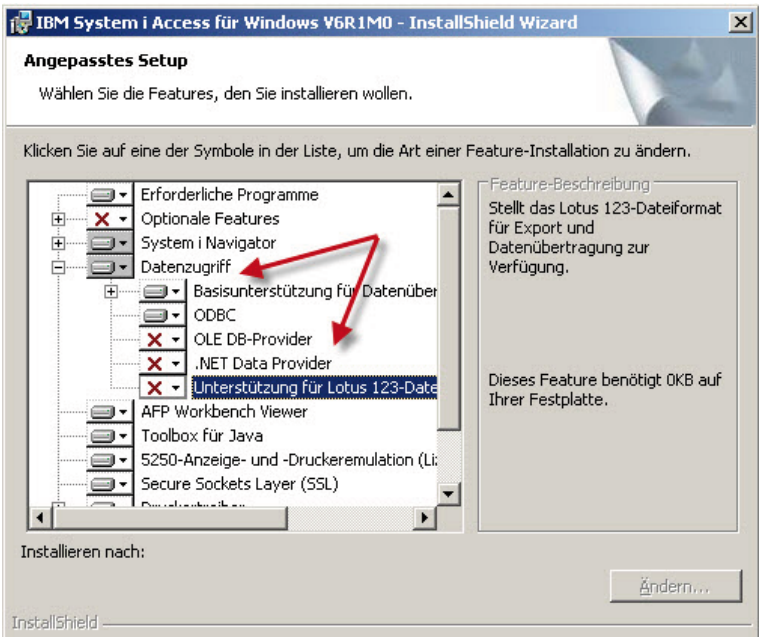


System-i-Navigator-Komponenten

Zuerst habe ich die Komponente „Erforderliche Programme“ ausgewählt. Um diese Komponente komme ich nicht herum, da hierüber alle Komponenten ausgewählt werden, die für einen Verbindungsaufbau vom System i Access für Windows zur System i nötig sind.

Danach habe ich noch die für den System i Navigator benötigten Komponenten ausgewählt.

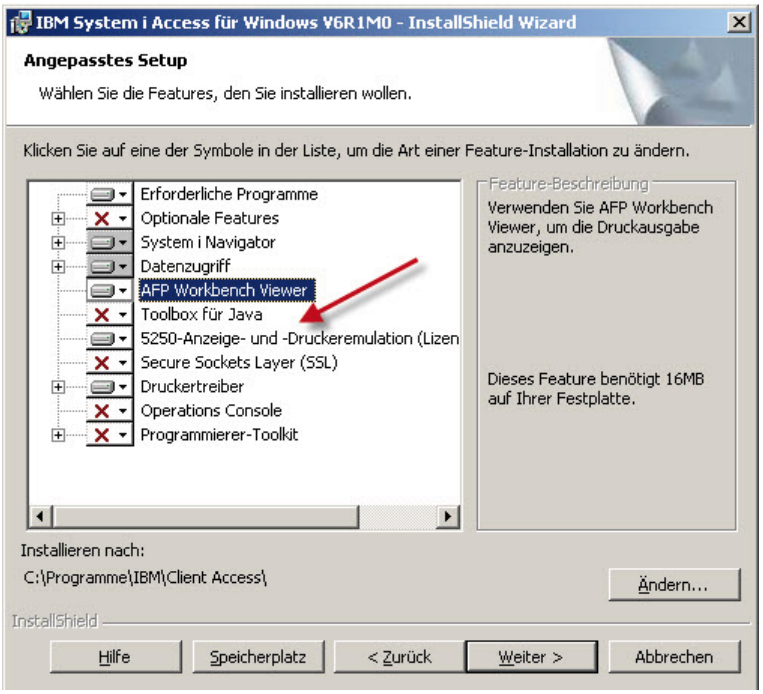
Im folgenden Dialog sehen Sie die Auswahl der Dateitransfer- und ODBC-Komponenten:



File-Transfer-Komponenten

Da ich die verschiedenen oben angegebenen Datenprovider nicht zur Installation ausgewählt habe, ist die Hauptkomponente „Basisunterstützung für Datenübertragung“ grau hinterlegt.

Jetzt erfolgt die Auswahl der Emulationskomponenten:

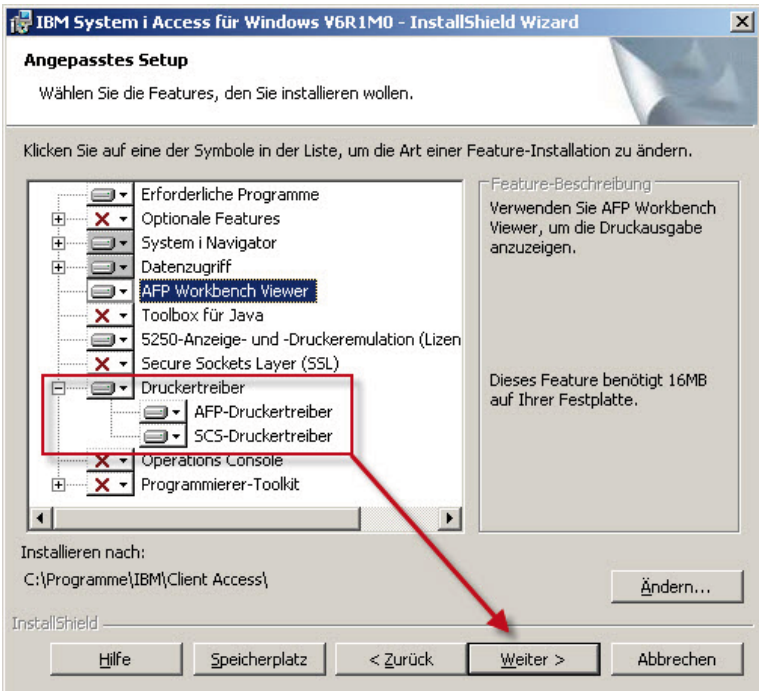


Bildschirm- und Druckeremulation

7.2.3.1
Seite 4

Auch in diesem Teil werden nicht alle Komponenten ausgewählt.

Jetzt bleibt lediglich der Punkt Druckertreiber zur Bearbeitung übrig:



Druckertreiber

Ich wähle hier die Hauptkomponente Druckertreiber komplett an, so dass alle Advanced Function Printing- (AFP-) und SCS-Druckertreiber für IBM-Drucker als Funktionsumfang des System i Access für Windows mit installiert werden.

Ich beende den Komponentenauswahldialog mit „Weiter“, so dass alle markierten Komponenten installiert werden.

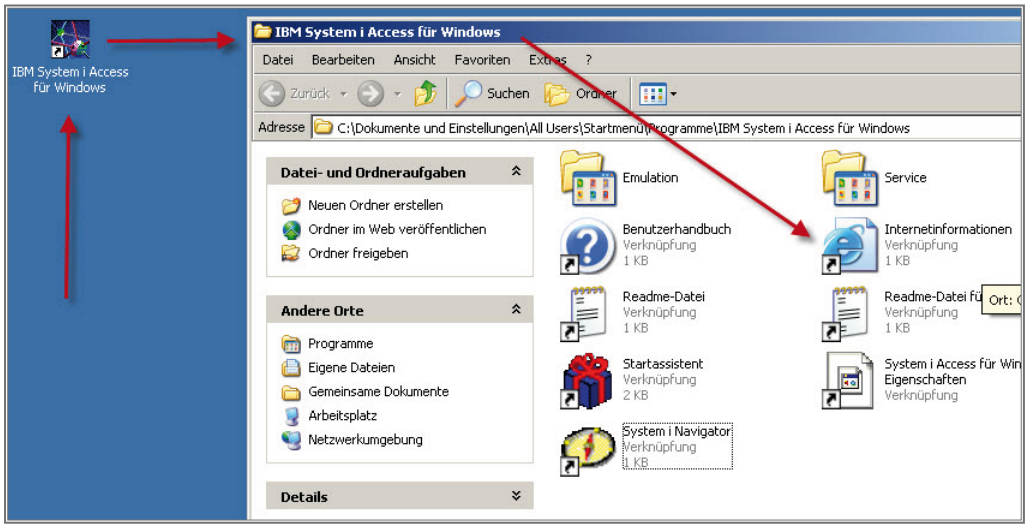
Service Packs

Nach der Installation sollten Sie das aktuelle Service Pack für System i Access für Windows einspielen. Neben der jeweils aktuellen Version des Service Packs finden Sie im Internet auch ältere PTFs für den Fall, dass Sie mit einer früheren Access-Version arbeiten.

Achtung: IBM Access für Windows ohne installierte Service Packs zu verwenden, ist nicht zu empfehlen, da eine Reihe von Access-Funktionen erst nach Installation des Service Packs fehlerfrei ausgeführt werden können.

Wichtig: Service Packs sind grundsätzlich kumulativ. Haben Sie in der Vergangenheit die Installation eines Service Packs vergessen, so reicht es, das jetzt aktuelle zu installieren.

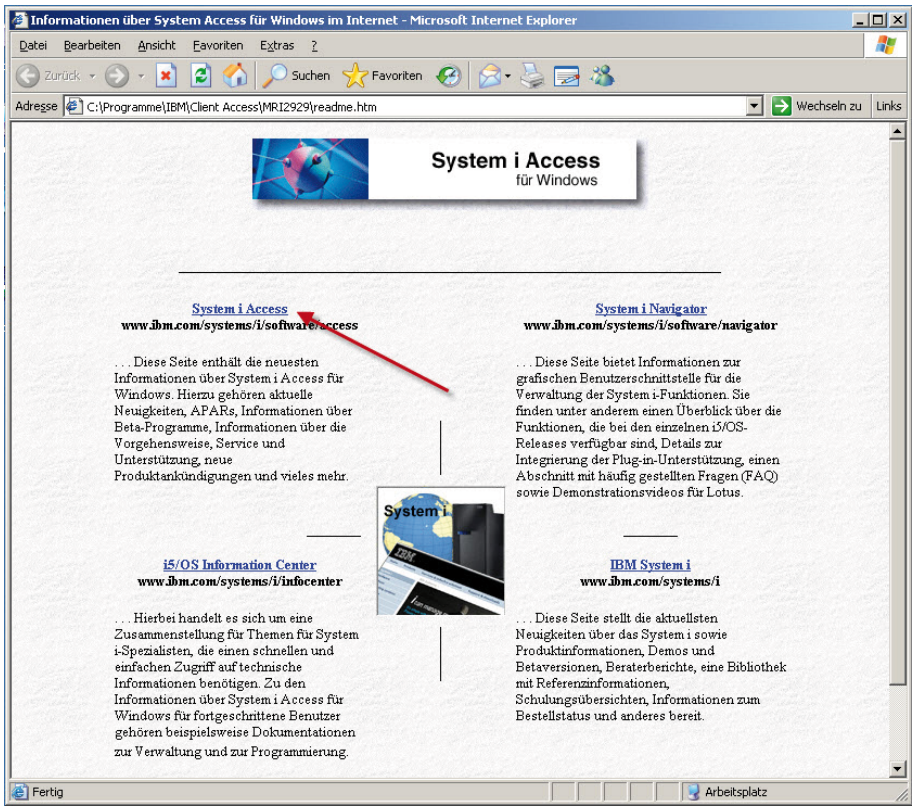
Um die Service Packs im Internet zu finden, öffnen Sie einfach die Programmgruppe System i Access für Windows.



Internetinformationen

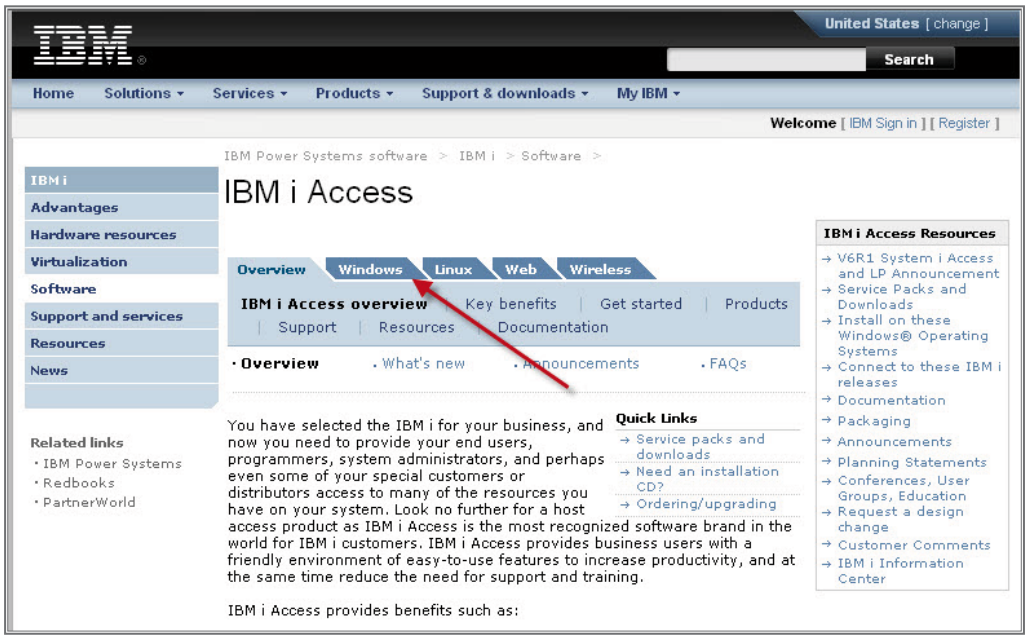
7.2.4 Seite 2

Starten Sie den Link „Internetinformationen“ und Sie sehen folgende Seite in Ihrem Browser.



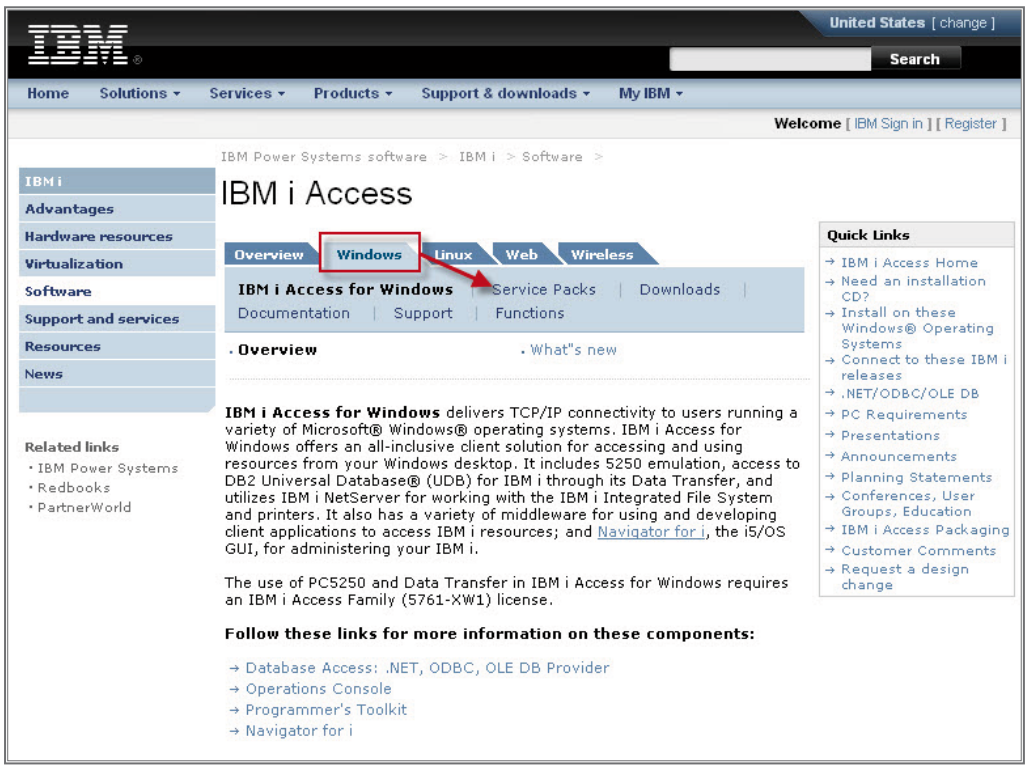
Link zur Access Homepage

Folgen Sie jetzt dem Link „System i Access“ und die entsprechende Homepage wird angezeigt.



IBM i Access Homepage

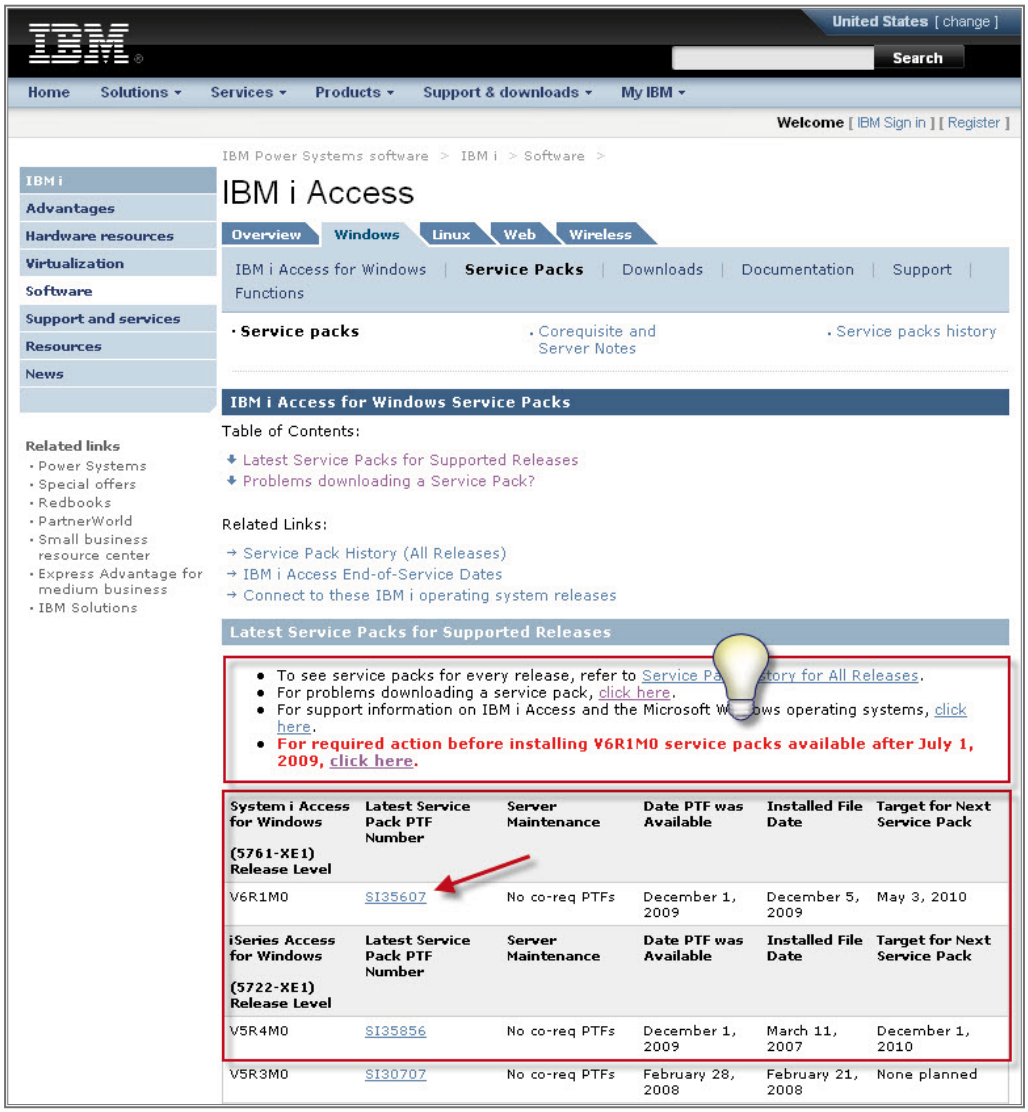
Jetzt klicken Sie auf den Reiter „Windows“.



Link zu „Service Packs 1”

Klicken Sie jetzt auf den bezeichneten Link „Service Packs“.

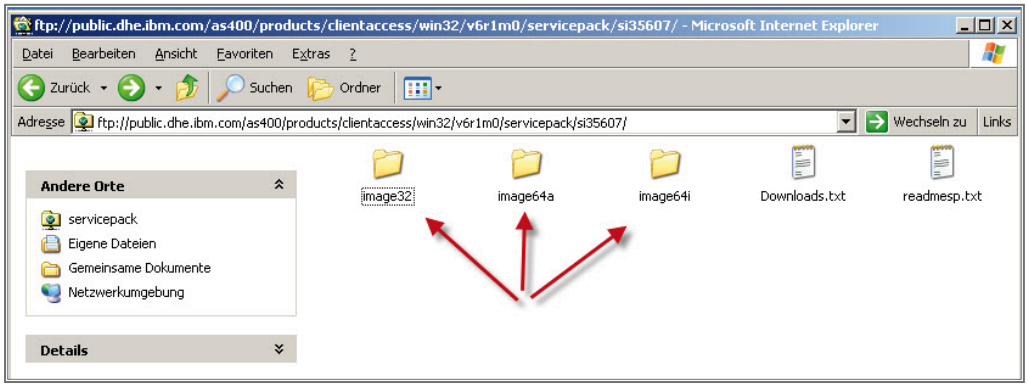
7.2.4
Seite 4



Link zu „Service Packs 2"

Achtung: Beachten Sie, ob es für die Installation des aktuellen Service Packs Vorbedingungen gibt, die Sie zu erfüllen haben.

Ansonsten klicken Sie auf den Link zum entsprechenden Service Pack Download.

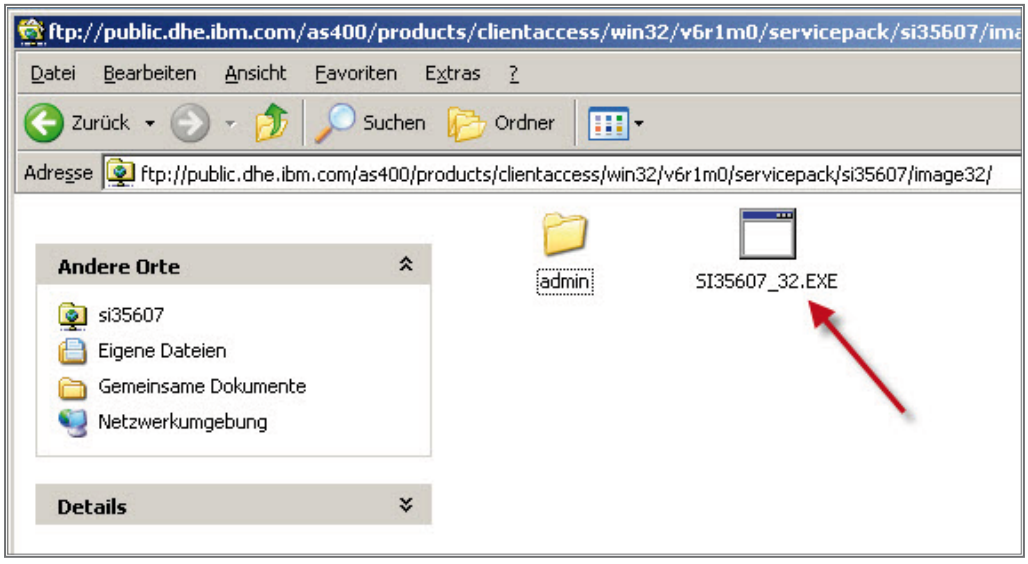


Ordner für Images

Jetzt sehen Sie, dass für jede Architekturvariante entsprechende Service Packs hinterlegt wurden.

Folgen Sie dem für Ihre Installation passenden Link.

Tipp: Lesen Sie die beiden Dateien Download.txt und readmesp.txt. Sie enthalten wichtige Informationen zur Installation von Servicepacks.



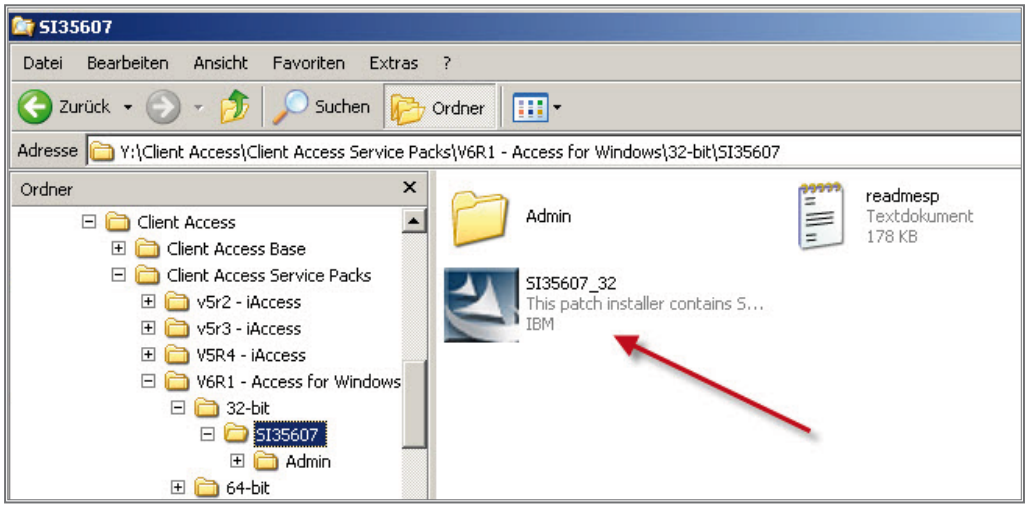
Das Service Pack herunterladen

Mit einem Doppelklick auf die gezeigte exe-Datei laden Sie das Service Pack herunter. Speichern Sie diese Datei auf Ihrem Arbeitsplatz-PC oder auf einem Fileserver Ihres Unternehmens.

Der Ordner „Admin“ enthält ein Administrationswerkzeug mit dem Namen Update.exe, dem wir uns später zuwenden werden.

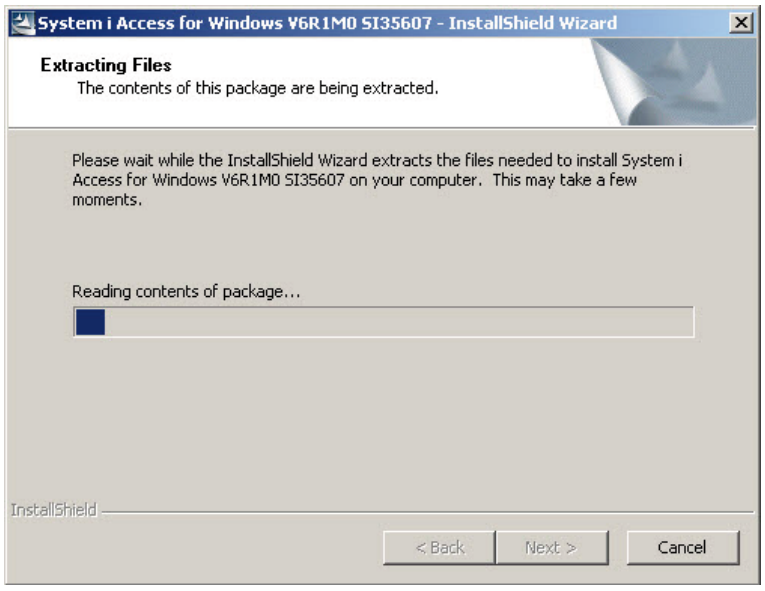
7.2.4
Seite 6

Ich habe das Service Pack auf unseren Fileserver heruntergeladen.



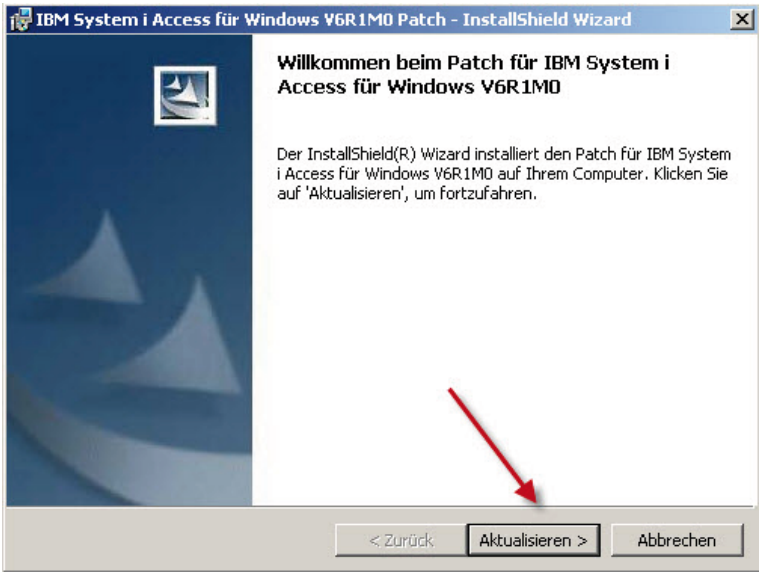
Service Pack installieren

Mit einem Doppelklick auf die angezeigte Datei beginnt der Installationsvorgang für das Service Pack. Zuerst wird das Servicepack in ein temporäres Verzeichnis entpackt.



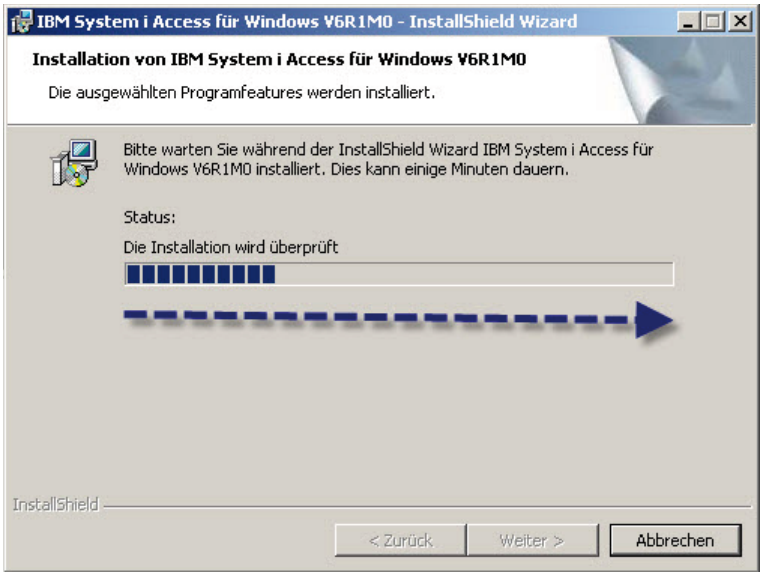
Service Pack entpacken

Nachdem dieser Arbeitsschritt erfolgreich beendet wurde, wird der Willkommen-Dialog angezeigt.



Willkommen

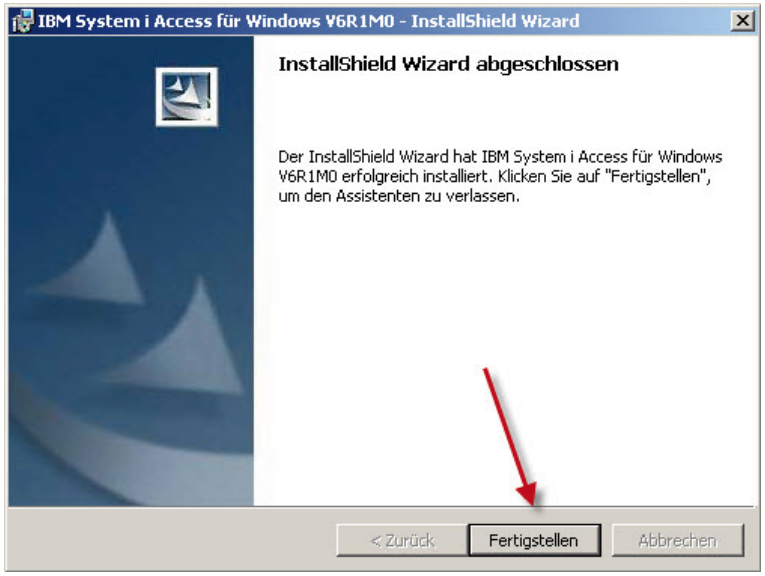
Bestätigen Sie diesen Dialog mit „Aktualisieren“ und der Updatevorgang beginnt.



Update beginnt

7.2.4
Seite 8

Kurze Zeit später wird der „Fertigstellen“-Dialog angezeigt.



Update beendet

Jetzt müssen Sie Ihren PC neu starten.

Neustart von Windows anfordern

Nachdem Ihr PC wieder zur Verfügung steht, kann die Auskonfiguration Ihres System-i-Access-Arbeitsplatzes beginnen.

Eine Antwortdatei erstellen

Unser erster Verarbeitungsschritt besteht jetzt darin, eine Antwortdatei (Response File) zu erstellen. Diese Datei wird erstellt, indem wir eine ganz normale interaktive Installation auf einem Windows PC ausführen. Die Besonderheit dieser Installation liegt darin, dass wir den Aufruf des Programms SETUP.EXE mit bestimmten Parametern steuern.

Schematisch sieht der Aufruf des Setup-Programms wie folgt aus:

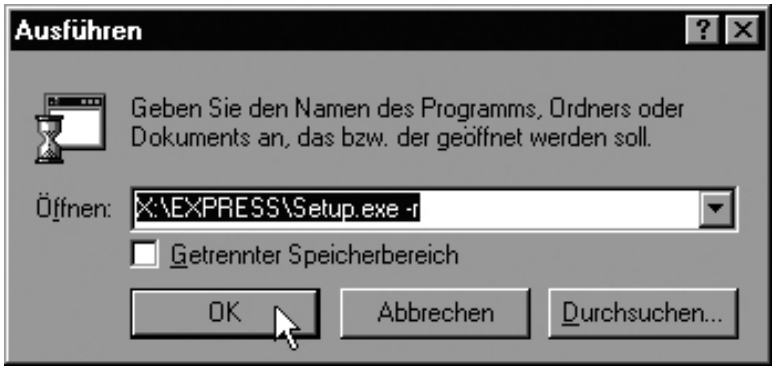
```
SETUP.EXE -r -f1LW\DIR\ANTWORT.ISS
```

Parameter	
-r	Steht für record = aufnehmen Jede interaktive Aktion des Installierenden wird in die Antwortdatei aufgenommen.
-f1	LW = Laufwerk DIR = Verzeichnis ANTWORT.ISS = Name der Antwortdatei Der Parameter „-f1" ist ein optionaler Parameter. Wird er gesetzt, so kann der Standort und Name der aufgezeichneten Antwortdatei genau beschrieben werden. Es ist notwendig, dass eine Antwortdatei die Namenserweiterung „ISS" trägt. Wird dieser Parameter nicht gesetzt, was möglich ist, so wird automatisch eine Antwortdatei mit dem Namen „SETUP.ISS" erzeugt. Diese Datei finden Sie nach Beendigung der Aufnahme im Windows-Installationsverzeichnis in der jeweiligen Windows-Version.

Wir werden jetzt eine vollständige Installation des iSeries Access Express durchführen und uns danach die aufgezeichnete Antwortdatei ansehen.

Aus dem Dialog „Ausführen" des Windows rufe ich das iSeries Access Express Setup-Programm auf:

7.2.4.1
Seite 2



Setup ausführen

Im „Ausführen"-Dialog habe ich hinter dem Aufruf des Programms SETUP.EXE den Parameter „-r" gesetzt. Da ich die Aufnahme der Antwortdatei an einem Windows NT 4.0-System durchführe und keinen spezifischen Standort für die Antwortdatei angegeben habe, erhalte ich die Datei „SETUP.ISS" als Antwortdatei. Die Datei SETUP.ISS werde ich dann im Verzeichnis WINNT meines PCs finden.

Im Weiteren folgt ein ganz normaler iSeries Access Express-Installationsdialog.

Der Willkommen-Dialog

Im „Willkommen"-Dialog betätige ich nur den Knopf „Weiter". Dann erscheint der Dialog „Lizenzinformationen".

Lizenzinformationen prüfen

Auch diesen Dialog bestätige ich, indem ich den Knopf „Ja“ drücke. Jetzt kann ich den „Installationstyp“ auswählen.

Angepasste Installation

7.2.4.1
Seite 4

Hier wähle ich die „Angepasste“ Installation aus. Warum wähle ich nicht die „Vollständige“ Installation, wenn ich alle Komponenten des iSeries Access Express installieren will? Nur indem ich die Angepasste Installation auswähle, erhalte ich in meiner Antwortdatei eine Liste aller iSeries Access Express-Komponenten und Sub-Komponenten. Diese Liste kann ich später für noch folgende unbeaufsichtigte Installationen, in denen nicht alle Komponenten installiert werden sollen, einfach mit einem Texteditor bearbeiten, sodass ich nicht noch weitere Aufnahmevorgänge realisieren muss.

Sie sehen hier jetzt den Standard-Installationspfad für iSeries Access Express.

Standard-Installationspfad

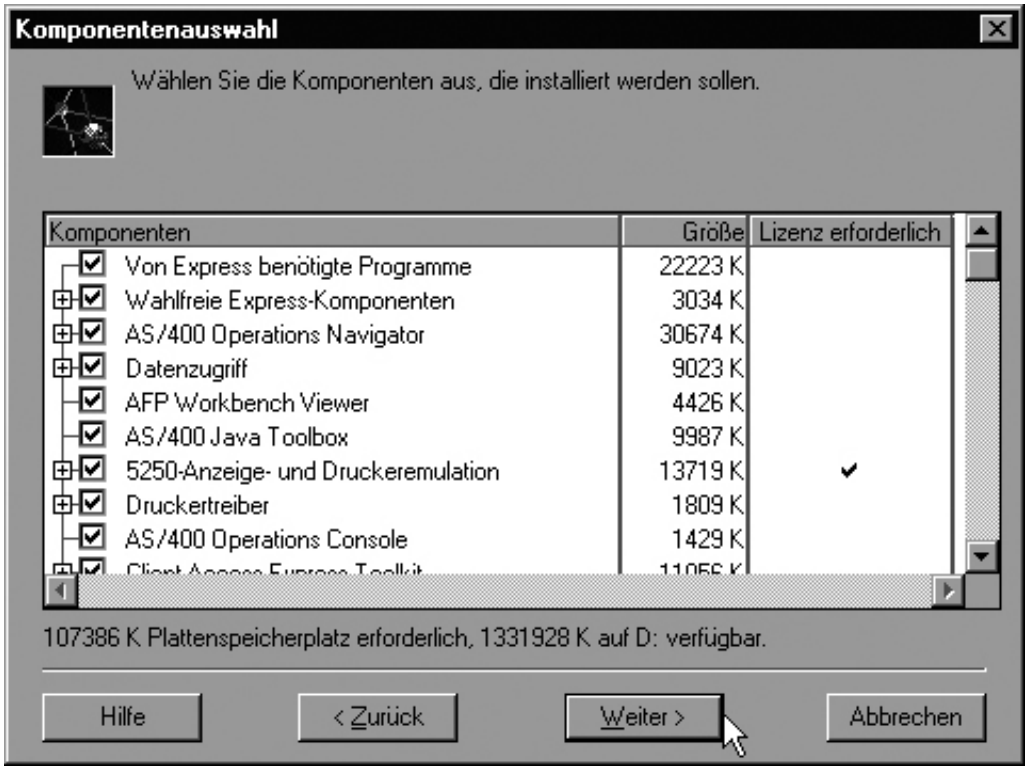
Mit Hilfe des Dialogs „Durchsuchen…“ ändere ich den Pfad.

Installationspfad wurde geändert

Da auf meinem Windows PC alle Programme auf Laufwerk „D:“ liegen, habe ich eine entsprechende Änderung des Pfads durchgeführt.

Jetzt erfolgt die Komponentenauswahl.

7.2.4.1
Seite 6



Komponentenauswahl

Wie Sie leicht erkennen können, habe ich alle nur möglichen Komponenten des iSeries Access Express ausgewählt und erhalte so in der Antwortdatei eine komplette Liste aller angewählten Komponenten.

Im folgenden Dialog wird der „Name“ des Programmordners für iSeries Access Express angegeben.

Programmordner benennen

Hier belasse ich es bei den Standard-Einstellungen des iSeries Access Express.

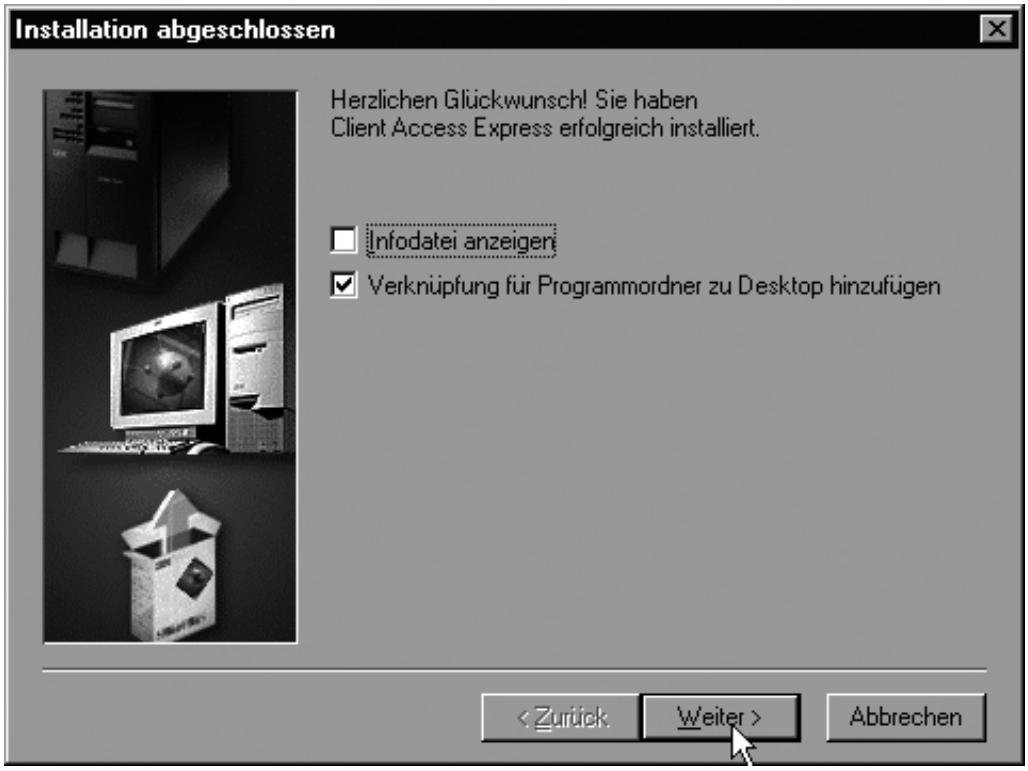
Es erfolgt noch einmal die Überprüfung der zur Installation vorgesehenen Komponenten.

7.2.4.1
Seite 8

Kopiervorgang starten

Ich betätige den Knopf „Weiter“ und der Installationsprozess beginnt.

Danach ist der Kopiervorgang innerhalb der Installation beendet.



Installation abgeschlossen

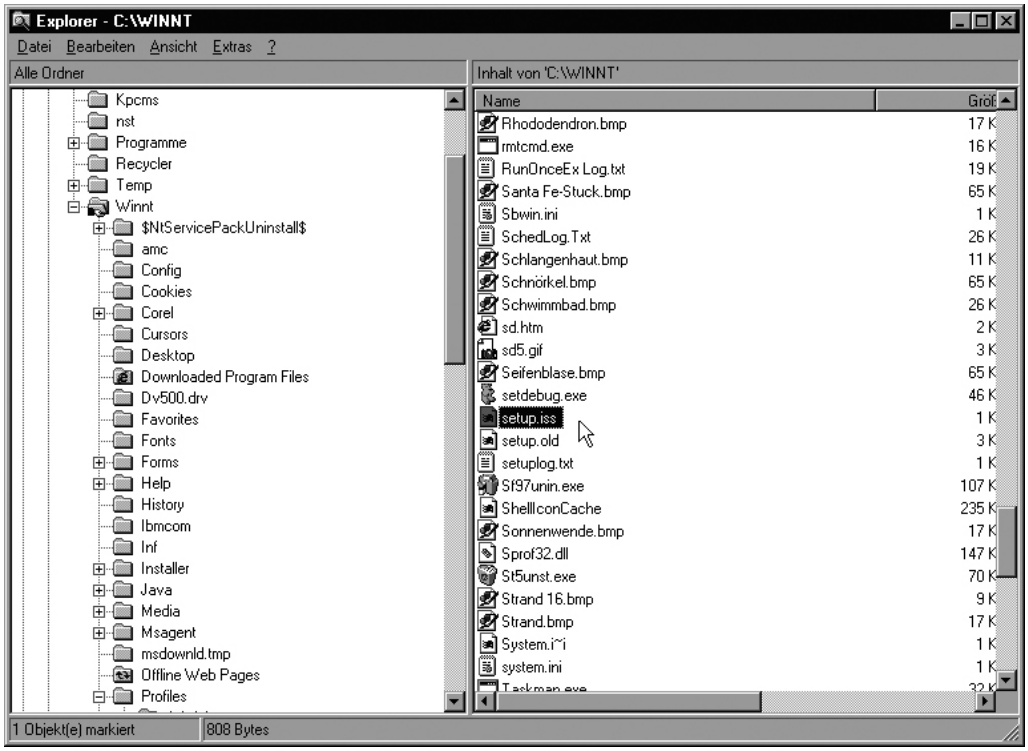
Ich deaktiviere hier den Parameter „Infodatei anzeigen".

Im folgenden Dialog bestätige ich durch eine entsprechende Eingabe, dass mein Windows PC jetzt heruntergefahren und erneut gestartet wird.

7.2.4.1
Seite 10

Computer neu starten

Nachdem mein Windows-System wieder verfügbar ist, sehe ich zuerst einmal im Windows-Explorer nach, ob meine Antwortdatei erstellt wurde.



SETUP.ISS wurde gefunden

Sie sehen im obigen Bild, dass im Verzeichnis WINNT meines Systems die Datei SETUP.ISS erzeugt wurde. Diese Datei werden wir uns einmal näher ansehen.

Die Antwortdatei SETUP.ISS ist eigentlich von ihrem Aufbau her nichts anderes als eine Windows-Initialisierungsdatei. Diese Datei ist in so genannte Paragraphen aufgeteilt, deren Namen man sehr leicht dadurch erkennt, dass sie in eckigen Klammern eingeschlossen sind.

Der erste Paragraph ist ein Header-Abschnitt, der für den Install Shield – das ist das Programm SETUP-EXE – benötigt wird. Für uns ist dieser Abschnitt weniger wichtig.

```
[InstallShield Silent]
Version=v5.00.000
File=Response File
```

In dem Paragraphen „DlgOrder“ wird die Dialog-Reihenfolge innerhalb des Install Shield benannt. Sehen Sie sich einmal die Dialog-Namen an, und vergleichen Sie diese Namen mit den Dialogen, die wir soeben bei der Aufnahmesitzung für unsere Antwortdatei durchlaufen haben. Sie werden feststellen, dass hier die Dialoge aufgezählt werden, die Sie oben als Hardcopys sehen konnten.

```
[DlgOrder]
Dlg0=SdWelcome-0
Count=9
Dlg1=SdLicense-0
Dlg2=SdOptionsButtons-0
Dlg3=SdAskDestPath-0
Dlg4=CwbComponentDlg-0
Dlg5=SdShowDlgEdit1-0
Dlg6=SdStartCopy-0
Dlg7=SdAskOptions-0
Dlg8=SdFinishReboot-0
```

Jetzt wird festgelegt, welche Aktionen aus den einzelnen Dialogen heraus ausgeführt werden sollen. Der erste Dialog des Installationsvorgangs ist der Begrüßungsdialog. In diesem Dialog kann der Anwender nichts anderes machen, als auf den Knopf „Weiter“ zu drücken. Durch „Result=1“ wird genau diese Aktion bei der unbeaufsichtigten Installation ausgelöst.

```
[SdWelcome-0]
Result=1
```

7.2.4.1
Seite 12

```
[SdLicense-0]
Result=1
```

Im folgenden Paragraphen wird der Installationstyp gesetzt. In unserem Fall ist das die „Angepasste“ Installation.

```
[SdOptionsButtons-0]
Result=103
```

Danach wird jetzt der Speicherort für den iSeries Access Express bestimmt.

```
[SdAskDestPath-0]
szDir=D:\Programme\IBM\iSeries Access
Result=1
```

Jetzt geht es los. Im Paragraphen „CwbComponentDLG“ werden jetzt alle während der Aufnahmesitzung ausgewählten Komponenten des iSeries Access Express aufgelistet.

```
[CwbComponentDlg-0]
CAOptional-type=string
CAOptional-count=4
CAOptional-0=CAOptional\DirUpdate
CAOptional-1=CAOptional\IRC
CAOptional-2=CAOptional\MAPI
CAOptional-3=CAOptional\OUG
Unity-type=string
Unity-count=14
Unity-0=Unity\Base
Unity-1=Unity\BasicOp
Unity-2=Unity\JobMan
Unity-3=Unity\SysConfig
Unity-4=Unity\Network
Unity-5=Unity\Security
Unity-6=Unity\UserGroups
Unity-7=Unity\Database
Unity-8=Unity\FileSys
Unity-9=Unity\Multimedia
Unity-10=Unity\Backup
Unity-11=Unity\AppDev
```

```
Unity-12=Unity\ManCentral
Unity-13=Unity\Admin
DataAccess\FileTransfer-type=string
DataAccess\FileTransfer-count=3
DataAccess\FileTransfer-0=DataAccess\FileTrans-
fer\DataXfer
DataAccess\FileTransfer-1=DataAccess\FileTrans-
fer\Excel
DataAccess\FileTransfer-2=DataAccess\FileTrans-
fer\WK4
DataAccess-type=string
DataAccess-count=3
DataAccess-0=DataAccess\FileTransfer
DataAccess-1=DataAccess\ODBC
DataAccess-2=DataAccess\OLEDB
Emulators\Standard\PCFont-type=string
Emulators\Standard\PCFont-count=1
Emulators\Standard\PCFont-0=Emulators\Standard\
PCFont\NoAddFonts
Emulators\Standard-type=string
Emulators\Standard-count=3
Emulators\Standard-0=Emulators\Standard\Base
Emulators\Standard-1=Emulators\Standard\PdfPdt
Emulators\Standard-2=Emulators\Standard\PCFont
Emulators-type=string
Emulators-count=1
Emulators-0=Emulators\Standard
PrinterDrivers-type=string
PrinterDrivers-count=2
PrinterDrivers-0=PrinterDrivers\AFP
PrinterDrivers-1=PrinterDrivers\SCS
Toolkit-type=string
Toolkit-count=2
Toolkit-0=Toolkit\Base
Toolkit-1=Toolkit\VBW
Component-type=string
Component-count=13
```

7.2.4.1
Seite 14

```
Component-0=Install
Component-1=Base
Component-2=CAOptional
Component-3=Unity
Component-4=DataAccess
Component-5=MDAC
Component-6=AFPViewer
Component-7=JRE
Component-8=JAVATB
Component-9=Emulators
Component-10=PrinterDrivers
Component-11=OpCon
Component-12=Toolkit
Result=1
```

Im Folgenden sehen Sie die Angabe des Programmordners für den iSeries Access Express.

```
[SdShowDlgEdit1-0]
szEdit1=IBM AS400 iSeries Access Express
Result=1
```

Der Kopiervorgang wird ausgelöst.

```
[SdStartCopy-0]
Result=1
[Application]
Name=iSeries Access
Version=CurrentVersion\Selectively_Installable_Components\Toolkit
VB Wizard
Company=IBM
```

Die Verknüpfung zum Desktop wird hinzugefügt.

```
[SdAskOptions-0]
Component-type=string
Component-count=1
Component-0=Verknüpfung für Programmordner zu Desktop hinzufügen
Result=1
```

Das System soll nach der Beendigung des Installationsvorgangs neu gebootet werden.

```
[SdFinishReboot-0]
Result=1
BootOption=3
```

Soll das System nicht erneut gebootet werden, setzen Sie den Parameter „BootOption“ auf den Wert „Null“ (0). Meine Erfahrung ist, dass bei Aufnahmevorgängen oftmals der Neustart des Systems nicht gefordert wird. Sollte in Ihrer Antwortdatei der Parameter „BootOption“ nicht auf den Wert „3“ gesetzt worden sein, so sollten Sie dies händisch nachtragen.

„BootOption“ auf „3“ – also Neustart des Systems – zu setzen, hat einen positiven Nebeneffekt: Wird eine Silent-Installation durchgeführt, so läuft diese asynchron im System. Nach Eingabe des entsprechenden Setup-Befehls steht das System sofort zur Weiterarbeit zur Verfügung. Von daher ist es manchmal relativ schwierig, zu bewerten, ob eine unbeaufsichtigte Installation wirklich zu Ende ist. Wird das System nach dem Ende der Installation gestartet, so sehen Sie als positives Zeichen, dass die Silent-Installation beendet wurde.

7.2.4.1
Seite 16

Die unbeaufsichtigte Installation durchführen

Für die unbeaufsichtigte Installation wird wiederum das Programm SETUP.EXE des iSeries Access Express aufgerufen. Auch hier ist es notwendig, einen speziellen Schalter zu setzen.

Im Folgenden sehen Sie den schematischen Aufbau des Befehls:

```
SETUP.EXE -s          -f1LW\DIR\ANTWORT.ISS
                      -f2LW\DIR\PROTOKOLL.LOG
```

Parameter	
-s	Steht für silent = unbeaufsichtigt Die Installation wird ohne interaktive Aktionen des Installierenden durchgeführt.
-f1	LW = Laufwerk DIR = Verzeichnis ANTWORT.ISS = Name der Antwortdatei Der Parameter „-f1" ist ein optionaler Parameter. Wird er gesetzt, so kann der Standort und der Name der aufgezeichneten Antwortdatei genau beschrieben werden. Es ist notwendig, dass eine Antwortdatei die Namenserweiterung „ISS" trägt. Wird dieser Parameter nicht gesetzt, was möglich ist, so wird automatisch eine Antwortdatei mit dem Namen „SETUP.ISS" gesucht. Diese Datei muss sich in dem Verzeichnis befinden, aus dem heraus das Programm SETUP.EXE aufgerufen wird.
-f2	Der Parameter „-f2" ist ein optionaler Parameter. Wird er gesetzt, so kann der Standort und der Name einer Protokolldatei für die unbeaufsichtigte Installation genau beschrieben werden. Wird dieser Parameter nicht gesetzt, was möglich ist, so wird automatisch eine Protokolldatei mit dem Namen „SETUP.LOG" erzeugt. Diese Datei befindet sich in dem Verzeichnis, aus dem heraus das Programm SETUP.EXE aufgerufen wird.

Auf meinem Windows PC wird das Installations-Image für den iSeries Access Express im folgenden Pfad gehalten:

F:\SWLager\CAExpress\Express

In diesem Pfad befindet sich das Programm SETUP.EXE.

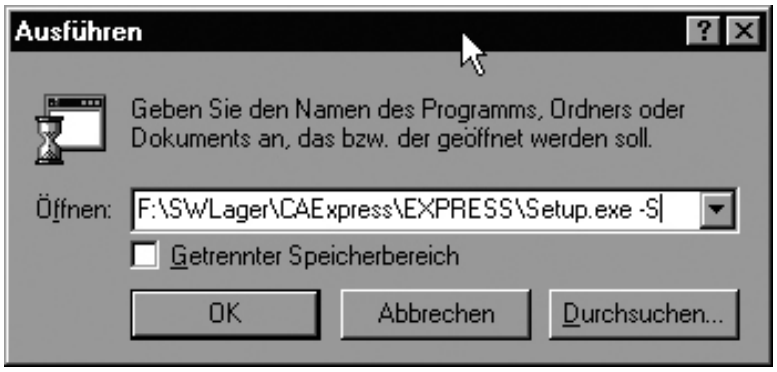
Ich werde also die Antwortdatei SETUP.ISS aus dem Windows NT-Installationsverzeichnis – hier WINNT – in den Pfad „F:\SWLager\CAExpress\Express" kopieren.

7.2.4.2
Seite 2

Dann rufe ich das Setup-Programm wie folgt auf:

F:\SWLager\CAExpress\Express\Setup.exe -s

Da sich die Datei SETUP.ISS im Pfad des Setup-Programms befindet, wird sie automatisch zur unbeaufsichtigten Installation herangezogen. Nach Beendigung der Installation wird im Pfad „F:\SWLager\CAExpress\Express“ die Protokolldatei SETUP.LOG erzeugt.



Aufruf der unbeaufsichtigten Installation

Das Protokoll der unbeaufsichtigten Installation

Wie schon vorher gesagt, wird am Ende der Silent-Installation eine Protokolldatei erzeugt. Da ich nicht mit dem f2-Parameter gearbeitet habe, befindet sich jetzt eine Datei mit dem Namen SETUP.LOG in dem Verzeichnis, aus dem heraus das Programm SETUP.EXE aufgerufen wurde.

Sehen wir uns den Inhalt dieser Datei einmal an. Auch diese Datei folgt wieder dem Aufbau einer Windows Initialisierungs-Datei.

```
[InstallShield Silent]
Version=v5.00.000
File=Log File
```

```
[Application]
Name=iSeries Access
Version=CurrentVersion\Selectively_Installable_Components\Toolkit
VB Wizard
Company=IBM
```

Der Paragraph „ResponseResult“ ist für uns der wichtige! Hier können wir nämlich sehen, wie die unbeaufsichtigte Installation beendet wurde.

```
[ResponseResult]
ResultCode=0
```

„ResultCode = 0“ ist die positive Bestätigung, dass der Silent Setup des iSeries Access Express gelungen ist.

Folgende „ResultCodes“ sind möglich:

0	Erfolg!
-1	Allgemeiner Fehler
-2	Ungültiger Installationsmodus
-3	Erforderliche Daten nicht in Antwortdatei gefunden
-4	Nicht genügend Speicher
-5	Datei nicht vorhanden
-6	Antwortdatei kann nicht erstellt werden
-7	Es kann nicht in die Protokolldatei geschrieben werden
-8	Ungültige Änderung in der Antwortdatei
-9	Kein gültiger Typ in der Liste (Zeichenfolge oder numerisch)
-10	Ungültiger Datentyp
-11	Unbekannter Fehler während des Setups
-12	Falsche Dialogfolge

7.2.4.3
Seite 2

-51	Ordner kann nicht erstellt werden
-52	Kein Zugriff auf Datei oder Ordner
-53	Ungültige Komponente selektiert

Ich hoffe für Sie, dass Sie immer den Wert „Null“ (0) sehen.

Was man weiß, was man wissen sollte

Haben Sie Ihre *.ISS- bzw. *.LOG-Dateien mit langen Namen versehen, wie es Windows 95/98/NT und 2000 zulassen, dann muss der gesamte Pfadname in Anführungsstriche gesetzt werden.

Beispiele:

SETUP.EXE -S -f1“c:\Silent Setup.iss“

SETUP.EXE -S -f1“c:\Silent Setup.iss“
-f2”c:\Install Protokolle\Silent Setup.log”

7.2.4.4

Seite 2

Installationsimages erzeugen

Wenn Sie in einer Umgebung arbeiten, in der viele gleichartige System i Access für Windows-Konfigurationen zu erstellen sind, so lässt sich diese Arbeit besser in den Griff bekommen, wenn Sie für alle gleichartigen Installationen ein sogenanntes Installationsimage erzeugen.

Bisher war die Erzeugung eines Images eine manuelle und fehleranfällige Angelegenheit, musste man doch die Dateien der Installations-CD zerpflücken und funktionsbezogen neu gliedern.

Mit System i Access für Windows V6R1 ist die Erzeugung eines Images gegenüber den Vorgängerversionen grundlegend geändert worden, da nun auch hier der MSI-Installer die Kontrolle übernommen hat.

Beginnen wir mit einer Begriffsklärung: Ein erzeugtes Installationsimage wird in der Nomenklatur des MSI-Installers ein Verwaltungsimage genannt. Beabsichtigen wir also ein Installationsimage zu erzeugen, so erzeugen wir ein Verwaltungsimage. Verwaltungsimages bilden den gesamten Funktionsumfang des System i Access ab.

7.2.5 Seite 2

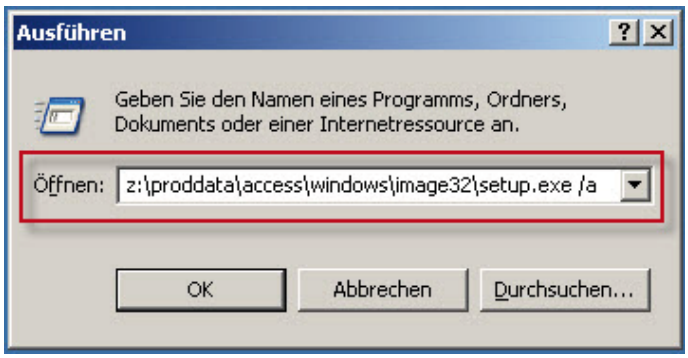
Verwaltungsimage erzeugen

Wie erzeugen wir ein Verwaltungsimage, das den kompletten Funktionsinhalt des System i Access bereitstellt?

Hier der entprechende Befehl: Setup.exe /a

Lassen Sie uns diesen Vorgang am Beispiel nachvollziehen. Ich werde das Verwaltungsimage aus dem im Pfad /QIBM/Proddata/Access/Windows/Image32 meines Power Systems gespeicherten Originalimage erzeugen.

Ich öffne den Dialog „Windows-Ausführen“ mit der Windows-Taste + R.

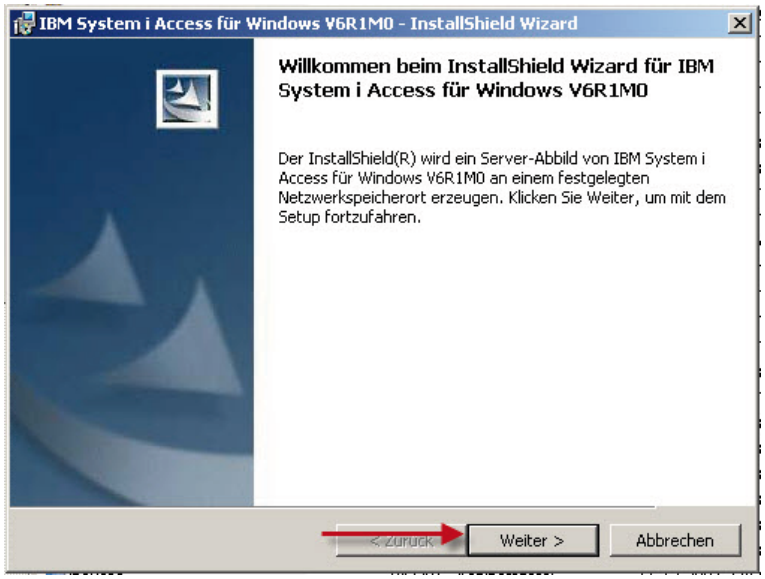


Neustart von Windows anfordern

Installationssprache

7.2.5.1
Seite 2

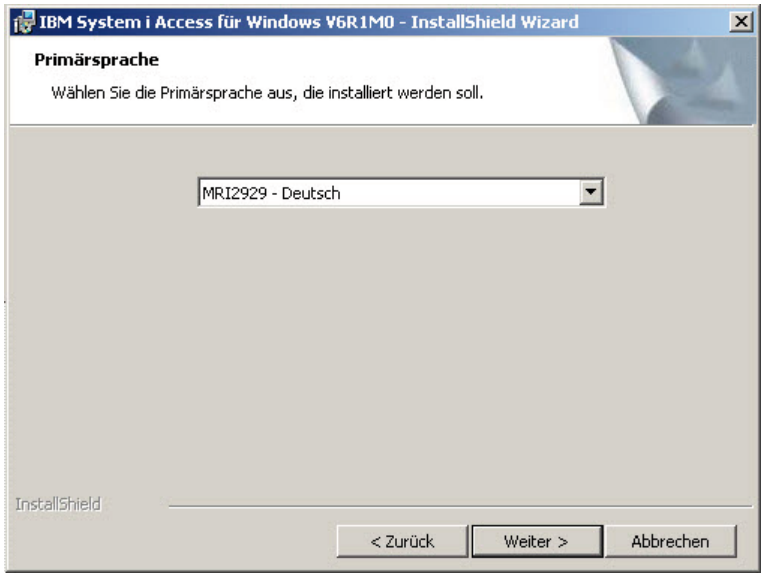
Sie werden jetzt aufgefordert, die Sprache für die Installationsdialoge festzulegen.



Willkommen

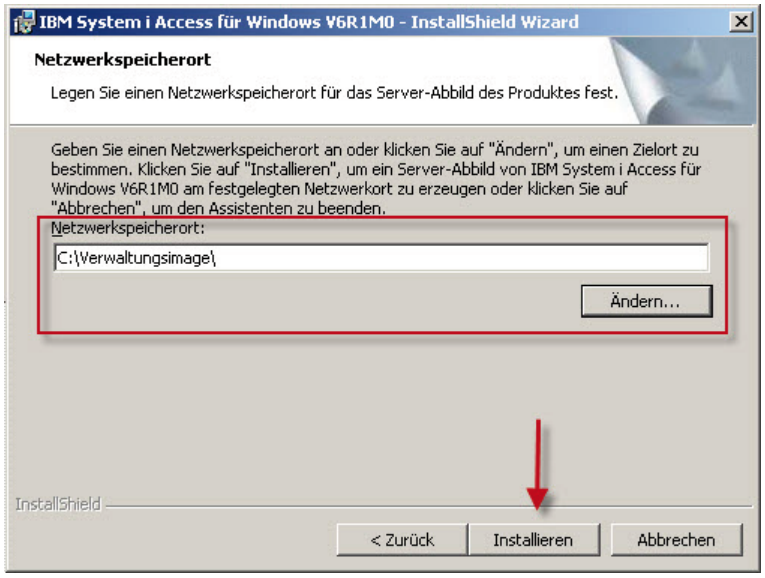
Den Willkommendialog beenden Sie mit „Weiter".

Keine Angst, Sie erstellen jetzt keine neue Installation! Die auf Ihrem Arbeitsplatz befindliche System i Access für Windows-Installation wird durch diesen Vorgang nicht berührt.



Primärsprache

Legen Sie jetzt die Primärsprache Ihres Verwaltungsimages fest und klicken Sie auf „Weiter".

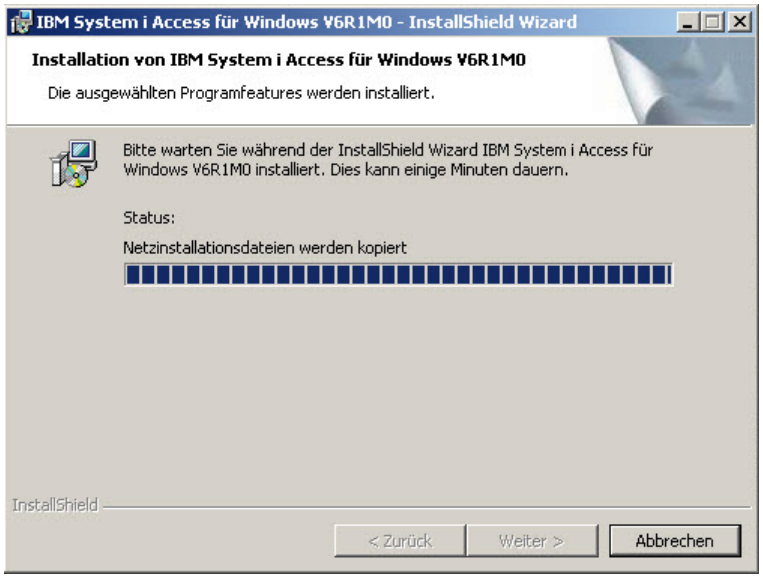


Netzwerkspeicherort

Jetzt legen Sie fest, in welchem Verzeichnis das Verwaltungsimage erzeugt werden soll. Beabsichtigen Sie, viele Clients zu installieren, ist eine Speicherung des Verwaltungsimages auf einem Fileserver sicherlich sinnvoll.

In diesem Beispiel verwende ich jedoch einen lokalen Pfad: c:\Verwaltungsimage.

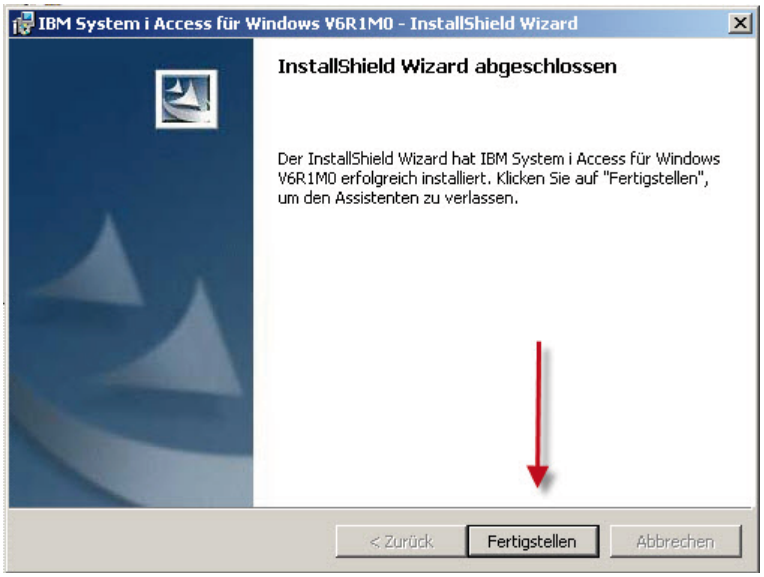
Ich beende den Dialog mit „Installieren“.



Erstellungsfortschritt

7.2.5.1
Seite 4

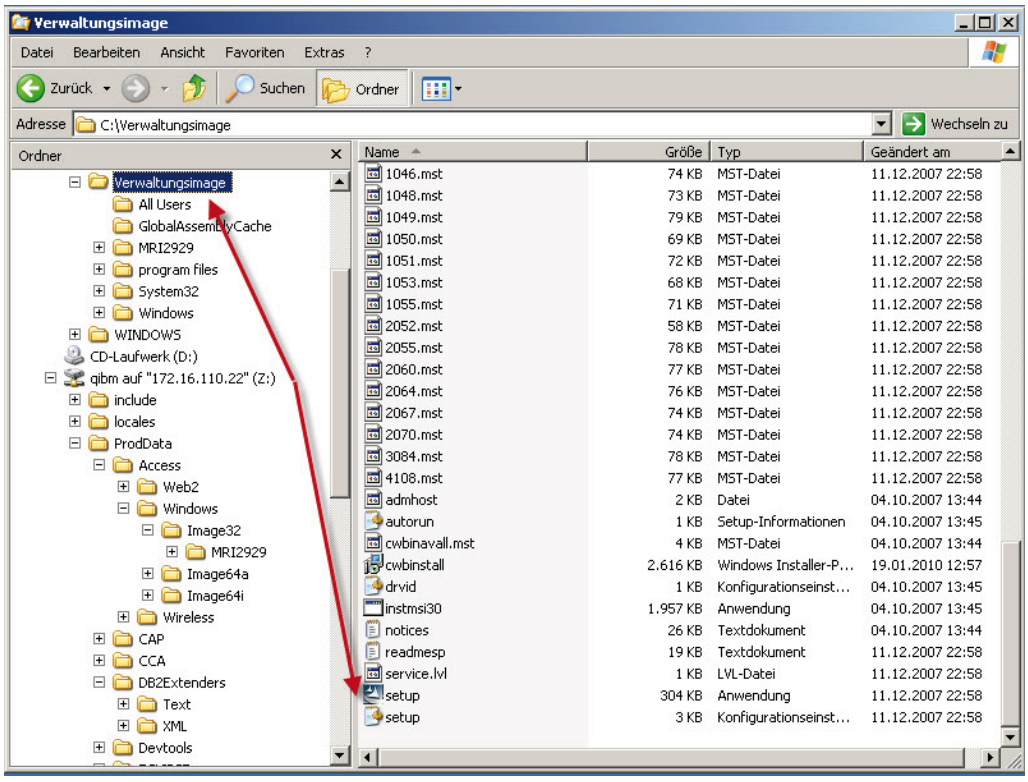
Der Fortschritt bei der Erstellung des Verwaltungsimages wird mit einem Laufbalken angezeigt.



Fertigstellen

Beenden Sie den Prozess mit „Fertigstellen“.

In der folgenden Abbildung sehen Sie das im Verzeichnis c:\Verwaltungsimage erstellte Verwaltungsimage.



Das fertige Verwaltungsimage

Rufe ich jetzt Setup.exe aus dem Verwaltungsimageverzeichnis auf, so wird der bereits am Anfang gezeigte Installationsdialog aufgerufen.

Beabsichtigen wir aus der Gesamtmenge aller Funktionen lediglich eine Teilmenge zu installieren, so wird der bereits bekannte Feature-Auswahldialog uns zu einer angepassten Client-Installation des System i Access führen.

7.2.5.1
Seite 6

Angepasste Installation mit Kommandozeilenparametern erzeugen

Nachdem ich das Verwaltungsimage erzeugt habe, geht es in diesem Abschnitt darum, eine angepasste Installation von nur einigen wenigen Access-Funktionen durchzuführen.

Grundsätzlich wäre es jetzt machbar, das Setup-Programm direkt aufzurufen und interaktiv eine Komponentenauswahl zu treffen. Ich möchte jedoch von diesem Weg abweichen und Ihnen zeigen, dass wir mittels einer geschickten Wahl von Kommandozeilenparametern die Installation so gestalten können, dass am Ende keine Dialoge erscheinen, so dass wir von einer stillen Installation (Silent Installation) ausgehen können.

Der Hauptschlüssel zu einer angepassten, nicht interaktiven Installation sind sogenannte Öffentliche Eigenschaften, die man dem Setup-Befehl zuordnen kann.

Um dem Setup-Programm Parameter zu übergeben, wird er folgendermaßen erweitert: setup.exe /v

Hinter den Parameter /v werden die Öffentlichen Eigenschaften einfach angehängt. Öffentliche Eigenschaften verwenden folgenden syntaktischen Formalismus: ÖFFENTLICHEEIGENSCHAFT=Wert

Formal betrachtet würde unser Aufruf des Setup-Programms dann wie folgt aussehen: setup.exe /vÖFFENTLICHEEIGENSCHAFT=Wert

Beachten Sie, dass zwischen /v und dem Namen der Öffentlichen Eigenschaft keine Leerstelle stehen darf!

Soll der Setup-Prozess durch mehrere Eigenschaften gesteuert werden, sieht der Aufruf folgendermaßen aus: setup.exe /v“ÖE1=Wert1 ÖE2=Wert2“

In diesem Fall müssen die Eigenschaften als Gesamtzeichenfolge durch Anführungszeichen dargestellt werden.

Dieser Sachverhalt gilt auch dann, wenn der Wert einer Eigenschaft ein Leerzeichen enthält.

setup.exe /v“ÖE1=Wert1‘

24. Ergänzung 9/2010

7.2.5.2
Seite 2

Welche Öffentlichen Eigenschaften können verwendet werden:

- CWBINSTALLTYPE
- CWBPC5250VERSION
- CWBSHOWWELWIZ
- CWBPRIMARYLANG
- CWBINSTALLPLUGINS
- CWBSILENTPLG
- ADDLOCAL (Eigenschaft von Windows Installer)
- REMOVE (Eigenschaft von Windows Installer)
- REBOOT (Eigenschaft von Windows Installer)
- REBOOTPROMPT (Eigenschaft von Windows Installer)
- INSTALLDIR (Eigenschaft von InstallShield)
- TARGETDIR

Lassen Sie mich mit der allgemeinen Erklärung für die Öffentlichen Eigenschaften beginnen. Danach werden wir Öffentliche Eigenschaften in einem Beispiel verwenden, um eine Installation von System i Access durchzuführen.

CWBINSTALLTYPE
Diese Eigenschaft legt den Installationstyp für eine Erstinstallation fest.

Geltungsbereich:
- Sie ist nur bei einer Erstinstallation gültig.

Standardwert:
- Der Standardinstallationstyp lautet „Complete“.

Werte:
- Complete (= Vollständig),
- Custom (= Angepasst),
- PC5250User (= PC5250-Benutzer)

Beispiel:
Für eine vollständige Installation geben Sie ein:

```
setup /vCWBINSTALLTYPE=Complete .
```

Zusätzliche Hinweise:
- Bei den Werten muss die Groß-/Kleinschreibung nicht beachtet werden.
- Die Eigenschaft CWBINSTALLTYPE wird bei einem Upgrade von einem Vorgängerrelease von System i Access für Windows sowie bei einer Wartungsinstallation ignoriert.

CWBPC5250VERSION
Diese Eigenschaft legt die Version für die PC5250-Anzeige- und -Druckeremulation fest.

Geltungsbereich:
Sie ist bei einer Erstinstallation, einem Upgrade und einer Wartungsinstallation (falls die PC5250-Anzeige- und -Druckeremulation noch nicht installiert wurde) gültig.

Standardwert:
Wenn die Eigenschaft nicht verwendet wird, wird die zu installierende Version des Emulators anhand der PC-Codepage ermittelt.

Werte:
- K
  Die Version der PC5250-Anzeige- und -Druckeremulation für Koreanisch wird installiert.
- P
  Die Version der PC5250-Anzeige- und -Druckeremulation für vereinfachtes Chinesisch wird installiert.
- S
  Die Standardversion der PC5250-Anzeige- und -Druckeremulation wird installiert.
- T
  Die Version der PC5250-Anzeige- und -Druckeremulation für traditionelles Chinesisch wird installiert.

Beispiel:
Zur Installation der koreanischen Version für PC5250 geben Sie ein:

```
setup /vCWBPC5250VERSION=K
```

Zusätzliche Hinweise:
- Falls PC5250 installiert ist und Sie eine Wartungsinstallation vornehmen, wird diese Eigenschaft ignoriert. Zur Installation einer anderen Version müssen Sie die aktuelle Version von PC5250 deinstallieren und dann die Wartungsinstallation unter Verwendung dieser Eigenschaft erneut starten.
- Falls PC5250 installiert ist und Sie ein Upgrade auf ein neues Release vornehmen, wird diese Eigenschaft ignoriert. Zur Installation einer anderen Version müssen Sie die aktuelle Version von PC5250 deinstallieren und dann eine Wartungsinstallation unter Verwendung dieser Eigenschaft starten.

- Durch das Festlegen dieser Eigenschaft wird das Feature „PC5250-Anzeige- und -Druckeremulation“ sogar dann für die Installation vorausgewählt, wenn Sie den Featurenamen nicht mit der Eigenschaft ADDLOCAL übergeben.

CWBSHOWWELWIZ
Diese Eigenschaft gibt an, wann der Startassistent angezeigt werden soll.

Geltungsbereich:
Sie ist bei einer Erstinstallation und einem Upgrade gültig.

Standardwert: Yes

Werte:
- Yes
  Der Startassistent wird bei jedem PC-Neustart angezeigt.
- No
  Der Startassistent wird nicht angezeigt.

Beispiel:
Damit der Startassistent bei jedem PC-Neustart angezeigt wird, geben Sie ein:

```
setup /vCWBSHOWWELWIZ=yes
```

CWBPRIMARYLANG
Diese Eigenschaft gibt die Standardprimärsprache an, die installiert werden soll.

Geltungsbereich:
Sie ist bei einer Erstinstallation, einem Upgrade und einer nicht überwachten Installation gültig.

Standardwert:
Die voreingestellte Sprache ist die Primärsprache, wenn das IBM Lizenzprogramm als Quelleninstallationsimage verwendet wird. Dies wird durch eine Überprüfung der Datei „mrisetup.inii“ im Quellenverzeichnis ermittelt. Wird die DVD oder ein anderes Medium als Installationsquelle verwendet, ist die voreingestellte Sprache die landessprachliche Version, die mit der Ländereinstellung des PCs übereinstimmt, bzw. die Auswahl des Benutzers in der Benutzerschnittstelle.

Werte:
- MRI29xx
  Die Sprachenressourcen für MRI29xx werden auf dem PC installiert und MRI29xx wird als Primärsprache für System i Access für Windows festgelegt. Der Platzhalter xx steht für die Sprachenkennung.

Beispiel:
Um die Primärsprache Deutsch auszuwählen und den Dialog für die Primärsprache zu überspringen, geben Sie ein:

```
setup /vCWBPRIMARYLANG=MRI2929
```

Zusätzliche Hinweise:
Falls diese Eigenschaft bei einer Installation mit einer Basisbenutzerschnittstelle, mit einer reduzierten Schnittstelle oder ohne Benutzerschnittstelle nicht über die Befehlszeile übergeben wird, wird die Standardsprache installiert.

Wenn die Eigenschaft CWBPRIMARYLANG in der Befehlszeile übergeben wird, wird der Dialog für die Primärsprache übersprungen.

Bei Upgrades für eine Installation ohne MSI wird der Dialog für die Primärsprache nicht angezeigt, und es wird dieselbe Sprache installiert.

CWBINSTALLPLUGINS

Diese Eigenschaft gibt an, ob der Installationsprozess Plug-ins für System i Navigator von Drittherstellern erkennen und installieren soll.

Geltungsbereich:
Sie ist nur bei Erstinstallationen, bei Wartungsinstallationen und bei Upgrades gültig.

Standardwert:
Die Erkennung und Installation von Plug-ins ist zulässig.

Werte:

- No
- Yes

Beispiel:
Damit Plug-ins erkannt und installiert werden, geben Sie ein:

```
setup /vCWBINSTALLPLUGINS=No
```

Zusätzlicher Hinweis:
Bei diesem Wert muss die Groß-/Kleinschreibung nicht beachtet werden.

CWBSILENTPLG
Diese Eigenschaft gibt an, welche Plug-ins von System i Navigator bei einer nicht überwachten Installation (mit Basisbenutzerschnittstelle, reduzierter Benutzerschnittstelle oder ohne Benutzerschnittstelle) installiert werden sollen.

Geltungsbereich:
Sie ist nur bei Erstinstallationen, bei Wartungsinstallationen und bei Upgrades gültig, die im Modus mit reduzierter Benutzerschnittstelle, mit Basisbenutzerschnittstelle oder ohne Benutzerschnittstelle ausgeführt werden.

Standardwert:
Keiner

Werte:
Der Wert ist ein Pfad zur Datei „cwbsilentplg.ini“, die eine Liste der zu installierenden Plug-ins enthält. Die Datei hat folgendes Format:

```
[Plugins]
hersteller.komponente=yes hersteller.komponente=no
```

Hierbei steht hersteller für den Firmennamen und komponente für den Plug-in-Namen.

Beispiel:
Um bei einem Upgrade oder einer Wartungsinstallation das Plug-in „IBM. BRMSPlugin“ zu installieren und das Plug-in „LOTUS.QUICKPLACE“ zu entfernen, erstellen Sie eine Datei namens „cwbsilentplg.ini“ im Pfad „C:\ temp“. Die Datei sollte die folgenden Informationen enthalten:

```
[Plugins]
IBM.BRMSPlugin=yes
LOTUS.QUICKPLACE=no
```

Geben Sie dann in der Befehlszeile Folgendes ein: setup

```
/vCWBSILENTPLG=c:\temp
```

Zusätzlicher Hinweis:
Falls Sie bei einer nicht überwachten Installation die Eigenschaft CWBSILENTPLG nicht angeben, werden keine Plug-ins installiert oder entfernt. Bei einer anderen Installation wird diese Eigenschaft ignoriert.

ADDLOCAL
Diese Eigenschaft listet die Features auf, die lokal installiert werden sollen.

Geltungsbereich:
ADDLOCAL ist bei einer Erstinstallation, einer Wartungsinstallation oder einem Upgrade gültig.

Standardwert:
Für die Eigenschaft ADDLOCAL gibt es keinen Standardwert. Falls die Öffentliche Eigenschaft nicht verwendet wird, bestimmen die im Installationsprogramm oder für die Umsetzung (sofern bereitgestellt) ausgewählten Optionen, welche Features installiert werden.

Werte:

Feature	Interner Name des Features
Alle	all
Erforderliche Programme	req
Verzeichnisaktualisierung	dir
Eingehender ferner Befehl	irc
Datenübertragung	dt
Excel-Add-in für Datenübertragung	dtexcel
ODBC	odbc
OLE DB-Provider	oledb
Unterstützung für Lotus 123-Dateiformat	lotus123
.NET Data Provider	dotnet
AFP Workbench Viewer	viewer
5250-Anzeige- und -Druckeremulation	emu
Secure Socket Layer (SSL)	ssl
AFP-Druckertreiber	afp
SCS-Druckertreiber	scs
Operations Console	oc
Header, Bibliotheken und Dokumentation	hld
System i Navigator (Basisunterstützung)	inav
Basisoperationen	inavbo
Ablaufsteuerung	inavwm
Konfiguration und Services	inavcfg
Netzwerke	inavnet
Verwaltung für integrierten Server	inavisa
Sicherheit	inavsec
Benutzer und Gruppen	inavug
Datenbanken	inavdb

Dateisysteme	inavfs
Sicherung	inavback
Befehle	inavcmd
Pakete und Produkte	inavpp
Überwachungen	inavmon
Logische Systeme	inavlog
AFP-Manager	inavafp
Anwendungsverwaltung	inavad
Toolbox für Java	tbj

Bei dem Namen des Features muss die Groß-/Kleinschreibung nicht beachtet werden.

Beispiel:
/vADDLOCAL=ALL

Zusätzliche Hinweise:
- Die Features müssen in der Tabelle „Feature“ aufgelistet sein.
- Bei den Featurenamen muss die Groß-/Kleinschreibung nicht beachtet werden.
- Geben Sie den Wert „ADDLOCAL=ALL“ nicht in der Eigenschaftentabelle ein, da hierdurch ein lokal installiertes Paket generiert wird, das nicht ordnungsgemäß entfernt werden kann.

REMOVE
Diese Eigenschaft gibt Features an, die entfernt werden sollen.

Geltungsbereich:
REMOVE ist nur gültig, wenn das Produkt installiert ist.

Standardwert:
Für die Eigenschaft REMOVE gibt es keinen Standardwert. Falls die Öffentliche Eigenschaft nicht verwendet wird, bestimmen die im Installationsassistenten ausgewählten Optionen, welche Features entfernt werden.

Werte:

Feature	Interner Name des Features
Alle	all
Erforderliche Programme	req
Verzeichnisaktualisierung	dir
Eingehender ferner Befehl	irc
Datenübertragung	dt
Excel-Add-in für Datenübertragung	dtexcel
ODBC	odbc

OLE DB-Provider	oledb
Unterstützung für Lotus 123-Dateiformat	lotus123
.NET Data Provider	dotnet
AFP Workbench Viewer	viewer
5250-Anzeige- und -Druckeremulation	emu
Secure Socket Layer (SSL)	ssl
AFP-Druckertreiber	afp
SCS-Druckertreiber	scs
Operations Console	oc
Header, Bibliotheken und Dokumentation	hld
System i Navigator (Basisunterstützung)	inav
Basisoperationen	inavbo
Ablaufsteuerung	inavwm
Konfiguration und Services	inavcfg
Netzwerke	inavnet
Verwaltung für integrierten Server	inavisa
Sicherheit	inavsec
Benutzer und Gruppen	inavug
Datenbanken	inavdb
Dateisysteme	inavfs
Sicherung	inavback
Befehle	inavcmd
Pakete und Produkte	inavpp
Überwachungen	inavmon
Logische Systeme	inavlog
AFP-Manager	inavafp
Anwendungsverwaltung	inavad
Toolbox für Java	tbj

Zusätzliche Hinweise:
Bei den Featurenamen muss die Groß-/Kleinschreibung nicht beachtet werden.

24. Ergänzung 9/2010

REBOOT
Diese Eigenschaft unterdrückt bestimmte Bedienerführungen für einen Warmstart des Systems.

Geltungsbereich:
REBOOT ist bei Erstinstallationen, bei Upgrades und bei Wartungsinstallationen gültig.

Standardwert:
Wenn Sie die Eigenschaft REBOOT nicht festlegen, wird am Ende der Installation ein ggf. erforderlicher Warmstart ausgeführt.

Werte:
- Force (oder F)
  Der Warmstart findet am Ende der Installation immer statt. Erfolgt die Installation ohne Benutzerschnittstelle, wird das System ohne Bestätigung erneut gestartet.
- Suppress (oder S)
  Der Warmstart am Ende der Installation wird unterdrückt.

Beispiel:
Damit das System am Ende einer Installation nicht erneut gestartet wird, geben Sie ein:

```
setup /v"REBOOT=Suppress"
```

oder

```
setup /v"REBOOT=S"
```

Zusätzliche Hinweise:
Das Installationsprogramm von System i Access für Windows plant immer dann einen Warmstart des Systems, wenn das Feature „Erforderliche Programme“ oder das Feature „AFP Viewer“ installiert oder erneut installiert wird.

Bei einer Erstinstallation, einem Upgrade oder einer erneuten Installation (Service-Pack oder Wartung) im nicht überwachten Modus startet Windows Installer den PC erneut, ohne eine Bestätigung anzufordern. Legen Sie diese Eigenschaft mit „S“ oder „Suppress“ fest, um einen Warmstart des Systems zu verhindern.

REBOOTPROMPT
Diese Eigenschaft bewirkt, dass bei einem Warmstart keine Bedienerführung für den Benutzer angezeigt wird.

Geltungsbereich:
REBOOTPROMPT ist nur bei einer Erstinstallation, einem Upgrade oder einer Wartungsinstallation mit einer Basisbenutzerschnittstelle, einer reduzierten Benutzerschnittstelle oder einer vollständigen Benutzerschnittstelle gültig.

Standardwert:
Falls die Eigenschaft REBOOTPROMPT nicht festgelegt ist und das Installationsprogramm einen Warmstart des Systems plant, wird am Ende der Installation ein Dialog ausgegeben.

Werte:
Suppress (oder S)
Jeder von Windows Installer durchgeführte Warmstart findet automatisch ohne Interaktion des Benutzers statt. Wenn Sie diese Eigenschaft festlegen, wird das Aufrufen von Bedienerführungen für den Benutzer hinsichtlich des Warmstarts unterdrückt.

Beispiel:
Um die Anzeige von warmstartbezogenen Bedienerführungen für den Benutzer zu unterdrücken, geben Sie ein:

```
setup /vREBOOTPROMPT=S
```

Zusätzlicher Hinweis:
Bei einer nicht überwachten Installation ist diese Eigenschaft nicht wirksam.

INSTALLDIR
Diese Eigenschaft legt das Installationsverzeichnis für System i Access für Windows fest.

Geltungsbereich:
INSTALLDIR ist nur bei einer Erstinstallation gültig.

Standardwert:
Falls die Eigenschaft INSTALLDIR nicht angegeben ist, wird System i Access für Windows im Verzeichnis „C:\Program Files\IBM\Client Access“ installiert.

Werte:
Beliebiger gültiger Pfad

Beispiel:
Um System i Access für Windows im Verzeichnis „C:\installationspfad“ zu installieren, geben Sie ein:

```
setup /vINSTALLDIR=X:\installationspfad.
```

Hierbei steht X für den Buchstaben des Ziellaufwerks und installationspfad für das Zielverzeichnis.

Wenn der Verzeichnispfad Leerzeichen enthält, verwenden Sie doppelte Anführungszeichen, und zwar jeweils für den Parameter „/v“ und die Eigenschaft INSTALLDIR. Bitte beachten Sie, dass den integrierten Anführungszeichen ein Backslash als Escapezeichen vorangestellt werden muss:

```
setup /v"INSTALLDIR=\"C:\mein installationspfad\""
```

TARGETDIR
Diese Eigenschaft legt das Zielverzeichnis für ein Verwaltungsimage fest, wenn eine Verwaltungsinstallation vorgenommen wird.

Geltungsbereich:
TARGETDIR ist bei einer Verwaltungsinstallation gültig.

Standardwert:
Falls die Eigenschaft TARGETDIR bei einer Verwaltungsinstallation nicht angegeben wird, verwendet Windows Installer als Wert das erste verbundene Netzlaufwerk, das gefunden wird.

Werte:
Gültiger Pfad

Beispiel:
Um das Zielverzeichnis für ein Verwaltungsimage festzulegen, geben Sie ein:

```
setup /a v/TARGETDIR=X:\verwaltungsimage
```

Hierbei steht X für den Buchstaben des Ziellaufwerks und verwaltungsimage für das Zielverzeichnis.

Wenn der Verzeichnispfad Leerzeichen enthält, verwenden Sie doppelte Anführungszeichen, und zwar jeweils für den Parameter „/v“ und die Eigenschaft TARGETDIR. Bitte beachten Sie, dass den integrierten Anführungszeichen ein Backslash als Escapezeichen vorangestellt werden muss:

```
setup /a /v"TARGETDIR=\"C:\mein verwaltungsimage\""
```

Nach dieser theoretischen Darstellung der gegebenen Steuerungsmöglichkeiten werden wir uns jetzt an ein konkretes Beispiel heranwagen. Ich hoffe, Sie erinnern sich daran, dass wir beabsichtigen, eine funktionell reduzierte Installation vorzunehmen.

Folgende Funktionen sollen installiert werden:

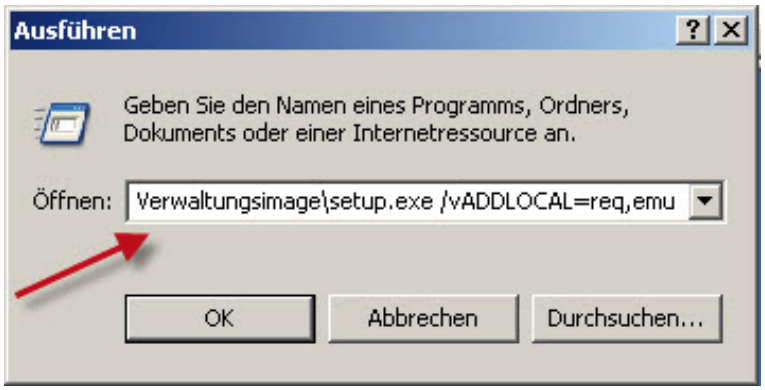
- Die sogenannten erforderlichen Features
- Die Bildschirmemulation

Um die Funktionsmenge einzugrenzen, verwenden wir die Öffentliche Eigenschaft ADDLOCAL.

Grundsätzlich wäre es jetzt denkbar, die Eigenschaftswerte req und emu anzufordern, so dass unser Setup-Befehl folgende Ausprägung hätte:

```
Setup /vADDLOCAL=req,emu
```

In meinem Client rufe ich diesen Befehl im „Ausführen"-Dialog auf:



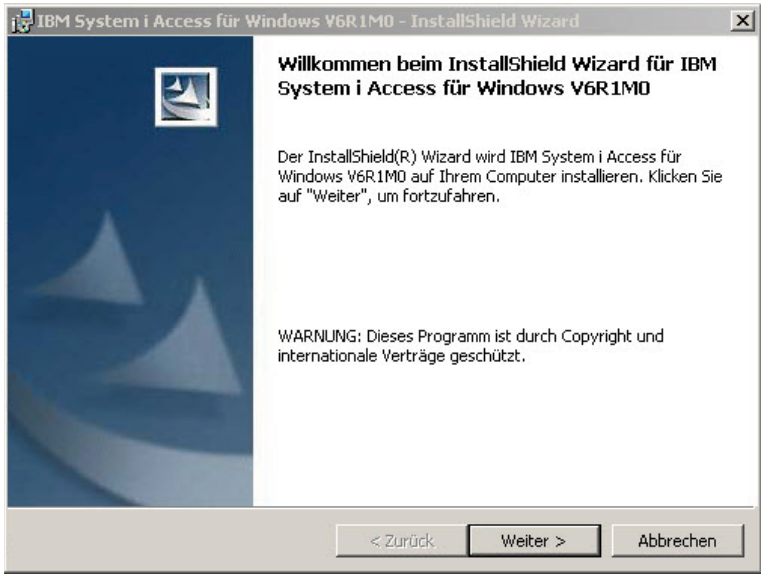
Reduziertes Setup aufrufen

Folgende Dialoge werden angezeigt:

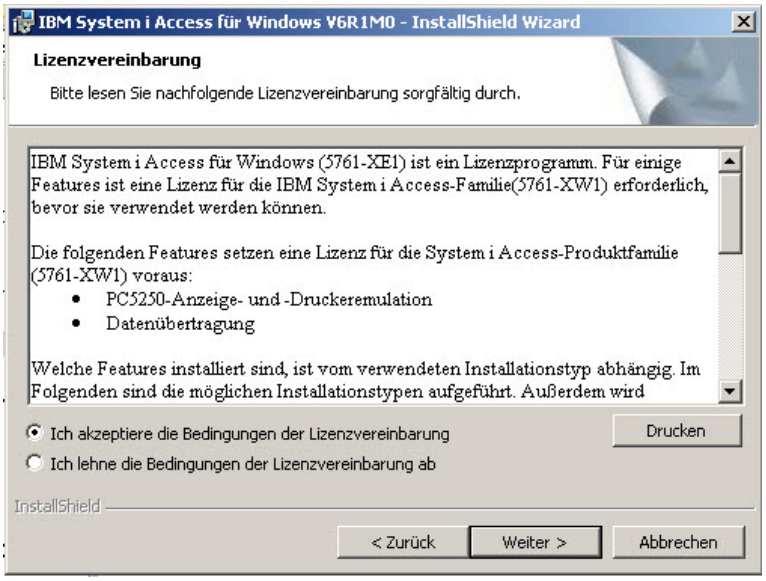
- Auswahl der Installationssprache

7.2.5.2
Seite 14

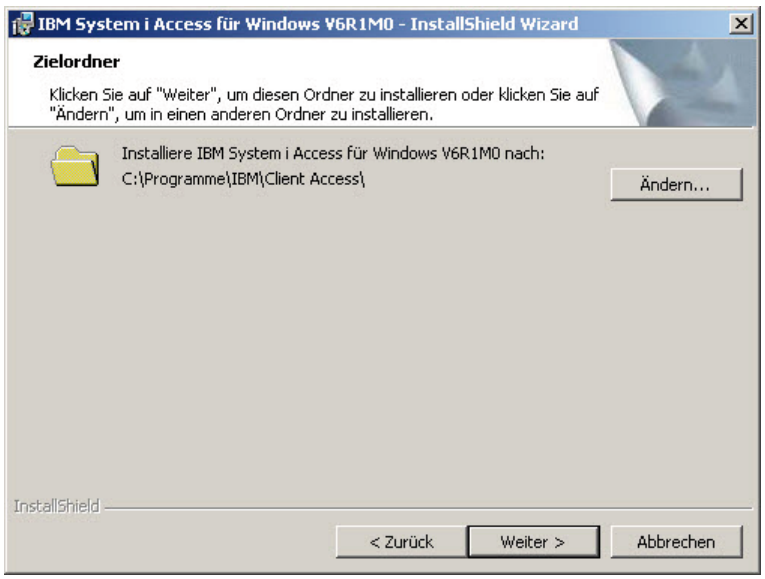
Willkommen-Dialog



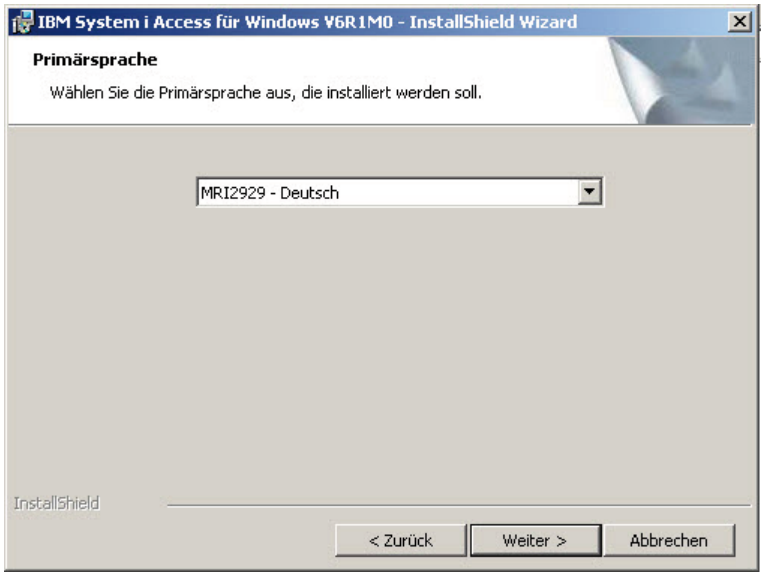
Lizenzabfrage



Installationspfad

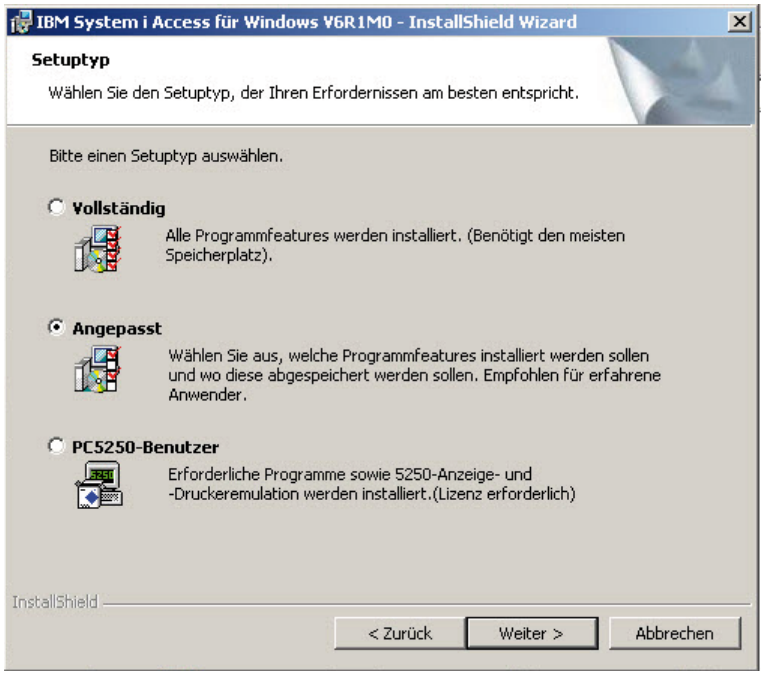


Primärsprache



7.2.5.2
Seite 16

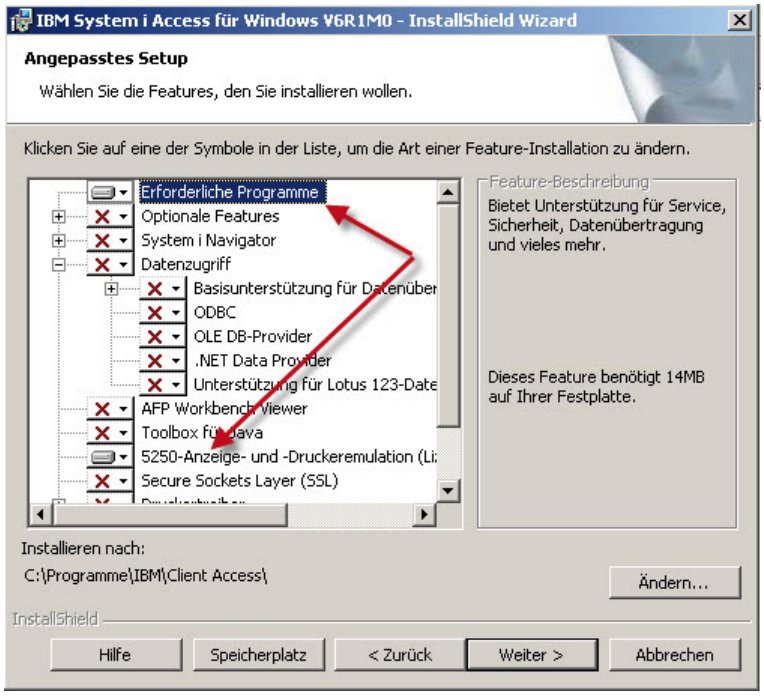
Setuptyp



Hier sehen Sie, dass die angepasste Installation bereits markiert wurde.

Gehen wir weiter, so sehen wir, dass die erforderlichen Features und die 5250-Anzeigefunktion bereits ausgewählt wurden. Alle anderen Features sind deaktiviert.

Mehr haben wir zu diesem Zeitpunkt nicht erreicht!



Komponentenauswahl

Jetzt geht es darum, die Dialoge, die wir bis jetzt angezeigt bekommen, zu unterdrücken.

Wenden wir uns als erstes der Unterdrückung des Dialogs zur Definition der Installationsprache zu. Um diesen Dialog zu unterdrücken gibt es den Befehlzeilenparameter /s.

Unser Setupbefehl sieht dann so aus:

```
C:\Verwaltungsimage\setup.exe /s /vADDLOCAL=req,emu
```

Jetzt wird der genannte Dialog unterdrückt.

Um alle weiteren Dialoge zu unterdrücken, verwenden wir den Befehlszeilenparameter /qn, so dass unser Setupbefehl folgendermaßen aussieht:

```
C:\Verwaltungsimage\setup.exe /s /v"ADDLOCAL=req,emu /qn"
```

Jetzt verschwinden sämtliche Installationsdialoge. Am Ende der Installation wird ein Warmstart durchgeführt.

Voilà, wir haben eine Silent-Installation erreicht!

Ich will Ihnen nicht vorenthalten, dass es zwei weitere Parameter gibt, die Sie unbedingt kennen sollten.

```
C:\Verwaltungsimage\setup.exe /s /v"ADDLOCAL=req,emu /qr"
```

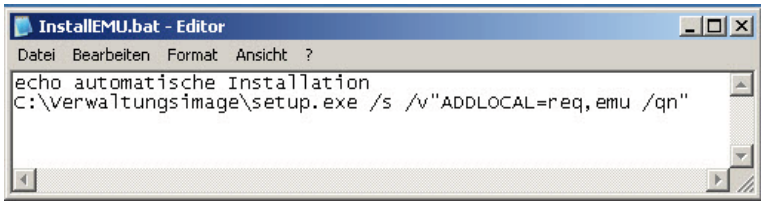
Installation mit reduzierter Umgebung. Bei dieser Variante wird lediglich der Fortschrittsanzeiger während der Installation angezeigt. Ein Warmstart wird nicht ausgeführt.

```
C:\Verwaltungsimage\setup.exe /s /v"ADDLOCAL=req,emu /qb"
```

Es wird eine Beendigungsnachricht ausgegeben und der Warmstart angefordert.

Um die Silent-Installation automatisch ausführen zu können, schreibe ich den Setup-Befehl in eine BAT-Datei mit dem Namen InstallEMU.bat. Diese Datei speichere ich in das Verzeichnis des Verwaltungsimages.

7.2.5.2
Seite 18



Automatische Ausführung

Diesen Arbeitsschritt wiederhole ich für jede funktionelle Installationsvariante, so dass ich am Ende für die jeweilige Installation nur noch das entsprechende Script zu starten brauche.

Servicepack und Installationsimage mischen

Installieren Sie System i Access-Arbeitsplätze auf die bisher beschriebene Art und Weise, so müssen Sie nach erfolgter Installation des System i Access das jeweils aktuelle Service Pack in einem weiteren Arbeitsschritt installieren.

Es geht jedoch einfacher! Warum mischen Sie nicht einfach das aktuelle Servicepack in das Installationsverzeichnis des System i Access für Windows hinein?

Wie das geht, beschreibe ich Ihnen jetzt.

Laden Sie sich von der IBM Servicepack-Downloadseite aus dem Verzeichnis Admin das Misch-Programm herunter:

Für die jeweiligen Prozessorarchitekturen heißen die Programme:

32-bit: UPDATE_32.EXE
64-bit a: UPDATE_64A.EXE
64-bit i: UPDATE_64I.EXE

Kopieren Sie die Datei UPDATE_XXX.EXE in das Stammverzeichnis des Verwaltungsimages.

Führen Sie den folgenden Befehl aus: Update /v"/a cwbinstall.msi".

Falls sich die Datei „cwbinstall.msi" des Service Packs in einem anderen Pfad als die Datei UPDATE_XXX.EXE befindet, müssen Sie den Pfad zur Datei „cwbinstall.msi" wie im folgenden Beispiel übergeben:

```
Update /v"/a y\ServicePacks\cwbinstall.msi"
```

Wenn der Pfad wie im folgenden Beispiel Leerzeichen enthält, verwenden Sie Anführungszeichen. Bitte beachten Sie, dass den inneren Anführungszeichen ein umgekehrter Schrägstrich als Escapezeichen vorangestellt werden muss:

```
Update /v"/a \"C:\Admin Image\cwbinstall.msi\""
```

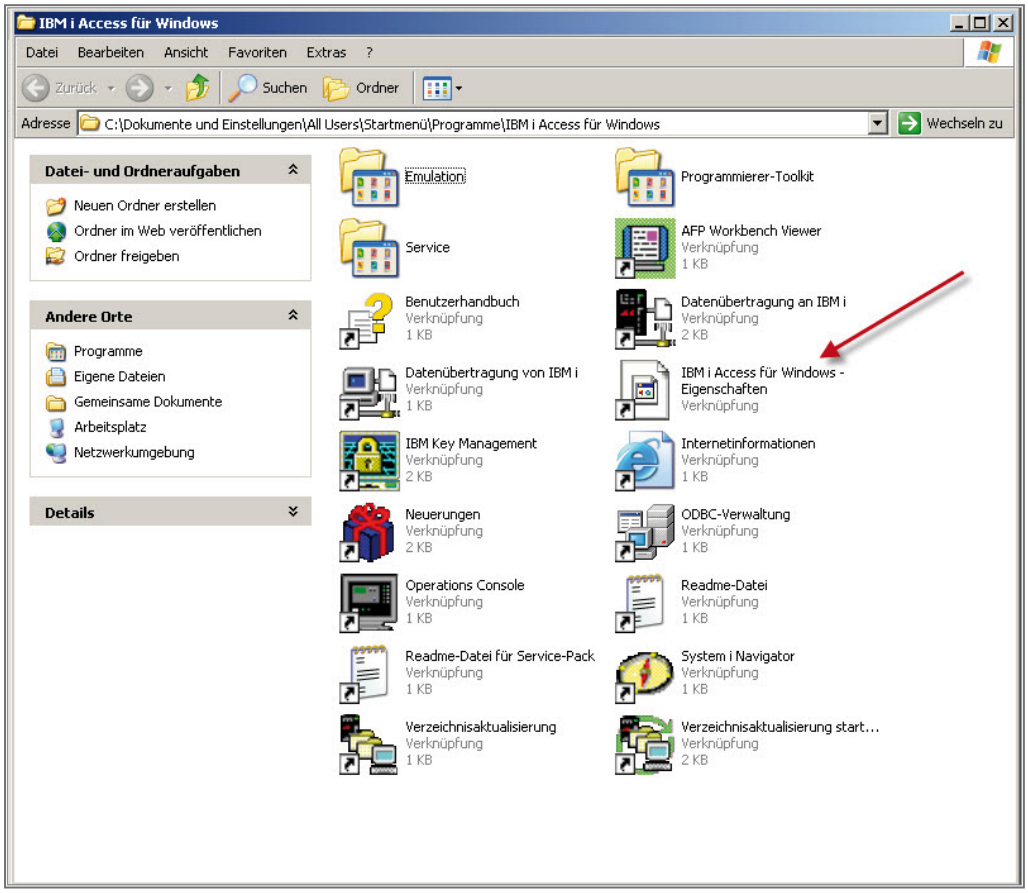
7.2.5.3
Seite 2

Die IBM i Access für Windows-Eigenschaften

Die IBM i Access-Eigenschaften steuern einige der i Access-Grund- und Benutzerfunktionen. Das für den Aufruf der Eigenschaften richtige Symbol finden Sie an zwei Standorten auf Ihrem Windows PC:

- Im Ordner Systemsteuerung
- Im Ordner IBM i Access

Das folgende Bild zeigt den i Access-Ordner, in dem sich das Symbol „i Access – Eigenschaften“ befindet.



Aufruf „i Access – Eigenschaften“

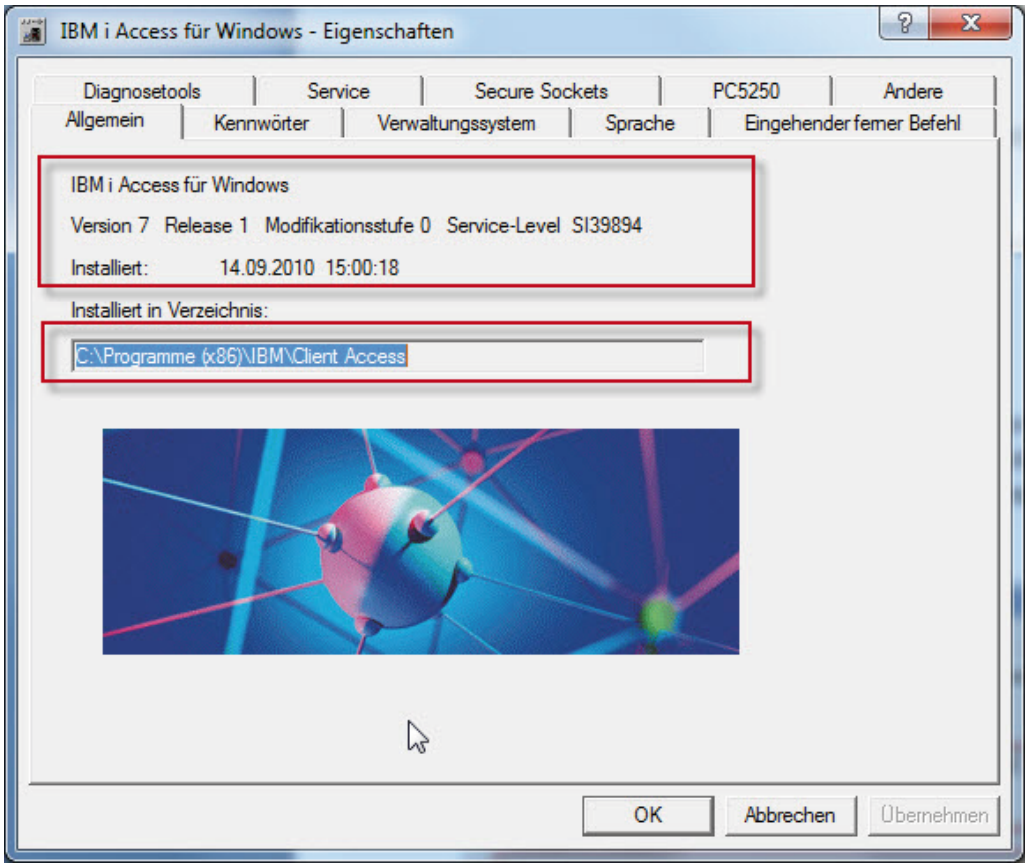
Ein Doppelklick auf das Symbol „i Access – Eigenschaften“ öffnet den Eigenschaften-Dialog.

In diesem Dialog finden Sie eine Reihe von Registerblättern, die im Folgenden kurz erläutert werden.

7.2.6 Seite 2

Allgemein

In diesem Registerblatt finden Sie allgemeine Angaben zur Installation, über den aktuellen Service Pack Level, über das Installationsverzeichnis des i Access und darüber, wann das Service Pack installiert wurde.



Aufruf „i Access – Eigenschaften“

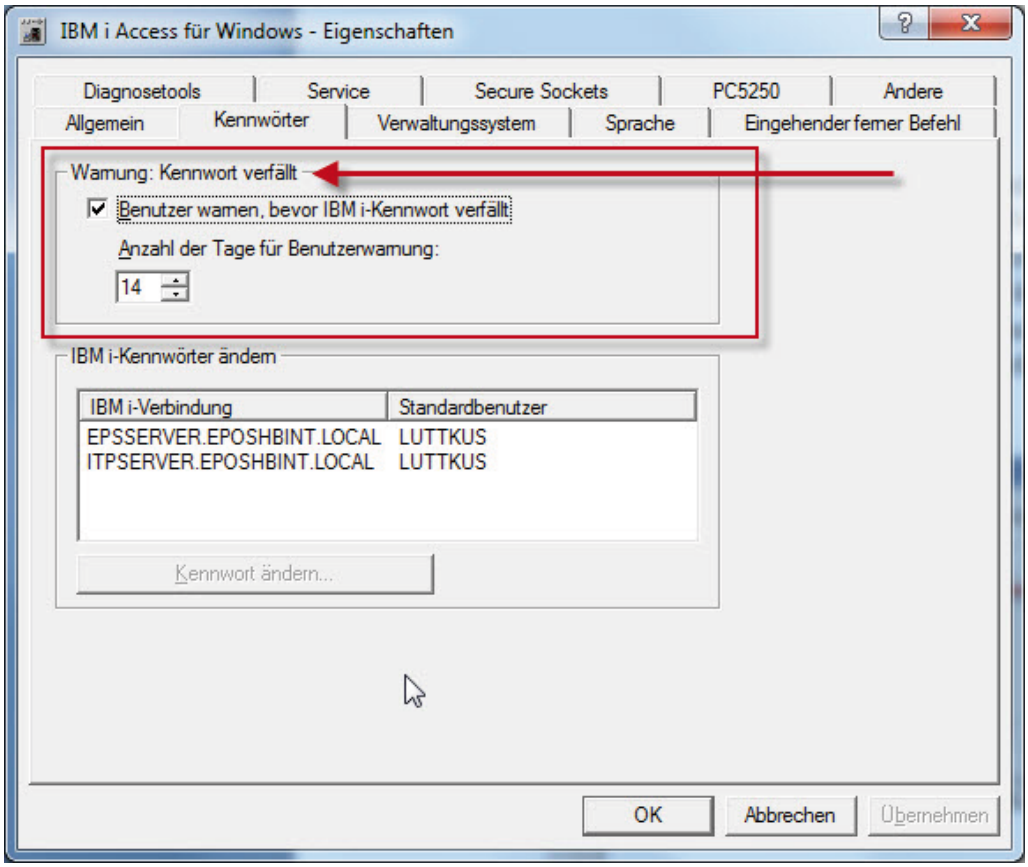
Interessante Angaben auf dieser Seite:

- i Access Produktkomponente:
  – hier i Access für Windows
- Version und Service Pack Stufe:
  – hier V7R7M0 SI39894
- Installationsdatum und Zeit:
  – hier 14.09.2010 und 15:00:18
- Installationsverzeichnis auf dem Client
  – hier c:\programme(x86)\ibm\client access

7.2.6.1
Seite 2

Kennwörter

Im i Access für Windows steht Ihnen ein Password Cache-Speicher zur Verfügung. Des Weiteren können Sie die Verbindungsanmeldung für die Bildschirmfunktion aus der Windows-Anmeldung übernehmen. Ändern Sie das Kennwort Ihres Benutzerprofils auf dem System, so wird dieser Automatismus unterbrochen. Der im Folgenden gezeigte Dialog bietet Ihnen die Möglichkeit, Nachrichten zu erhalten, wenn dieser Fall eingetreten ist.



Register Kennwörter

Parameter:
Benutzer warnen,...: Wünschen Sie eine Warnung bei Kennwortänderung? Falls ja, so sollten Sie diese Auswahl markieren.

Anzahl der Tage... Wie lange vor dem Verfallen des Kennwortes sollen Warnungen an Sie ausgegeben werden? Der Default-Wert ist 14 Tage.

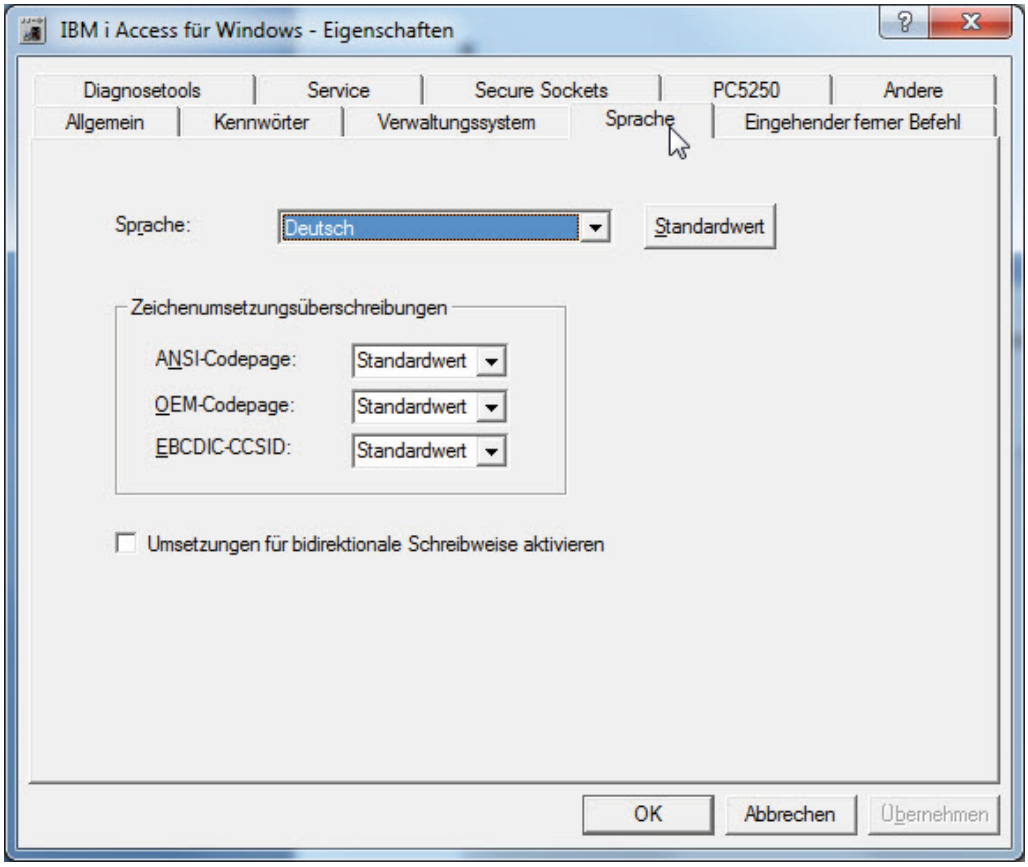
IBM i-Kennwörter ändern: Markieren Sie die angezeigte Systemverbindung und drücken Sie dann auf den Knopf „Kennwort ändern“. Im nachfolgenden Dialog erhalten Sie die Möglichkeit, das Kennwort des angezeigten Benutzerprofils manuell auf der iSeries zu ändern.

7.2.6.2

Seite 2

Sprache

Sprachversion des i Access ändern



Register Sprache

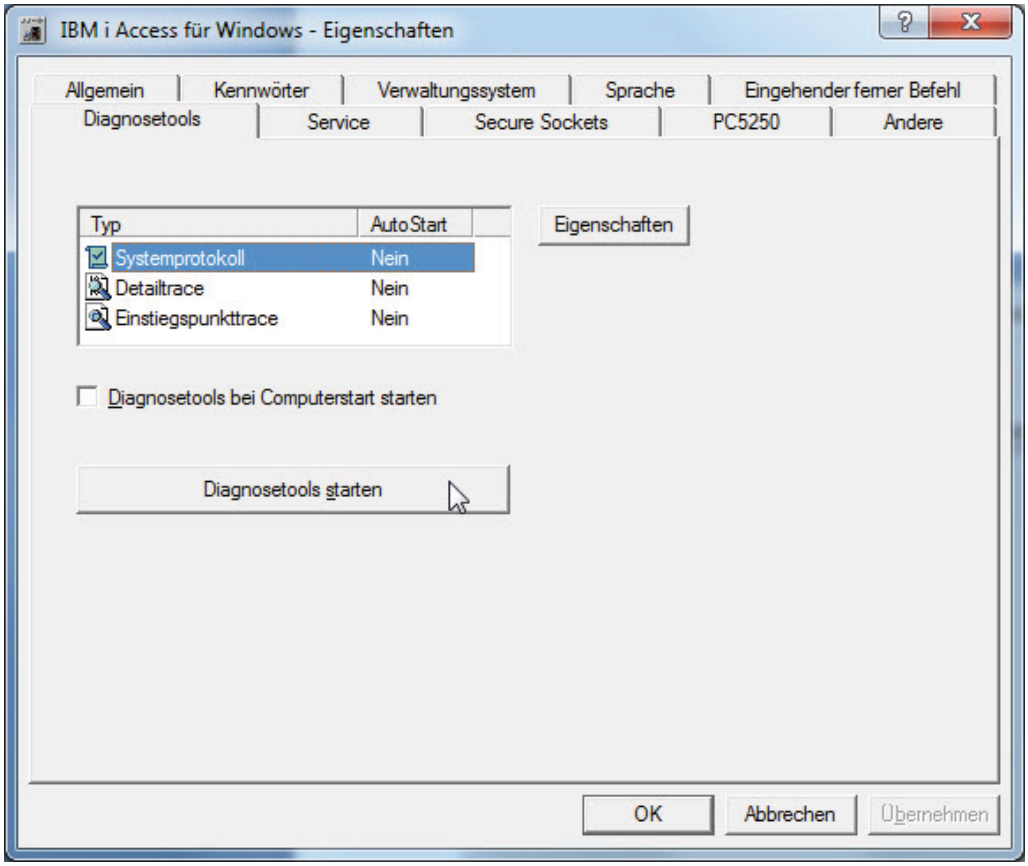
Arbeiten Sie in einer multinationalen iSeries-Umgebung, so kann mit Hilfe dieses Registerblatts die sprachliche Umgebung für den i Access-Client geändert werden.

Weiterhin wird Ihnen die Gelegenheit gegeben, Zeichenumsetzungstabellen u. a. für den Dateitransfer anzugeben, die von den Standardeinstellungen abweichen.

7.2.6.3

Seite 2

Diagnosetools



Register Diagnosetools

Zeigt Ihre i Access für Windows–Installation immer wiederkehrende oder auch sporadische schwer nachzuvollziehende Fehler, so besteht mit Hilfe dieses Dialogs die Möglichkeit, die Diagnosetools (Ablaufverfolgungen, Protokolle etc.) zu starten.

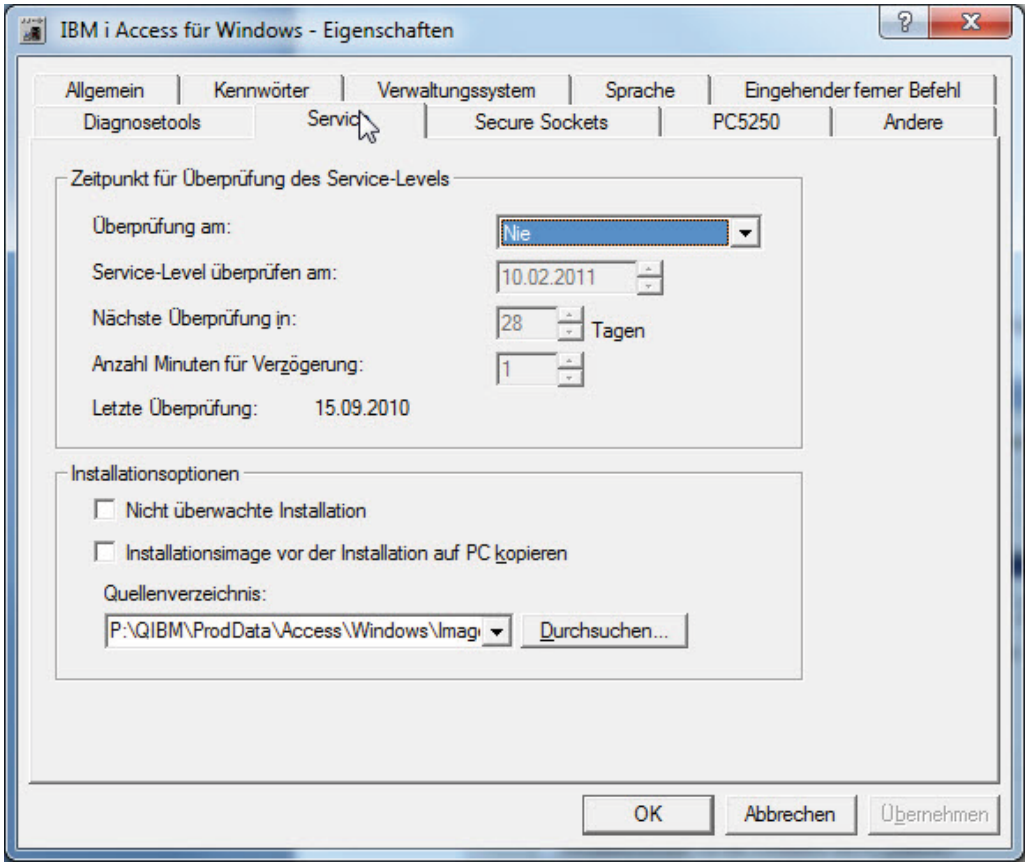
Der Start kann sowohl manuell vorgenommen werden (Knopf Diagnosetools starten), als auch automatisiert werden (Option Diagnosetools beim Computerstart starten).

7.2.6.4

Seite 2

Service

Das Register Service steuert zentral die Service Pack–Aktualisierung.



Register Service

Parameter:
Überprüfung am: Hier wird das Intervall eingestellt, in dem es zu Überprüfungen der Aktualität des Service Packs kommt. Drei Werte sind möglich:

Immer: Bei jedem Start wird die Aktualität des Service Pack überprüft. Alle folgenden Parameter sind deaktiviert.

Nie: Es finden keine Überprüfungen statt. Alle folgenden Parameter sind deaktiviert.

Regelmäßig: In bestimmten Zeitabständen wird eine Überprüfung der Aktualität des Service Packs gestartet. Jetzt sind die folgenden Parameter aktiv.

Service-Level überprüfen am: Wann soll die nächste Überprüfung stattfinden?

Nächste Überprüfung in: In welchem Zeitabstand finden die Überprüfungen statt?

7.2.6.5
Seite 2

Anzahl Minuten für Verzögerung: Da es bei einigen Windows-Clients zu Synchronisierungsfehlern in der Startphase des Clients kommt, wenn automatisch eine Service Pack-Überprüfung stattfindet, kann hier ein Verzögerungsfaktor eingestellt werden. Sie sehen hier den Defaultwert von einer Minute.

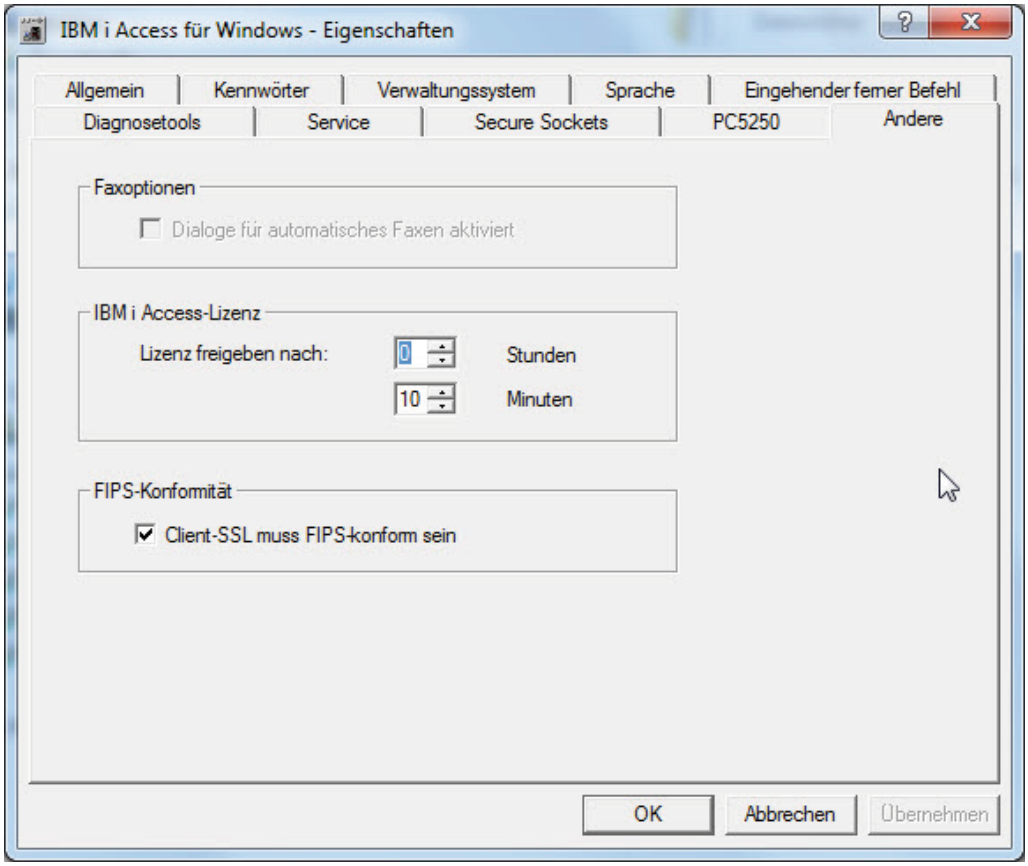
Letzte Überprüfung: Wann hat die letzte Überprüfung stattgefunden?

Installationsoptionen:
Nicht überwachte Installation: Wenn es ein neues Service Pack gibt, können Sie hier wählen, wie die Installation durchgeführt werden soll. Markieren Sie diesen Punkt, so läuft die Service Pack-Installation im Hintergrund Ihres Systems ab, ohne dass irgendwelche Dialoge angezeigt werden.

Installationsimage...: Hier können Sie wählen, ob das komplette Image der Service Pack-Objekte auf den Client kopiert werden soll.

Quellenverzeichnis: Wo ist der Standort des aktuellen Service-Packs? Als Defaultwert wird bei dieser i Access-Version das Installationsverzeichnis des Produkts angezeigt. Spielen Sie Ihre Service Packs in ein Netzwerkverzeichnis ein, das beispielsweise auf einem File-Server liegt, so können Sie direkt und zentral die Service Packs von diesem einspielen.

Andere



Register Andere

Parameter

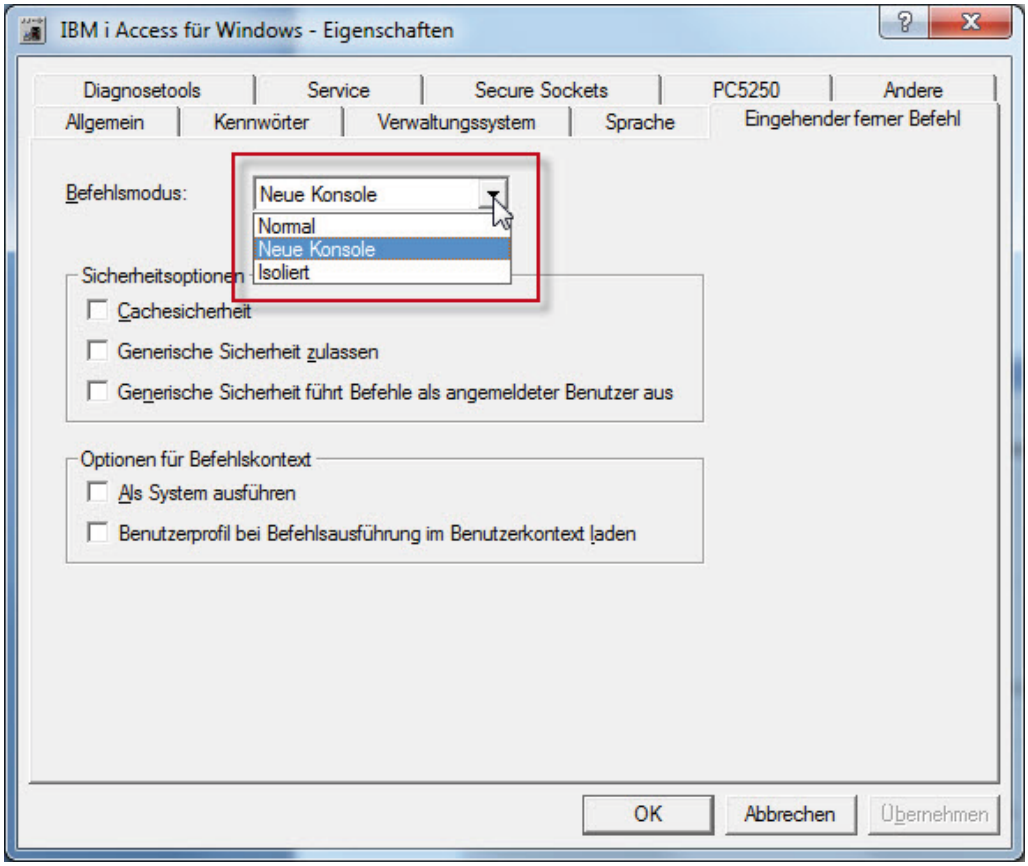
Faxoptionen
Arbeiten Sie mit einer Faxeinheit auf dem System, so können mit Hilfe des i Access die Eingabedialoge für Faxnummer, Empfänger etc. angezeigt werden. Dieser Parameter ist als Standard aktiviert. Deaktivieren Sie diese Option und nutzen die Faxeinheit des Systems, so haben Sie nicht die Möglichkeit einer dialog-orientierten Eingabe.

IBM i Access Lizenz
Immer dann, wenn Sie eine i Access für Windows-Verbindung beenden, wird Ihre zugewiesene Lizenz für einen kurzen Zeitraum als noch zugeweisen betrachtet. Sollten Sie eine Verbindung beenden und sie innerhalb von 10 Minuten wieder aufnehmen, so muss Ihnen die Lizenz für den Zugriff nicht erneut vergeben werden. Eine erneute Anmeldung ist dann, zumindest was die grundsätzliche Verbindung angeht, nicht mehr notwendig. Soll dieses Intervall vergrößert werden, so können Sie hier einen entsprechenden Zeitraum einstellen.

7.2.6.6

Seite 2

Eingehender ferner Befehl



Register Eingehender ferner Befehl

Parameter

Befehlsmodus
Sobald das Programm „Eingehender ferner Befehl“ einen Befehl empfängt, erstellt es einen neuen Prozess für die Ausführung dieses Befehls. Es gibt unterschiedliche Methoden, einen neuen Prozess auszuführen. Das Programm „Eingehender ferner Befehl“ bietet dem Benutzer drei unterschiedliche Werte als Befehlsmodus an. Jeder Modus erstellt den Prozess auf eine andere Weise.

Normal
Im Modus „Normal“ wird das eigene Konsolfenster des Programms „Eingehender ferner Befehl“ verwendet, wenn für den Befehl ein Konsolfenster erforderlich ist. Außerdem ist im Modus „Normal“ der neue Prozess, in dem der Befehl ausgeführt wird, ein Kindprozess des Prozesses von „Eingehender ferner Befehl“. Hierdurch kann die Befehlsausgabe erfasst und an den Benutzer zurückgesendet werden, der den Befehl abgesetzt hat.

Neue Konsole
Im Modus „Neue Konsole“ wird der Befehl in einem eigenen Konsolfenster anstatt im Konsolfenster von „Eingehender ferner Befehl“ ausgeführt. Wie beim Modus „Normal“ wird der neue Prozess als Kindprozess des Prozesses von „Eingehender ferner Befehl“ ausgeführt. Hierdurch kann die Befehlsausgabe erfasst und an den Benutzer zurückgesendet werden, der den Befehl abgesetzt hat. Das erstellte neue Konsolfenster kann auf dem Desktop angezeigt werden. Möglicherweise dauert die Ausführung des Befehls etwas länger, weil zunächst das neue Konsolfenster erstellt werden muss. Interaktive Konsolanwendungen, die in einem eigenen Konsolfenster ausgeführt werden, halten jedoch die Ausführung nachfolgender Befehle nicht zurück, wie dies möglicherweise im Modus „Normal“ (also im Konsolfenster von „Eingehender ferner Befehl“) der Fall wäre. Ein Beispiel für eine solche Konsolanwendung ist der Texteditor edit.com, der mit Windows ausgeliefert wird.

Isoliert
Im Modus „Isoliert“ wird der Befehl in einem eigenen Konsolfenster und nicht im Konsolfenster von „Eingehender ferner Befehl“ und zudem isoliert von dem Prozess von „Eingehender ferner Befehl“ ausgeführt. Daher kann die Ausgabe des Befehls, wenn es sich bei diesem um ein ausführbares Programm handelt, häufig nicht erfasst und an den Benutzer zurückgesendet werden, der den Befehl abgesetzt hat. Anstelle der Ausgabe des Befehls wird möglicherweise eine Nachricht an den Befehlsabsender gesendet, die angibt, dass der Befehl keine Ausgabe erzeugt hat. Wird die Ausgabe des Befehls nicht an den Befehlsabsender zurückgegeben, ist es viel komplizierter zu ermitteln, ob der Befehl wie erwartet ausgeführt wurde. Einige Anwendungen werden jedoch nicht korrekt oder überhaupt nicht verarbeitet, wenn sie nicht isoliert von dem Prozess von „Eingehender ferner Befehl“ ausgeführt werden. Daher ist bei solchen Anwendungen möglicherweise die Verwendung dieses Modus erforderlich. Ein Beispiel hierfür ist die Anwendung xcopy.exe von Windows NT, die nur im Modus „Isoliert“ ausgeführt werden kann.

Als System ausführen
Diese Option bewirkt, dass das Programm „Eingehender ferner Befehl“ jeden Befehl im Kontext des Kontos ausführt, unter dem der i Access für Windows-Service „Ferner Befehl“ gemäß seiner Konfiguration ausgeführt wird. In der Standardeinstellung handelt es sich hierbei um das Systemkonto. Wird diese Option nicht angegeben, wird jeder Befehl in dem Kontext der Benutzer-ID ausgeführt, die den Befehl gesendet hat. Die meisten Befehle funktionieren erwartungsgemäß, wenn sie im Kontext des Benutzerkontos ausgeführt werden. Bei einigen Befehlen ist dies jedoch möglicherweise nicht der Fall, weshalb unter Umständen diese Option verwendet werden muss. Die Funktionsweise eines bestimmten Befehls kann bei der Ausführung unter verschiedenen Konten variieren, da die Konten möglicherweise unterschiedlich konfiguriert sind.

Cache-Sicherheit
Diese Option bewirkt, dass das Programm „Eingehender ferner Befehl“ mit Hilfe des NT-Sicherheitsmanagers eine Benutzeranmeldung durchführt. Dies wird jedoch nur ein Mal für jede Benutzer-ID durchgeführt, denn die Benutzer-ID und das Kennwort werden im Cache gespeichert und erneut verwendet, sobald derselbe Benutzer wieder einen Befehl sendet.

Generische Sicherheit zulassen
Diese Option bewirkt, dass das Programm „Eingehender ferner Befehl“ Befehle akzeptiert und ausführt, die ohne Sicherheitswerte – also durch Eingabe von *NONE für Benutzer-ID und Kennwort – gesendet wurden. (Unter Windows 95/98 können Sie diese Funktion nutzen, indem Sie für die Benutzer-ID und das Kennwort auf der Seite „Eingehender ferner Befehl“ der Eigenschaften für i Access für Windows das Platzhalterzeichen (*) verwenden.) Dies ist beispielsweise dann sinnvoll, wenn mit RUNRMTCMD Befehle von einem Programm gesendet werden, das für den Windows NT-PC keine gültige Benutzer-ID und kein gültiges Kennwort abrufen kann. Es muss jedoch auf folgendes hingewiesen werden: Wenn diese Option verwendet wird, ist JEDER BENUTZER in der Lage, JEDEN BEFEHL auf dem PC auszuführen, wenn RUNRMTCMD oder ein anderer REXEC-ähnlicher Client verwendet wird, der das Senden einer leeren Benutzer-ID und eines leeren Kennworts mit dem Befehl zulässt! Dies gilt insbesondere dann, wenn der i Access-Service „Ferner Befehl“ zur Ausführung unter dem Systemkonto konfiguriert ist, da das Systemkonto alle Berechtigungen auf dem PC besitzt. Daher ist es bei Verwendung dieser Option unter Umständen sinnvoll, den Zugriff auf den Befehl RUNRMTCMD auf Ihren Systemen IBM iSeries auf anerkannte Benutzer oder Programme einzuschränken bzw. andere Sicherheitsmaßnahmen zu ergreifen, um die absichtliche oder versehentliche Ausführung destruktiver Befehle auf dem PC zu verhindern.

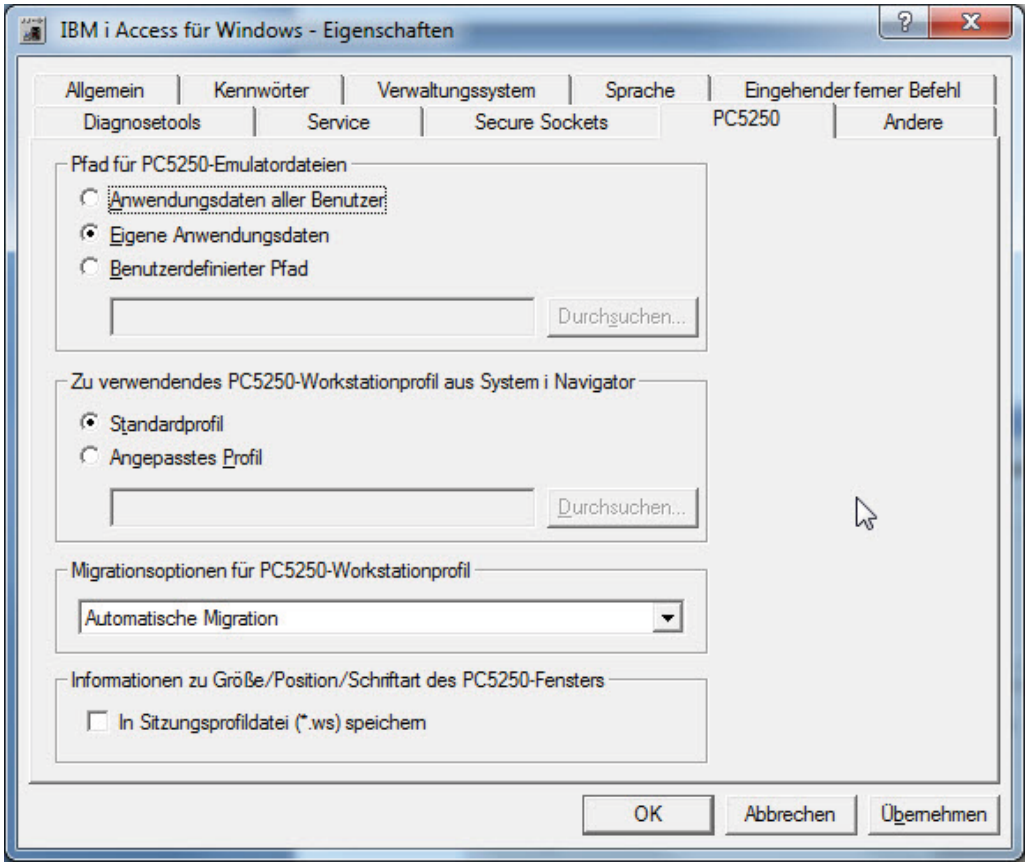
Generische Sicherheit führt Befehle....
Ähnlich wie die Option „Generische Sicherheit zulassen“ lässt diese Option das Absetzen von Befehlen zu, bei denen für Benutzer-ID und Kennwort der Wert *NONE zur Verfügung gestellt wird. Bei Verwendung dieser Option versucht i Access jedoch, die Benutzer-ID und das Kennwort des Benutzers abzurufen, der gegenwärtig an dem Windows NT-System angemeldet ist, auf dem der Befehl ausgeführt werden soll. Werden Benutzer-ID und Kennwort gefunden, wird der Befehl im Kontext dieses Benutzers ausgeführt. Wenn kein Benutzer angemeldet ist oder die Benutzer-ID und das Kennwort nicht gefunden werden, die Option „Generische Sicherheit zulassen“ jedoch ebenfalls angegeben wurde, wird der Befehl in dem Kontext ausgeführt, in dem der Service gestartet wurde (normalerweise „System“). Können diese Werte nicht ermittelt werden und wurde die Option „Generische Sicherheit zulassen“ NICHT angegeben, wird der Befehl nicht ausgeführt und die Ausgabe in der Spool-Datei gibt an, dass die Benutzer-ID fehlte oder ungültig war. Diese

26. Ergänzung 3/2011

7.2.6.7
Seite 4

Option ist dann nützlich, wenn RUNRMTCMD ohne Benutzer-ID und Kennwort ausgeführt werden soll, der Befehl jedoch im Kontext eines angemeldeten Benutzers ausgeführt werden soll, damit auf die in diesem Benutzerkontext zugeordneten Ressourcen zugegriffen werden kann. Zu solchen Ressourcen können unter anderem Netzlaufwerke und Netzwerkdrucker gehören.

PC5250



Register PC5250

In diesem Register können Sie den Standort aller für die PC5250-Bildschirmemulation wichtigen Konfigurationsdateien festlegen.

Bildschirmsitzungen werden durch mehrere Konfigurationsdateien beschrieben. Hier die charakteristischen Namenserweiterungen dieser Dateien:

*.ws – das sogenannte Bildschirmprofil: Festlegen des Layouts des Bildschirms, Farben, Cursor-Form etc.

*.kmp – Tastaturlayout und Tastenbelegungen

*.bar – Definition der Funktionsleiste

*.mac – Makro-Dateien

*.cae – Steuerung des Anmeldungsverhaltens bei der Verbindungsaufnahme

In dem oben gezeigten Register sehen Sie, welche Möglichkeiten hier bestehen.

7.2.6.8
Seite 2

Pfad für PC5250-Emulatordateien: Bildschirmbezogene Konfigurationsdaten werden im Pfad Benutzer\<Name des Benutzers>AppData\Roaming\IBM\Client Access\Emulator\Private abgelegt. Dieses ist der Standardwert. Nachteil: Die Konfigurationsdaten werden lokal auf jedem Client abgelegt.

Eigene Dateien: Die Konfigurationsdateien werden im Verzeichnis „Eigene Dateien" des Windows Desktop abgelegt. Der Nachteil besteht auch hier darin, dass die Daten lokal abgelegt werden. Es ermöglicht jedoch einem Windows-Benutzer, der nicht lokaler Administrator ist, Konfigurationsarbeiten am i Access durchzuführen. Dazu gehört auch das Ausführen der Funktion „Selektive Installation".

Benutzerdefinierter Pfad: Hier kann ein eigener Pfad für die Speicherung der Profildaten vergeben werden. Ich habe hier einen Netzwerkpfad auf einen Dateiserver eingegeben, so dass alle Bildschirm-Konfigurationsdaten zentral auf einem Fileserver abgelegt werden. Vorteil: Konfigurationsdaten werden zentral im Netzwerk abgelegt, können somit auch zentral gesichert werden und von einem Administrator zentral gepflegt werden.

Verwaltungssystem

Um diese Funktion zu erklären, muss ich ein wenig ausholen.

Die Netzwerkfähigkeiten des Systems sind in den letzten Jahren beträchtlich gestiegen. Es gibt jedoch eine ganze Reihe von Anwendern, die diesen Entwicklungen relativ skeptisch gegenüberstehen, da sie das Gefühl haben, dass die i-Systeme den Clients fast schutzlos ausgeliefert sind.

Dieses Gefühl kommt einerseits aus einer tradierten AS/400-Objektberechtigungsstruktur, die darauf vertraute, dass die Anwender ausschließlich über die 5250-Schnittstelle das System betreten und dann von Menüs und Bildschirmen in ihren Möglichkeiten derart eingeschränkt werden (Stichwort: keine Systembildschirme), dass ein differenziertes Objektrecht nicht notwendig ist.

Heute wissen wir, dass ein derartiges Vorgehen riesige Sicherheitslücken in unsere Systeme reißt, denn heute kommen Anwender nicht nur über 5250-Bildschirme ins System, sondern viel mehr mit Zugriffsmethoden, die außerhalb des 5250-Datenstroms angesiedelt sind: ODBC, JDBC, ADO.Net, FTP, SQL etc.

Das hat dazu geführt, dass sich eine ganze Reihe von Tool-Anbietern im Markt befindet, die diese Sicherheitslücken abdichtet und kontrolliert.

Jedoch fast unbemerkt hat sich seit OS/400 V5R1 mit dem iSeries Navigator ebenfalls ein Werkzeug eingeschlichen, das Client-Schnittstellen zum System bearbeiten und kontrollieren kann.

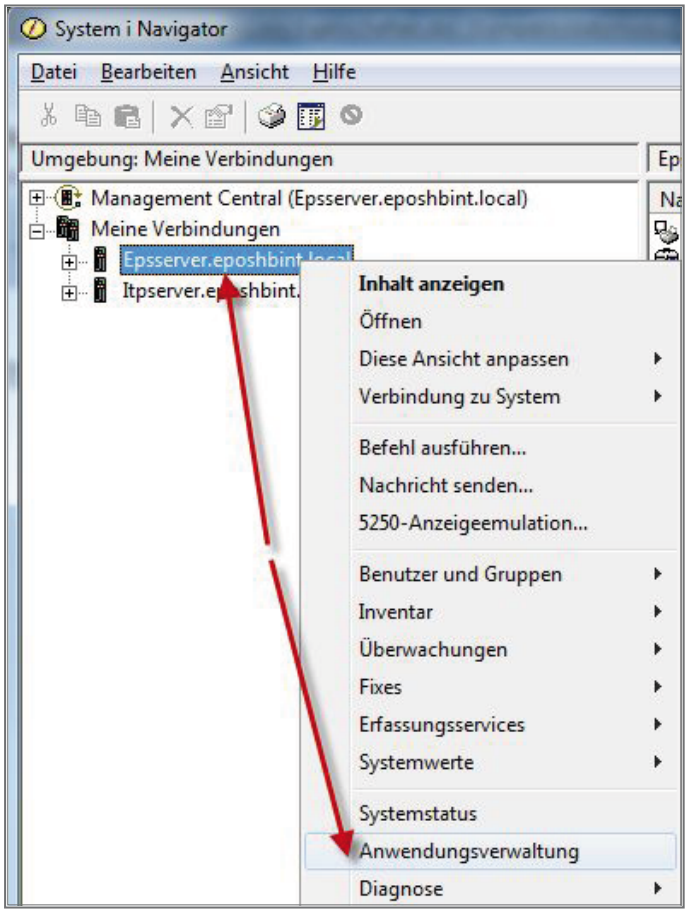
Dieses Werkzeug wird unter dem Begriff „Anwendungsverwaltung“ im i Navigator geführt.

Was leistet die Anwendungsverwaltung? Sie bearbeitet nicht das Objektrecht von Benutzerobjekten auf dem System, sondern kümmert sich um Funktionen, die die Anwendung und Sicherheit des Betriebssystems bedrohen können. Die Anwendungsverwaltung bearbeitet also Funktionen und deren Schnittstellen zum Betriebssystem. Die Sicherheitsdaten, die dabei erzeugt werden, werden im OS/400 gespeichert – also nicht lokal auf dem PC, auf dem der iSeries Navigator installiert ist.

Um in die Anwendungsverwaltung zu gelangen, gibt es mehrere Wege: einerseits über die Erstellung bzw. Änderung eines Benutzers oder einer Benutzergruppe, andererseits über die Systemebene. Ich werde Ihnen den zweiten Weg hier zeigen, da er mir als der bessere erscheint. Warum? Weil ich nicht für jeden einzelnen Benutzer seine Rechte festlegen möchte, sondern eher eine globale Steuerung suche, die nur über die Systemebene ausgeführt werden kann.

7.2.6.9
Seite 2

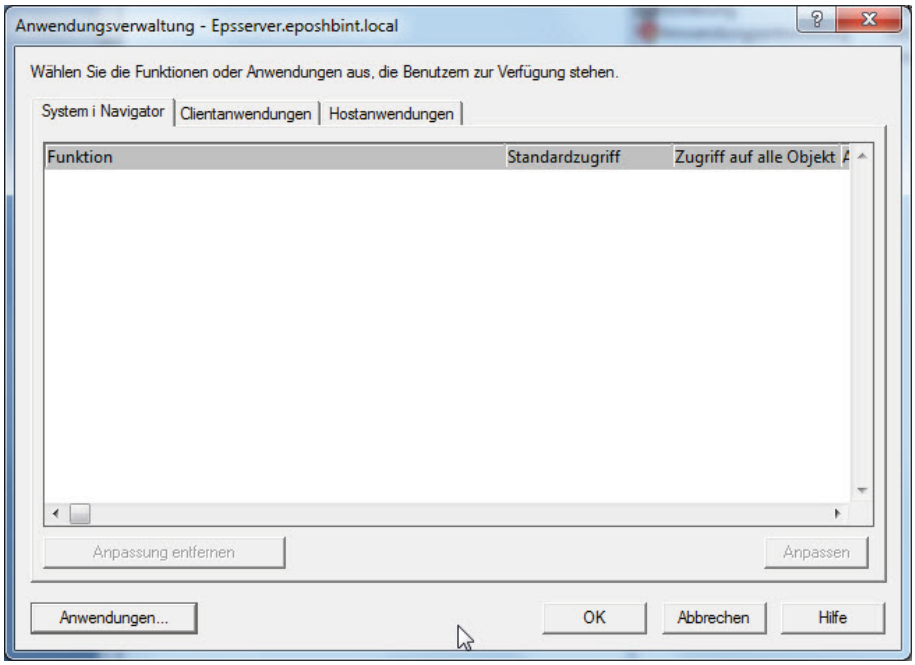
Ich werde Ihnen jetzt anhand einer Systemverbindung zur Adresse EPSServer.eposhbint.local zeigen, wie man in die Anwendungsverwaltung gelangt. Dazu öffne ich den iSeries Navigator, klicke mit der rechten Taste auf die Verbindung EPSServer.eposhbint.local, so dass das Kontextmenü für diese Verbindung angezeigt wird.



Kontextmenü für Systemverbindung

Sie sehen hier den vorletzten Menüpunkt „Anwendungsverwaltung“, den ich jetzt auswähle.

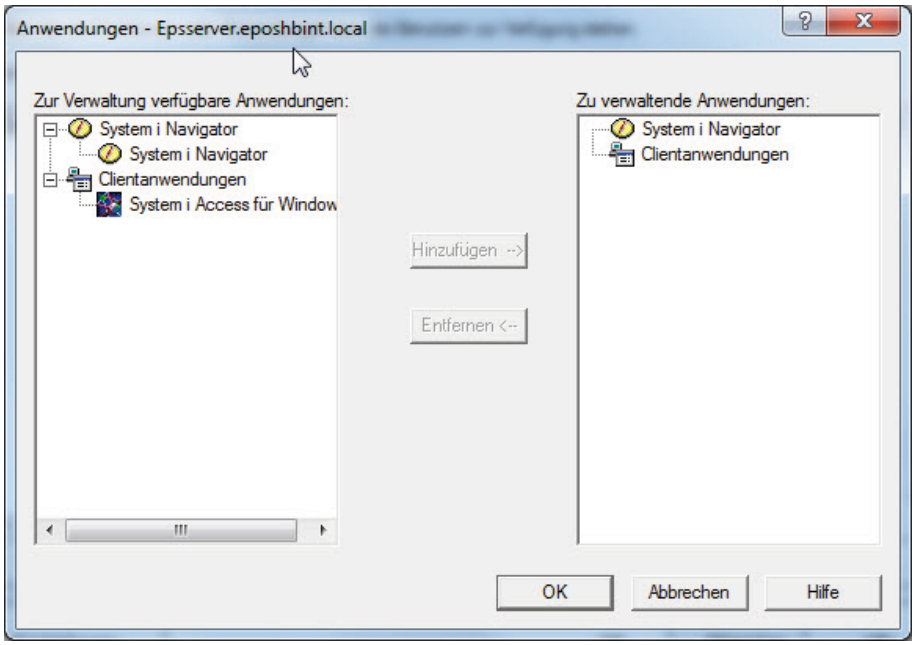
Daraufhin öffnet sich nachfolgender Dialog. Wenn Sie die „Anwendungsverwaltung“ das erste Mal öffnen, so ist dieses Fenster leer.



Anwendungsverwaltung

Zuerst müssen die Anwendungen definiert werden, gegen deren Funktionen wir unser System zu schützen gedenken. Aus diesem Grund klicke ich auf den Knopf „Anwendungen…“.

Jetzt öffnet sich ein weiterer Dialog.

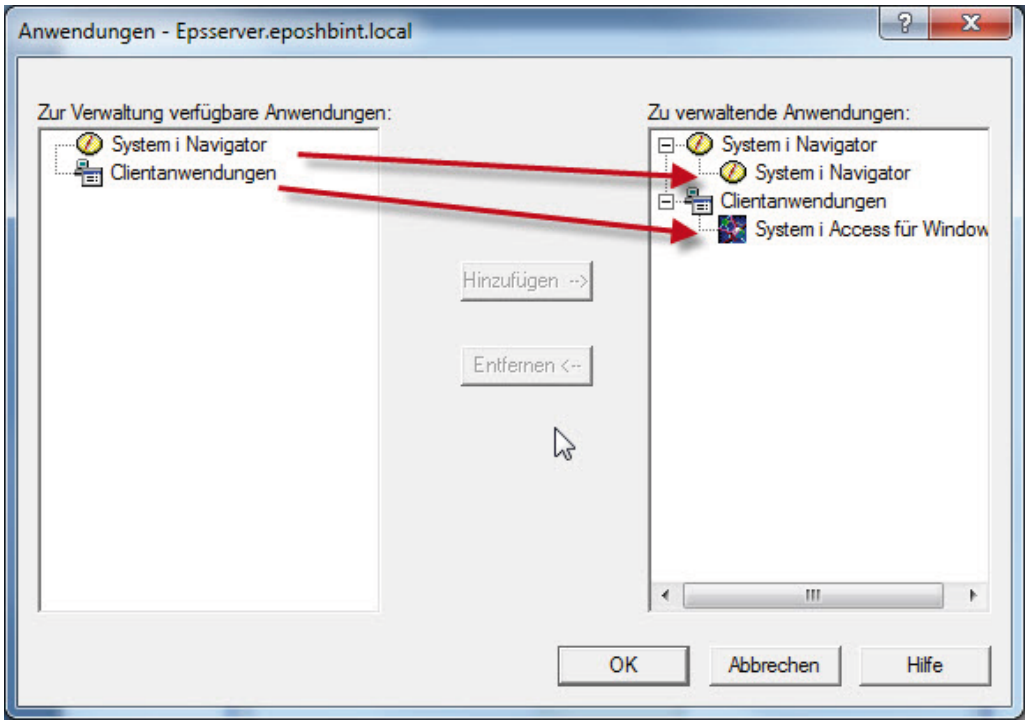


Anwendungen

7.2.6.9
Seite 4

Auf der linken Seite sehen Sie die Programme, gegen deren Funktionen wir unser System zu schützen gedenken. Ich markiere die erste Anwendung – hier „iSeries Navigator“ – und klicke auf den Knopf „Hinzufügen“.

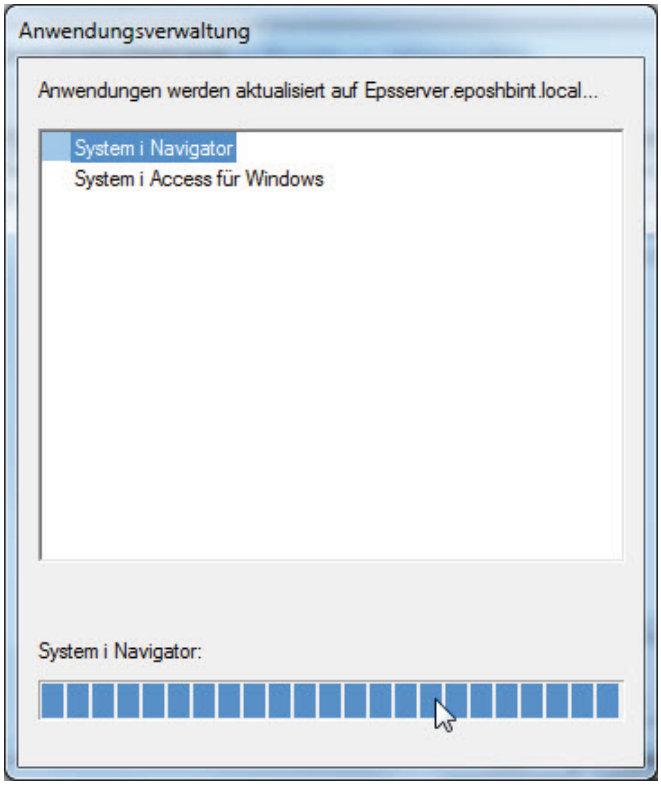
Gleichermaßen verfahre ich mit der zweiten Anwendung in der Gruppe „Clientanwendungen“.



Anwendungen sind ausgewählt

Dann wird der gesamte Dialog mit „OK“ bestätigt.

Die eben angewählten Anwendungen werden jetzt bezüglich ihrer Funktionen analysiert.

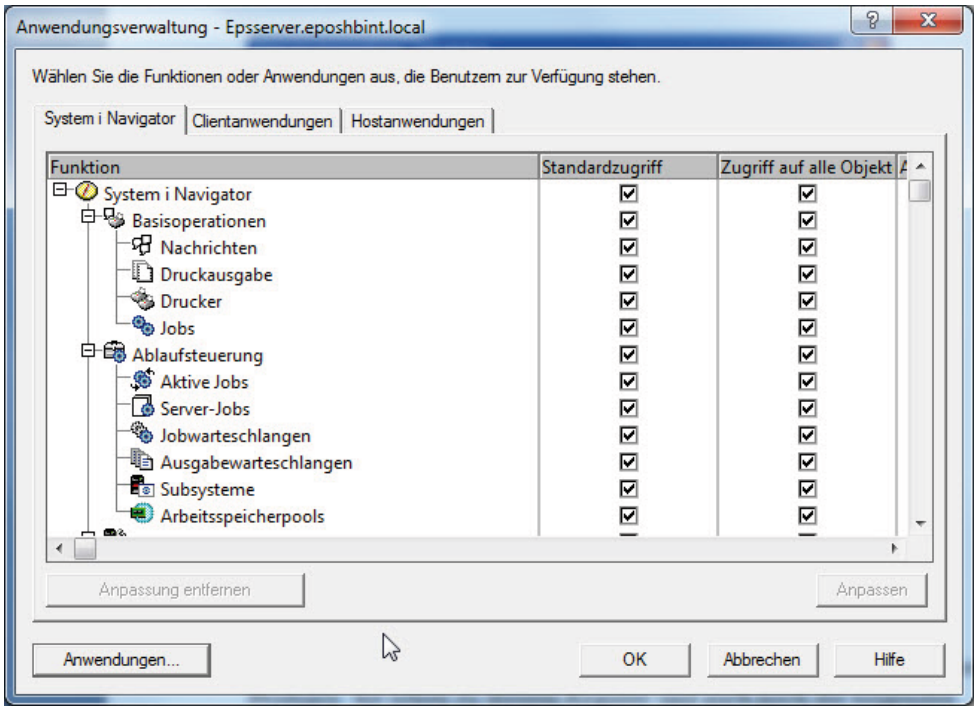


Anwendungsanalyse

Am Fortschrittsbalken erkennen wir den Fortschritt der Analyse.

Ist die Analyse beendet, so kehren wir in das Anfangsfenster der „Anwendungsverwaltung“ zurück.

Wie Sie sehen, hat sich dieses Fenster erheblich verändert.



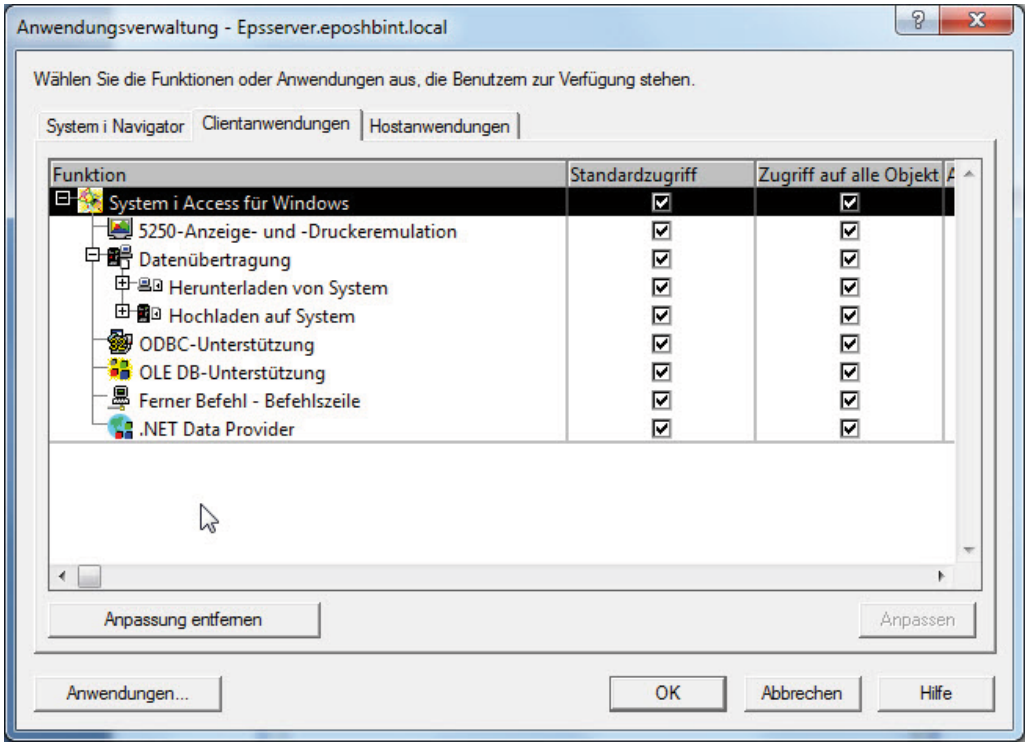
Gefüllte Anwendungsverwaltung

7.2.6.9
Seite 6

Es haben sich drei Register gebildet.

Das Register „System i Navigator“ beschreibt die Funktionen eben jenes Produkts. Sie sehen zu diesem Register auch noch die folgenden zwei Spalten: „Standardzugriff“ und „Zugriff auf alle Objekte“. Die Häkchen in der Spalte „Standardzugriff“ besagen, dass ein Benutzer, der die jeweilige Funktion nutzen möchte, mittels seines Benutzerprofils im System geprüft wird – d. h., es wird geprüft, ob er diese Funktion überhaupt auslösen darf. Die Häkchen in der Spalte „Zugriff auf alle Objekte“ besagen, dass auch Benutzer, die das *ALLOBJ-Recht besitzen, diese Funktion nutzen können. Das ist interessant! Man kann also an dieser Stelle einem Administrator der iSeries untersagen, eine bestimmte Funktion zu verwenden.

Sie werden später sehen, dass wir diese Rechte individuell für Benutzer und Benutzergruppen einschränken oder gewähren können.

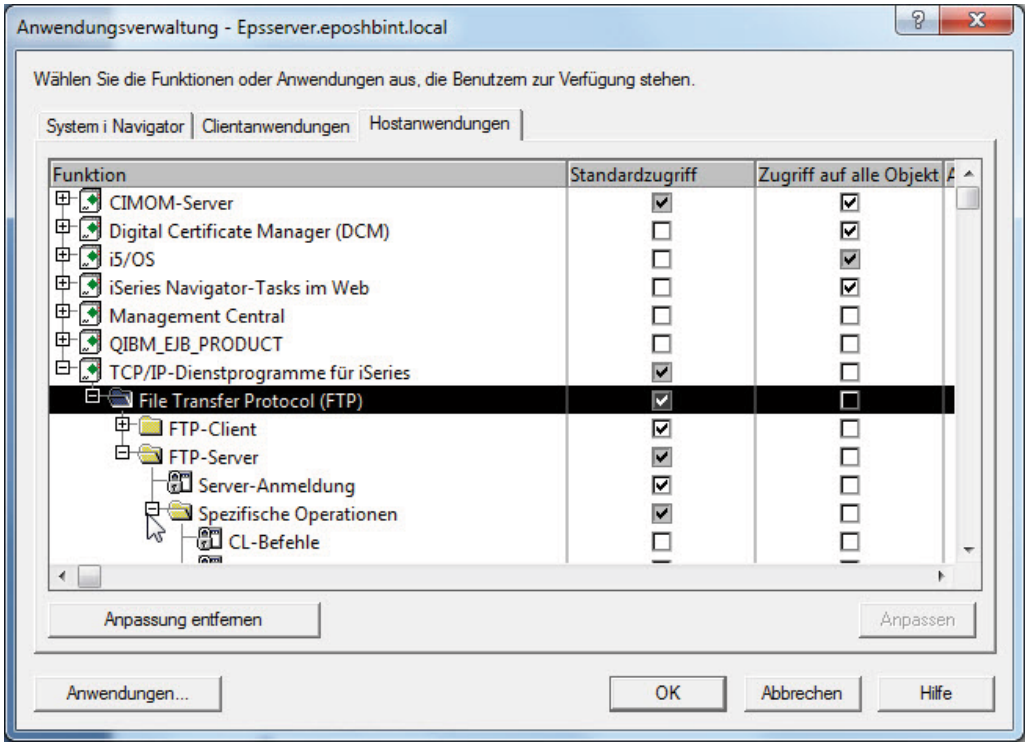


Register „Clientanwendungen“

Hier sehen Sie das Register „Clientanwendungen“, in welchem wir die Funktionen des Client Access / i Access sozusagen entschärfen können.

Auch hier ist eingestellt, dass grundsätzlich alle Benutzer des Systems diese Funktionen erreichen können.

Das letzte Register bezieht sich auf Funktionen im Betriebssystem.



Register „Hostanwendungen“

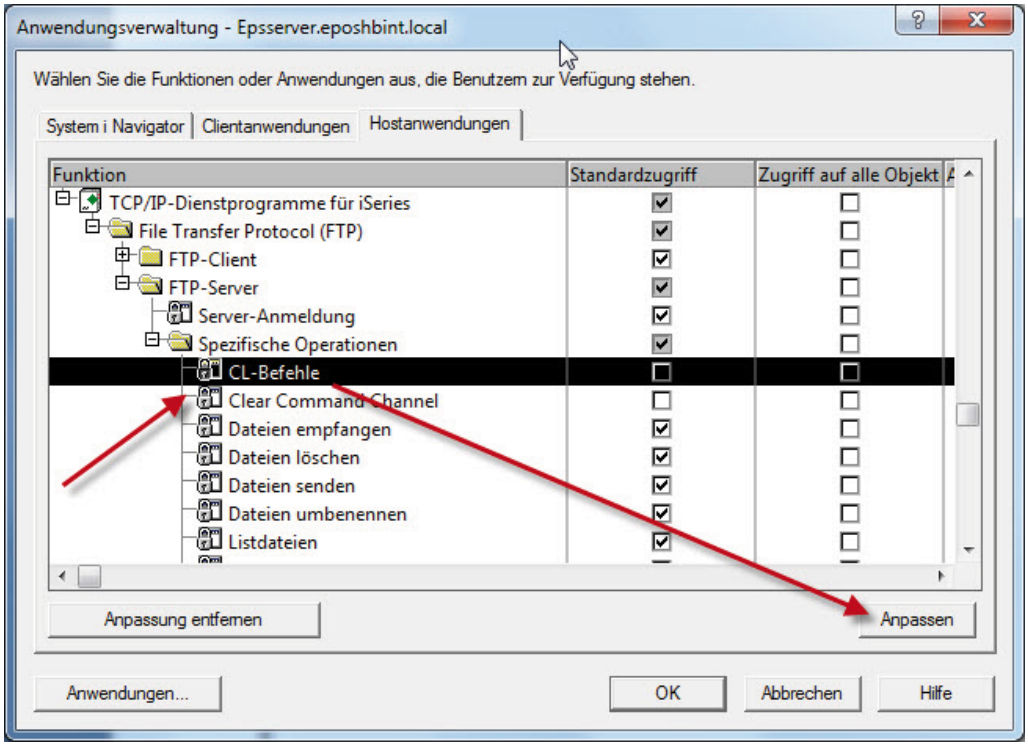
Interessant bei den „Hostanwendungen“ ist, dass nicht alle Administratoren automatisch das Recht besitzen, auf diese systemnahen Funktionen zugreifen zu können.

Bei den TCP/IP-Netzwerkfunktionen werden derzeit nur die FTP-Funktionen geschützt. Das ist ein wenig enttäuschend, denn die Brisanz von Telnet- und SMTP-Zugriffen wird dabei komplett ignoriert.

Ich möchte Ihnen jetzt zeigen, wie man die Remote-Command-Schnittstelle des FTP-Servers entschärft. Dies ist eine mächtige Funktion des iSeries FTP-Servers, die es möglich macht, CL-Befehle mit Hilfe eines FTP-Clients an den FTP-Server der iSeries zu übergeben. Bedenken Sie, dass auch ein CALL-Befehl zur Gruppe der CL-Befehle gehört.

Ich markiere also mit der Maus die Funktion „TCP/IP Dienstprogramme ➜ FTP-Server ➜ Spezifische Operationen ➜ CL-Befehle“.

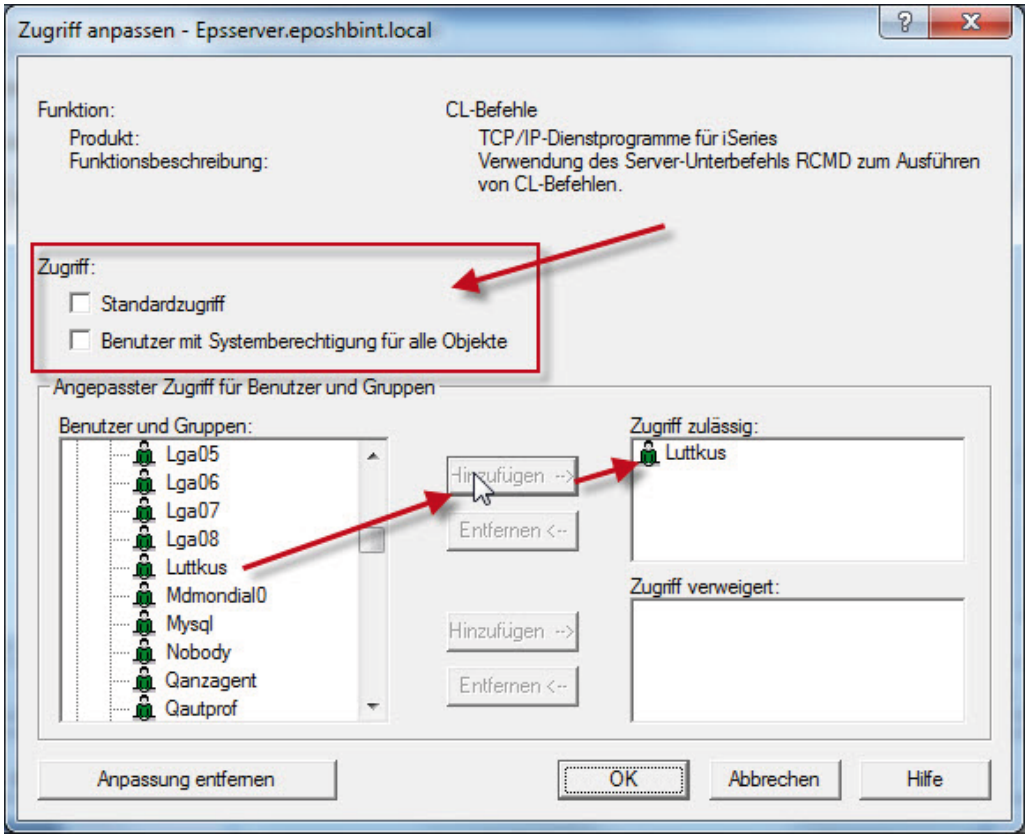
7.2.6.9
Seite 8



FTP-Funktion einschränken

Dann klicke ich auf den Knopf „Anpassen“. Jetzt öffnet sich folgender Dialog.

Auf der linken Seite des nachfolgenden Fensters sehen Sie die im Betriebssystem i registrierten „Benutzer und Gruppen“.



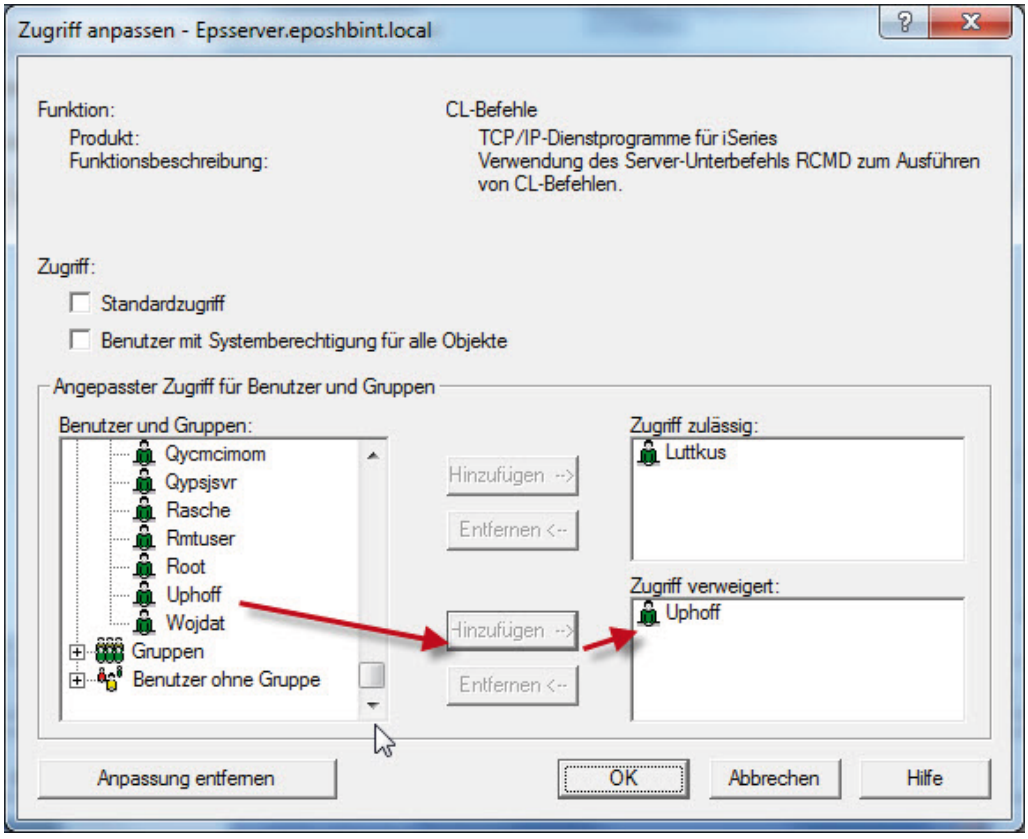
Zugriff anpassen 1

In der Benutzeranzeige wähle ich den Benutzer „Luttkus“ und klicke auf „Hinzufügen“.

Sonstige Zugriffe werden deaktiviert.

7.2.6.9
Seite 10

Danach wähle ich einen weiteren Benutzer – „Uphoff“ – aus.

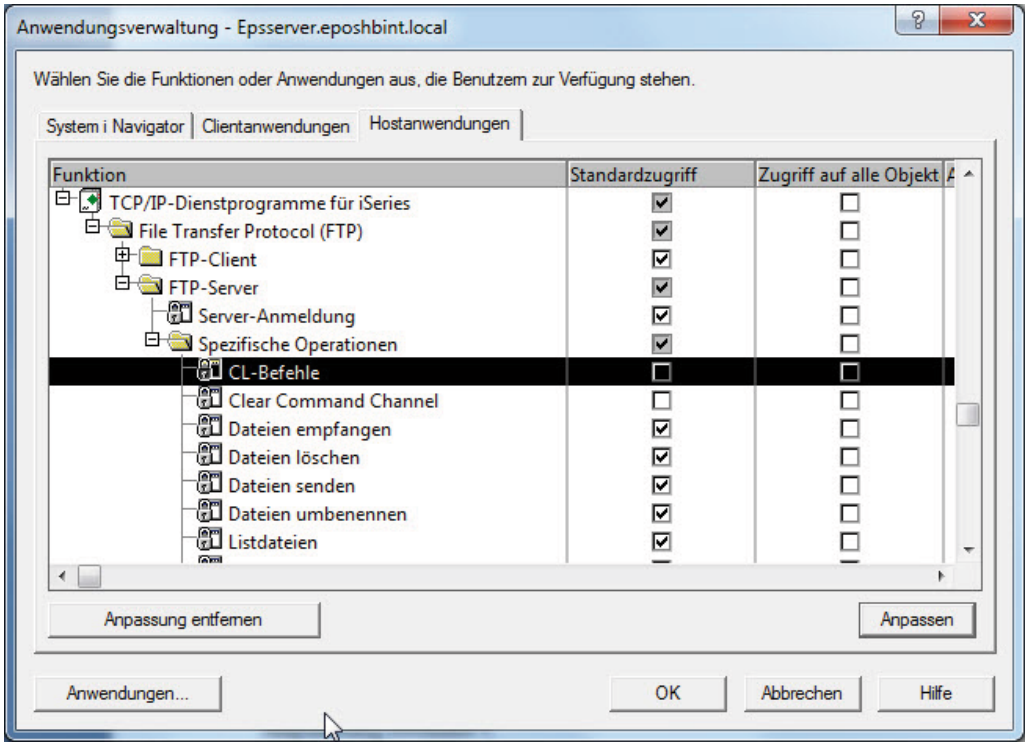


Zugriff anpassen 2

Uphoff übernehme ich durch den unteren Knopf „Hinzufügen“ in die Liste „Zugriff verweigert“.

Auch hier werden alle Standardzugriffe deaktiviert.

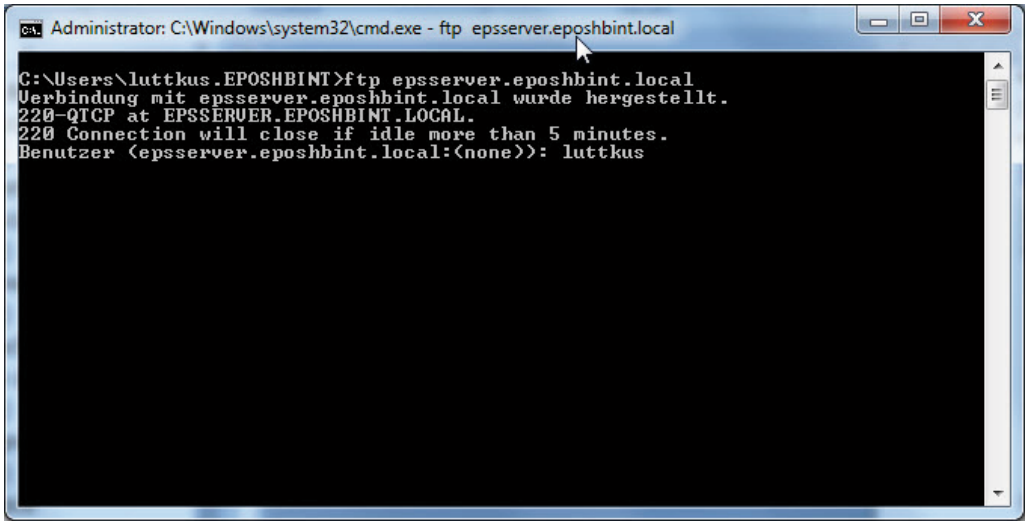
Danach verlasse ich den gesamten Dialog mit „OK“.



Anpassung beendet

Auf meinem Windows-PC öffne ich jetzt die Befehlszeile zum Betriebssystem, indem ich über „Start ➜ Ausführen“ den Befehl cmd.exe ausführe.

Dann rufe ich meinen FTP-Client für das System EPSServer auf.



FTP aufrufen

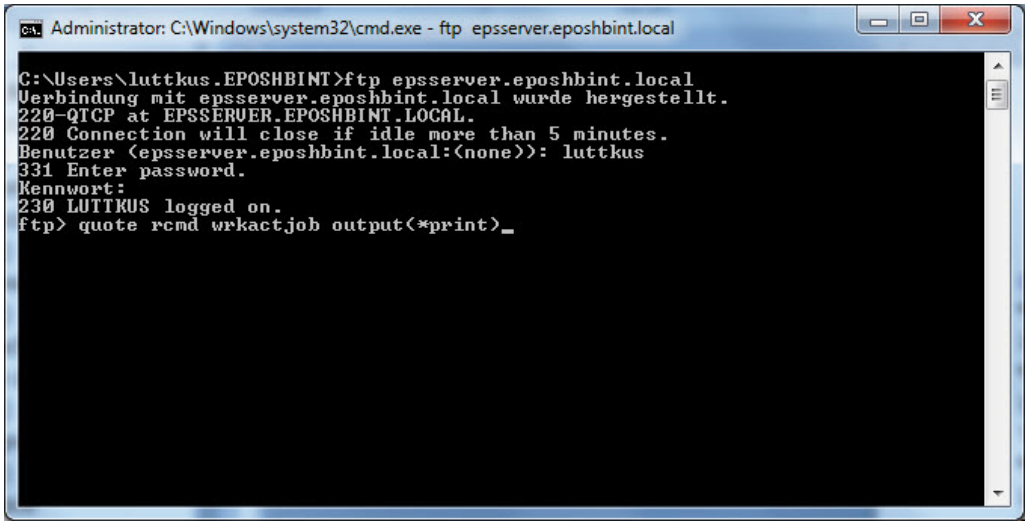
Ich werde aufgefordert, mich anzumelden.

Ich melde mich mit dem Namen „Luttkus“ an und bestätige den Vorgang der Passwort-Angabe.

7.2.6.9
Seite 12

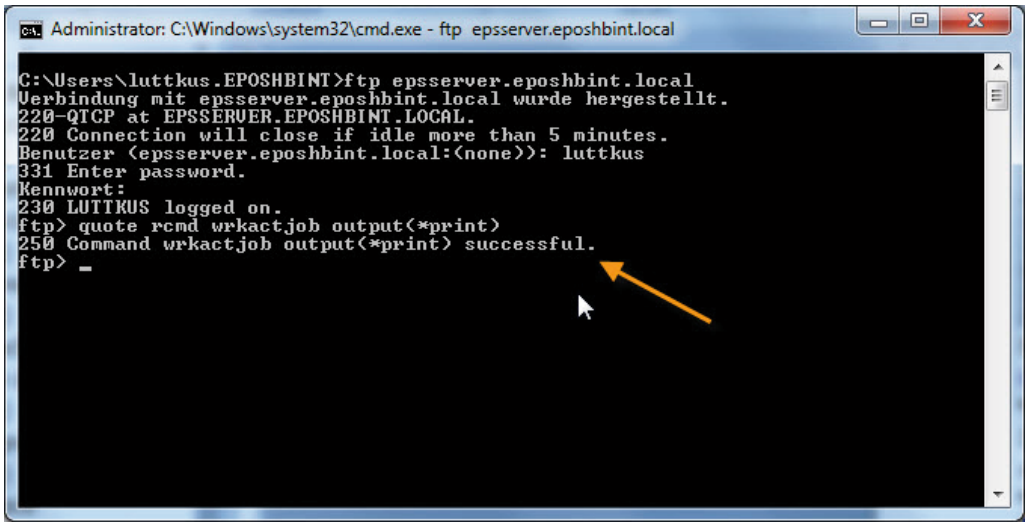
Ich gebe folgende FTP-Befehlsfolge ein:

```
Quote rcmd wrkactjob output(*print)
```



Remote Command aufrufen 1

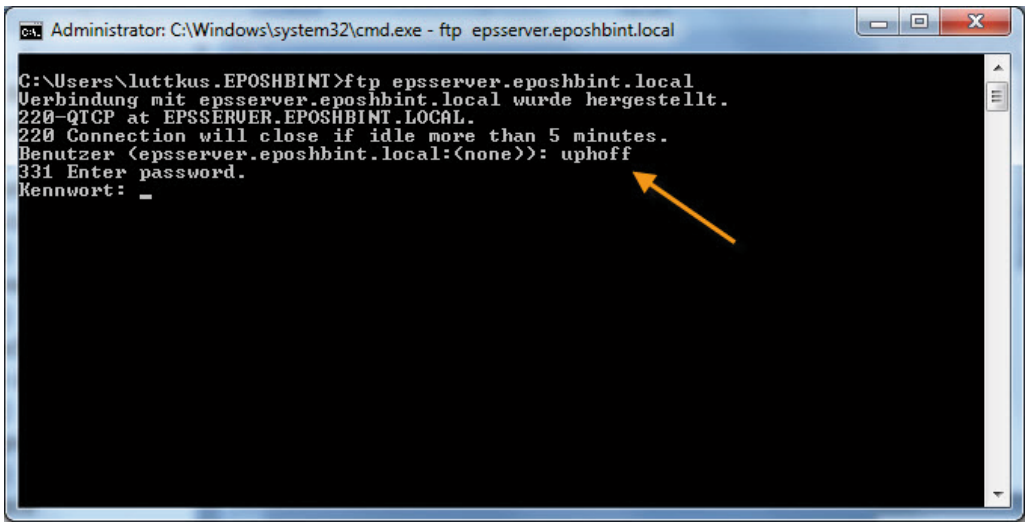
Dieser FTP-Befehl soll im OS/400 den Befehl WRKKACTJOB ausführen und die erzeugte Ausgabe in eine Liste drucken.



WRKACTJOB wurde ausgeführt

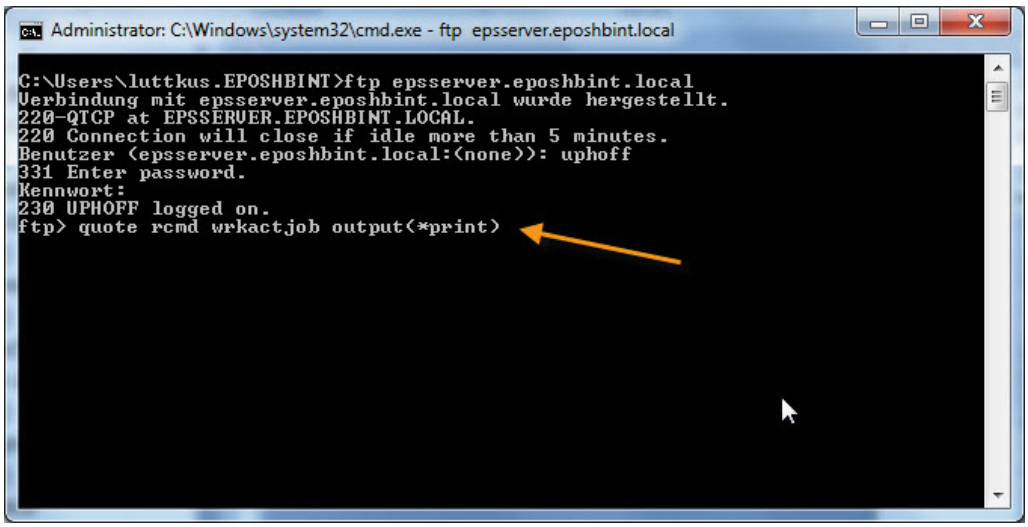
Sie sehen, dass der Befehl erfolgreich ausgeführt wurde. Ich habe dem Benutzer „Luttkus“ die Ausführung dieser Funktion in der Anwendungsverwaltung gestattet, was eigentlich auch nicht anders zu erwarten gewesen wäre.

Ich wiederhole den gesamten Vorgang für den Benutzer „Uphoff“.



FTP von Luttkusb

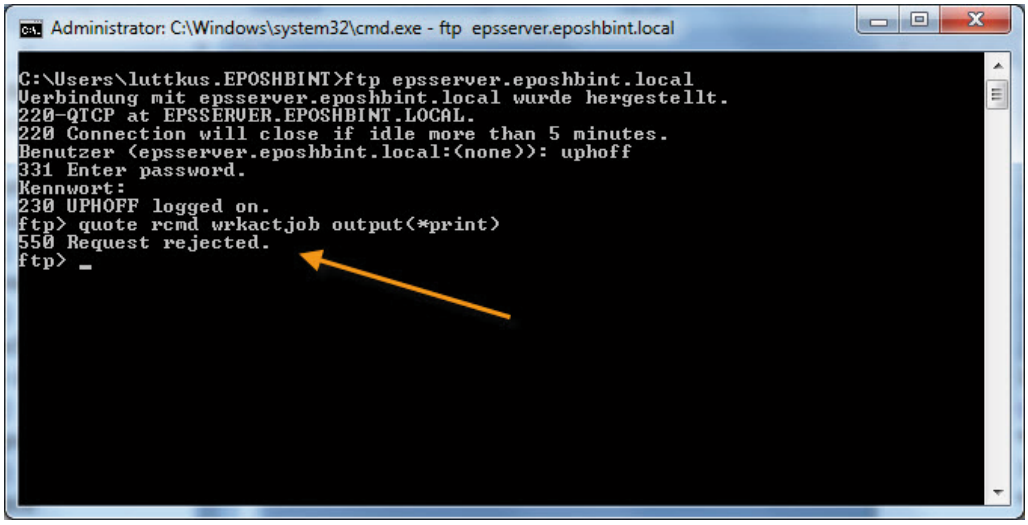
Auch der Benutzer „Uphoff“ führt den Befehl Quote rcmd wrkactjob output(*print) aus.



Remote Command aufrufen 2

7.2.6.9
Seite 14

Sehen Sie sich jetzt im unteren Bild an, was geschehen ist.



WRKACTJOB-Ausführung verweigert

Die Ausführung der FTP-Befehlssequenz wurde verweigert.

Folgendes habe ich Ihnen bisher noch nicht verraten: Der Benutzer „Uphoff“ ist eine Kopie des Benutzers „Luttkus“ und beide gehören der Benutzerklasse *SECOFR an, sind also Administratoren des Systems.

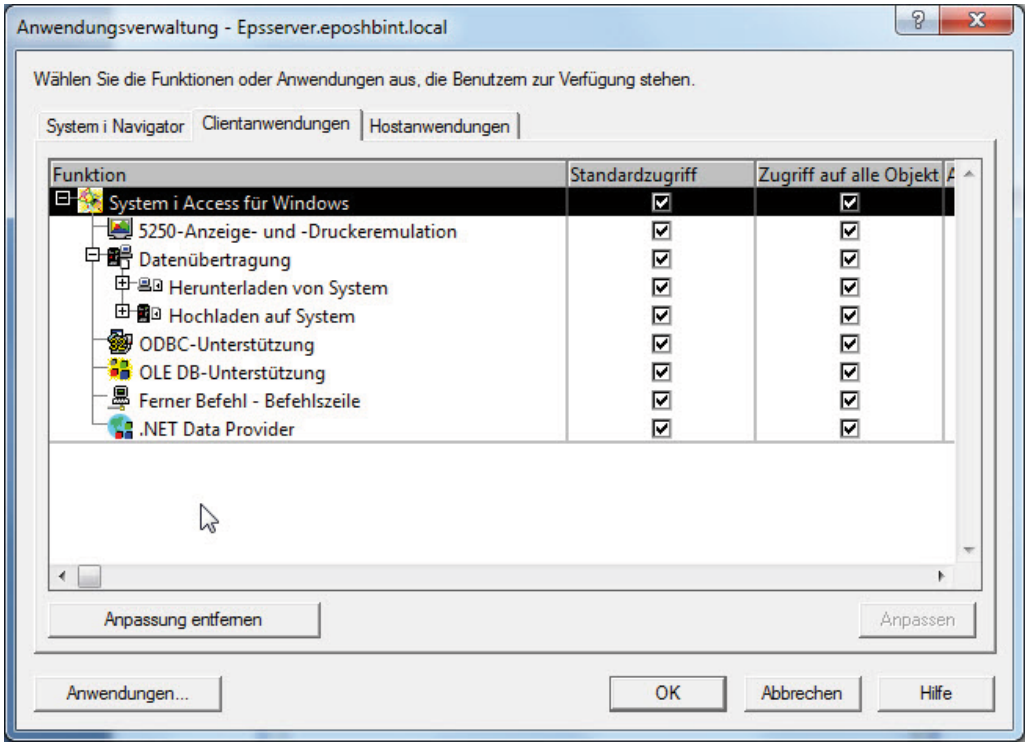
Sie sehen also, dass Sie sich mit der Anwendungsverwaltung auch gegen Benutzer schützen können, die bezüglich des Objektrechts auf höchster Ebene im OS/400 arbeiten.

Im Folgenden zeige ich Ihnen zur Übersicht noch einmal die komplett expandierten Register der Anwendungsverwaltung.

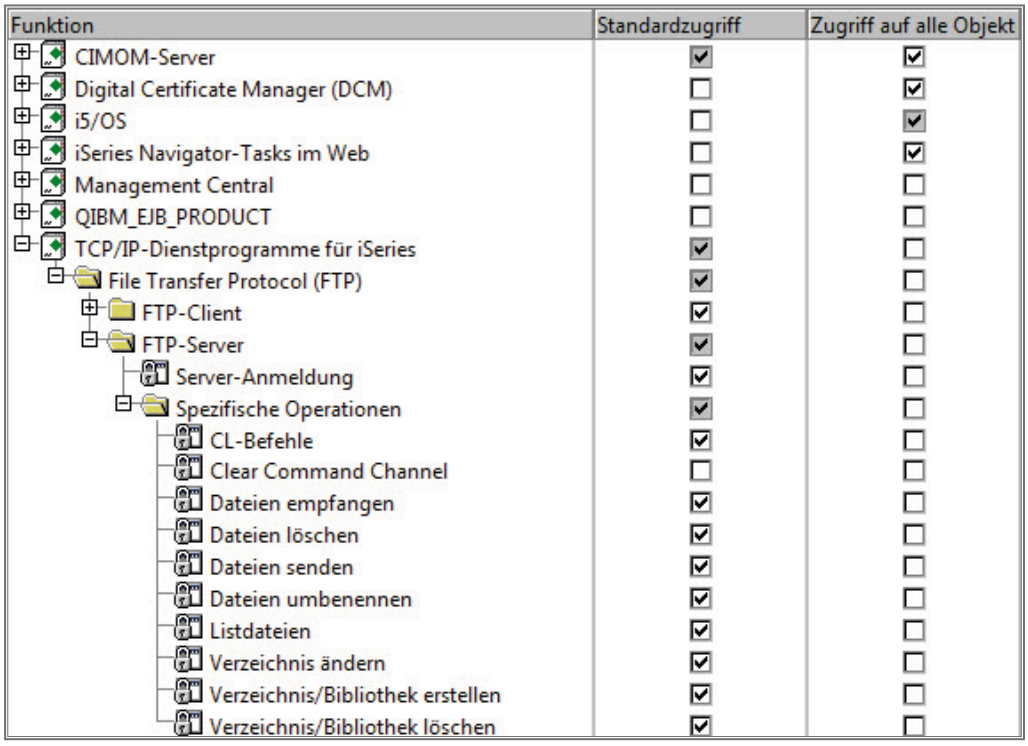
Funktion	Standardzugriff	Zugriff auf alle Objekt
Epsserver.eposhbint.local in Meine Verbindungen	☑	☑
Basisoperationen	☑	☑
Nachrichten	☑	☑
Druckausgabe	☑	☑
Drucker	☑	☑
Jobs	☑	☑
Ablaufsteuerung	☑	☑
Aktive Jobs	☑	☑
Server-Jobs	☑	☑
Jobwarteschlangen	☑	☑
Ausgabewarteschlangen	☑	☑
Subsysteme	☑	☑
Arbeitsspeicherpools	☑	☑
Konfiguration und Service	☑	☑
Systemwerte und Zeitmanagement	☑	☑
Systemprotokoll	☑	☑
Hardware	☑	☑
Software	☑	☑
Fix-Inventar	✔	✔
Erfassungsservices	✔	✔
Netzwerk	☑	☑
TCP/IP-Konfiguration	☑	☑
Remote Access Services	☑	☑
Server	☑	☑
IP-Richtlinien	☑	☑
Enterprise Identity Mapping	☑	☑
Internet	☑	☑
IBM Network Stations (nur Clients vor V5R3M0)	☑	☑
Verwaltung für integrierten Server	☑	☑
Sicherheit	☑	☑
Berechtigungslisten	☑	☑
Richtlinien	☑	☑
Netzwerkauthentifizierungsservice	☑	☑
Cryptographic Services Key Management	☑	☑
Erkennung von unbefugtem Zugriff	☑	☑
Benutzer und Gruppen	☑	☑
Datenbanken	☑	☑
Schemata	☑	☑
Database Navigator-Masken	☑	☑
SQL-Leistungsüberwachungen	☑	☑
Momentaufnahme für SQL-Planungscache	☑	☑
ODBC-Datenquellen (nur Clients vor V5R1M0)	☑	☑
Transaktionen	☑	☑
Dateisysteme	☑	☑
Integrated File System	☑	☑
Root	☑	☑
QOpenSys	☑	☑
QDLS	☑	☑
QSYS.LIB	☑	☑
QFileSvr.400	☑	☑
QLANSrv	☑	☑
QNetWare	☑	☑
QOPT	☑	☑
QNTC	☑	☑
Dateifreigaben	☑	☑
Multimedia (nur Clients vor V5R1M0)	☑	☑
Sicherung	☑	☑
Anwendungsentwicklung	☑	☑
AFP Manager	☑	☑
Anwendungsverwaltung	☑	☑

Register System i Navigator

7.2.6.9
Seite 16



Register Clientanwendungen



Register Host-Anwendungen mit FTP

Anwendungsverwaltung mit zentralen Verwaltungssystemen

Mit OS/400 V5R2 und dem entsprechenden Navigator wurde die Anwendungsverwaltung noch einmal erweitert und zwar für solche Unternehmen, die mehrere i-Systeme betreiben.

Ein Nachteil der bisherigen Anwendungsverwaltung war, dass man die Berechtigungsdaten auf jedem System separat legen musste.

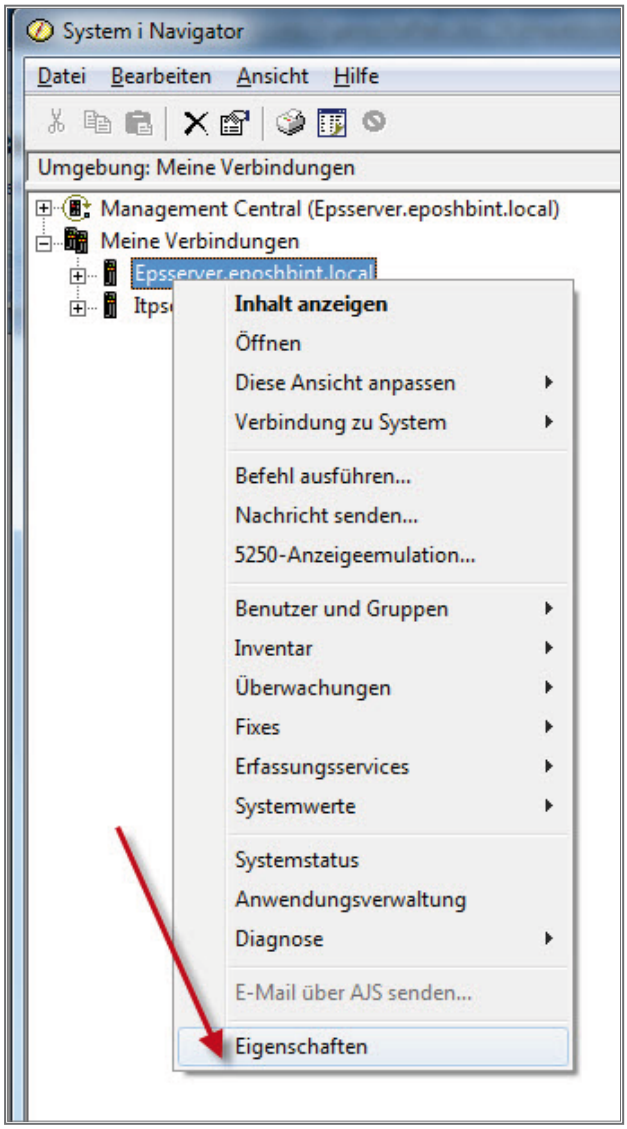
Dieser Nachteil wurde mit OS/400 V5R2 zumindest für die Absicherung von i Access-Funktionen behoben, indem neben der lokalen Anwendungsverwaltung eine zentrale Anwendungsverwaltung eingeführt wurde.

Lokal bedeutet in diesem Zusammenhang, dass Berechtigungsdaten nur auf einem System wirksam werden können. Zentral heißt, dass im Rahmen eines Verwaltungssystems Berechtigungsdaten zentral für mehrere i-Systeme verwaltet werden.

Der erste Arbeitsschritt, um die zentrale Anwendungsverwaltung einzurichten, besteht darin, ein verwaltendes System zu benennen.

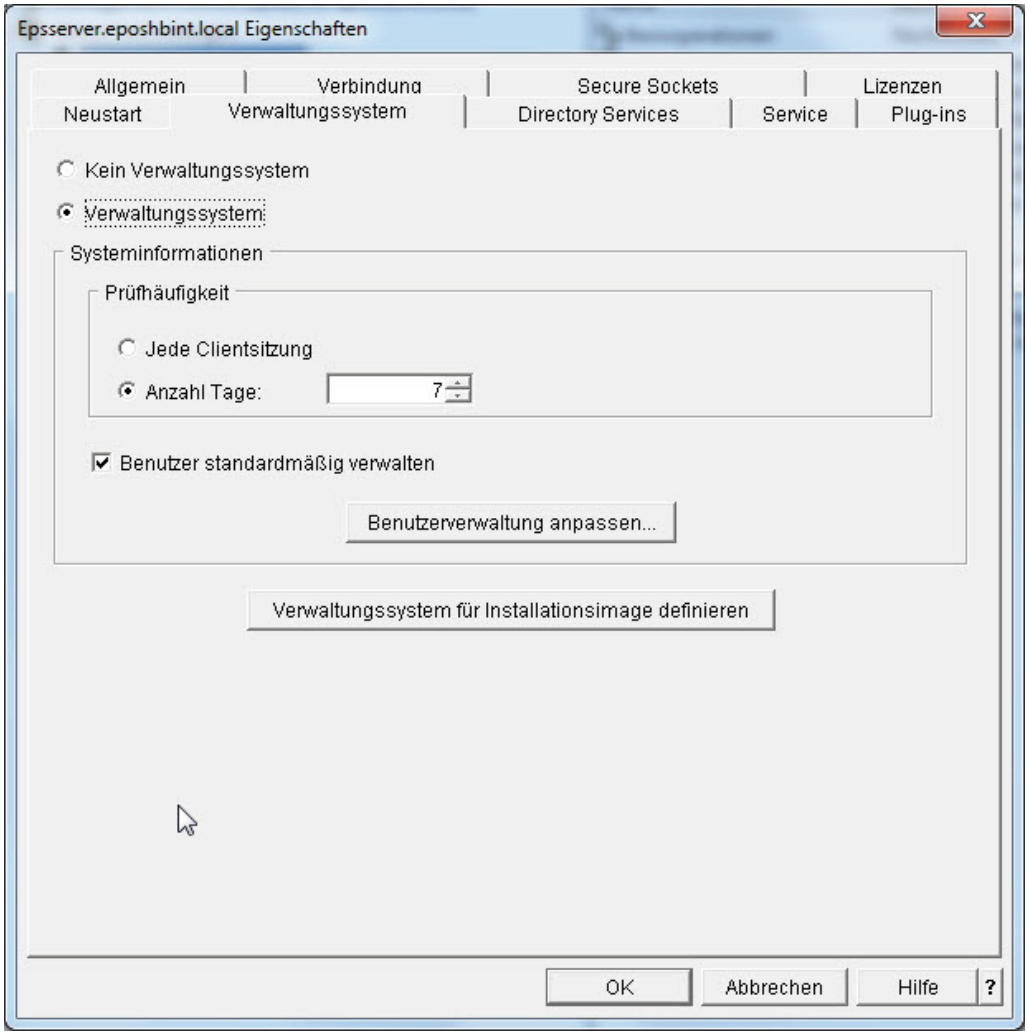
Dies erreiche ich dadurch, dass ich mit der rechten Maustaste im i Navigator das System auswähle, welches ich zum verwaltenden System erklären möchte. In meinem Fall handelt es sich wieder um das System hinter der Verbindung EPSServer.eposhbint.local.

7.2.6.9.1
Seite 2



Verbindungseigenschaften

Ich wähle den Menüpunkt „Eigenschaften“ des Verbindungskontextmenüs für das System EPSServer.eposhbint.local an.



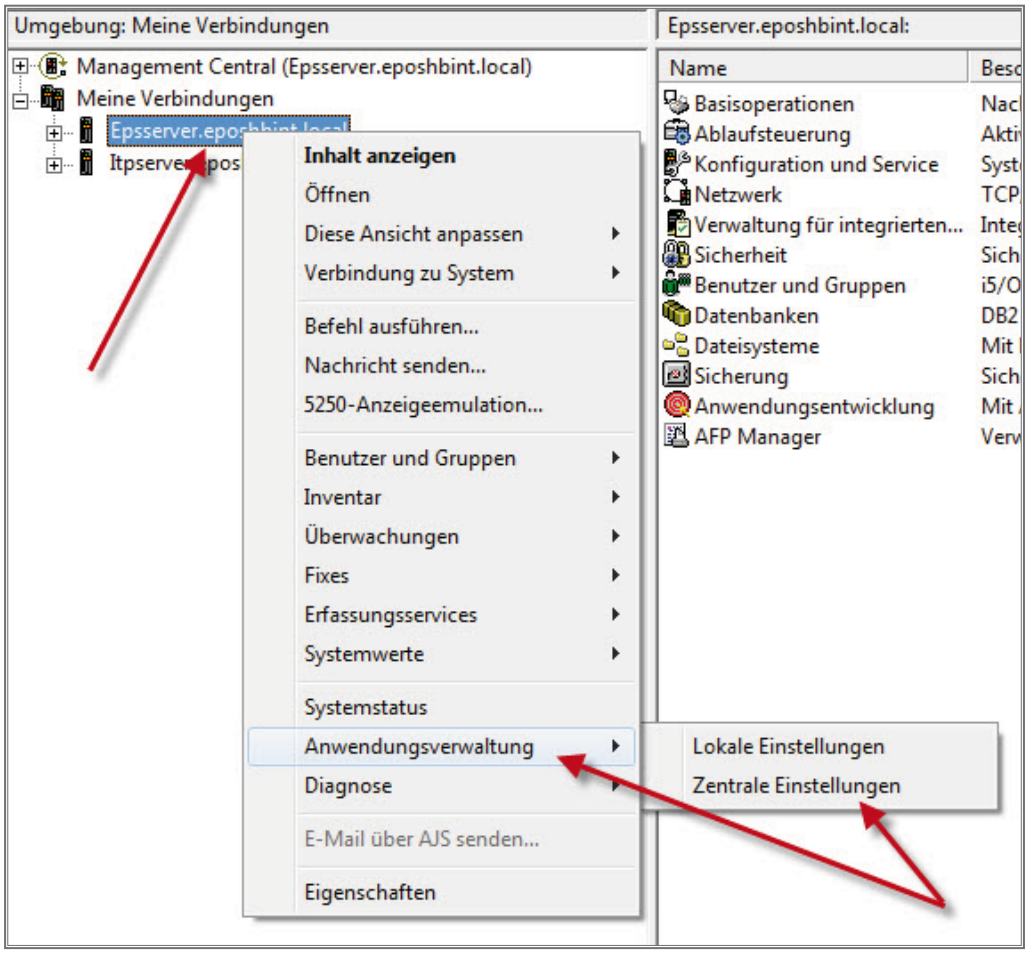
Eigenschaften „Verwaltungssystem“

In den „Eigenschaften“ klicke ich auf das Registerblatt „Verwaltungssystem“. Ich aktiviere den Button „Verwaltungssystem“.

Alle anderen Parameter belasse ich bei den Standard-Einstellungen.

Jetzt muss der i Navigator erst einmal geschlossen werden. Beim erneuten Aufruf expandiere ich wieder das Kontextmenü für die Systemverbindung EPSServer.eposhbint.local.

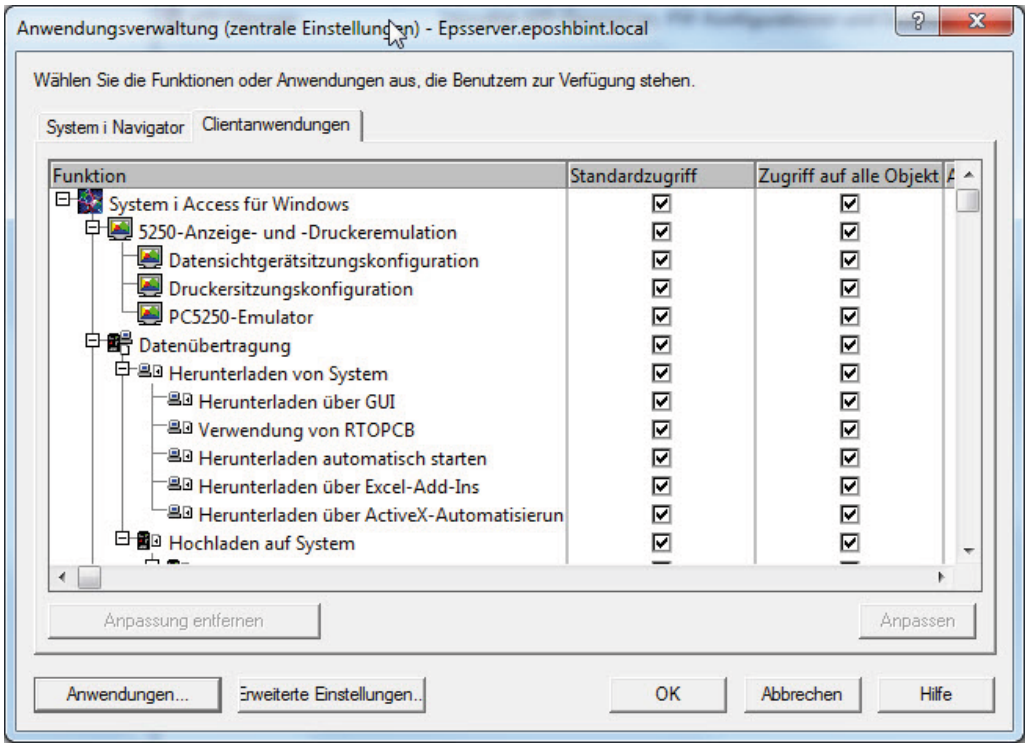
7.2.6.9.1
Seite 4



Anwendungsverwaltung

Ich klicke auf „Anwendungsverwaltung“; sehen Sie jetzt, was passiert. Ein weiteres Menü öffnet sich.

Zusätzlich zu den lokalen Einstellmöglichkeiten sehen Sie einen Eintrag „Zentrale Einstellungen“. Diesen Punkt wähle ich an.



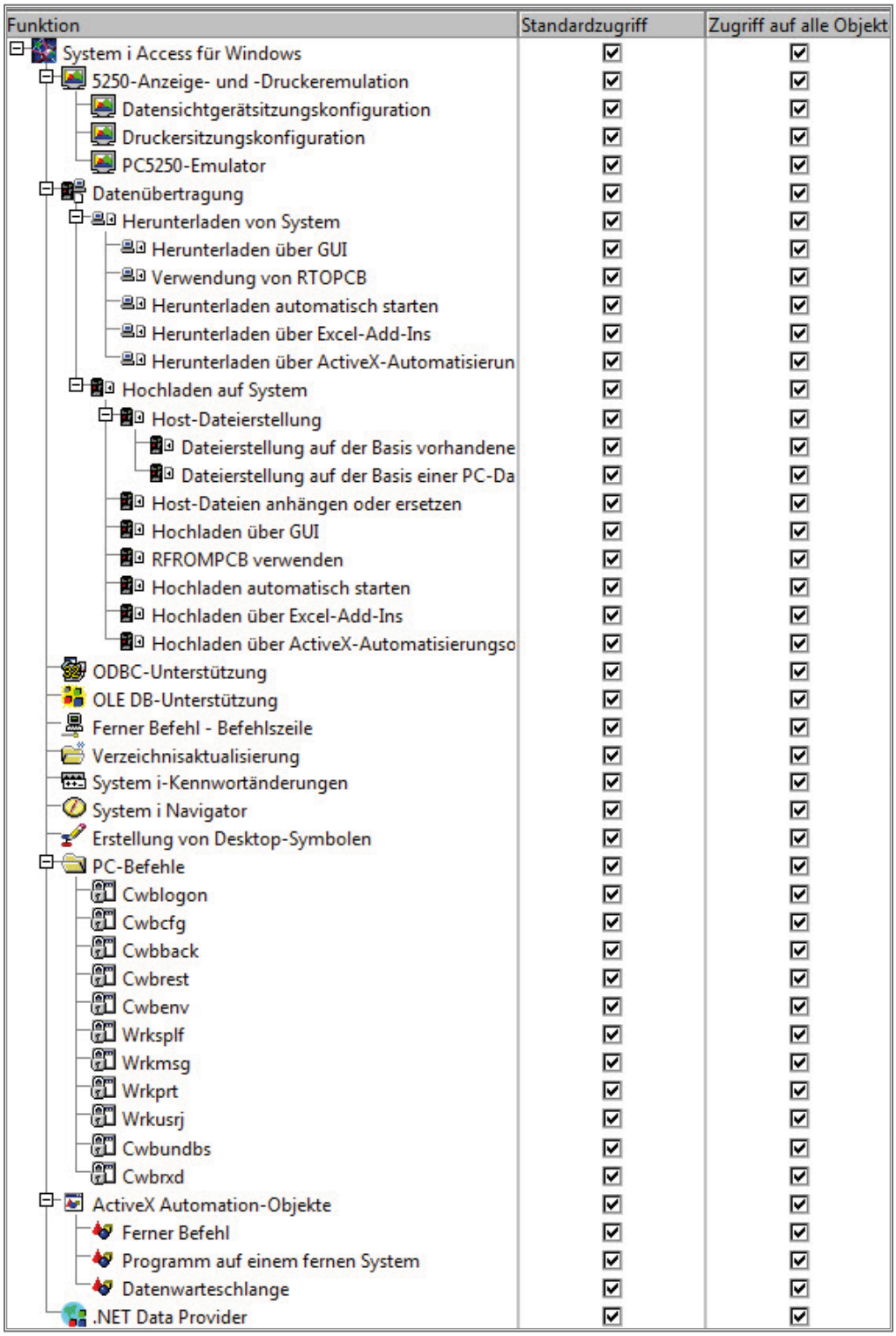
Zentrale Einstellungen

Dieser Dialog hat zwei Registerblätter. Das Register „iSeries Navigator“ ist derzeit leer und soll wohl in einer späteren Version noch aktiviert werden.

Aber das Register „Clientanwendungen“ ist prall gefüllt.

Vergleichen Sie die im Folgenden aufgelisteten kompletten Funktionen der Clientanwendungen mit denen der lokalen Anwendungsverwaltung.

7.2.6.9.1
Seite 6

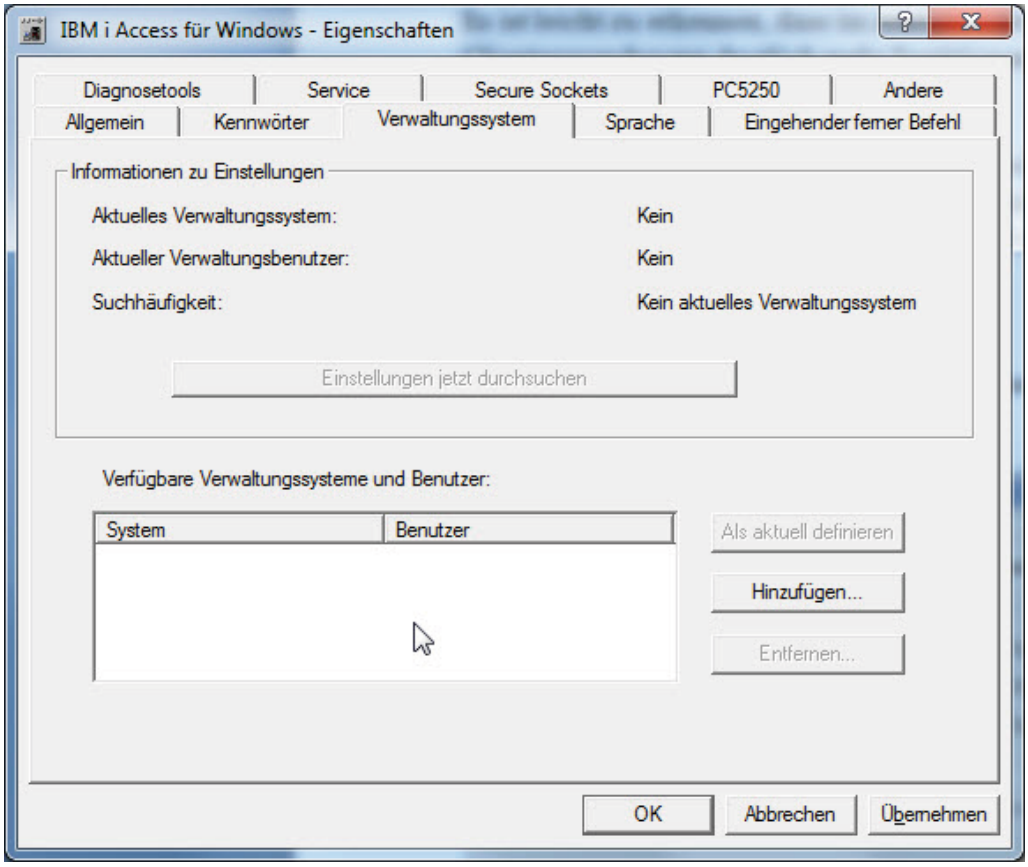


Clientanwendungen zentral

Es ist leicht zu erkennen, dass im zentralen Bereich der Clientanwendungen deutlich mehr Funktionen des i Access entschärft werden können, als im lokalen Bereich.

Ein letzter Arbeitsschritt ist jedoch noch auf einem Client, gegen den sich das Betriebssystem i absichern will, zu leisten. Ich muss dem Client mitteilen, welches sein verwaltendes System ist, von dem er seine Berechtigungsdaten bekommt.

In der i Access-Programmgruppe öffne ich das Symbol „IBM i-Access – Eigenschaften".



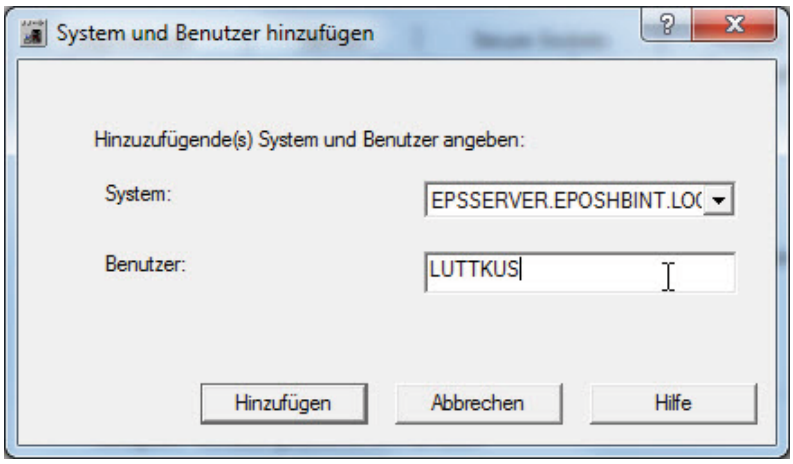
i Access-Eigenschaften

Hier klicke ich auf das Registerblatt „Verwaltungssystem".

Sie sehen, dass dieser Client bisher keinem Verwaltungssystem zugeordnet wurde. Das werde ich jetzt nachholen, indem ich auf den Knopf „Hinzufügen…" klicke.

7.2.6.9.1
Seite 8

Daraufhin öffnet sich der folgende kleine Dialog.

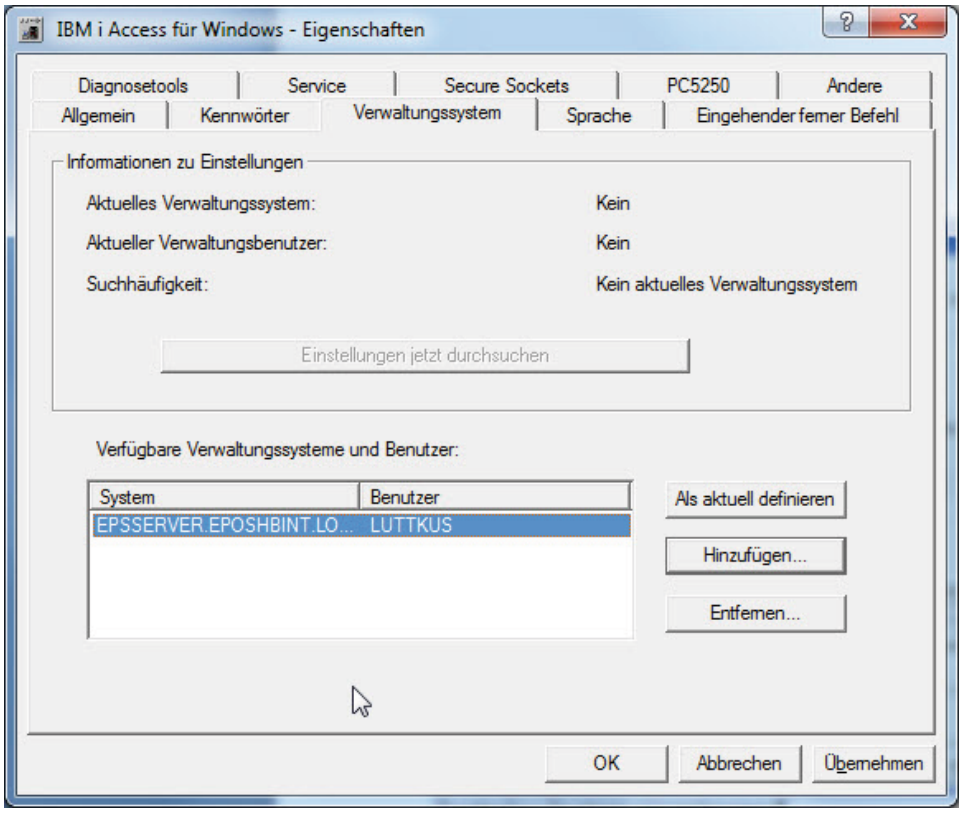


System und Benutzer hinzufügen 1

Aus der Pulldown-Liste des Parameters „System“ wähle ich die Adresse EPSSERVER…. aus – also die Adresse des Systems, das wir zum verwaltenden System gemacht haben.

Im Parameter „Benutzer“ trage ich einen Benutzer des Systems ein, der das Recht hat, auf die Sicherheitsdaten des Systems EPSSERVER zuzugreifen.

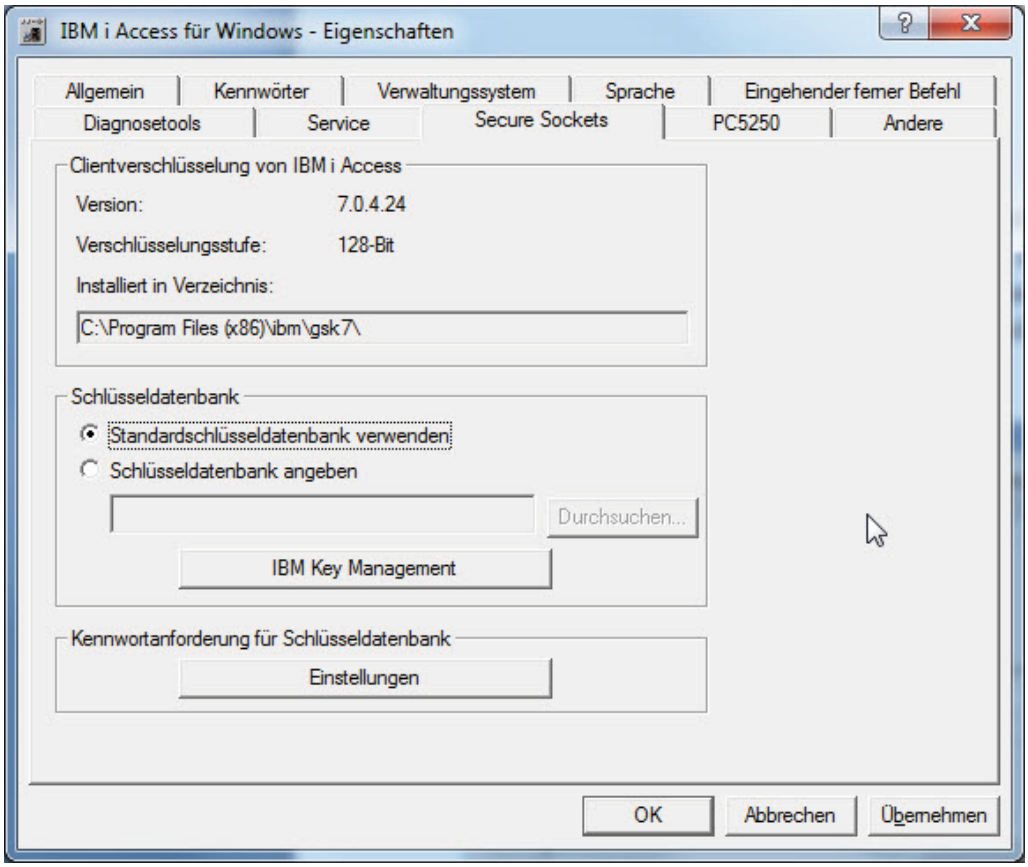
Ich beende den Dialog mit „Hinzufügen“.



Zentrales System eingetragen

Secure Sockets

Secure Sockets Layer (SSL) ist ein verbreitetes Sicherheitsschema, das dem PC-Client die Authentifizierung des Servers sowie die Verschlüsselung aller Daten und Anforderungen ermöglicht. i Access für Windows ermöglicht eine SSL-Kommunikation mit dem Server mit der 128-Bit-Verschlüsselungsstufe.



Register Secure Sockets

Secure Sockets Layer (SSL) überprüft die Identität durch zwei Methoden: die Clientauthentifizierung und die Serverauthentifizierung. Die Serverauthentifizierung findet statt, wenn der Client die Identität der Serveranwendung anhand des Serverzertifikats überprüft, das an die Clientanwendung übergeben wurde. Die Clientauthentifizierung findet statt, wenn der Server die Identität des Clients anhand des Clientzertifikats überprüft, das an die Serveranwendung übergeben wurde.

Wenn eine Clientauthentifizierung ausgeführt werden soll, muss zunächst eine Serverauthentifizierung stattfinden.

7.2.6.10
Seite 2

Clientauthentifizierung
Der PC5250-Telnet-Client unterstützt die erforderliche SSL-Clientauthentifizierung. Bei V5R1 und höheren Releases verwenden Sie Digital Certificate Manager (DCM), um die SSL-Eigenschaften des Telnet-Servers so zu ändern, dass die Clientauthentifizierung erforderlich ist.

Serverauthentifizierung
iSeries Navigator enthält eine Option zum Herunterladen der Zertifizierungsinstanz. Bei Verwendung der Zertifizierungsinstanz zum Erstellen von System- oder Clientzertifikaten automatisiert diese Option den Vorgang, mit dem die Zertifizierungsinstanz auf den PC heruntergeladen wird.

Um Secure Sockets Layer (SSL) verwenden zu können, muss die i Access für Windows-Komponente für Secure Sockets Layer (SSL) auf dem PC installiert worden sein. Die Komponente für SSL wird im Menü für die Komponentenauswahl während einer Installation von i Access für Windows oder bei Verwendung der Funktion "Selektive Installation" nach einer Installation angezeigt.

So installieren Sie SSL mit der Funktion "Selektive Installation":

1 Klicken Sie nacheinander auf Start > Programme > IBM i Access für Windows > Selektive Installation.

2 Befolgen Sie die Anweisungen, die in der Anzeige ausgegeben werden. Klicken Sie im Dialog "Komponentenauswahl" auf das Kästchen neben "Secure Sockets Layer", um dieses Kästchen auszuwählen. Befolgen Sie anschließend die Anweisungen, um die Installation vollständig auszuführen.

Installations-Images erzeugen

Wenn Sie in einer Umgebung arbeiten, in der viele gleichartige iSeries-Access-für-Windows-Konfigurationen zu erstellen sind, so lässt sich diese Arbeit einfacher gestalten, wenn Sie für alle gleichartigen Installationen ein sogenanntes Installations-Image erzeugen.

Bisher war die Erzeugung eines Images eine manuelle und fehleranfällige Angelegenheit, musste man doch die Dateien der Installations-CD zerpflücken und funktionsbezogen neu gliedern.

Mit iSeries Access für Windows ist die Erzeugung eines Image ein Kinderspiel geworden, auch wenn das Auffinden dieser Funktion keines ist.

Legen Sie dazu zuerst Ihre iSeries-Access-für-Windows-Installations-CD in das CD-Rom-Laufwerk Ihres Windows-PC und öffnen Sie den Explorer.

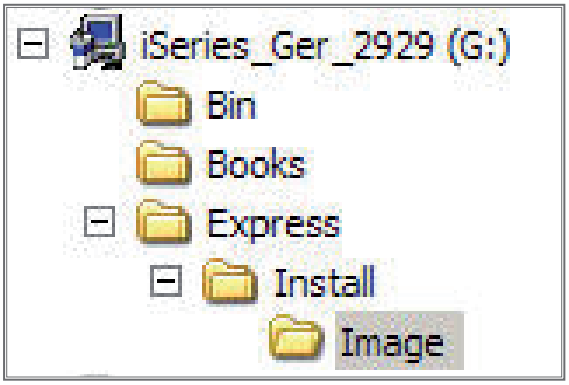


Image-Builder finden 1

Öffnen Sie den oben gezeigten Pfad: „LW:\Express\Install\Image“.

In diesem Pfad suchen Sie nach der Datei „cwbinimg.bat“.

Image-Builder finden 2

Ein Doppelklick auf diese Datei startet die Erstellung eines Installations-Images.

Eingangs sehen Sie das iSeries-Access-für-Windows-Logo, das auch während der Installation sowie des Upgrade eines Arbeitsplatzes angezeigt wird.

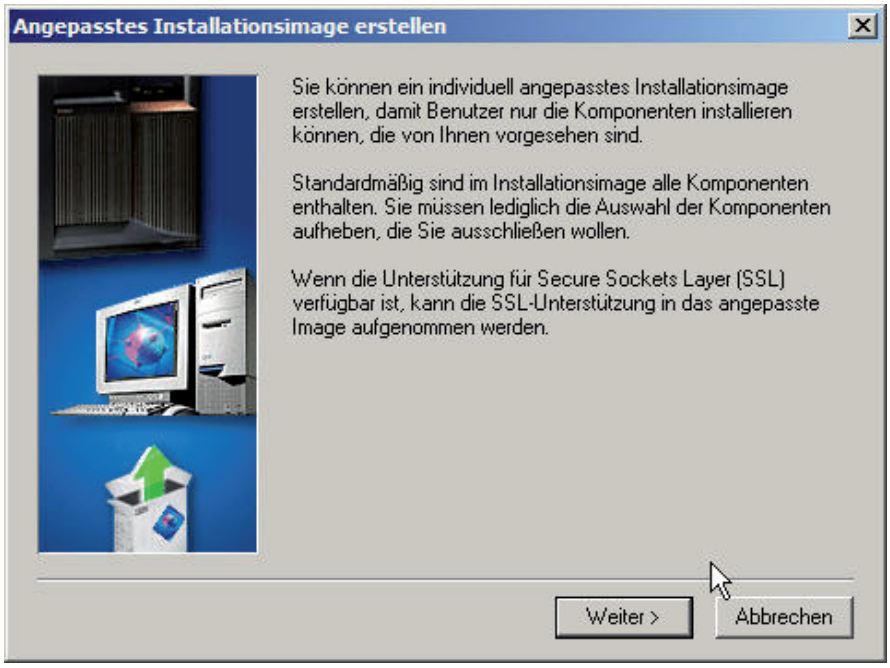
7.2.7
Seite 2

Image-Builder Logo

Keine Angst! Sie erstellen jetzt keine neue Installation. Die auf Ihrem Arbeitsplatz befindliche Client-Access- oder iSeries-Access-für-Windows-Installation wird durch diesen Vorgang nicht berührt.

Initialisierungsfortschritt

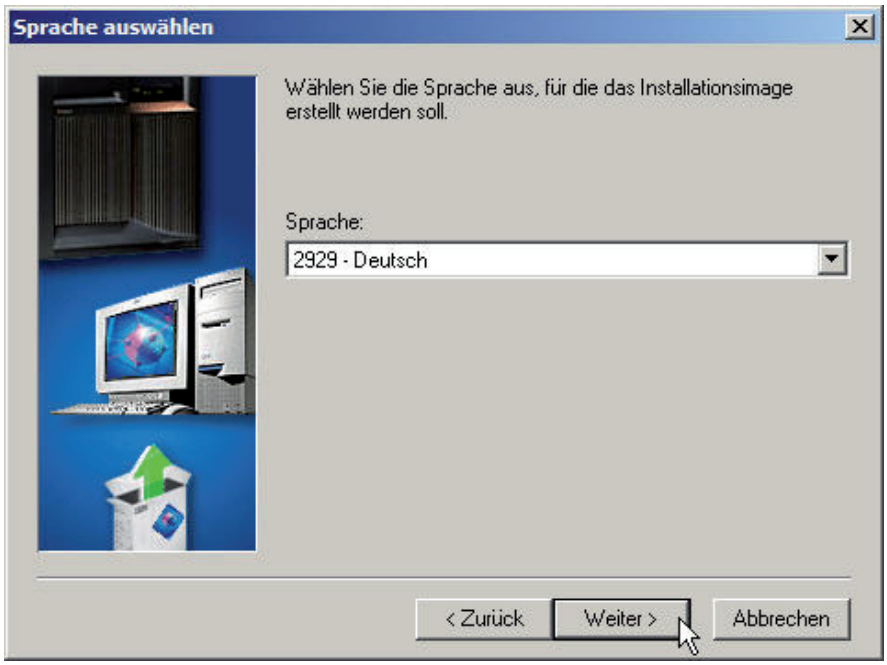
Der Fortschrittsbalken zeigt Ihnen den Fortgang des Ladeprozesses des Image-Builders an.



Beginn der Image-Erstellungsprozess

Zu Beginn des Build-Prozesses steht eine kleine Erläuterung, die aussagt, was in der Folge für Sie zu tun ist: Anfänglich sind alle Komponenten des iSeries Access für Windows ausgewählt. Ihre Aufgabe besteht darin, die Komponenten, die nicht installiert werden sollen, abzuwählen.

Wählen Sie jetzt die Sprache aus.



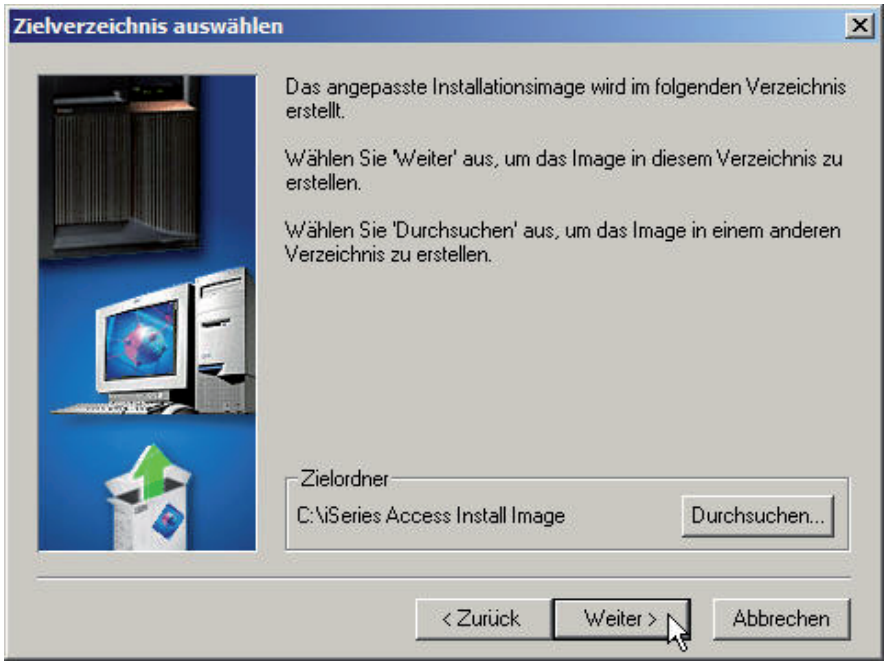
Sprache auswählen

7.2.7
Seite 4

Verfügen Sie über eine deutschsprachige AWI-Installations-CD, so können Sie hier nur „2929 – Deutsch“ auswählen.

Wenn Sie den Image-Builder aus dem IFS-Installationspfad der iSeries ausführen und Sie mehrere Sekundärsprachen installiert haben, dann können Sie auch Images für die Sprachvarianten der installierten Sekundärsprachen erstellen.

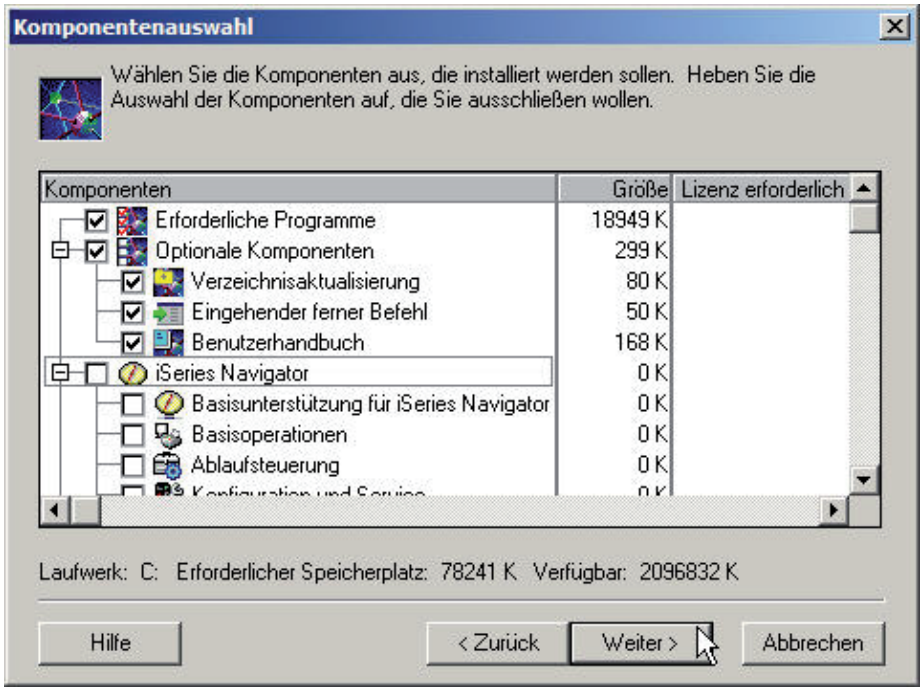
Als nächster Schritt erfolgt die Abfrage, in welches Zielverzeichnis das Image kopiert werden soll.



Zielverzeichnis für Image angeben

Ich habe es hier bei der Standardangabe belassen.

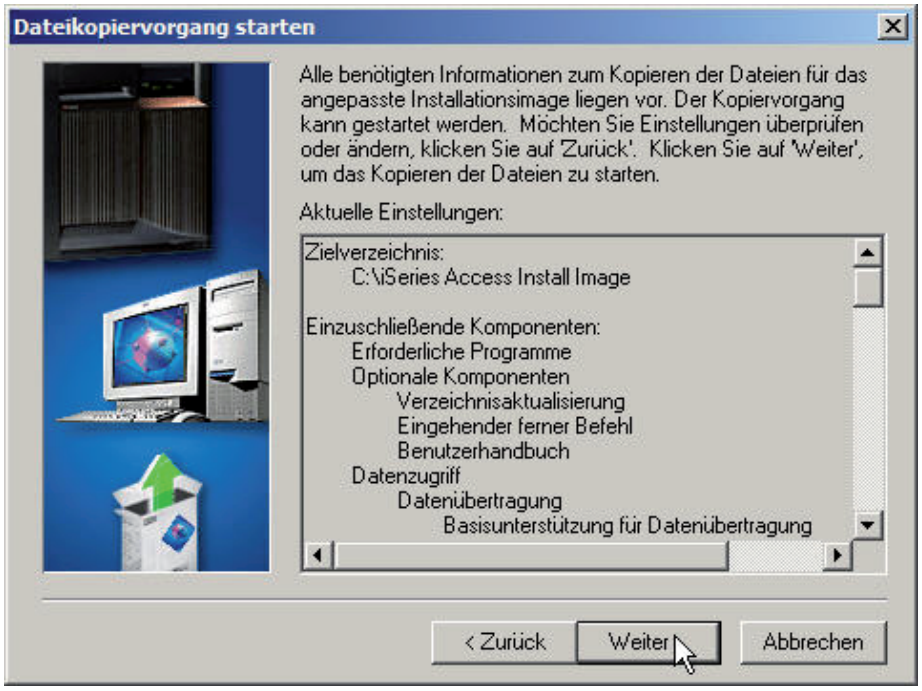
Dann erfolgt die Auswahl der iSeries-Access-für-Windows-Komponenten, die in das Image aufgenommen werden sollen.



Komponentenauswahl

Dieser Dialog ist Ihnen sicherlich aus der Installationsroutine für Client Access für Windows bekannt, so dass hier keine Erläuterung notwendig ist.

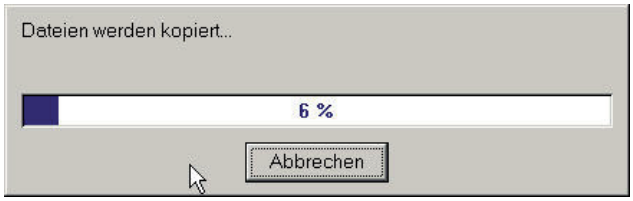
Anschließend kann noch einmal geprüft werden, welche Komponenten Bestandteil des Image sein sollen und welche Komponenten vom Image ausgeschlossen wurden.



Dateien ins Image-Verzeichnis kopieren

7.2.7
Seite 6

Dieser Dialog wird mit „Weiter“ bestätigt.



Kopierfortschritt

Anhand des Fortschrittsbalkens kann beobachtet werden, wie die Erstellung des Image vorangeschritten ist.

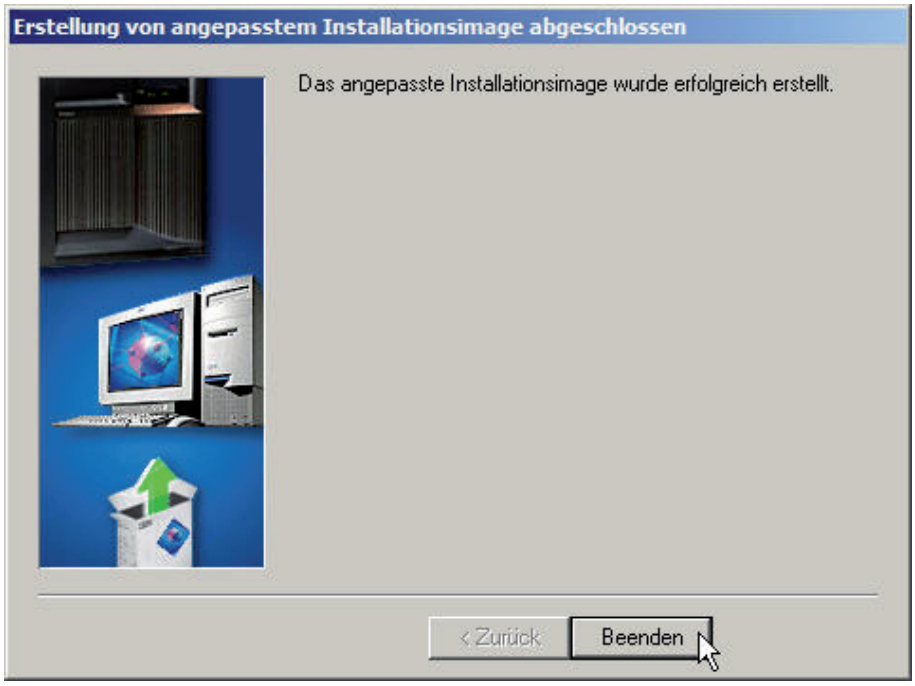
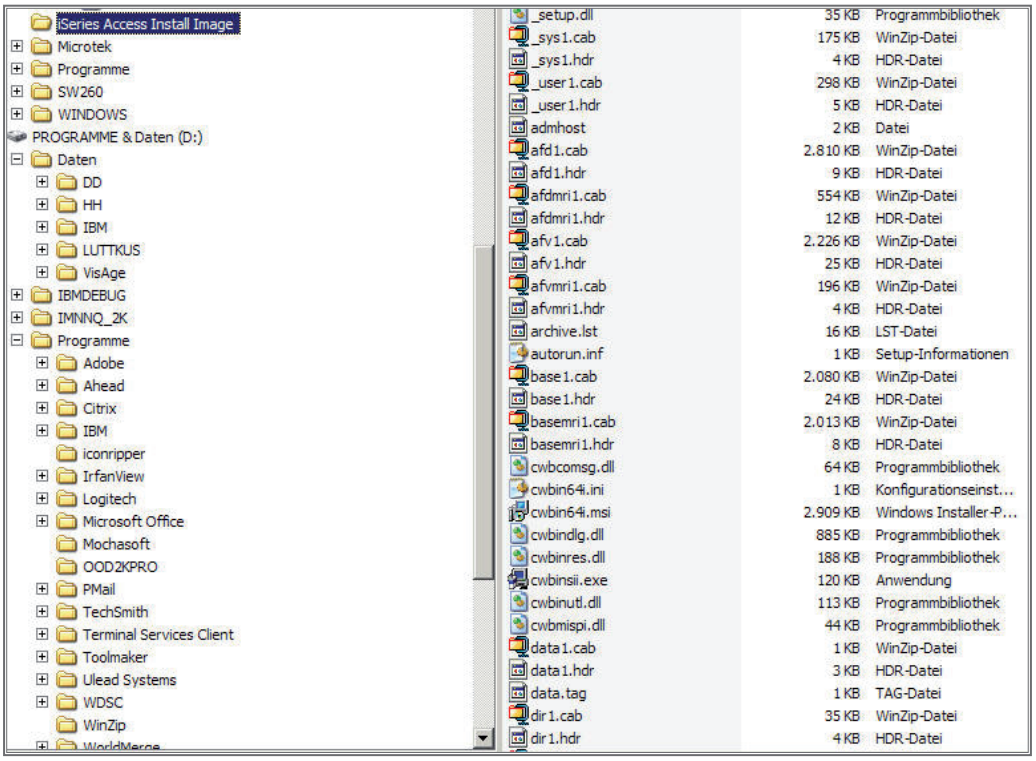


Image wurde erstellt

Sollten keine Fehler beim Erstellen des Images aufgetreten sein, so wird obiger Dialog angezeigt, der mit „Beenden“ geschlossen wird.

Im Image-Verzeichnis finden Sie dann alle Objekte des erstellten Images:



Das Image-Verzeichnis

Außerdem finden Sie im Image-Verzeichnis eine Datei „Setup.exe“, mit der das Image auf einen beliebigen Arbeitsplatz installiert werden kann.

7.2.7
Seite 8

Service-Pack und Installations-Image mischen

Auf Ihrem System i befindet sich – wie bereits beschrieben – ein Installations-Image des iSeries Access für Windows. Sie finden es in folgendem Pfad:

/QIBM/ProdData/CA400/Express/Install/Image

Wenn Sie iSeries Access-Arbeitsplätze aus diesem Verzeichnis installieren und wenn Ihnen IBM Service-Packs zur Verfügung gestellt werden, dann müssen Sie nach erfolgter Installation des iSeries Access das jeweilig aktuelle Service-Pack in einem weiteren Arbeitsschritt installieren.

Allerdings geht das auch einfacher! Mischen Sie einfach das aktuelle Service-Pack in das Installationsverzeichnis des iSeries Access für Windows hinein. Wie das geht, beschreibe ich Ihnen im Folgenden:

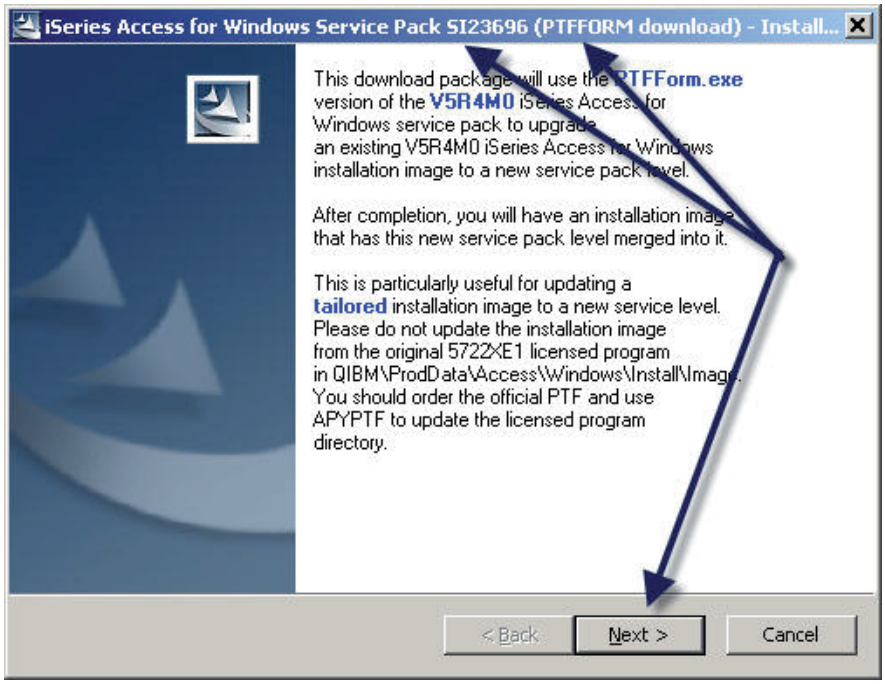
Laden Sie zuerst einmal das Programm FTFFORM von folgender ULR herunter:

- V5R4:
  ftp://ftp.software.ibm.com/as400/products/clientaccess/win32/v5r4m0/servicepack/si23696/tailored
- V5R3:
  ftp://ftp.software.ibm.com/as400/products/clientaccess/win32/v5r3m0/servicepack/si23492/tailored
- V5R2:
  ftp://ftp.software.ibm.com/as400/products/clientaccess/win32/v5r2m0/servicepack/si23978/tailored

In unserem Beispiel habe ich PTFFORM. EXE in das Verzeichnis hineingeladen, in dem mein aktuellstes Service-Pack für iSeries Access für Windows liegt.

Jetzt starte ich PTFFORM.

7.2.7.1
Seite 2

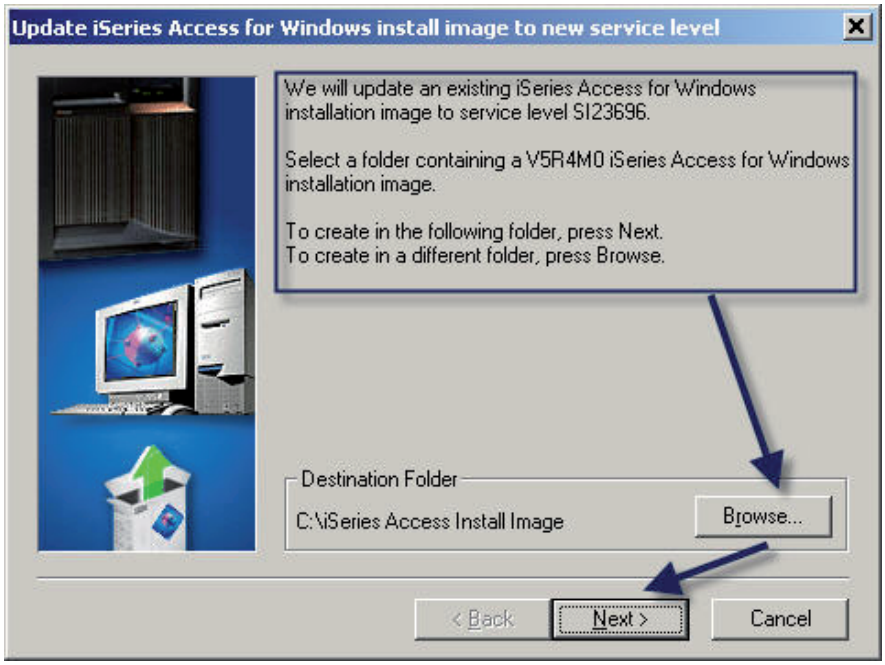


PTFFORM starten

Sie sehen in der Titelzeile, dass das Service-Pack SI23696 mit PTFFORM installiert wird.

Im ersten Verarbeitungsschritt wird das Service-Pack entpackt.

Dann beginnt das eigentliche Update des bestehenden iSeries Access-Installations-Images.

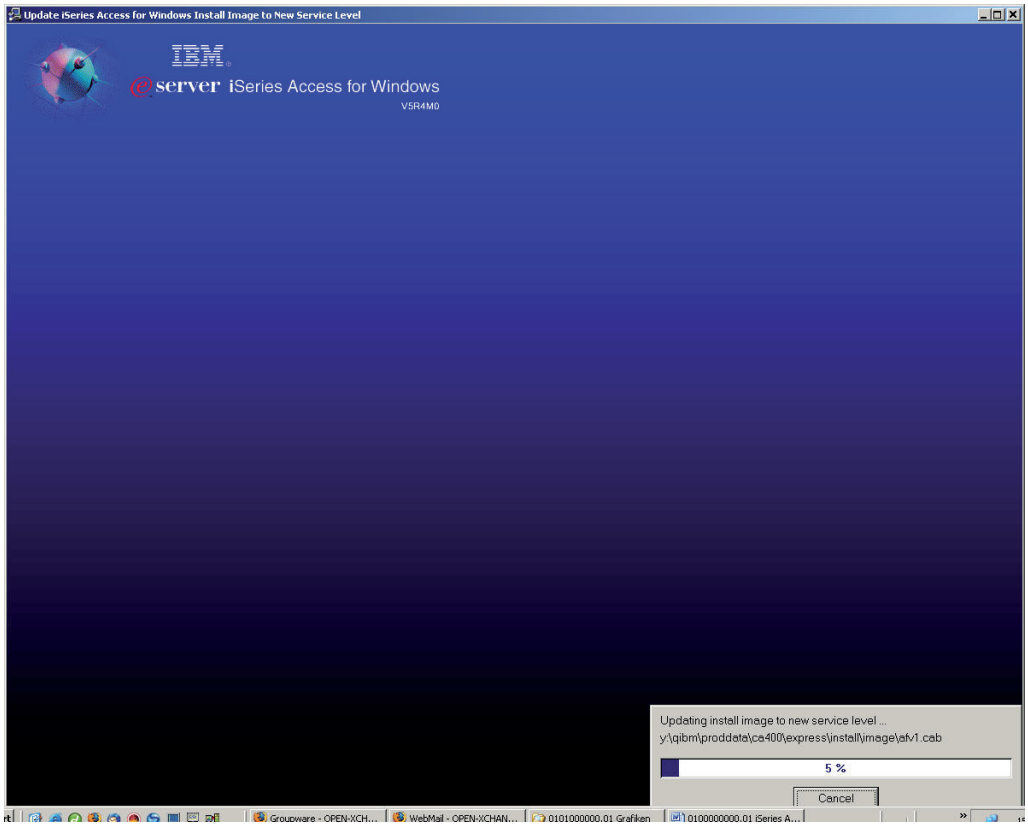


Nach bestehendem Image suchen

In oben gezeigtem Dialog werden wir aufgefordert, den Standort des zu aktualisierenden Install-Image anzugeben. Mit Hilfe des Buttons „Browse…“ suchen wir auf dem System i nach dem entsprechenden Verzeichnis:

/QIBM/ProdData/CA400/Express/Install/Image

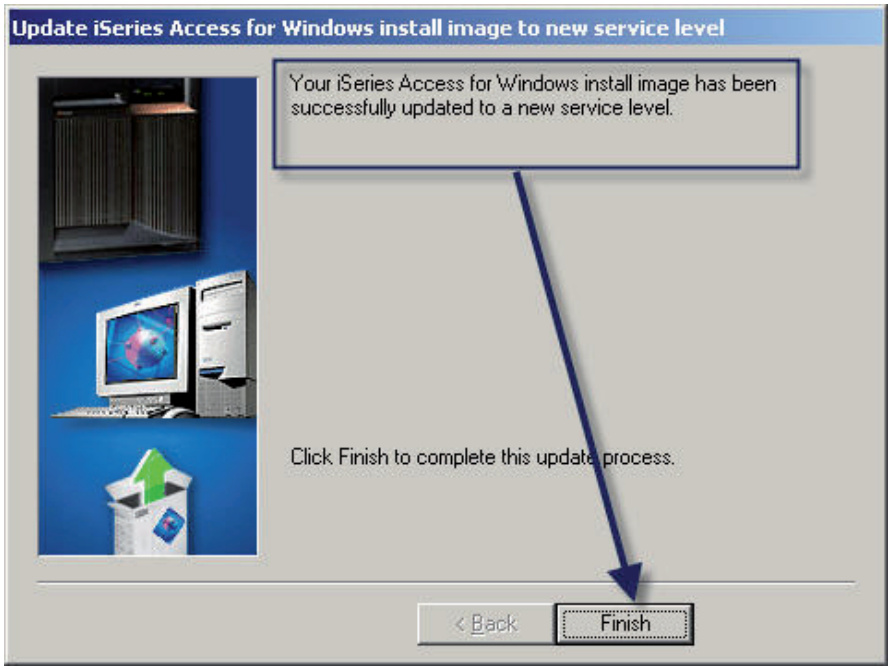
Der Dialog wird mit „Next“ beendet; anschließend beginnt der Aktualisierungsvorgang.



Aktualisierung des Image

Nachdem der Aktualisierungsprozess erfolgreich beendet wurde, erscheint folgendes Fenster:

7.2.7.1
Seite 4



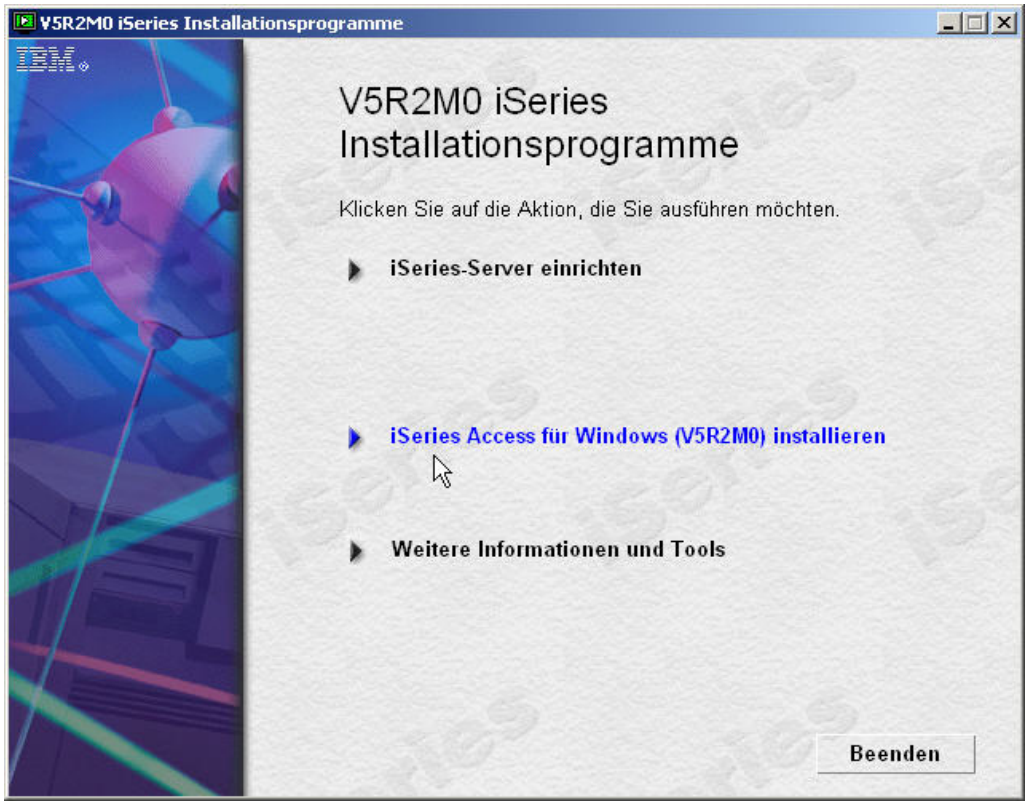
Aktualisierung beendet

Wenn jetzt Arbeitsplätze direkt aus dem System-Installationspfad „/QIBM/ProdData/CA400/Express/Install/Image“ installiert werden, so beinhaltet die Installation automatisch das letzte Service-Pack.

Sollten Sie individuelle Installations-Images in eigenen Verzeichnissen angelegt haben, so lassen sie sich auf dem gleichen Weg installieren.

Upgrade zu iSeries Access für Windows

Um die Migration zu iSeries Access für Windows zu beginnen, lege ich einfach die Installations-CD in das entsprechende Laufwerk. Sofern ein Autostart zugelassen ist, öffnet sich der folgende Dialog:



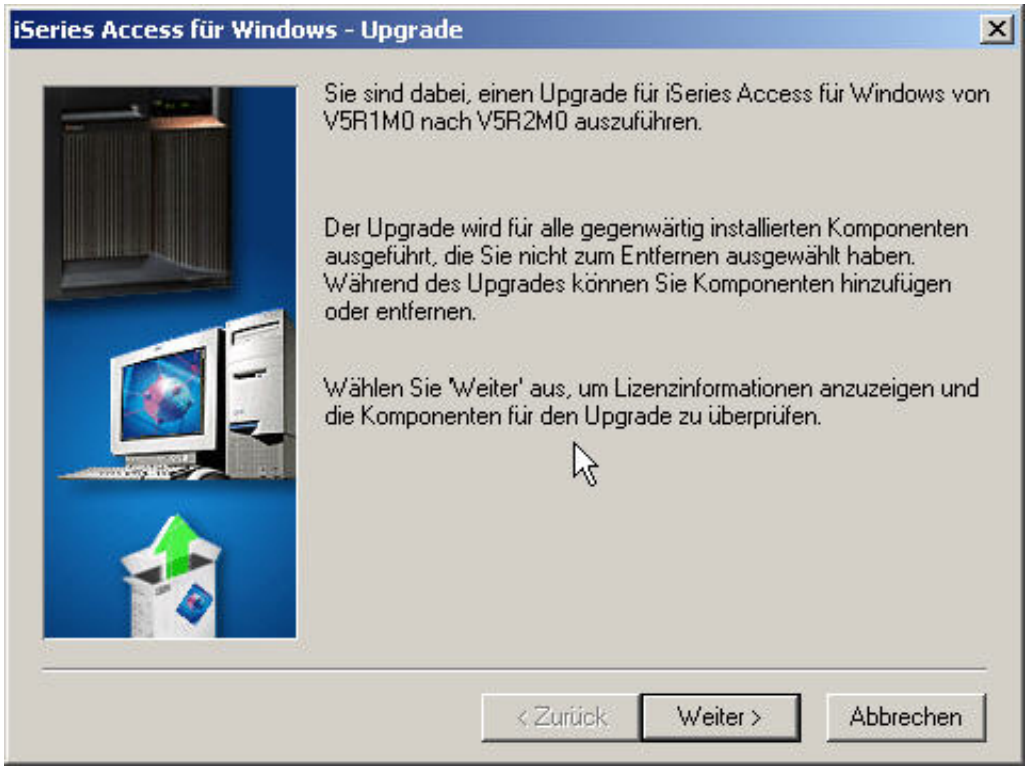
Beginn der Installation/Upgrade

Ich klicke doppelt auf den Menüpunkt „iSeries Access für Windows installieren“ und die Migration kann beginnen. Auf die gleiche Weise wird übrigens die Komplett-Installation auf einem neuen Arbeitsplatz gestartet.

Sollte bei Ihnen der automatische Start nicht klappen, so starten Sie einfach aus dem Pfad „LW:\express“ heraus, der sich auf der Installations-CD des Programms „Setup.exe“ befindet. Danach öffnet sich ebenfalls der oben dargestellte Dialog.

Da sich bereits eine Vorgängerversion des iSeries Access auf meinem Arbeitsplatz befindet, wird dieser Zustand durch den nachfolgenden Dialog angezeigt.

7.2.8
Seite 2

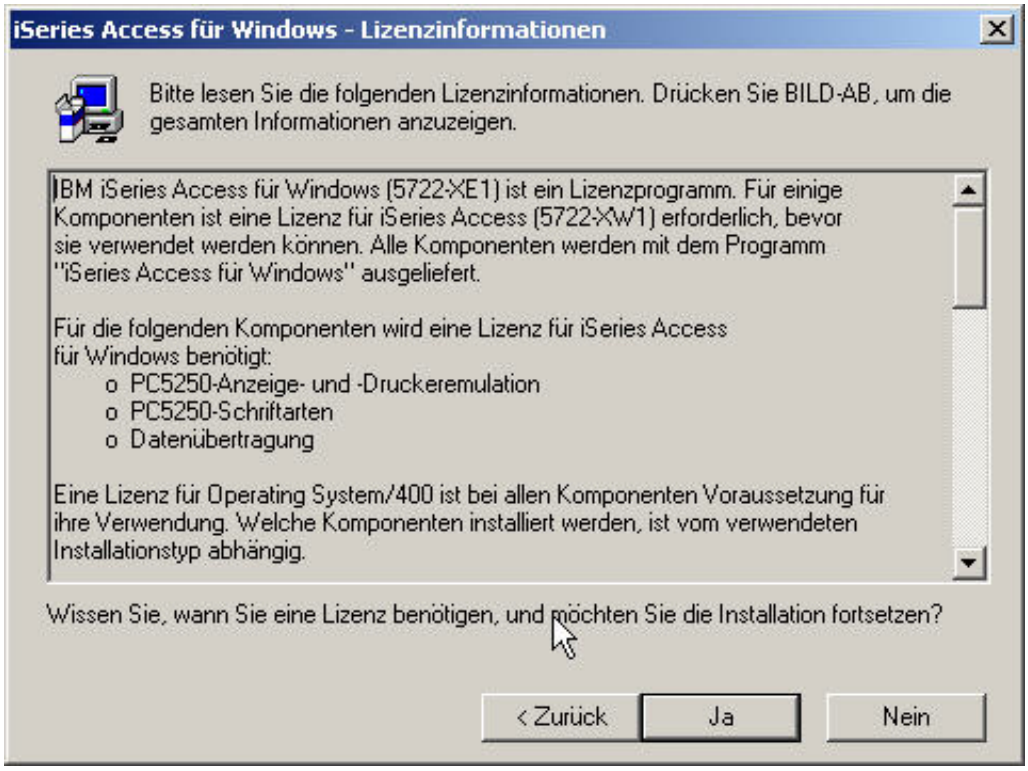


Hinweis auf den Upgrade

Bei diesem Dialog handelt es sich um einen Hinweis darauf, was den Installierenden beim Upgrade erwartet. Sie erfahren hier, von welcher Client Access-Version Ihr Upgrade startet (hier V5R1) und dass Sie bereits installierte Komponenten entfernen oder neue hinzufügen können.

Mit „Weiter“ wird der Dialog beendet.

Es folgt danach die Anzeige der „Lizenzinformationen“, die man von den Installationsgängen des Client Access Express kennt.

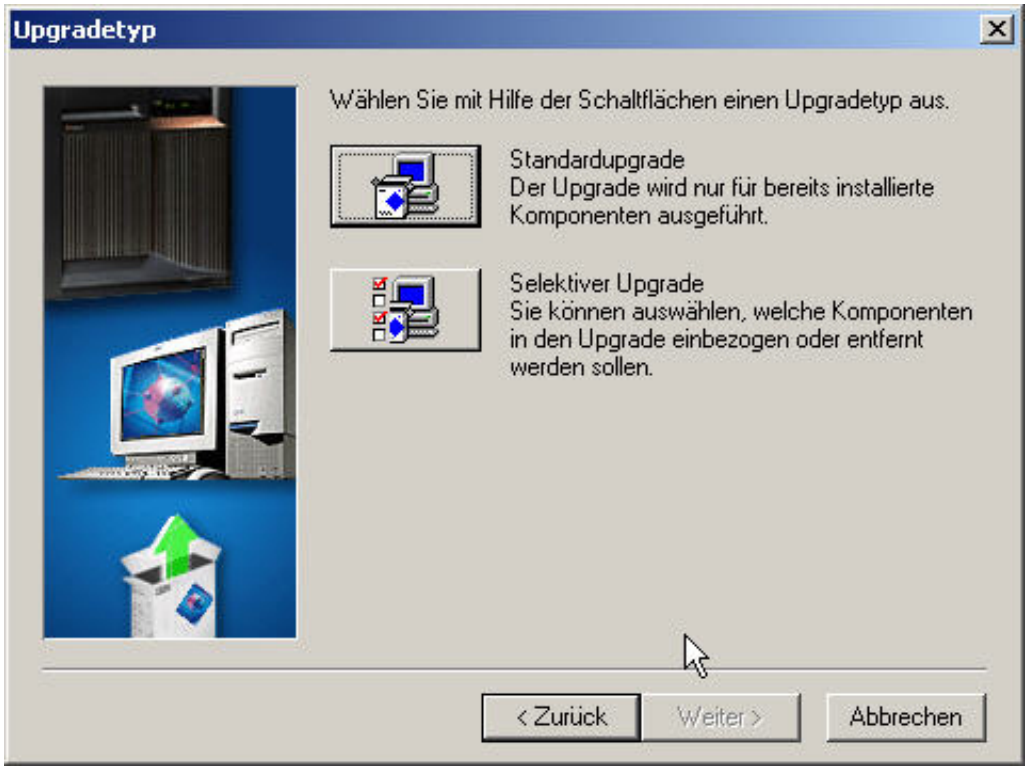


Lizenzinformationen

Diese Lizenzinformationen werden mit „Ja“ bestätigt, so dass der Upgrade ungehindert weitergehen kann.

Jetzt werden wir gefragt, auf welche Weise der Upgrade durchgeführt werden soll.

7.2.8
Seite 4



Installationstyp

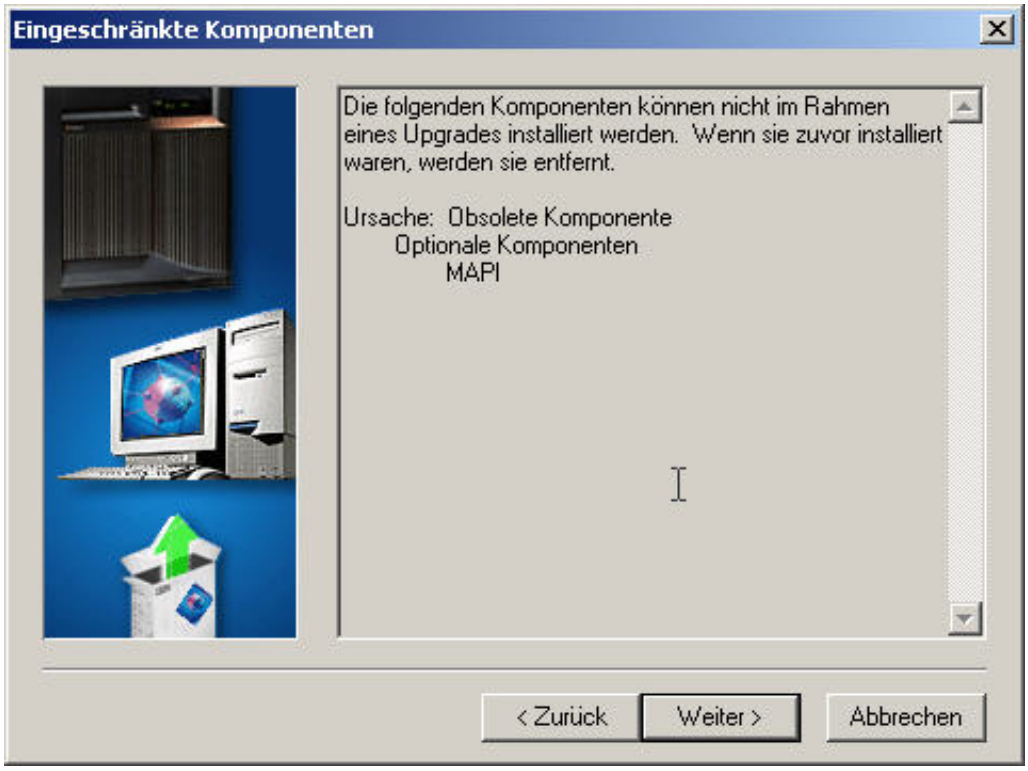
Es gibt zwei Installationstypen:

Standardupgrade: Alle in der alten Version installierten Komponenten des Client Access werden auf den neuen iSeries Access-Versionsstand aktualisiert. Es werden keine Komponenten-Auswahldialoge angezeigt.

Selektiver Upgrade: Eine alte Version wird aktualisiert, jedoch können alte Komponenten entfernt und neue hinzugefügt werden. Ein Komponenten-Auswahldialog wird angezeigt.

Ich wähle „Standardupgrade“.

Daraufhin erscheint der Dialog „Eingeschränkte Komponenten“.



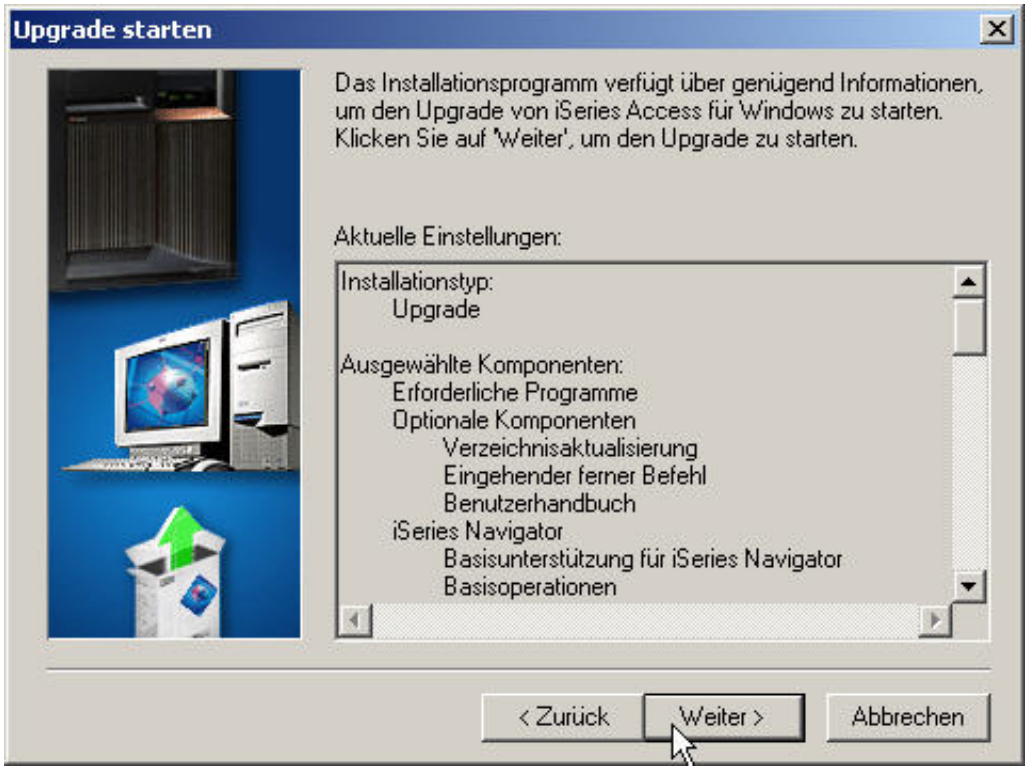
Eingeschränkte Komponenten

In diesem Fenster werden alte Client Access-Komponenten angezeigt, die unter der neuen iSeries Access-Version nicht mehr bestehen und daher aus dem System entfernt werden müssen.

Dieser Dialog wird mit „Weiter“ bestätigt.

Danach beginnt der eigentliche Upgrade-Vorgang mit dem Fenster „Upgrade starten“. In diesem Fenster wird angezeigt, welche Komponenten für den Upgrade des iSeries Access für Windows berücksichtigt werden.

7.2.8
Seite 6



Upgrade starten

Nach dem Upgrade sollten Sie noch das jeweils aktuelle Service Pack für iSeries Access für Windows anlegen. Dann muss Windows beendet und erneut gestartet werden. Anschließend ist der Upgrade formal beendet.

Greifen Sie mit den aktualisierten Funktionen auf alte Konfigurationsdateien zu. Sie werden benachrichtigt, dass die Konfigurationsdateien (Bildschirme, Drucker etc.) auf das neue iSeries Access für Windows-Format umgestellt werden und die Konfigurationsdateien nicht mehr in alte Client Access-Arbeitsplätze exportiert werden können.

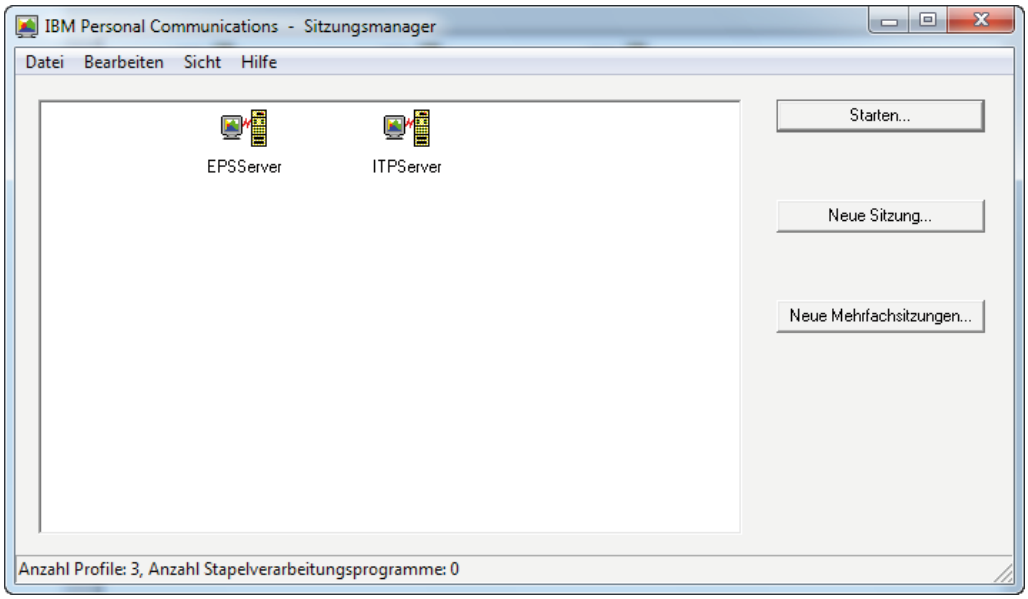
i Access für Windows – Dienste

27. Ergänzung 6/2011

7.3
Seite 2

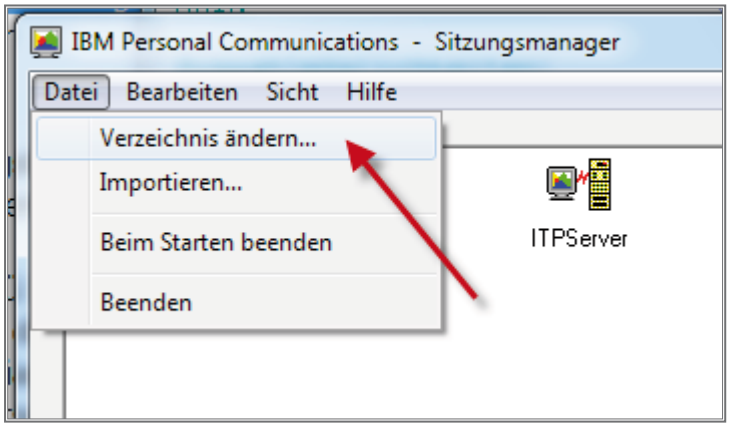
PC5250-Bildschirmsitzung konfigurieren

Um eine Bildschirmsitzung zu konfigurieren, klicken Sie in der Programmgruppe „IBM System i Access für Windows ➜ Emulation“ doppelt auf „Sitzung starten oder konfigurieren“ und es öffnet sich der „Sitzungsmanager“.



Der „Sitzungsmanager“

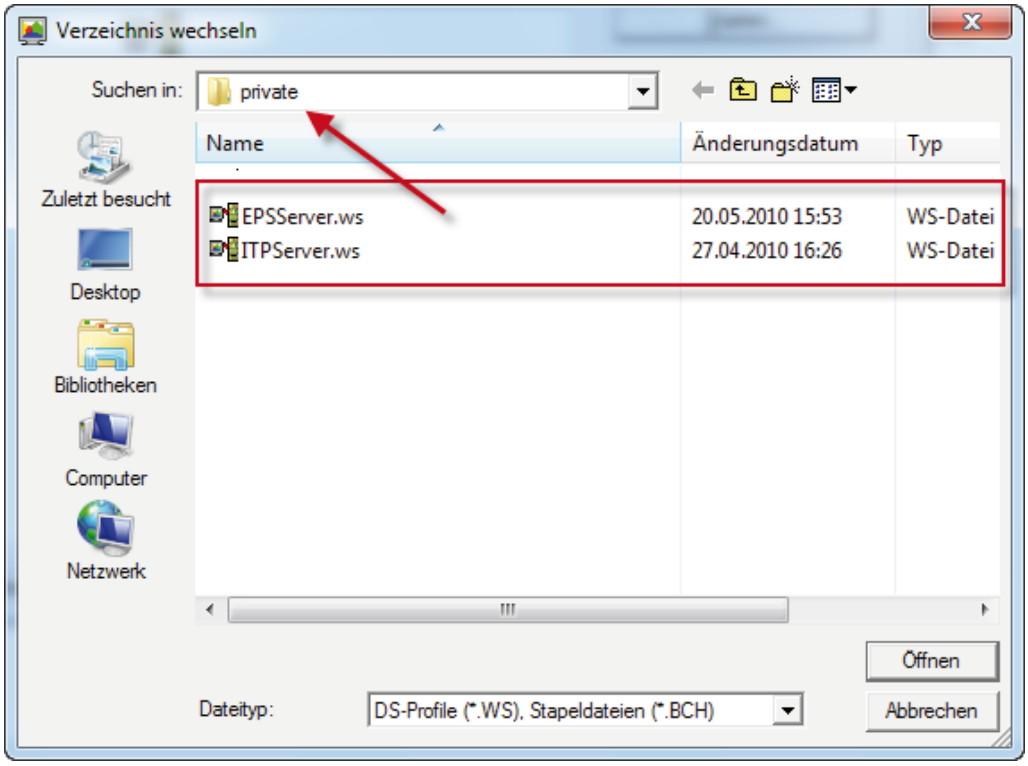
Mit dem „Sitzungsmanager“ kann ein Administrator Sitzungsprofile verwalten, die lokal oder im Netzwerk abgelegt sind.



Verzeichnis der gespeicherten Profile ändern

Öffnen Sie das Menü „Datei ➜ Verzeichnis ändern“, so finden Sie die im bezeichneten Pfad gefunden Sitzungsprofile in der Zentralanzeige des Programms wieder.

7.3.1
Seite 2

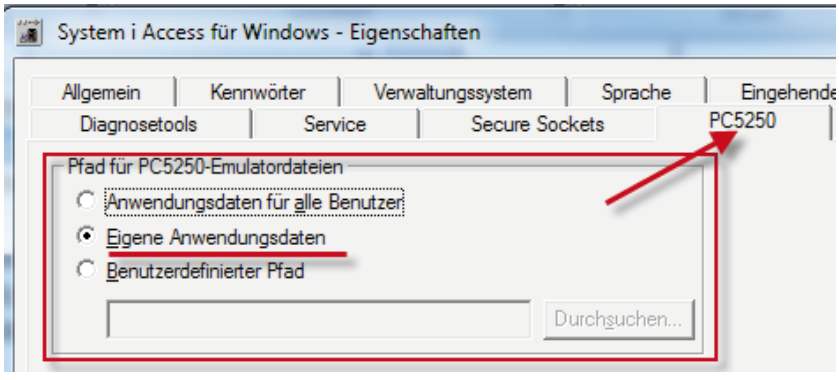


Verzeichnis der gespeicherten Profile

Seit System i Access V6R1 befinden sich alle persönlichen Objekte der eigenen Emulation im Verzeichnis des Benutzers, der die Bildschirmemulation konfiguriert hat.

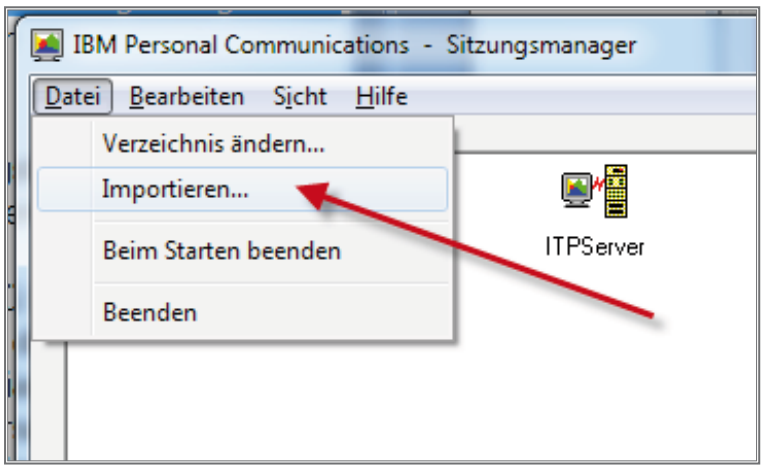
Hier: Luttkus\AppData\Roaming\IBM\Client Access\Emulator\private

Wo die Bildschirmprofile jedoch im Einzelfall abgespeichert werden, entscheidet sich in den Client Access-Eigenschaften (Ordner „System I Access für Windows ➜ Eigenschaften“ im Register „PC5250“.



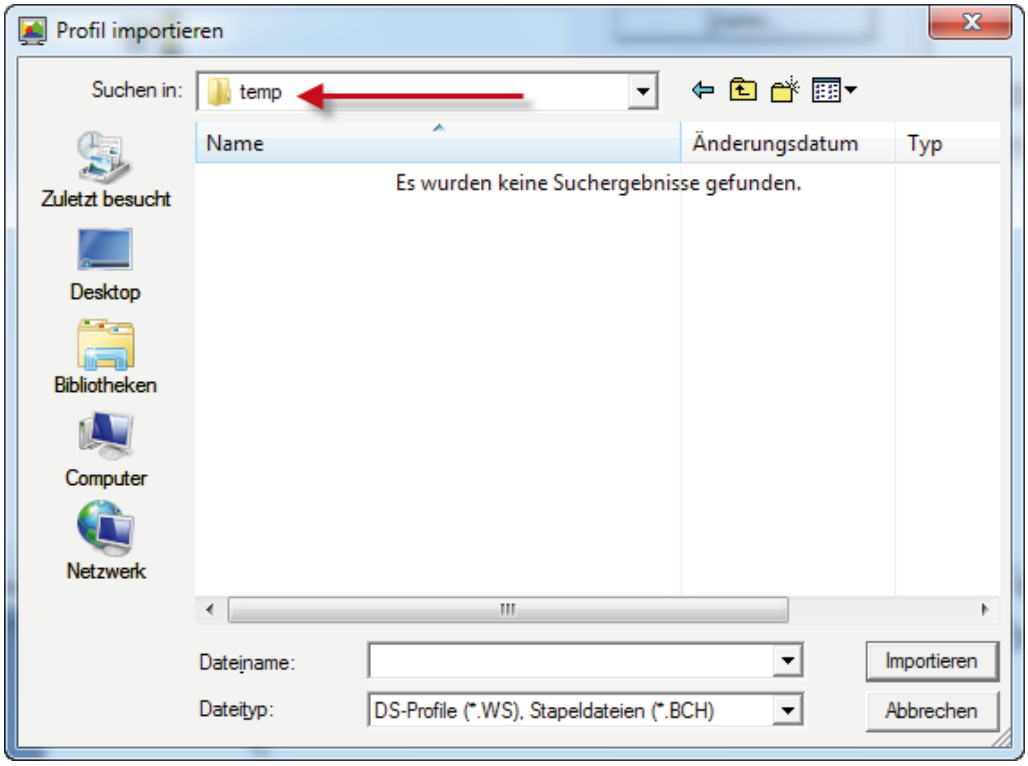
PC5250-Eigenschaften

Hier sehen Sie die Standardeigenschaften.



Bildschirmobjekte importieren

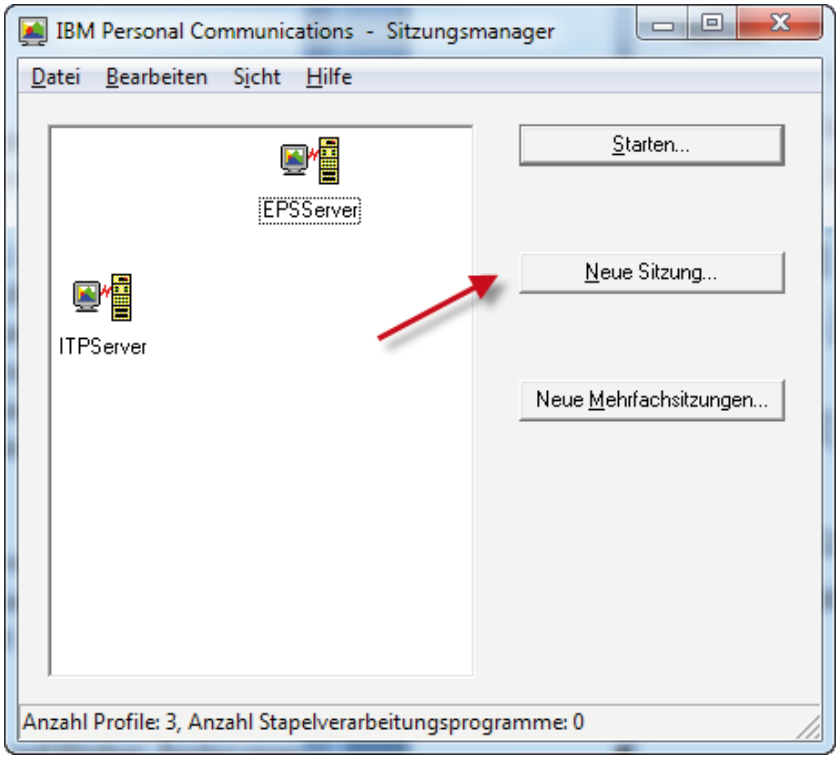
Öffnen Sie das Menü „Datei ➜ Importieren“, so übertragen Sie bestehende Sitzungsprofile und die davon abhängigen Dateien in den gesetzten Zielpfad (siehe Pfeil in der Abb. unten).



Importpfad setzen

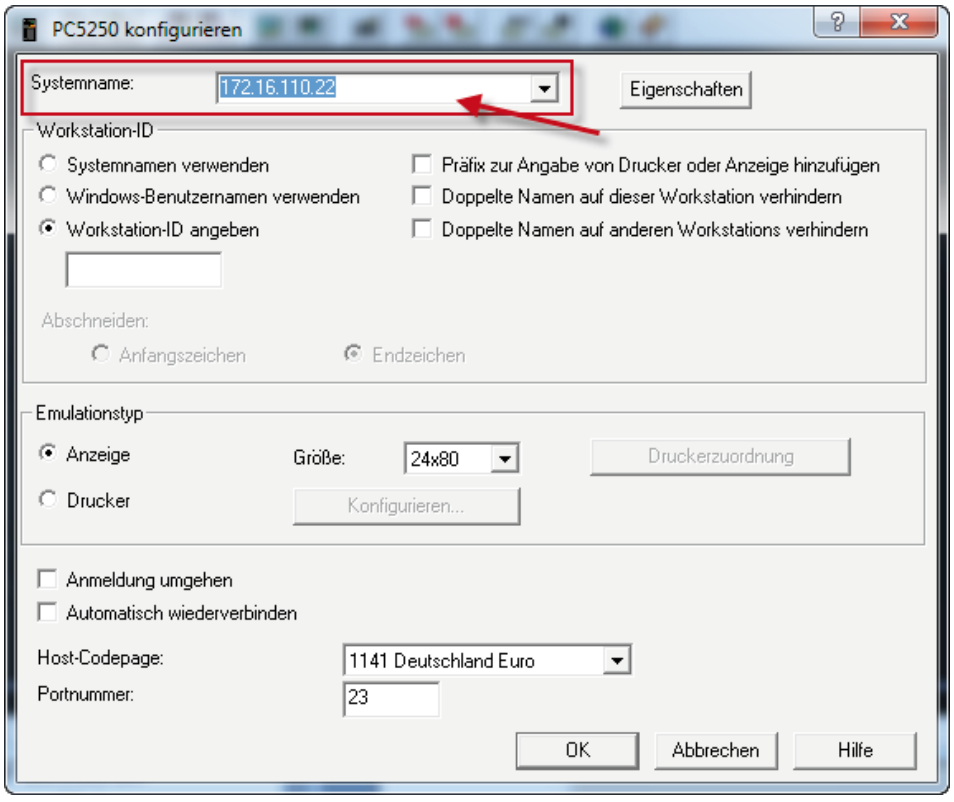
Um jetzt mit der Konfiguration einer Bildschirmemulation zu beginnen, klicken wir auf den Button „Neue Sitzung…“

7.3.1
Seite 4



Beginn der Bildschirmkonfiguration

Es öffnet sich folgendes Fenster:

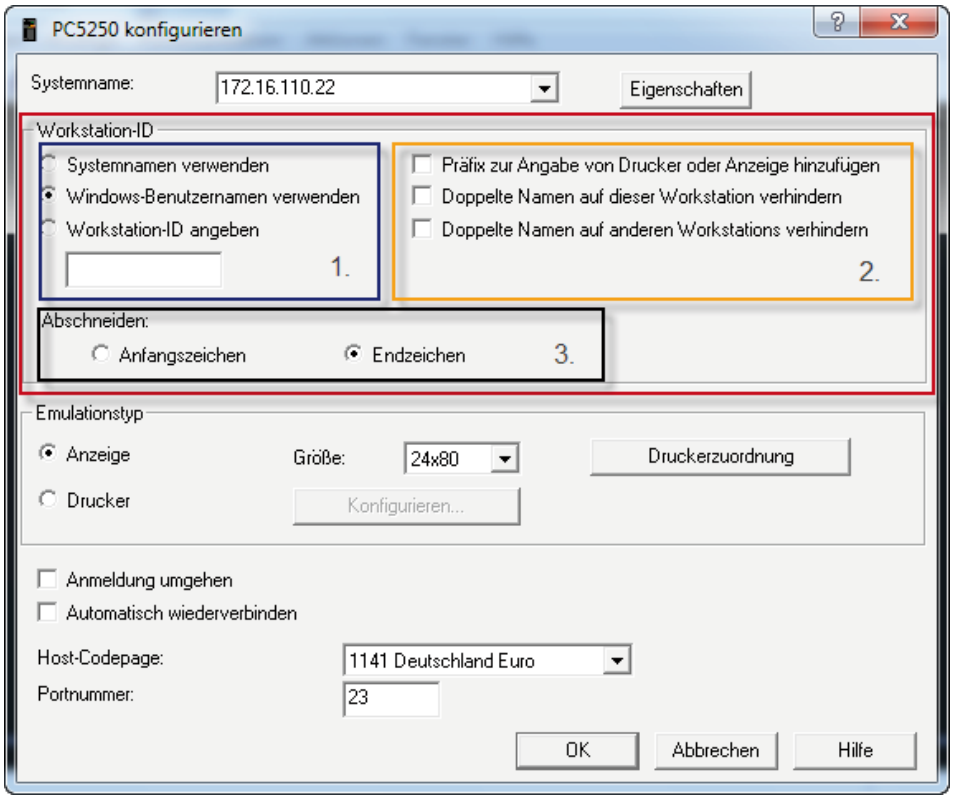


Bildschirmsitzung konfigurieren 1

Im Dialog für die Verbindungskonfiguration geben Sie u. a. die Verbindungsdaten zum System i an.

Parameter Systemname
Tragen Sie hier den Hostnamen Ihres System i ein, so wie er in der Host-Tabelle Ihres Windows-PCs oder im Domain Name Service hinterlegt wurde. Es ist durchaus möglich, hier die numerische IP-Adresse Ihres System i anzugeben.

Parameter Workstation-ID



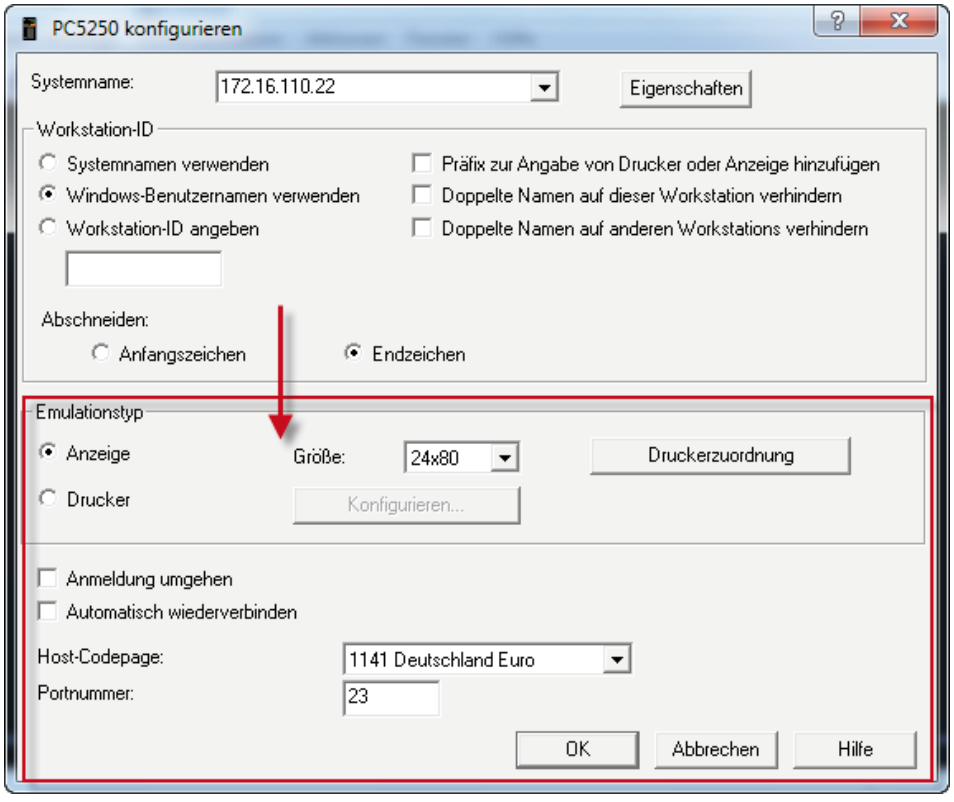
Bildschirmsitzung konfigurieren 2

1. Markieren Sie den Parameter „Systemnamen verwenden“, so wird der Computername eines Windows-Netzwerk-Clients als Datenstations-ID-Namen an das System i übergeben und ein entsprechender Bildschirm wird generiert. Markieren Sie hingegen den Parameter „Windows-Benutzernamen verwenden“, so wird der Benutzername, mit dem Sie sich an Ihrem Windows PC angemeldet haben, als Bildschirmeinheitenname verwendet. Verwenden Sie „Workstation-ID“ angeben, so können Sie Ihrem Bildschirm gezielt einen Namen geben bzw. indem Sie gar nichts angeben, veranlassen, dass ein Bildschirmname frei generiert wird. In diesem Fall wird das System i Bildschirmnamen verwenden, die mit QPADEV beginnen.

2. Dieser Parameter bewirkt a) dass jeder Sitzung ein Buchstabe vorangestellt wird, der kennzeichnet, ob es sich bei der Emulation um einen Drucker (P) bzw. einen Bildschirm (S) handelt, b) dass Namensduplikate auf Ihrem PC vermieden werden und c) dass Namensduplikate bei Bildschirmeinheiten auf dem System i vermieden werden.
3. Sind die Windows-Namen länger als für System i-Objekte erlaubt, so können Sie mit Hilfe des Parameters „Abschneiden" definieren, ob sie die ersten oder die letzten zehn Zeichen eines Windows-Namens beibehalten wollen.

Da Windows-Benutzernamen und Systemnamen eindeutig sind, jedoch von jedem Windows-PC mehr als eine Emulationsverbindung zu einer System i aufgebaut werden kann, muss die Eindeutigkeit eines Einheitennamens erzwungen werden. Zu diesem Zweck können optional die Parameter „Doppelte Namen auf dieser Datenstation verhindern" und „Doppelte Namen auf anderen Datenstationen verhindern" markiert werden. Durch diese Parameter wird sichergestellt, dass Einheitennamen bezogen auf eine Arbeitsstation bzw. bezogen auf ein System IBM i eindeutig erzeugt werden. Die Eindeutigkeit des Namens wird dadurch erzwungen, dass an die Windows-Namen Suffixe (1, 2....) angehängt werden.

Weitere Parameter



Bildschirmsitzung konfigurieren 3

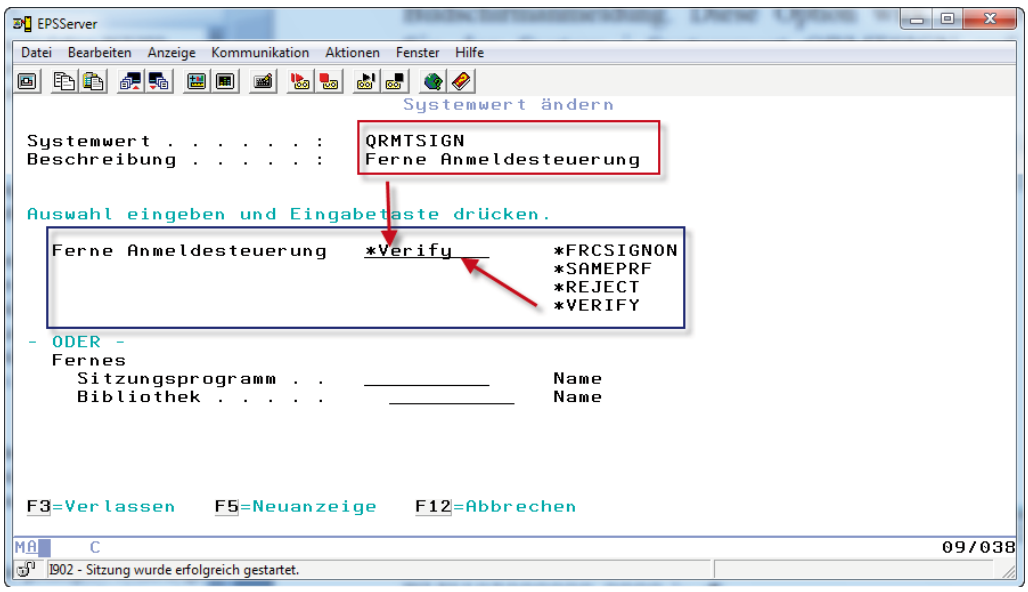
Anzeige der Größe
Dieser Parameter enthält als Standardwert 24x80. Um automatisch in das größere Format umschalten zu können, sollten Sie hier den Wert 132x27 verwenden.

Zeichenumsetzungstabelle Host
Dieser Wert steuert die Zeichenumsetzung zur PC-Oberfläche hin. Er ist nicht verändert worden. Die Zeichensatz-ID für Deutsche Zeichen mit dem Euro-Symbol (€) ist 1141.

Automatisch Wiederverbinden
Sollte die TCP/IP-Sitzung unterbrochen werden, so wird die Sitzung erneut aufgenommen, sobald das System i wieder zur Verfügung steht.

Port-Nummer
Dieser Wert beschreibt den TCP/IP–Port, über den der Emulationsbildschirm mit der System i kommuniziert. Ändern Sie diesen Wert nicht, denn dieser Port wird ebenfalls per Default vom System i erwartet. Eine einseitige Änderung würde dazu führen, dass keine Bildschirmsitzung zustande kommt.

Anmeldung umgehen
Markieren Sie diesen Punkt, um die Bildschirmanmeldung zu umgehen. In diesem Fall ist die Verbindungsanmeldung zugleich die Bildschirmanmeldung. Diese Option wirkt nur, sofern Sie den System i-Systemwert QRMTSIGN auf „*VERIFY“ gesetzt haben.



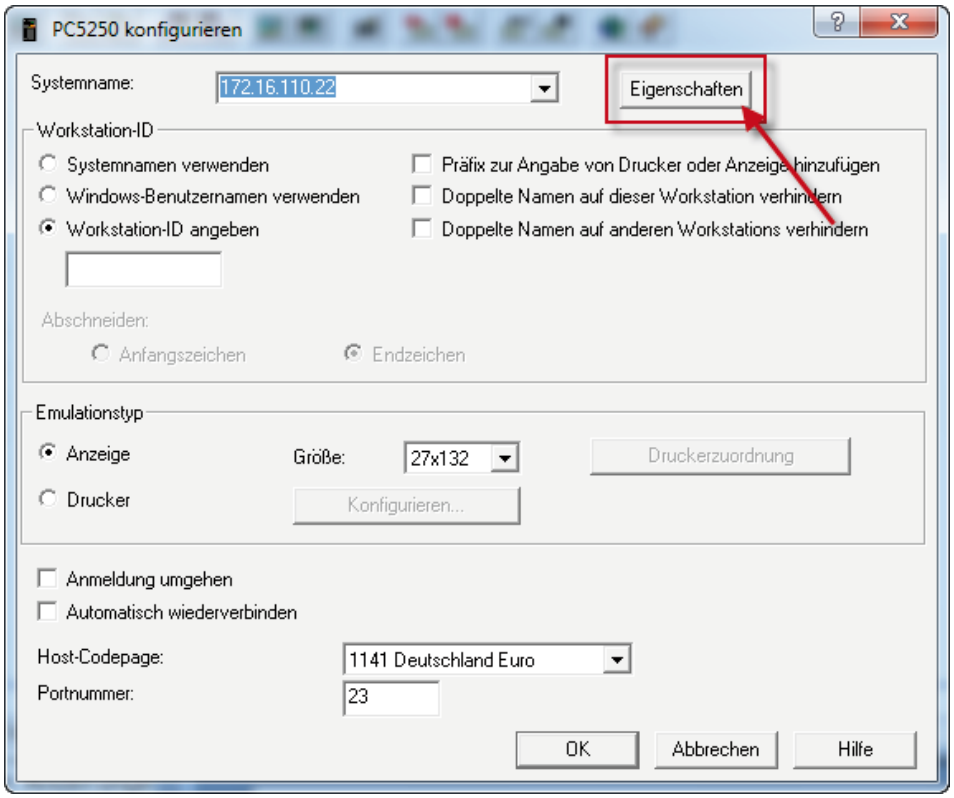
Systemwert QRMTSIGN

7.3.1
Seite 8

Automatisch wiederverbinden

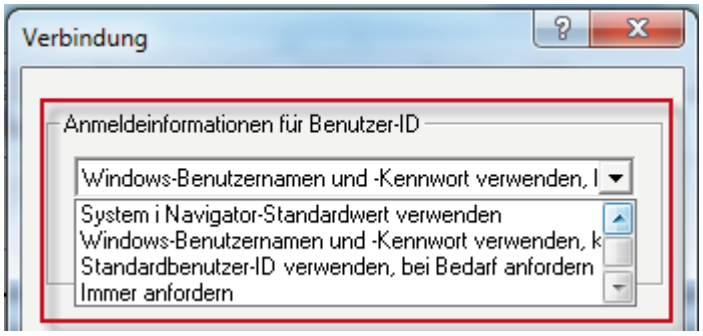
Haben Sie diesen Punkt markiert, so wird für den Fall, dass die Verbindung zum System i unterbrochen wird, die Verbindung automatisch wieder aufgebaut, sobald eine erneute Verbindungsaufnahme zum System i möglich ist.

Definition eines emulierten Druckers



Systemverbindungseigenschaften

Eigenschaften
Mit Hilfe des Knopfes „Eigenschaften“ definieren Sie Sicherheitseinstellungen für die Verbindungsherstellung zwischen Ihrem PC und dem System i.



Anmeldeinformationen

System i Navigator – Standardwert verwenden
Die Anmeldeinformationen zum Zweck der Verbindungsaufnahme sind im System i Navigator hinterlegt. Außerdem wird im System i Navigator automatisch eine Verbindung in der derzeit aktiven Umgebung angelegt.

Windows-Benutzernamen und -Kennwort verwenden
Der Windows-Benutzer, mit dem Sie sich an Ihrem PC angemeldet haben, existiert unter dem gleichen Namen und mit dem gleichen Kennwort im System i. Ist diese Bedingung gegeben, so wird keine Verbindungsanmeldung angezeigt. Dadurch können wir unter Windows endlich die lästigen Mehrfachanmeldungen im Bereich der Bildschirme abstellen. Es muss jedoch garantiert sein, dass ein Windows-Benutzerkonto mit den gleichen Anmeldungsdaten versehen wird wie ein System i-Benutzerprofil.

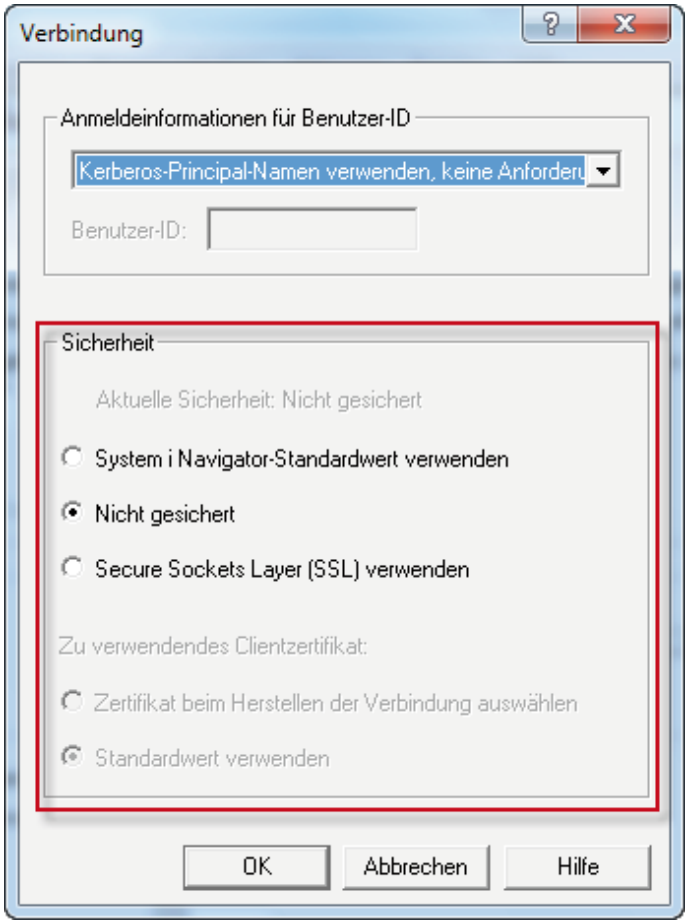
Wichtig! (A) Der Benutzerkontoname in Windows muss absolut identisch sein mit dem Namen des System i-Benutzerprofils. (B) Das Kennwort des Windows-Benutzerkontos muss absolut identisch sein mit dem Kennwort des System i-Benutzerprofils!

Standardbenutzer-ID verwenden…
Sie definieren einen Standardbenutzer für die Verbindungsanmeldung.

Kerberos-Prinzipalnamen verwenden …
Die Verbindungsanmeldung wird durch einen Kerberosserver vorgenommen.

Immer anfordern
Beim ersten Start einer Emulation auf Ihrem PC wird ein Verbindungsanmeldungsdialog angezeigt.

7.3.1
Seite 10



Sicherheitseinstellungen

Sicherheit
Arbeiten Sie mit dem System i Navigator, so wird die Anmelde- und Sicherheitseinstellung für diese Sitzung im System i Navigator hinterlegt. Sollte es im Navigator noch keine definierte Verbindung zu diesem System geben, so wird in der aktiven Umgebung des System i Navigator eine Verbindung mit diesen sicherheitsrelevanten Daten erzeugt.

Achtung!

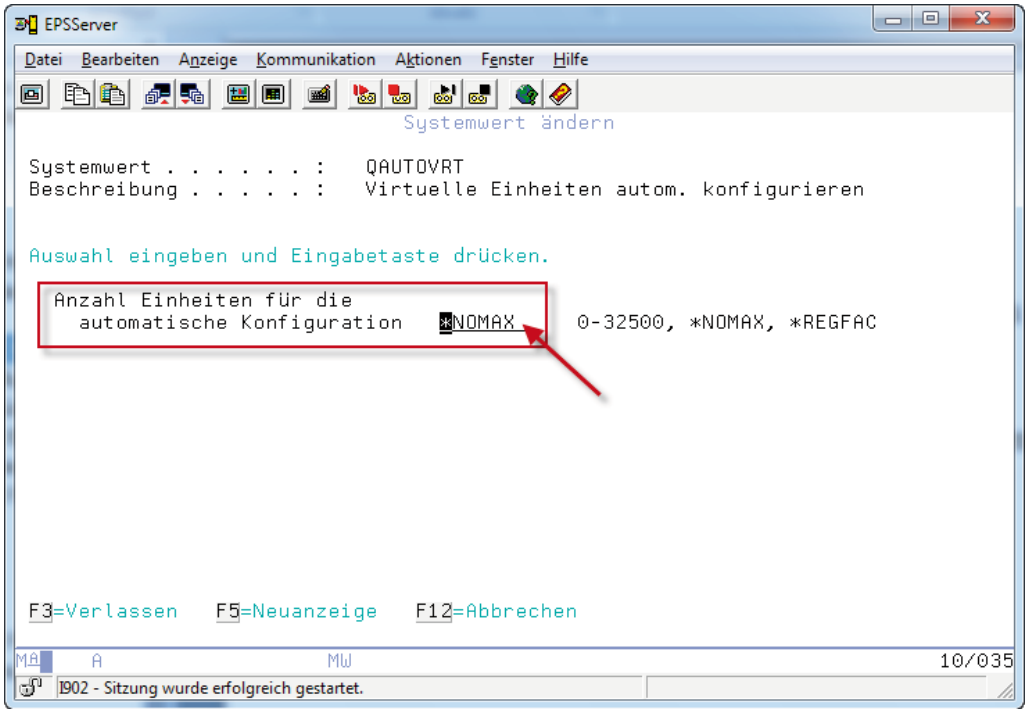
Diese Einstellungen können zu einem gewissen „Wildwuchs“ von Verbindungen im Navigator führen.

Ich empfehle hier die Einstellung „Nicht gesichert“ zu verwenden. Dann gelten die Anmeldedaten lediglich für die gerade konfigurierte Bildschirmsitzung und es werden auch keine Verbindungen im System i Navigator erzeugt.

Arbeiten Sie unter TCP/IP mit sogenannten Secure Socket Layern (SSL), einem kryptographischen Verschlüsselungsalgorithmus, so können Sie dieses Merkmal für die Bildschirmkommunikation anfordern.

Der Systemwert QAUTOVRT

Damit die Verbindungsaufnahme zwischen Emulation und System i gelingen kann, muss der Systemwert QAUTOVRT richtig gesetzt werden.



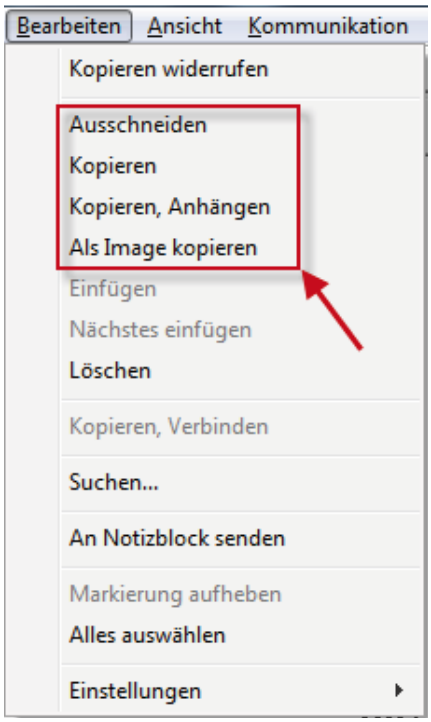
WRKSYSVAL QAUTOVRT

Wichtig ist, dass dieser Systemwert nicht auf 0 gesetzt wird. Stünde der Wert auf 0, so würde dadurch verhindert, dass das i/OS auf Anforderung automatisch Bildschirmsitzungen erzeugen könnte.

7.3.1
Seite 12

Das Menü „Bearbeiten“

Das Menü „Bearbeiten“ enthält mittlerweile eine Reihe interessanter Funktionen, um mit am Bildschirm angezeigten Daten zu arbeiten. Hierbei handelt es sich vornehmlich um Funktionen, die bei der Dokumentation von Bildschirmdialogen und deren Inhalt nützlich sind.



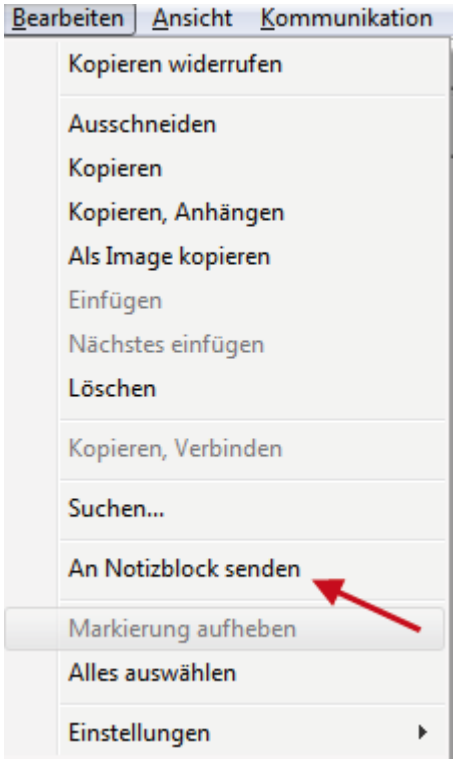
Menü „Bearbeiten“

Wenden wir uns zunächst dem Komplex „Kopieren von Bildschirmdaten ins Windows Clipboard“ zu.

Die Bearbeitungsfunktionen „Kopieren“, „Kopieren, Anhängen“ und „Ausschneiden“ sprechen für sich selbst. Diese Funktionen haben jedoch die Eigenschaft, die vom Bildschirm kopierten Daten in ein reines Textformat zu konvertieren, so dass diese Texte in der Folge in einem Textbearbeitungsprogramm weiterver- oder bearbeitet werden können.

Ist eine Weiterbearbeitung der Bildschirmdaten nicht gewünscht, sondern sollen die Bildschirmdaten einen Bildschirmverarbeitungsschritt dokumentieren, so wählen Sie die Funktion „Als Image kopieren“. Jetzt kann der „abfotografierte“ Bildschirm als Image in einen Dokumentationstext eingefügt werden.

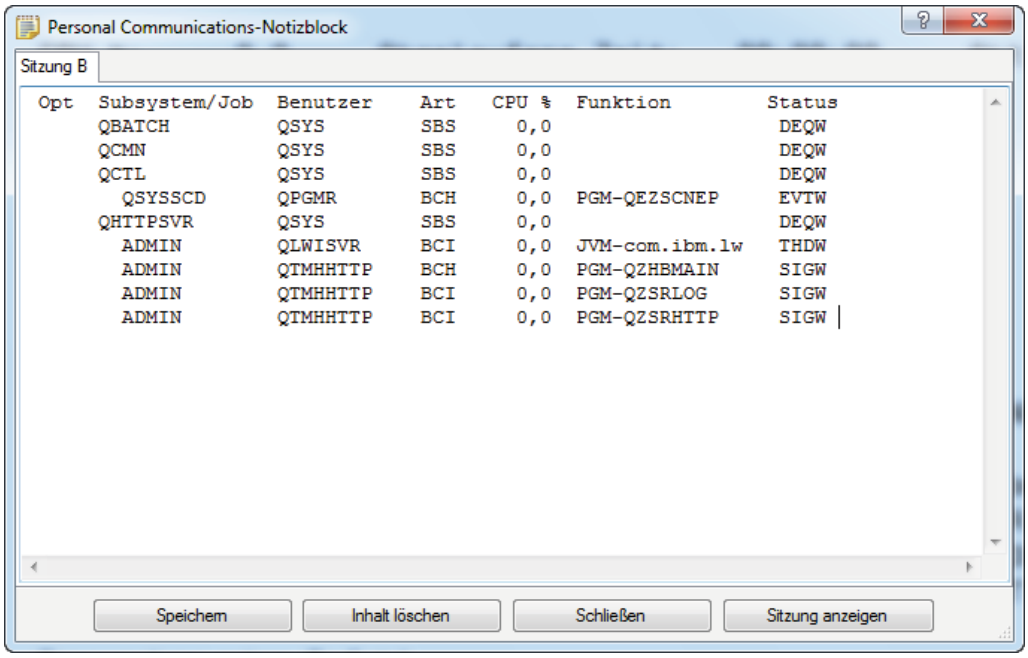
7.3.2 Seite 2



Der Notizblock

Sollen Bildschirminhalte sofort als Text bearbeitet werden können, bevor Sie in ein weiterverarbeitendes Programm übernommen werden, bietet sich die Funktion „An Notizblock senden“ an.

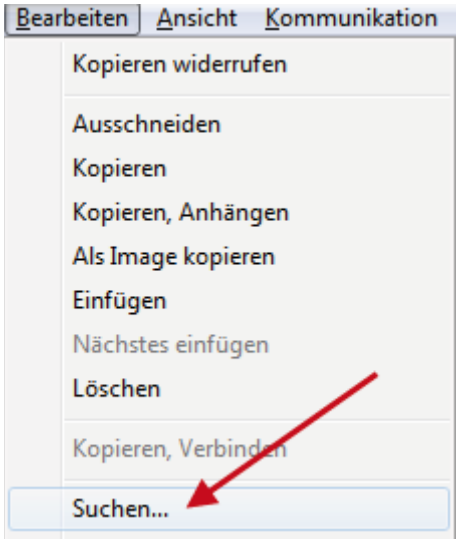
Ich habe Bildschirminhalt markiert und wähle „Bearbeiten ➜ An Notizblock senden“.



Inhalt des Notizblocks

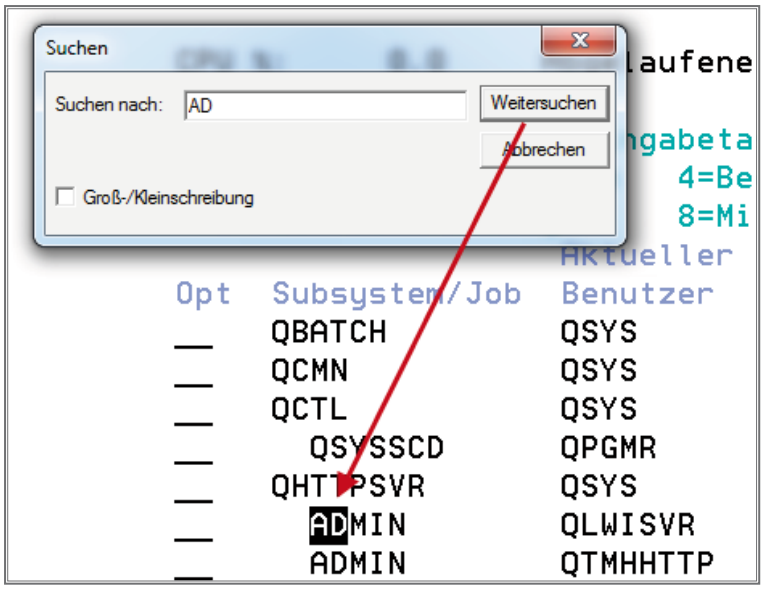
Der Notizblock ist ein einfacher Texteditor, der grundlegende Bearbeitungsfunktionen wie Ausschneiden, Kopieren und Einfügen bereitstellt.

– Speichern: Speichert den Inhalt des Notizblocks.
– Inhalt löschen: Löscht den Inhalt des Notizblocks.
– Schließen: Beendet die aktuelle Notizblocksitzung.
– Terminalsitzung anzeigen: Zeigt die zugeordnete Terminalsitzung an.



Suchen…

Mit der Funktion „Suchen…“ können Sie nach angezeigten Zeichenfolgen auf dem Bildschirm suchen.



… und finden

7.3.2
Seite 4

Ich habe nach der Zeichenfolge „Ad“ im angezeigten Bildschirm gesucht. Jetzt wird der Fundort markiert.

Kleiner Wermutstropfen: Die Suchen-Funktion durchsucht nur die angezeigten Daten. Daten, die in einem virtuellen Bereich des Bildschirms bereits vorhanden sind, werden nicht angezeigt.

Die Einstellungen

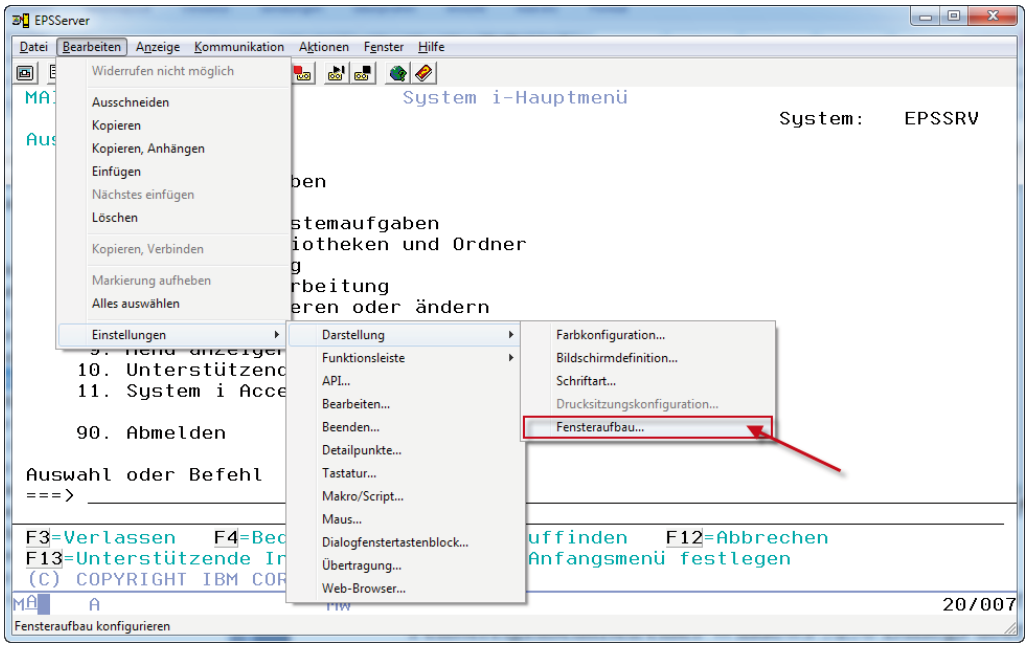
Im Menü „Einstellungen“ wird das Verhalten der Bildschirmemulation eingestellt, so dass sich jeder Benutzer „seinen“ Bildschirm auf „seine“ Arbeitsweise einstellen kann.

7.3.3 Seite 2

Das Menü „Einstellungen ➜ Darstellung“

Mittlerweile sind alle Layout-Konfigurationen in das Menü „Bearbeiten“ aufgenommen worden, so dass es nicht mehr wie in früheren Versionen notwendig ist, die Layout-Funktionen über mehrere Menüs sozusagen „zusammenzusammeln“.

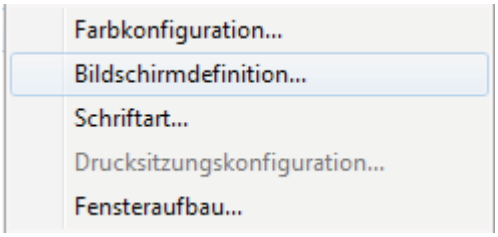
Ich öffne in meinem neuen Bildschirm das Menü „Bearbeiten ➜ Einstellungen ➜ Darstellung“.



PC5250 Einstellungen

Hier sehen Sie den Zugriff auf die Bearbeitung der Fenstereigenschaften eines Windows 5250-Dialogs über die Menüs „Bearbeiten ➜ Einstellungen ➜ Darstellung ➜ Fensteraufbau“.

In diesem Menü wird die Bildschirmoberfläche des Emulationsclients definiert. Die in der folgenden Abbildung zu sehenden Menüpunkte besprechen wir:



Die Menüoptionen des Menüs „Darstellung“

7.3.3.1

Seite 2

Die Menüauswahl „Bildschirmdefinition“

Mit Hilfe dieser Option sind Sie in der Lage, folgende Bildschirmbestandteile zu definieren:

– Die Cursorform

– Die Mauszeigerform

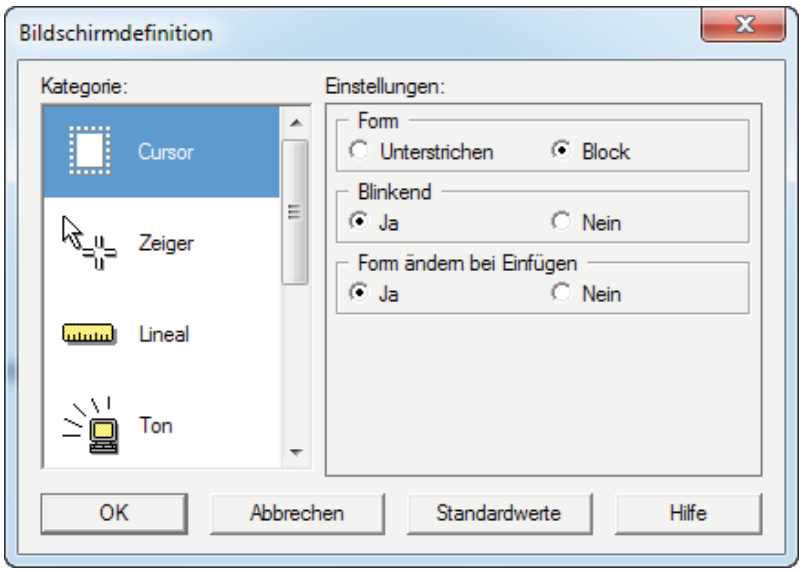
– Das Bildschirmlineal

– Ton: Akustische Signale bei Bildschirmereignissen

– Anzeigemodus: Definition der 5250-Statuszeile

– Palette: Farbschemata im 5250-Bildschirm

Da diese Konfigurationsmenüs für sich sprechen, beschränke ich mich darauf, die Cursordefinitionsfunktion zu beschreiben.



Den System i-Cursor definieren

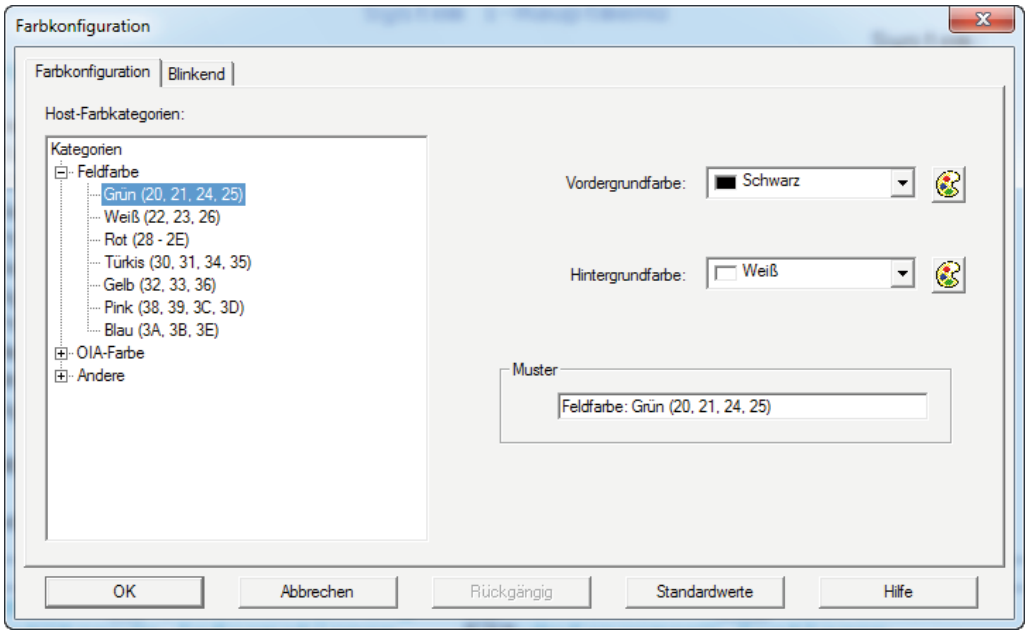
Traditionell sieht man auf Twinax-Bildschirmen sehr oft einen blinkenden Block-Cursor. Die Default-Einstellung des PC5250-Clients definiert jedoch einen Cursor, der als Unterstrich dargestellt wird. Um jetzt einen Blockcursor zu definieren, erledigen Sie folgende Arbeitsschritte:

1. Im Bereich „Einstellungen“ markieren Sie den Radio-Button „Block“.
2. Bei „Blinkend“ wählen Sie den Radio-Button „Ja“.
3. Soll sich die Cursorform zu einem Halbblock ändern, wenn Sie die „Einfügen“-Taste drücken, so setzen Sie den Parameter „Form ändern bei Einfügen“ auf „Ja“.

7.3.3.2
Seite 2

Die Menüauswahl „Farbkonfiguration“

Soll die Standardfarbgebung des Emulationsbildschirms geändert werden, so gehen Sie in diese Menüauswahl.



Emulationsfarben ändern

Grundsätzlich funktioniert dieses Fenster so, dass für jede Farbe, die auf einem Standard Twinax-Bildschirm dargestellt wird, eine Ersatzfarbe für die Emulationsanzeige ausgewählt werden muss.

Die Feldfarbe
Hiermit werden alle Ein- und Ausgabefelder eines Bildschirms beschrieben sowie der generelle Hintergrundbereich.

Die OIA-Farbe
Hiermit wird die Darstellung des Statuszeilenbereichs des Bildschirms beschrieben – die letzte Zeile des Bildschirms.

Farben anderer Bereiche
Die übrigen Bereiche des Bildschirms. Hierunter fällt die Darstellung der Spaltentrennzeichen, der Blinkattribute etc.

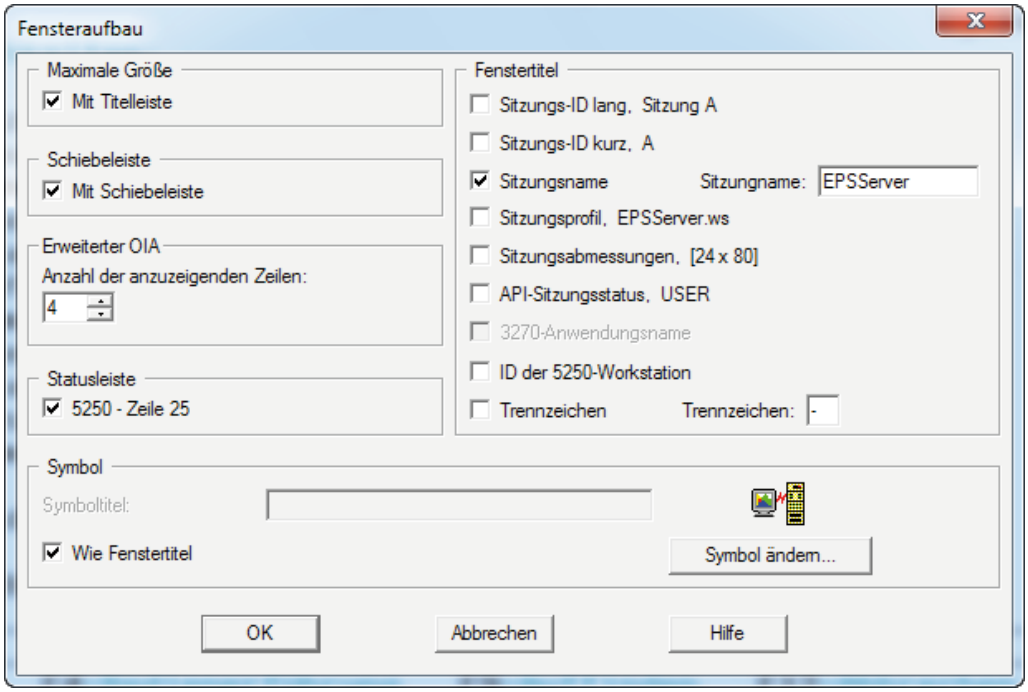
Wenn Sie einen positiven Monochrombildschirm definieren wollen, also einen Bildschirm, bei dem der Hintergrund generell weiß dargestellt wird und alle Zeichen schwarz, so setzen Sie für jede Farbe unter „Feldfarbe“ die „Hintergrundfarbe“ auf „Weiß“ und die „Vordergrundfarbe“ auf „Schwarz“.

7.3.3.3
Seite 2

Analog gehen Sie in den Kategorien „OIA-Farbe“ und „Andere Farben“ vor.

Alternativ können Sie im geöffneten Emulationsbildschirm auf den Bereich, den Sie einfärben möchten, einen Mausklick durchführen und dann in der Farbenleiste die entsprechende Farbkombination für Vordergrund und Hintergrund auswählen.

Die Menüauswahl „Fensterkonfiguration“



Das Fenster-Layout des Emulationsbildschirms einstellen

Parameter
Maximale Größe: Markieren Sie den Wert „Ohne Titelleiste“ und expandieren Sie das Emulationsfenster, so wird die Titelleiste nicht mehr angezeigt.

Schiebeleiste: Markieren Sie den Wert „Ohne Schiebeleiste“, so werden die für ein Windows-Fenster typischen Schiebeleisten nicht mehr angezeigt.

Markieren Sie einmal beide Werte zusammen in der beschriebenen Art und expandieren Sie das Fenster auf maximale Größe. Das Ergebnis ist beeindruckend! Das Emulationsfenster enthält keine der für Windows typischen Elemente. Sie haben den Eindruck, als würden Sie vor einem ganz alten Twinax-Terminal sitzen.

Statusleiste: System- und Programm-Nachrichten eines 5250-Bildschirms werden in dessen Zeile 25 angezeigt. Sollen diese Meldungen jedoch in der Statuszeile des Fensters angezeigt werden, in dem der Bildschirm angezeigt wird, so aktivieren Sie diesen Parameter.

Fenstertitel: In diesem Bereich beschreiben Sie das Aussehen der für Windows-Fenster typischen Kopfzeile. Standardeinstellung für diesen Bereich ist, dass der Wert „Sitzung A -[24x80]“ ausgegeben wird. Wollen Sie diesen Wert überschreiben, so demarkieren Sie den Parameter „Sitzungs-ID lang“ und markieren stattdessen den Parameter „Sitzungsname“. Dann tragen Sie in das dahinterliegende Feld einen eigenen Namen ein. Alle weiteren Parameter, die markiert sind, entwerten Sie.

7.3.3.4
Seite 2

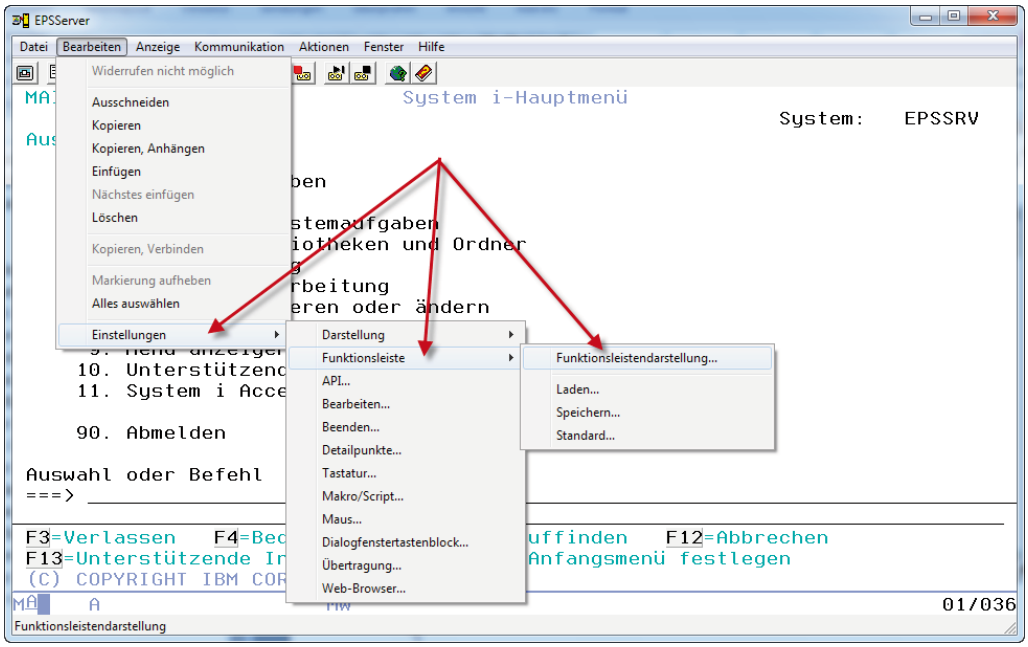
Symbol: In diesem Bereich wird Ihnen ein Standard-Icon für Ihre Bildschirmsitzung vorgeschlagen. Dieses Symbol kann dann in der Folge in einem eigenen oder in einem bereits bestehenden Windows-Ordner oder auf dem Windows-Desktop abgelegt werden. Die Symbol-Unterschrift kann entweder aus dem Fenstertitel übernommen oder selbst vergeben werden.

ID der Workstation: In der Titelleiste wird der Computername Ihres Windows PCs eingeblendet.

Bestätigen Sie alle vorgenommenen Änderungen mit „OK“.

Die Menüauswahl „Funktionsleiste...“

Die Funktionsleiste oder auch Buttonleiste kann in ihrem Layout und ihrer Funktionalität gesteuert werden.



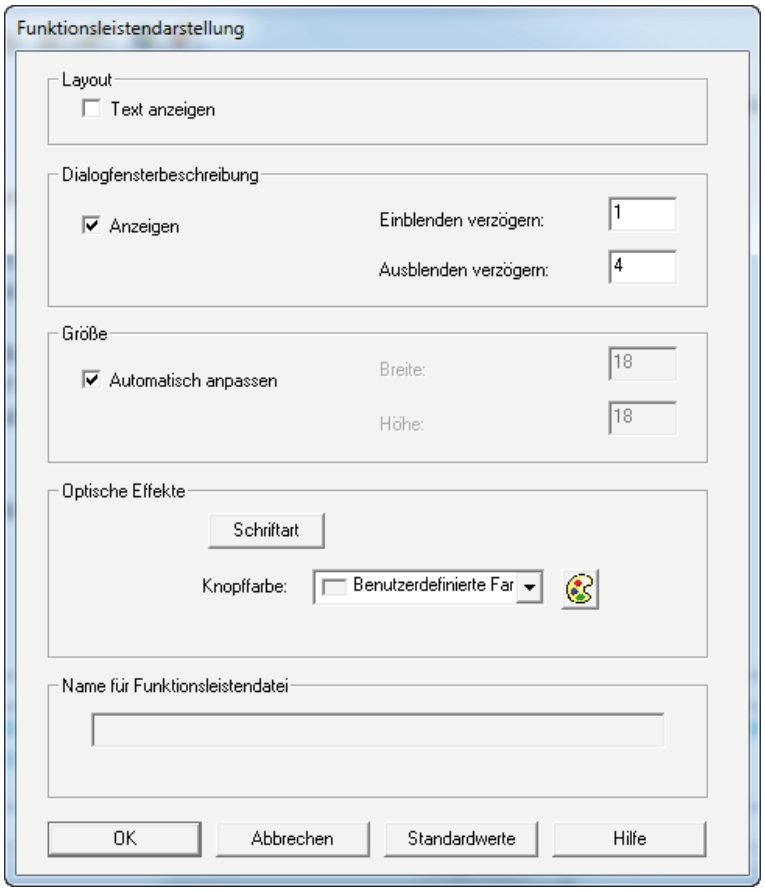
Die Funktionsleiste definieren

Mit Hilfe der Funktionsleiste kann der Anwender auf einfache Art Funktionen aufrufen, die ansonsten in den Menüs des Emulationsfensters „versteckt“ sind. Hier können so nützliche Funktionen wie das Erstellen einer Hardcopy des Bildschirms, das Kopieren der Anzeige ins Windows Clipboard, aber auch so gefährliche Funktionen wie die Umdefinition der Tastatur oder der Aufruf eines File-Transfers aufgerufen werden.

Mit Hilfe des Untermenüs „Funktionsleiste definieren“ können Sie Funktionen in die Buttonleiste aufnehmen bzw. abwählen.

7.3.3.5
Seite 2

Mittels des Untermenüs „Darstellung der Funktionsleiste“ beeinflussen Sie die graphische Darstellung der Buttonleiste.



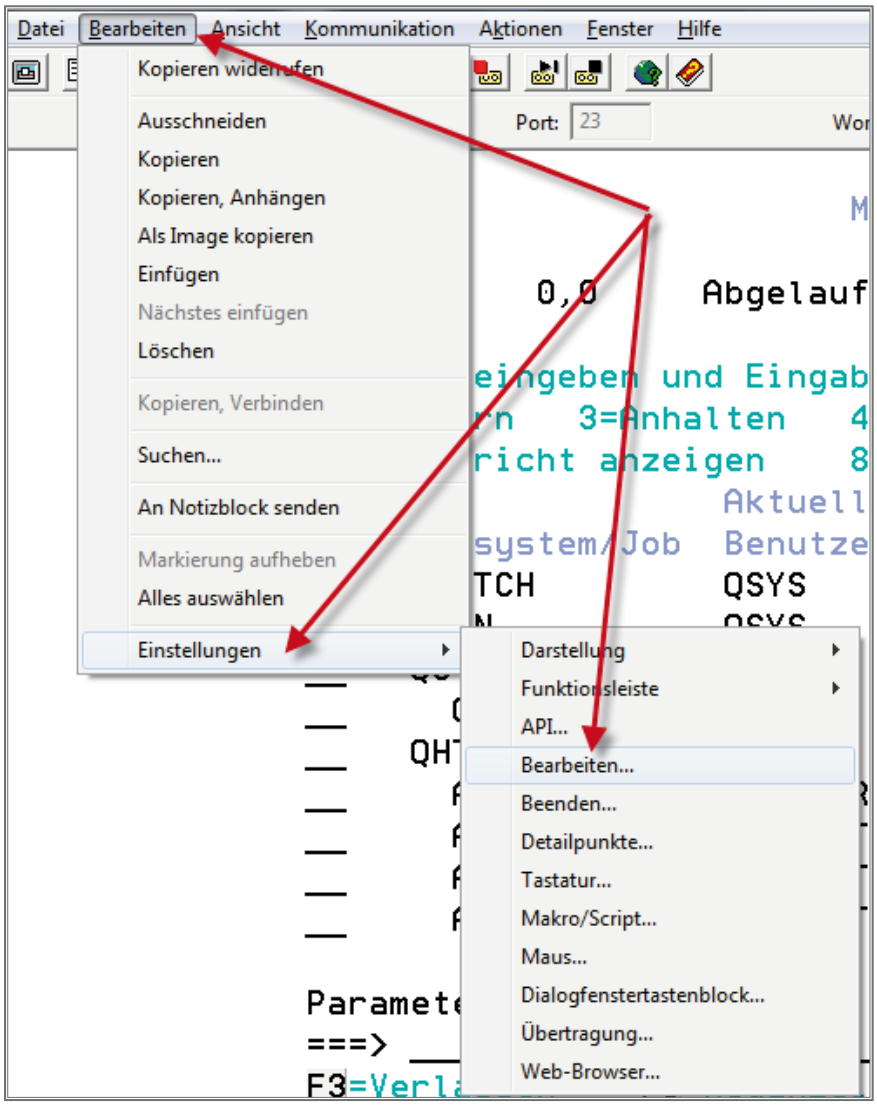
Die Funktionsleiste verkleinern

In der Default-Einstellung nehmen die Funktionsbuttons doch einen sehr großen Raum ein. Um die Buttons kleiner darzustellen, gibt es eine ganz einfache Möglichkeit.

Öffnen Sie einfach die Funktionsleistendarstellung und wählen Sie im Bereich „Layout“ die Funktion „Text anzeigen“ ab. Dadurch wird neben der Symboldarstellung keine Textdarstellung der Funktion realisiert, was zur Folge hat, dass die Funktionsknöpfe kleiner dargestellt werden.

Das Menü „Bearbeiten“

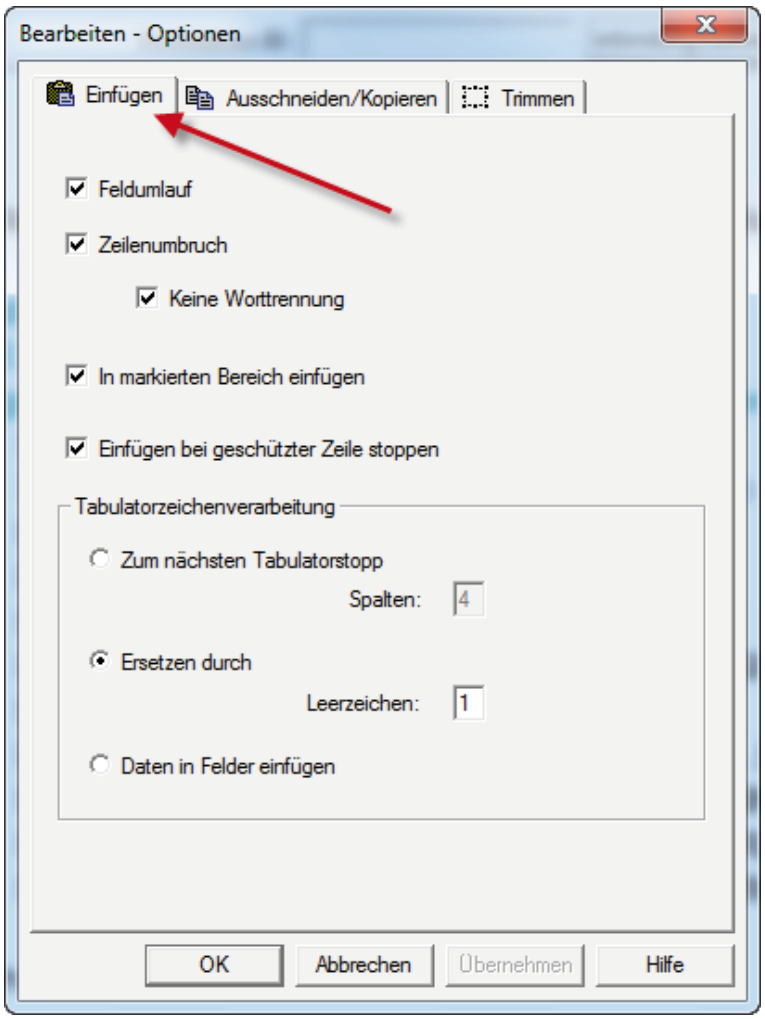
Mit Hilfe des Menüs „Bearbeiten“ steuern Sie das Verhalten der Bildschirmemulation gegenüber dem Windows Clipboard.



Bearbeiten…

7.3.3.6
Seite 2

Haben Sie diese Option ausgewählt, so öffnet sich folgender Dialog:



Einfügen von Daten

Hier kann gesteuert werden, wie Text vor und nach geschützten Feldern eingefügt und wie Text mit Tabulatoren nach dem Einfügen angezeigt wird.

Feldumbruch
Klicken Sie auf Feldumbruch, wenn eingefügte Daten, die auf ein geschütztes Feld treffen, auf das nächste ungeschützte Feld verschoben werden sollen. Wenn Sie dieses Markierungsfeld nicht aktivieren, gehen alle Daten verloren, die auf ein geschütztes Feld treffen.

Zeilenumbruch
Aktivieren Sie dieses Markierungsfeld, wenn kopierter Text in mehrere Zeilen eingefügt werden soll.

Keine Worttrennung
Aktivieren Sie dieses Markierungsfeld, wenn Wörter nicht auf mehrere Felder und Zeilen aufgeteilt werden sollen. Der in Felder eingefügte Text wird an Wortgrenzen getrennt und mit dem nächsten Wort im nächsten Feld fortgesetzt. Wenn ein Wort in ein Feld eingefügt wird, die Länge des Felds jedoch nicht zur Aufnahme des Worts ausreicht, wird ein möglichst großer Teil des Worts in das Feld eingefügt. Der verbleibende Teil wird in das folgende Feld eingefügt.

Hinweis: Die Option für die Worttrennung steht nicht zur Verfügung, wenn die Optionen für den Feldumbruch und den Zeilenumbruch deaktiviert sind.

In markierten Bereich einfügen
Aktivieren Sie dieses Markierungsfeld, damit der Text der Zwischenablage in den markierten Bereich – falls vorhanden – eingefügt wird. Wenn der eingefügte Text aus mehr Zeichen besteht, als der markierte Bereich aufnehmen kann, wird der eingefügte Text am Ende des markierten Bereichs gestoppt.

Ist kein Bereich markiert, erfolgt die Einfügung des Texts an der aktuellen Position.

Einfügen bei geschützter Zeile stoppen
Aktivieren Sie dieses Markierungsfeld, wenn die Einfügung von Text an einer geschützten Zeile der Emulatoranzeige stoppen soll. Wenn Sie dieses Feld nicht aktivieren, wird die Einfügung fortgesetzt.

Tabulatorzeichenverarbeitung
Zum nächsten Tabulatorstopp: Es kann ausgewählt werden, dass Text, der Tabulatoren enthält, an angegebenen Tabulatorstopps ausgerichtet wird. Wenn Sie z. B. „Zum nächsten Tabulatorstopp 4“ auswählen, wird der Text mit Tabulatoren an der Spaltenposition eingefügt, die dem nächsten Vielfachen von vier entspricht.

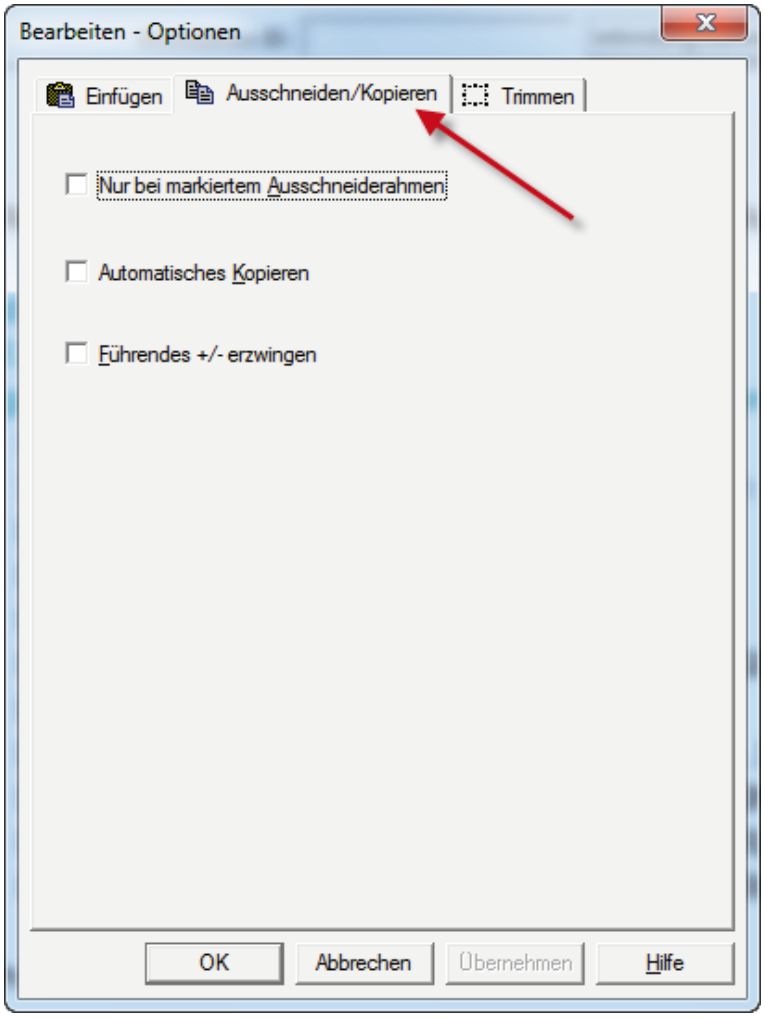
Tabulatorstopp durch n Leerzeichen ersetzen: Tabulatorstopps können durch eine bestimmte Anzahl Leerzeichen ersetzt werden. Beispielsweise wird bei Auswahl von „Tabulatorstopp durch 3 Leerzeichen ersetzen“ jeder Tabulatorstopp im ursprünglichen Text durch drei Leerzeichen ersetzt.

Standardmäßig wird jedes Tabulatorzeichen durch ein Leerzeichen ersetzt.

Daten in Felder einfügen
Diese Option steht nur für 5250-Sitzungen zur Verfügung.

Sie können Text mit Tabulatorstopps in nachfolgende ungeschützte Felder einfügen. Wird bei dieser Option ein Tabulatorzeichen erkannt, wird der nachfolgende Text in das nächste ungeschützte Feld der Emulatorsitzung eingefügt.

7.3.3.6
Seite 4



Ausschneiden/Kopieren von Daten

In diesem Register wird das Verhalten der Ausschneiden- bzw. Kopierfunktion eingestellt

Nur bei markiertem Ausschneiderahmen
Aktivieren Sie dieses Markierungsfeld, wenn nur der Inhalt des Ausschneiderahmens kopiert werden soll, der in der Sitzungsanzeige markiert wurde. Wenn kein Ausschneiderahmen markiert ist, wird standardmäßig der gesamte Inhalt der Anzeige kopiert.

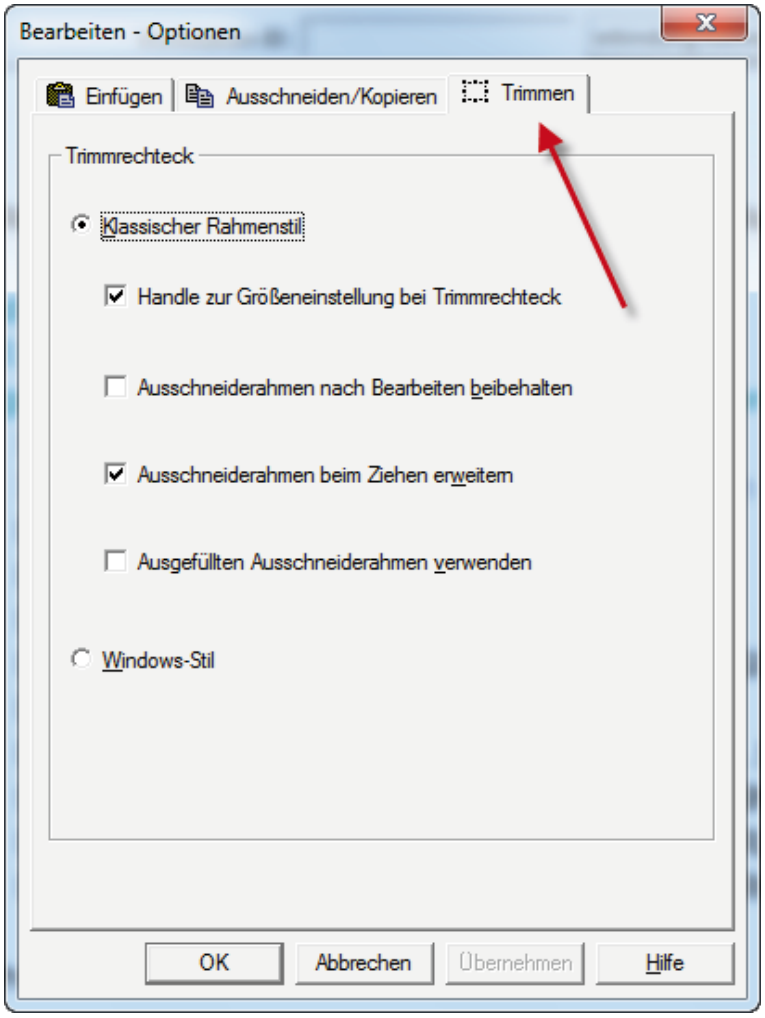
Führendes +/- erzwingen
Diese Option steht nur für 5250-Sitzungen zur Verfügung.

Bei einem als numerisch mit Vorzeichen definierten Feld erzwingt diese Option, dass das Plus- bzw. Minuszeichen am Anfang und nicht am Ende des Felds steht.

Automatisches Kopieren
Mit Hilfe dieser Option können Sie den ausgewählten Text automatisch in die Zwischenablage kopieren.

Nachdem Sie einen Bereich auf dem Bildschirm markiert haben, wird der ausgewählte Text automatisch in die Zwischenablage kopiert. Wenn ein vorhandener ausgewählter Bereich in einen anderen Anzeigenbereich verschoben wird, wird der Text in dem neu ausgewählten Bereich automatisch in die Zwischenablage kopiert.

Anmerkung: Wenn Sie dieses Markierungsfeld auswählen, wird auch die Funktion „Ausschneiderahmen nach Bearbeiten beibehalten“ ausgewählt.



Trimmen von Daten

7.3.3.6
Seite 6

Klassischer Rahmenstil
Diese Option ermöglicht es dem Benutzer, normale Rahmen wie z. B. Ausschneiderahmen zu zeichnen.

Handle zur Größeneinstellung bei Trimmrechteck: Aktivieren Sie dieses Markierungsfeld, wenn der Ausschneiderahmen mit Markierungspunkten angezeigt werden soll, die eine Änderung der Größe ermöglichen.

Ausschneiderahmen nach Bearbeiten beibehalten: Wenn Sie diese Option aktivieren, bleibt der Ausschneiderahmen auch nach der Bearbeitung aktiv.

Ausschneiderahmen beim Ziehen erweitern: Durch diese Option geht der Ausschneiderahmen in Zeichengrenzen über, während er in der Größe verändert wird.

Ausgefüllten Ausschneiderahmen verwenden: Durch diese Option ändert sich die Darstellung des Ausschneiderahmens. Anstelle der normalen Umrissgrenze wird der Ausschneidebereich als ausgefüllter Rahmen angezeigt.

Windows-Stil
Diese Option ermöglicht es dem Benutzer, für den Text einen soliden Markierungsblock auszuwählen . Wenn diese Option ausgewählt ist, werden standardmäßig ausgefüllte Ausschneiderahmen ohne Handles zur Größeneinstellung gezeichnet.

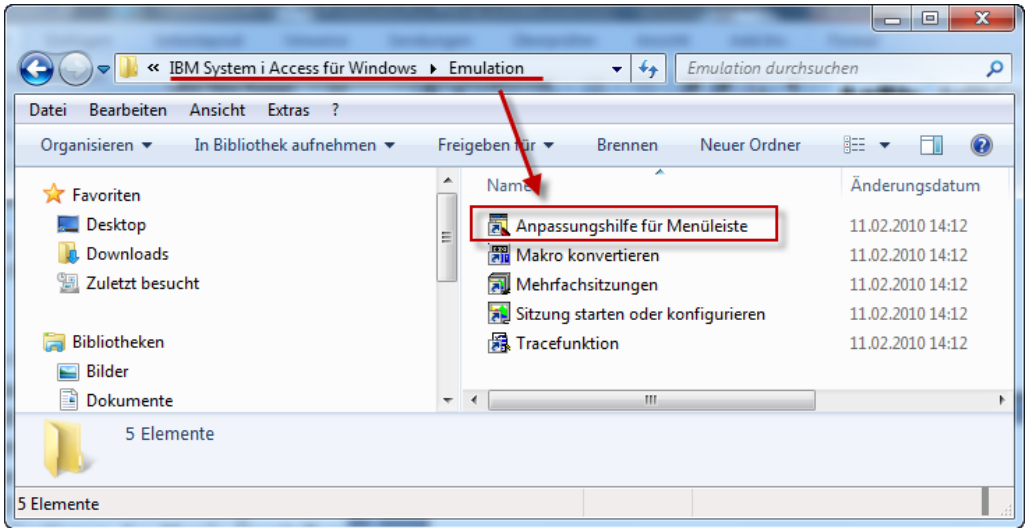
Anmerkung: Die Widerrufsfunktion im Bearbeitungsmenü ist für diese Funktionalität nicht aktiviert.

Menüleiste bearbeiten

Eine der größten Gefahrenquellen für Fehlbedienungen ist sicherlich die Funktionsleiste. Dort können Funktionen aufgerufen werden, die möglicherweise nicht allgemein zur Verfügung stehen sollen. Es ist ohne Weiteres möglich, alle (!) Knöpfe aus der Leiste zu entfernen.

Die zweite Gefahrenquelle ist die Menüleiste. Hier verbergen sich Funktionen, die den Betrieb eines Emulationsbildschirms insgesamt in Frage stellen können. Gehen Sie einfach einmal in das Menü Kommunikation; hier können Sie Verbindungsparameter verändern. Oder gehen Sie in das Menü Datei; an dieser Stelle können Sie einfach neue Sitzungen aufrufen ...

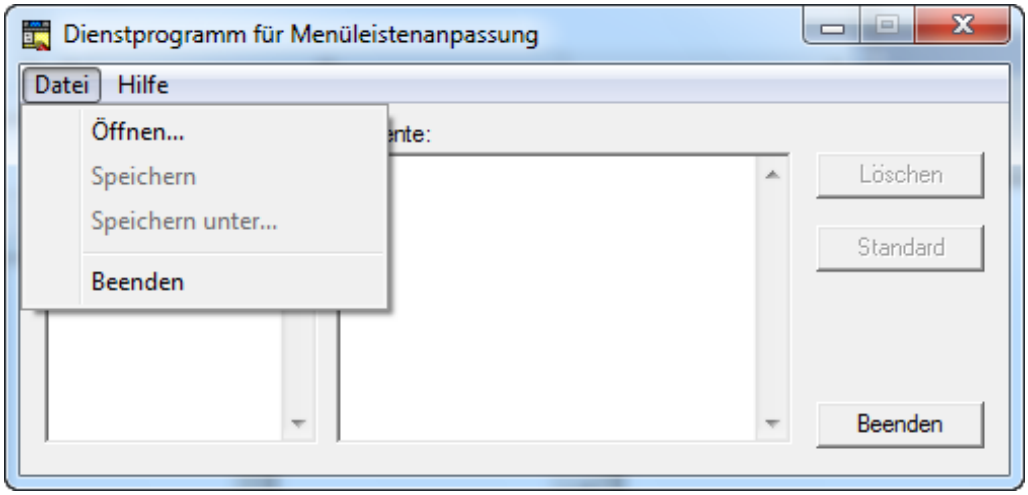
Zum Glück gibt es im Emulationsordner des System i Access eine Funktion, die hier Abhilfe bringen kann: die „Anpassungshilfe für Menüleiste“.



Anpassungshilfe für Menüleiste

7.3.4
Seite 2

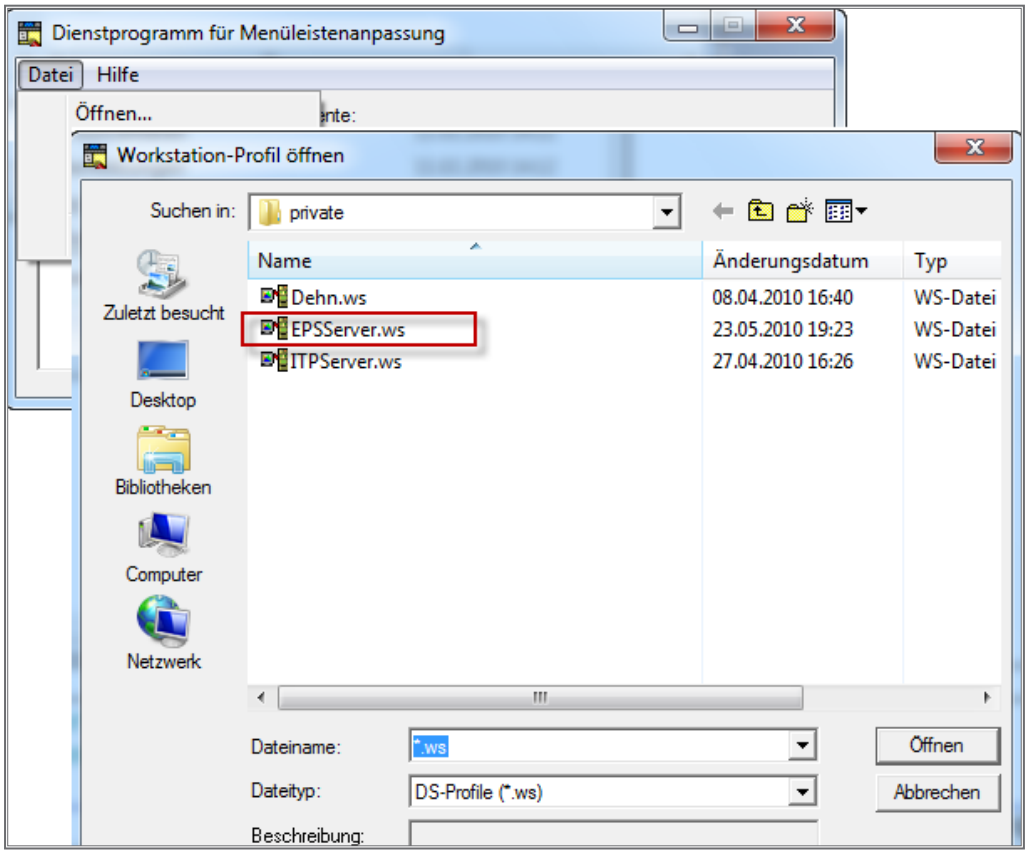
Führen Sie einen Doppelklick auf dieses Symbol aus und folgendes Fenster öffnet sich:



Der Menüeditor

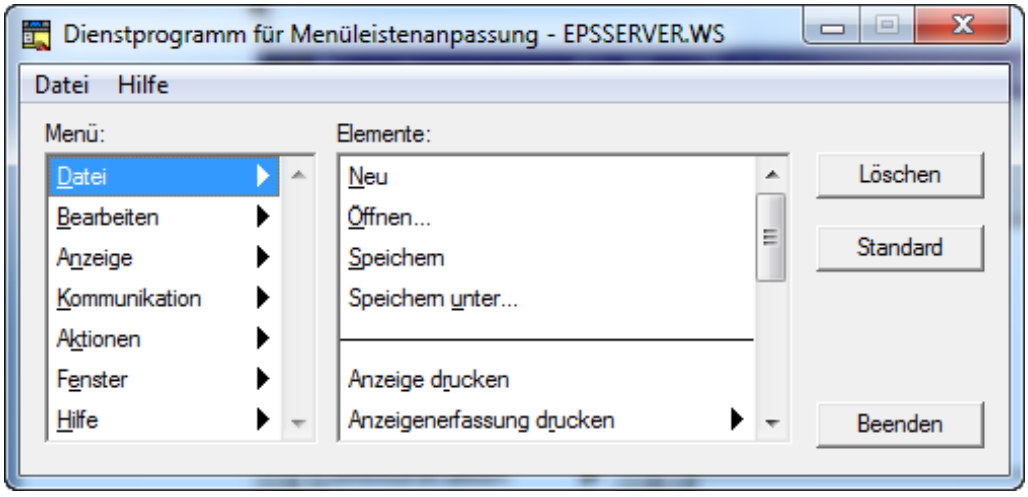
Sie sehen hier den Menüeditor, der er ermöglicht, aus einem bestehenden Menü Punkte herauszunehmen (Button „Löschen“) bzw. Standardeinstellungen des Emulationsfenstermenüs wieder herzustellen (Button „Standard“).

Um das Menü einer Bildschirmsitzung zu bearbeiten, gehen Sie in das Menü „Datei ➜ Öffnen“.



Profil öffnen

Im Dialog „Datenstationsprofil öffnen“ markieren Sie den zu bearbeitenden Bildschirmdialog und drücken dann den „OK“-Button.

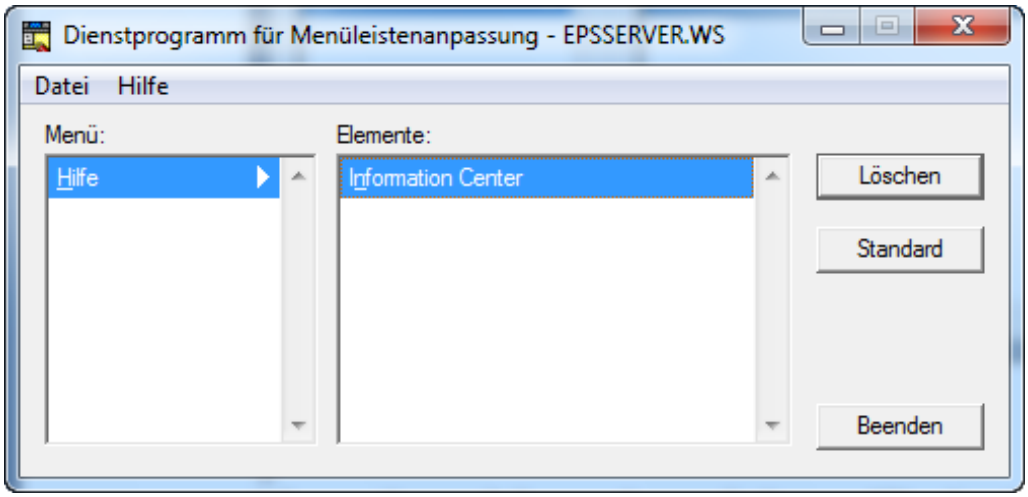


Die Menüstruktur der Sitzung EPSServer.WS

7.3.4
Seite 4

Sie sehen jetzt die aktuelle Menüstruktur der angewählten Bildschirmsitzung. Hier EPSServer.WS. Im List-Bereich „Menü“ erkennen Sie die Hauptmenüpunkte, die als Menüleiste des Bildschirms erscheinen. Im List-Bereich „Elemente“sehen Sie die Menüauswahlpunkte des markierten Hauptmenüeintrags.

Einträge können sowohl aus den Menüeinträgen (links) als auch aus den Menüpunkten (rechts) gelöscht werden.

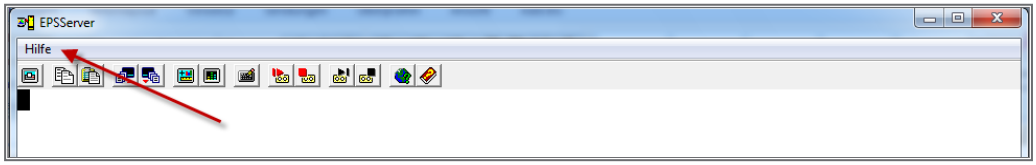


Das reduzierte Menü

Sie sehen hier, dass ich der Bildschirmsitzung radikal alle Menüpunkte entzogen habe – mit Ausnahme des Menüpunkts „Hilfe“ und der darin befindlichen Auswahl „Information Center“.

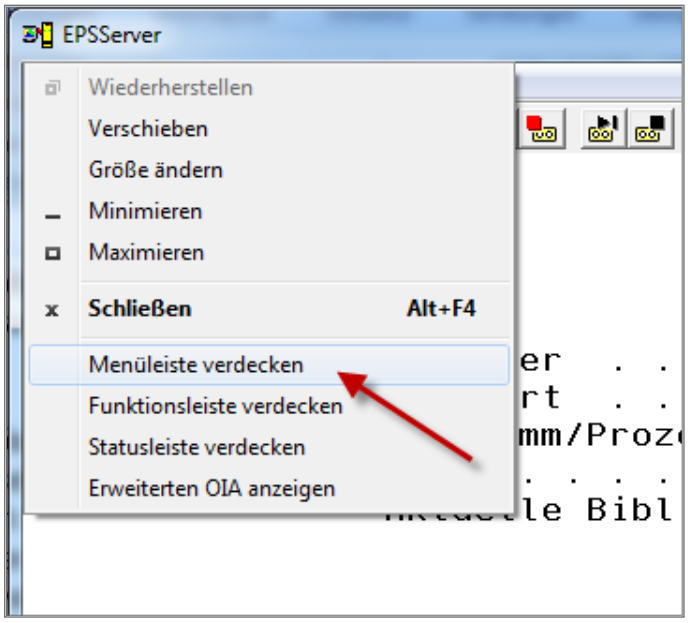
Mit dem „Speichern“-Dialog im Menü „Datei“ wird das modifizierte Bildschirmprofil weggeschrieben.

Rufen Sie jetzt diesen Bildschirm auf, so sieht er wie folgt aus:



Der entschärfte Bildschirm

Da nun das Menü und die Funktionsleiste so gut wie keine Funktionalität mehr besitzen, ist es nicht mehr notwendig, sie anzuzeigen. Lassen sich diese beiden Komponenten auch noch entfernen? Entfernen nicht, aber verstecken!



Die Menüleiste verdecken

Klicken Sie mit der Maus auf das Systemmenü des geöffneten Bildschirmfensters. Dort finden Sie zwei Menüpunkte:
– Menüleiste verdecken
– Funktionsleiste verdecken

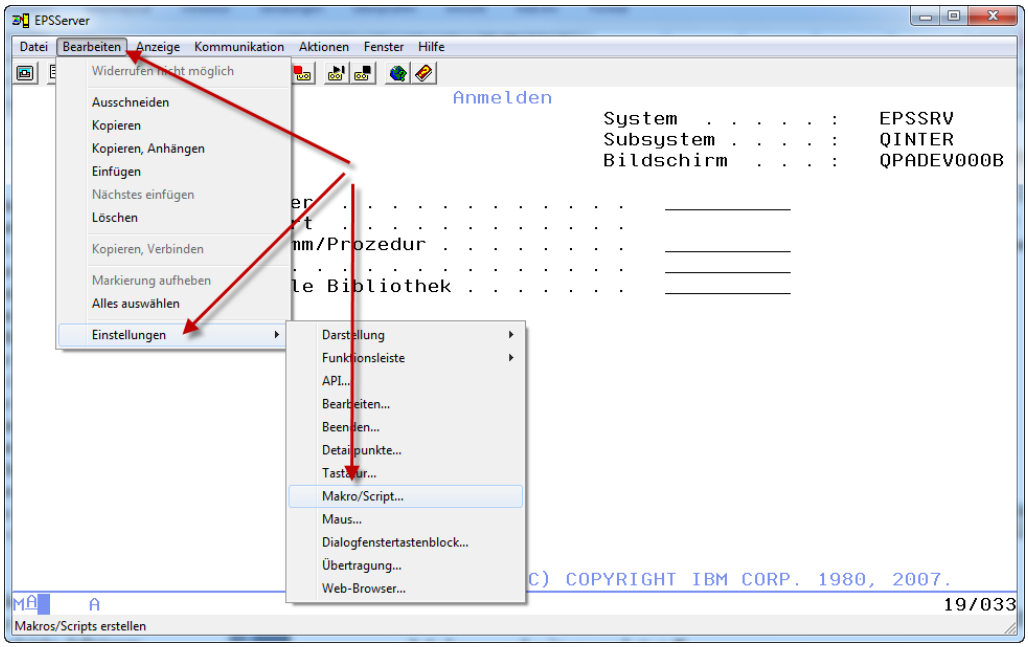
Je nachdem, welche Komponente Sie verdecken wollen, markieren Sie entweder den ersten, den zweiten oder nacheinander beide Werte. Sodann werden die ausgewählten Komponenten nicht mehr angezeigt.

7.3.4
Seite 6

Makros erstellen

Eine bei traditionellen Twinax-Terminals häufig genutzte Funktion ist die Aufzeichnung von Tastaturbefehlen, die dann per Funktionstaste aufgerufen werden können. Ich kenne viele System i-Administratoren, die oft genutzte Befehle wie WRKACTJOB (Mit aktiven Jobs arbeiten), WRKSYSSTS (Mit dem Systemstatus arbeiten) oder WRKSPL (Mit Druckausgaben arbeiten) in dieser Art auf eine Funktionstaste legen. Dadurch wird bei der Eingabe dieser Befehle viel Zeit gespart.

Eine ähnliche Funktion gibt es bei der 5250-Emulation des System i Access. Öffnen Sie einfach einmal eine Bildschirmsitzung und wählen Sie dann den Menüpunkt „Bearbeiten ➜ Einstellungen ➜ Makro/Script“ aus. Sie werden folgende Abbildung angezeigt bekommen:



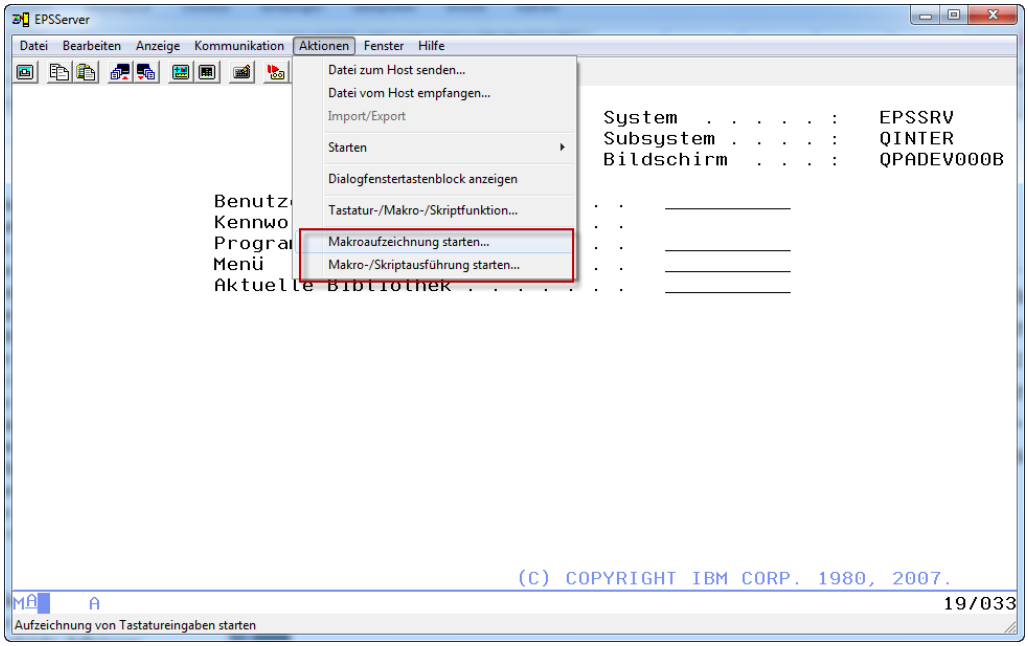
Menü „Bearbeiten ➜ Einstellungen“

Sie sehen hier eine Auswahl, die zum Bereich „Mit Makros arbeiten“ gehört:

– Makro/Skript
  Hier finden Sie einen einfachen Editor, um Makros bzw. Skripte zu erstellen. Was ist der Unterschied zwischen Makros und Skripten? Seit System i Access besteht die Möglichkeit, 5250-Makros in der Form von Visual Basic-Skripten zu erstellen. Diese Objekte werden einfach Skripte genannt. Makros, die mit der Makrosprache der 5250-Emulation erstellt wurden, werden einfach als Makros bezeichnet.

7.3.5
Seite 2

Weitere Makrofunktionen finden Sie im Menü „Aktionen“.

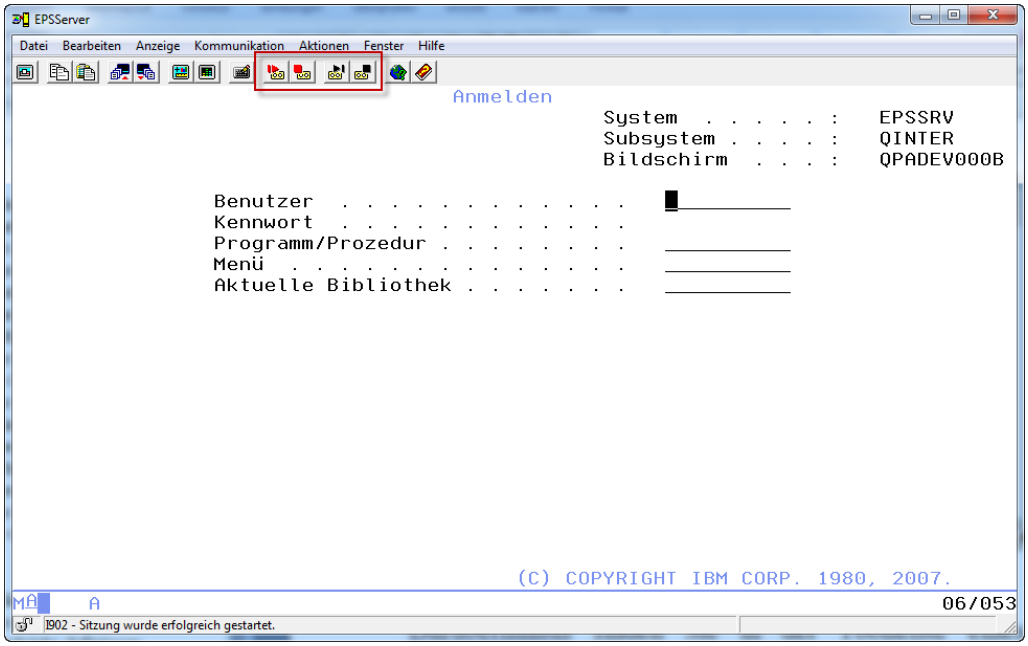


Das Menü „Aktionen“

– Makroaufzeichnung starten
  Dieser Punkt spricht für sich. Bildschirmaktionen werden aufgenommen während sie durchgeführt werden.

– Makro-/Skriptausführung starten
  Haben wir Makros bzw. Skripte erstellt, so können Sie u. a. mit Hilfe dieses Menüpunkts ausgeführt werden.

Aber darüber hinaus gibt es noch weitere Möglichkeiten der Makro- bzw. Skriptausführung, die Sie jetzt kennlernen werden.



Makrofunktionen in der Toolleiste

In der Toolleiste eines Bildschirms finden Sie ebenfalls Makroverarbeitungsoptionen.

Von links nach rechts betrachtet sind folgende Funktionen realisiert:
- Makro aufzeichnen
- Makroaufzeichnung stoppen und Makro sichern
- Makro/Skript starten
- Makroausführung stoppen

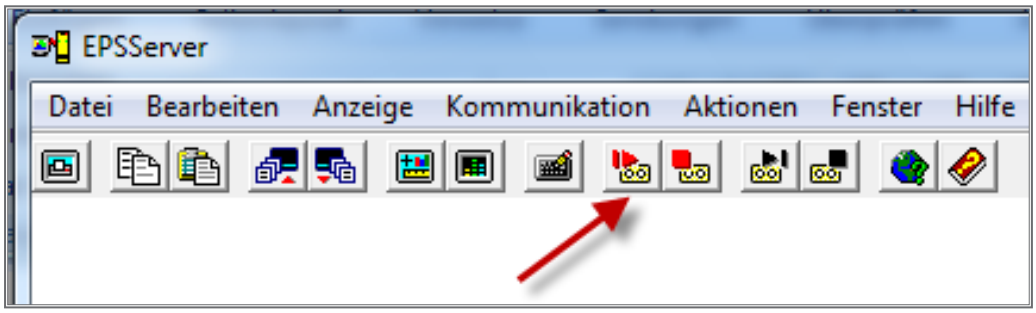
7.3.5 Seite 4

Makro aufzeichnen

Lassen Sie uns damit beginnen, einige Makros aufzuzeichnen. Wir wollen drei Makros erzeugen:

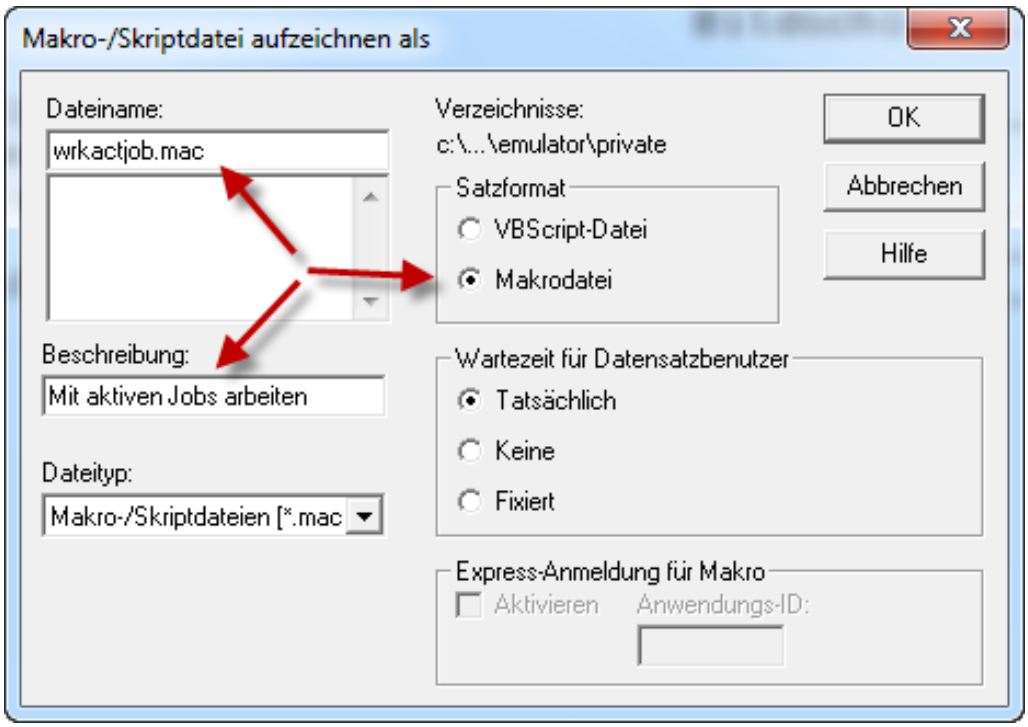
1. Makro für WRKACTJOB
2. Makro für WRKSYSSTS
3. Makro für WRKSPLF

Ich beginne die Makroaufzeichnung, indem ich auf den gezeigten Knopf der Funktionsleiste drücke.



Makroaufzeichnung starten

Jetzt öffnet sich folgender Dialog:



Makrofunktionen in der Toolleiste

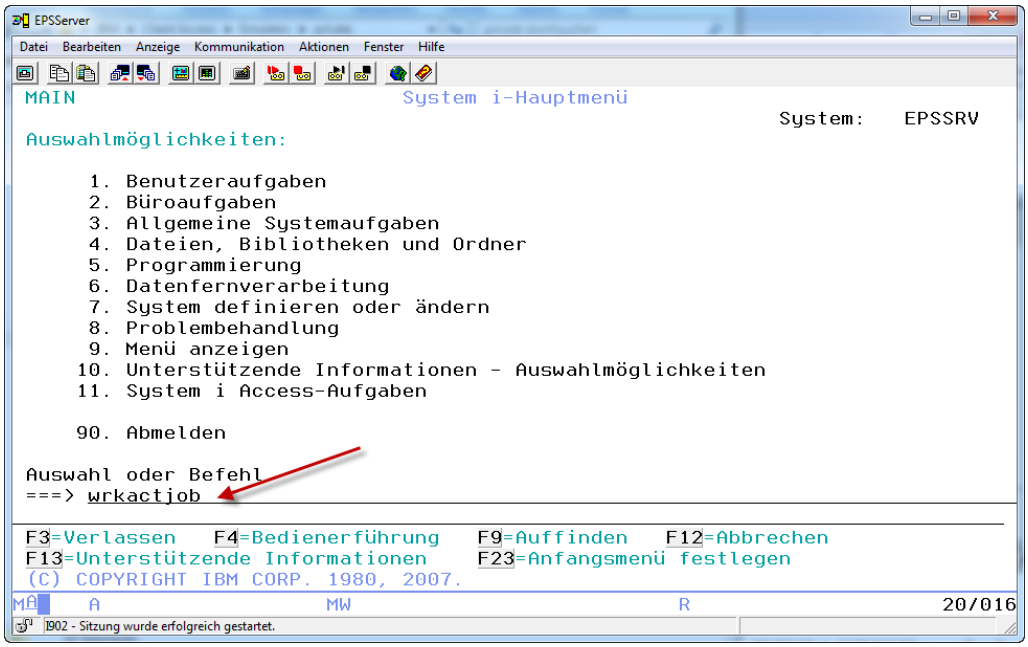
7.3.5.1
Seite 2

Im Parameter „Dateiname“ trage ich den Namen des Makros ein – in diesem Fall WRKACTJOB.MAC.

Dann hinterlege ich im Parameter „Beschreibung“ einen Langtext – „Mit aktiven Jobs arbeiten“. In der Auswahlgruppe Satzformat markiere ich den Radiobutton „Makrodatei“. Das Makro wird jetzt in der Form eines 5250-Makros abgespeichert. Den gesamten Dialog bestätige ich mit „OK“.

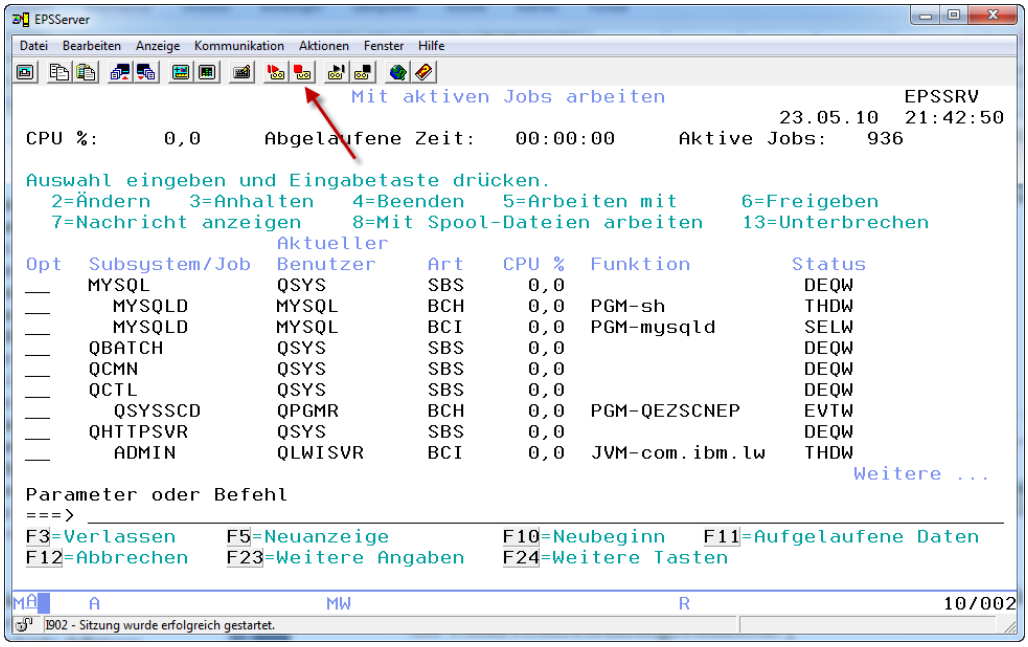
Jetzt wird das Makro im Verzeichnis „.../Emulator/private“ abgespeichert.

Damit ist die Makroaufzeichnung gestartet.



Befehl eingeben

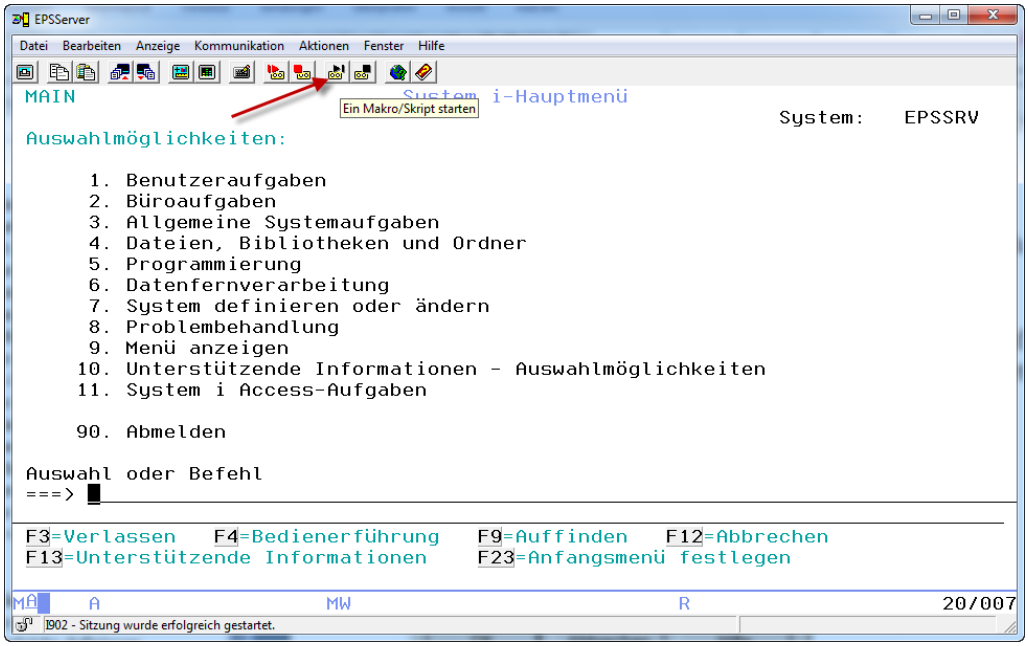
Jetzt gebe ich den Befehl WRKACTJOB auf der Befehlszeile des Bildschirms ein, bestätige die Eingabe durch Datenfreigabe und gelange in die Anzeige des Befehls WRKACTJOB.



Makroaufzeichnung beenden

Über den oben abgebildeten Funktionsknopf beende ich die Makroaufzeichnungsfunktion.

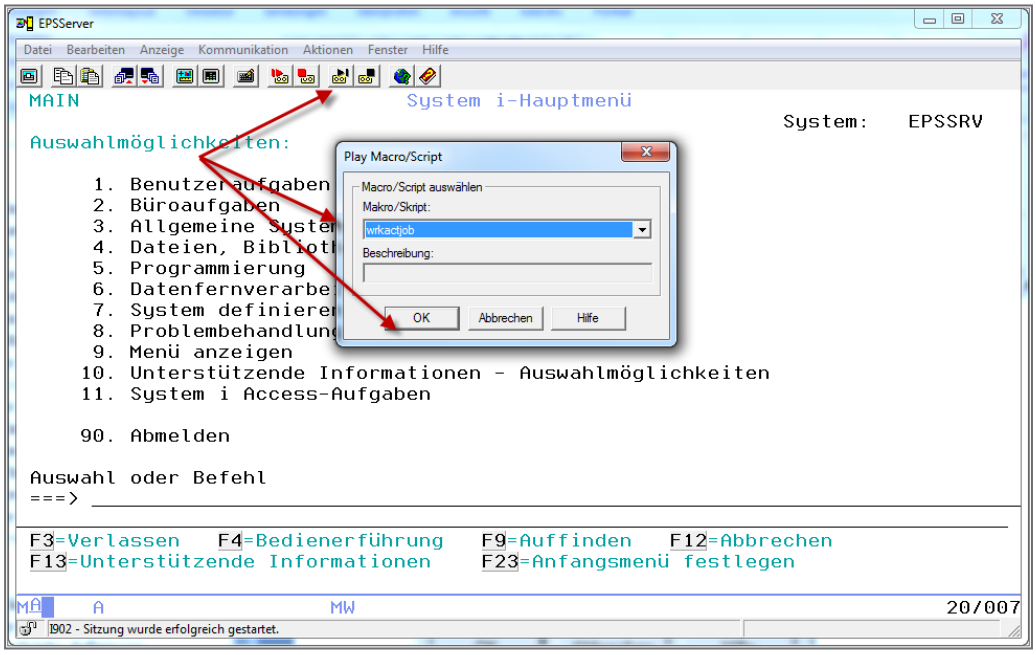
Zur Probe drücke ich jetzt den Funktionsknopf „Makros/Skript wiedergeben“.



Makro starten

7.3.5.1
Seite 4

Folgender Dialog öffnet sich:



Makro ausführen

Mit Hilfe des Pulldown-Menüs in der Combobox „Makro/Skript“ wähle ich mein gerade erstelltes Makro WRKACTJOB.MAC aus und bestätige die Auswahl mit „OK“. Sie sehen dann auf dem Bildschirm, wie der Befehl auf der Befehlszeile eingegeben und anschließend ausgeführt wird. Dieser Vorgang lässt sich in einer statischen Hardcopy nicht anzeigen.

Glauben Sie mir. Es funktioniert.

Diese Schritte wiederhole ich für die anderen oben angegebenen Befehle.

Makro modifizieren

Jetzt werden wir eines unserer Makros verändern. Meine Absicht ist es, folgende Funktion zu realisieren:

Es soll bei der Ausführung des Makros WRKSPLF.MAC nur die Druckausgabe für den Systemeigner der System i, den Benutzer QSYS, angezeigt werden.

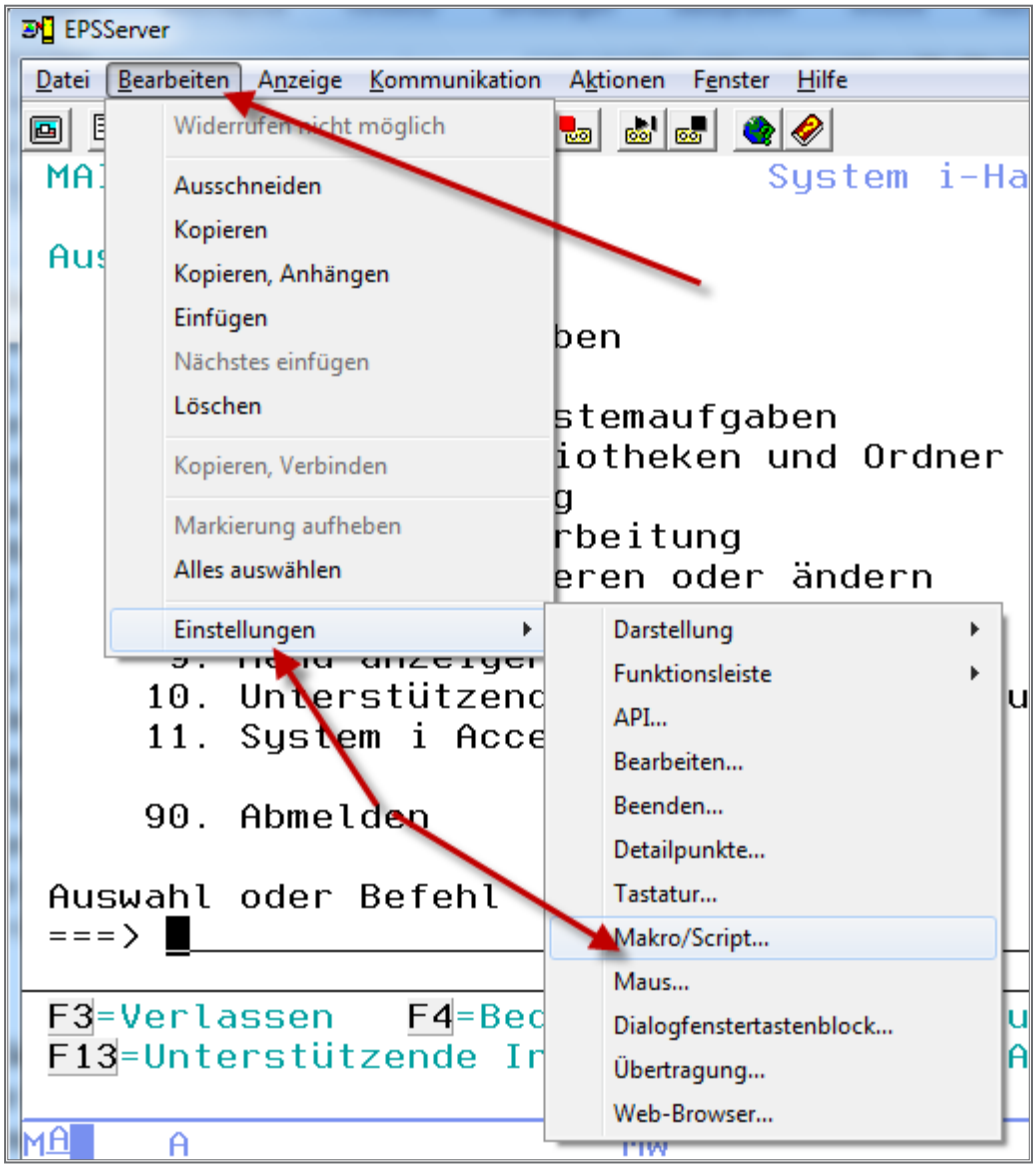
Dazu muss ich die Syntax des WRKSPLF-Befehls der System i kennen. Um nur die Druckausgaben für den Benutzer QSYS anzuzeigen, muss folgender Befehl eingegeben werden:

```
WRKSPLF SELECT(QSYS)
```

Wir könnten nun einfach ein neues Makro aufzeichnen, das den Befehl WRKSPLF aufruft und dann die Funktion F4=Bedienerhilfe ausführt, so dass wir dann den entsprechenden Benutzer eintragen könnten.

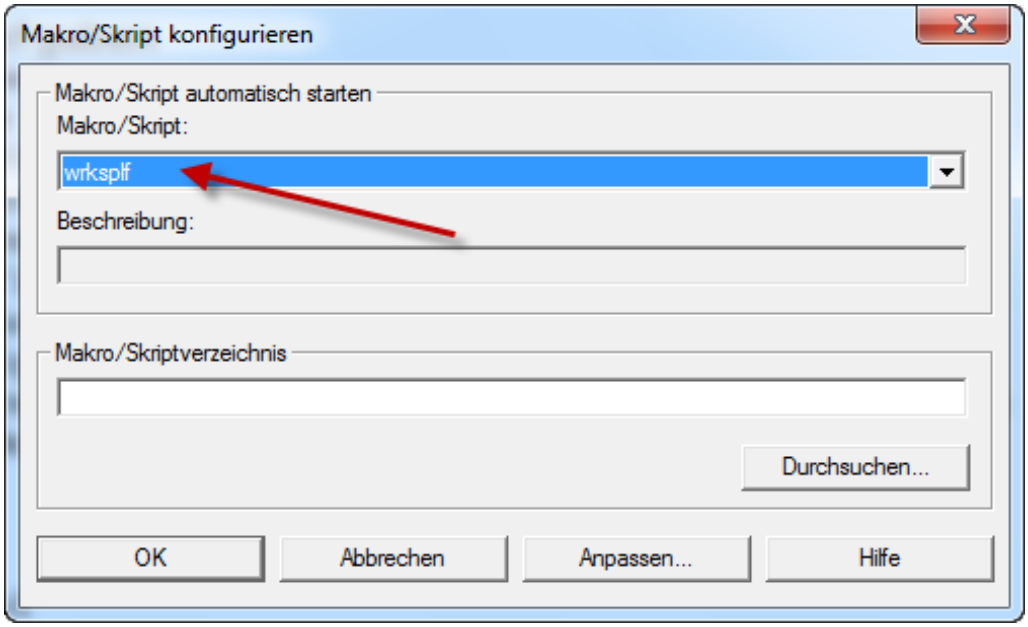
So werde ich jedoch nicht vorgehen! Ich werde das bestehende Makro WRKSPLF.MAC in den Makroeditor laden.

7.3.5.2
Seite 2



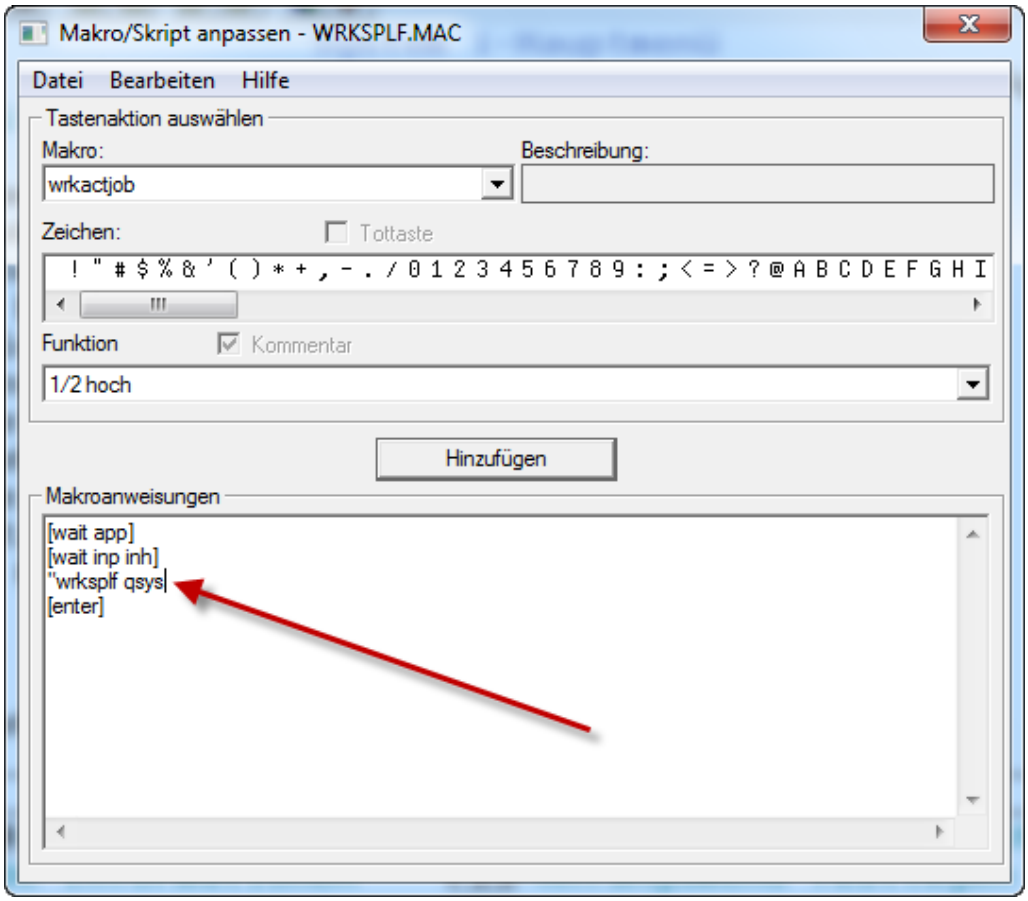
Makro bearbeiten

Diesen erreiche ich dadurch, dass ich in das Menü „Makro/Skript konfigurieren“ gehe, wo sich folgender Dialog öffnet:



Makro auswählen

Hier wähle ich das Makro WRKSPLF aus und öffne den Makroeditor durch Drücken des „Anpassen...“-Knopfs.



Der Makroeditor

Mit Hilfe der Funktionsgruppe „Tastenaktion auswählen“ können wir bestimmte Zeichen des System i-Zeichensatzes bzw. Funktionen der 5250-Emulation durch Drücken des Knopfs „Hinzufügen“ in den Bereich „Makroanweisungen“ übernehmen.

Hier eine kleine Übersicht, was alles mit Makros möglich ist:

Funktionen
Eine Funktion aus der Liste der Tastenaktionen auswählen, um Funktionen wie die folgenden einzufügen:
– Host-Befehle (z. B.: Löschen)
– Bearbeitungsbefehle (z. B.: Rückschritt)
– Verschiedene Befehle von Personal Communications (z. B.: Sicht anzeigen)

Makros
Ein großes Makro kann in kleinere Stücke unterteilt werden. Diese Submakros können innerhalb eines Makros aufgerufen werden. Um ein Makro in ein anderes Makro einzufügen, muss es aus der Liste im Feld „Tastenaktion auswählen“ ausgewählt werden.

Zeichen
Um Zeichen einzufügen, die nicht über die Tastatur eingegeben werden können, wählen Sie das gewünschte Zeichen in der Liste der Tastenaktionen aus.

Zeichenfolgen
Zeichenfolgen, die häufig eingegeben werden, wie z. B. „logon system1“, können eingefügt werden. Dazu die Zeichenfolge mit dem Präfix " (Anführungszeichen) eingeben – z. B.: "logon system1.

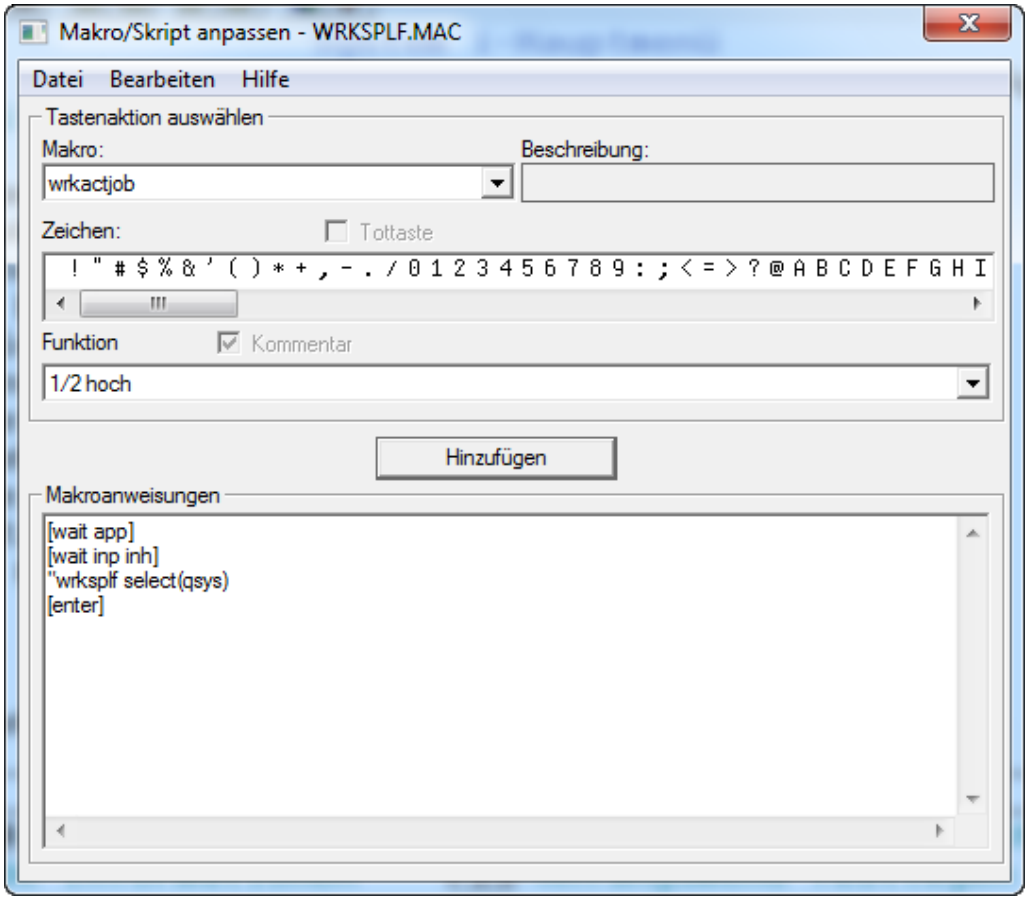
Wartebedingungen
Es können Wartebedingungen eingefügt werden, die das Makro unterbrechen, bis eine bestimmte Bedingung erfüllt ist – z. B.: „Warten bis der Cursor an Position (1,1) steht“.

Andere Anweisungen
Es gibt verschiedene andere Anweisungen, die verwendet werden können:

GOTO – LABEL – EXIT – RUN – NORMAL – MIN – MAX – REMARK

Es können auch die Befehle SEND oder RECEIVE verwendet werden, um eine Dateiübertragung einzuleiten.

Im Bereich Makroanweisungen ändere ich jetzt den Befehl WRKSPLF in der oben angegebenen Form.

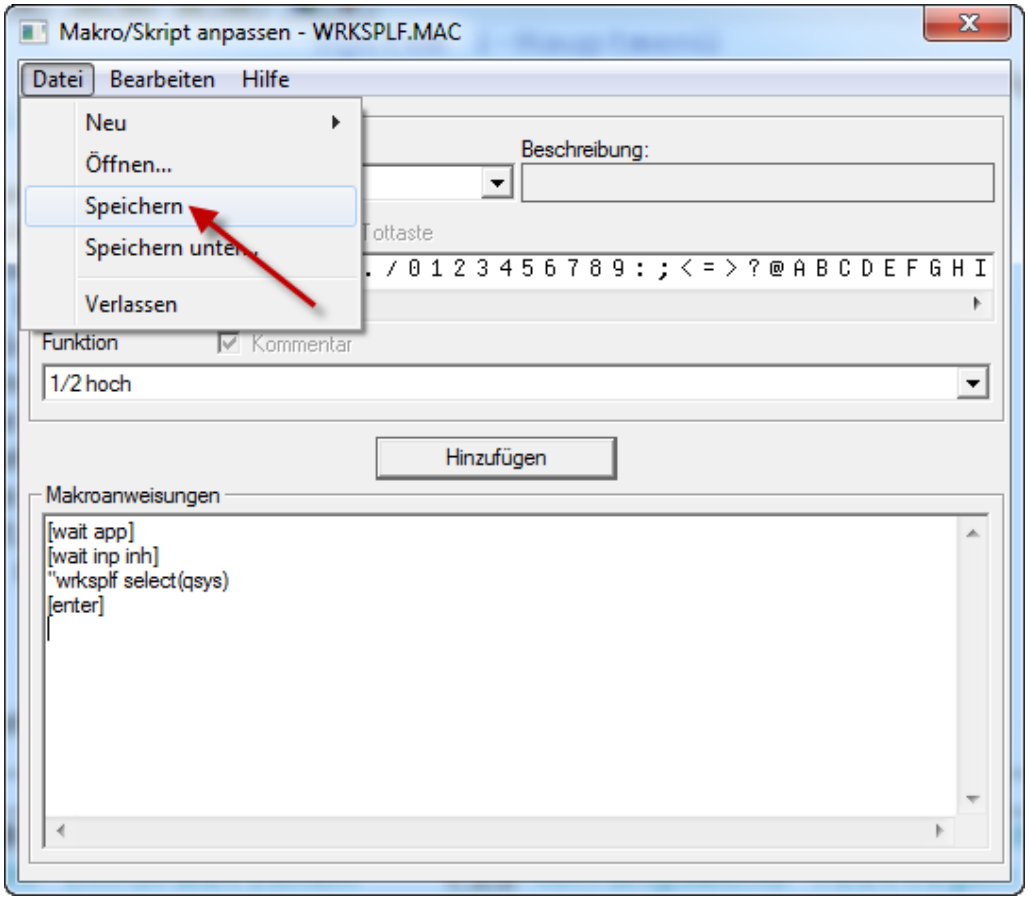


WRKSPLF ändern

Sie sehen, dass ich für WRKSPLF die Parameterübergabe geändert habe.

Jetzt muss das Makro nur noch abgespeichert werden.

7.3.5.2
Seite 6



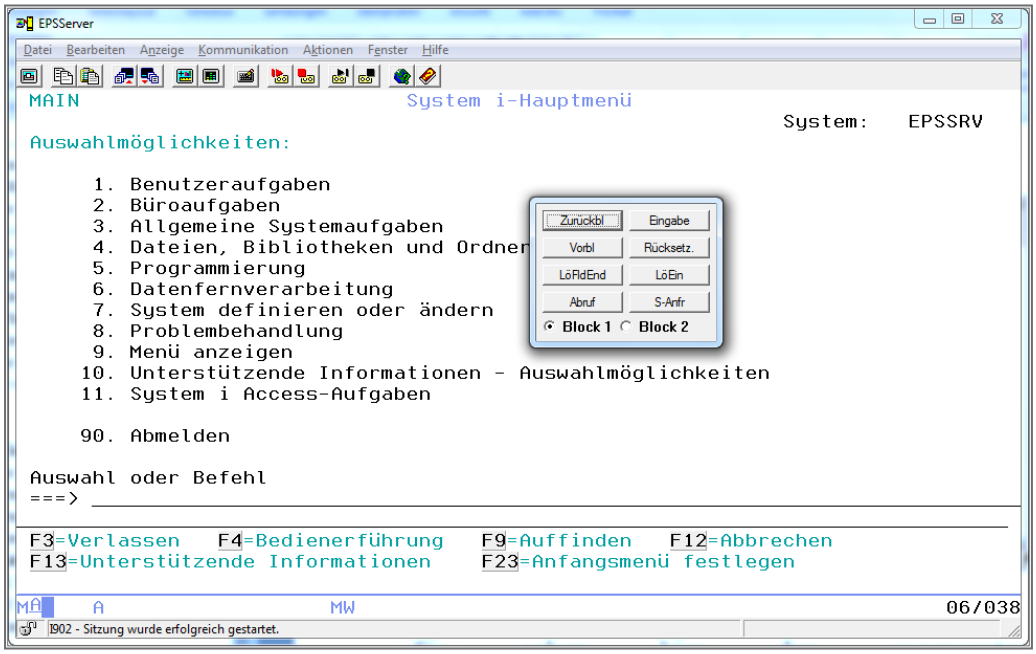
Makro sichern

Danach muss das Makro getestet werden. Auch hier setze ich wieder Ihr Vertrauen dafür voraus, dass dieses Makro funktionsfähig ist.

Dialogtastenblock erstellen

Eine weitere Möglichkeit, Makros auszuführen, besteht darin, sogenannte Dialogtastenblöcke zu erstellen. Mit Hilfe derartiger Dialogelemente sind wir nicht nur in der Lage Makros auszuführen, es können auch bestimmte Sondertasten ausgeführt werden, die sonst nur über die Mehrfachbelegung der Tastatur zu erreichen sind, oder es können Emulationsfunktionen direkt angesprochen werden.

Ich öffne einfach einmal meine Emulation und führe einen Rechtsklick auf den Bildschirmhintergrund aus.



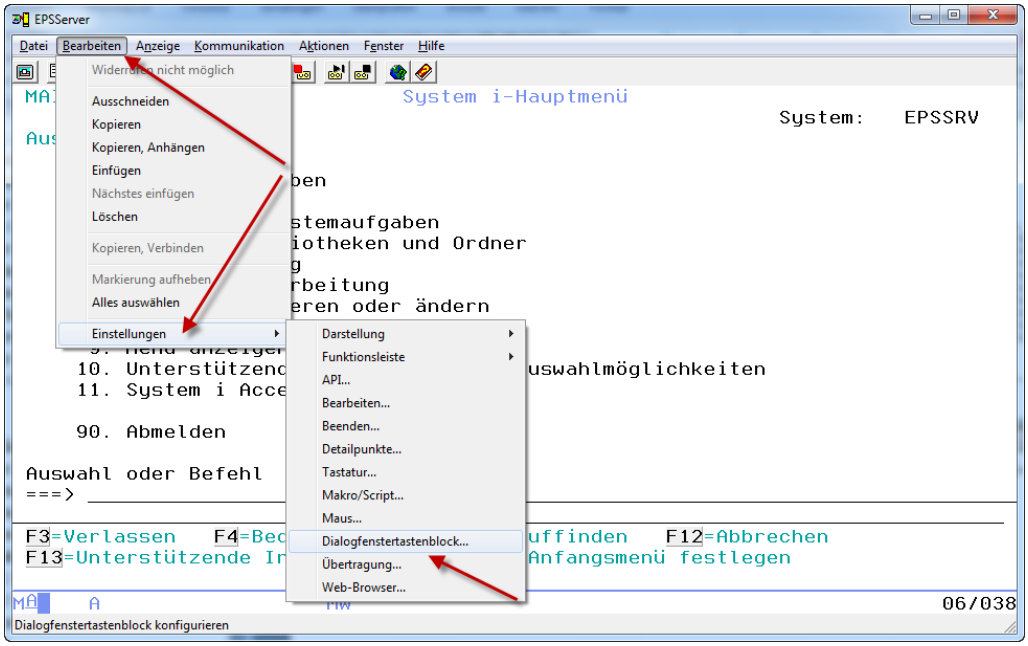
Der Standard-Dialogtastenblock

Sie sehen hier den sogenannten Systemdialogtastenblock, der Bestandteil jeder Emulation ist. Dieser Block zeichnet sich dadurch aus, dass er in zwei Ebenen angelegt ist, zwischen denen wir mit den Radio-Buttons „Block 1“ und „Block 2“ hin und herschalten können.

Ich werde jetzt einen neuen Dialogtastenblock definieren, aus dem heraus die im vorherigen Abschnitt erstellten Makros aufgerufen werden sollen.

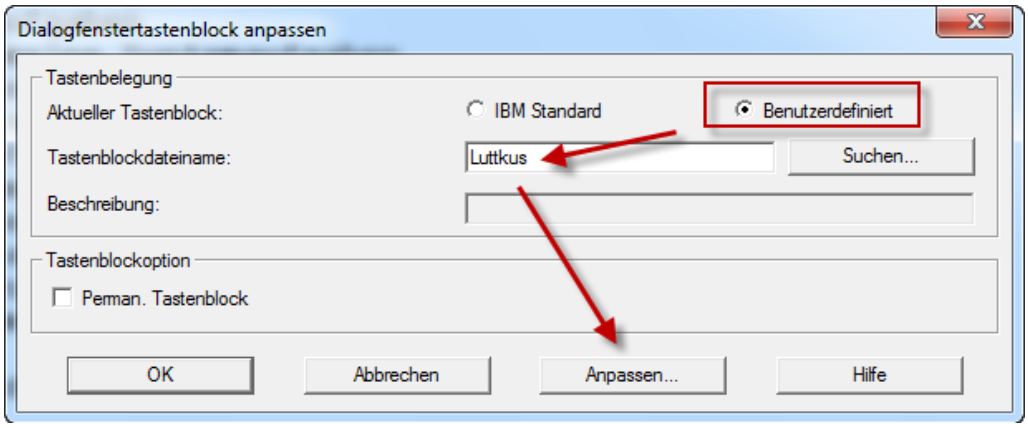
Dazu öffne ich das Menü „Bearbeiten ➜ Einstellungen ➜ Funktionen ➜ Dialogfenstertastenblock konfigurieren...“

7.3.6
Seite 2



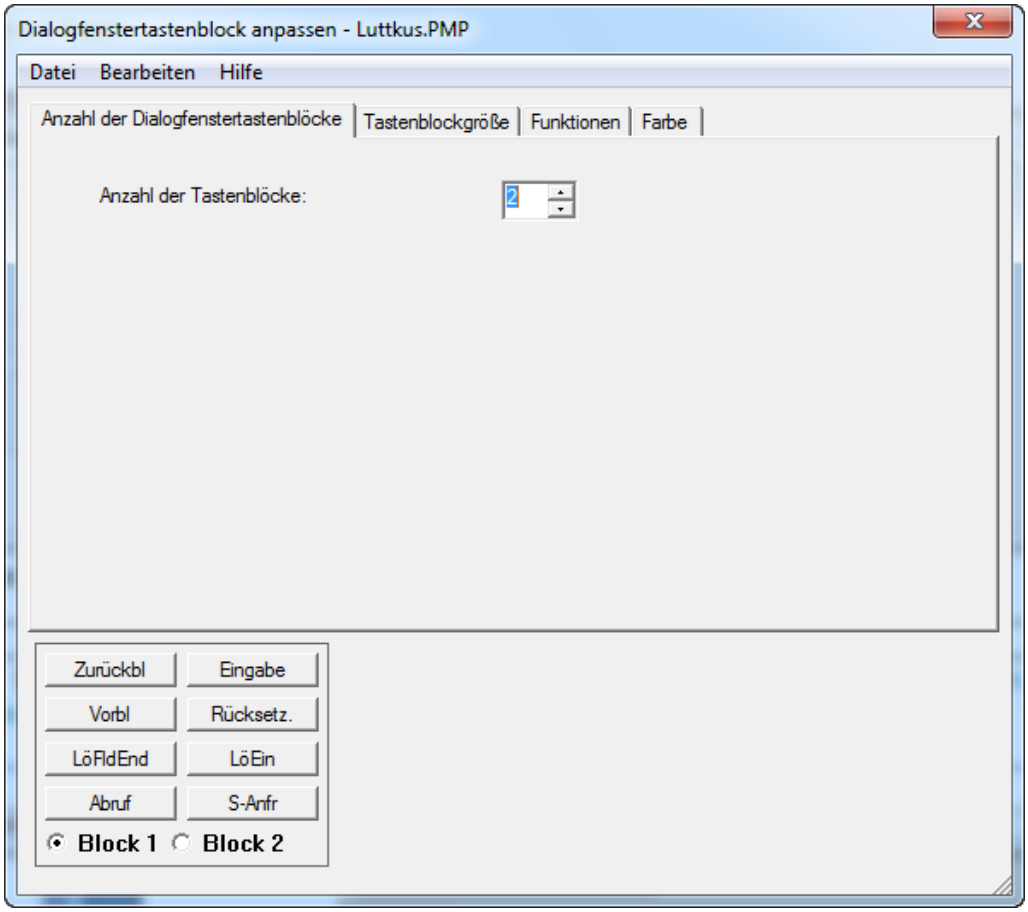
Einen neuen Dialogtastenblock definieren

Nachdem ich diese Menüpunkte ausgewählt habe, öffnet sich das Fenster „Dialogfenstertastenblock anpassen“.



Dialogfenstertastenblock anpassen

Ich stelle den Parameter „Aktueller Tastenblock“ auf „Benutzerdefiniert“ um, gebe dem neuen Tastenblock einen Namen und drücke den „Anpassen...“-Knopf.

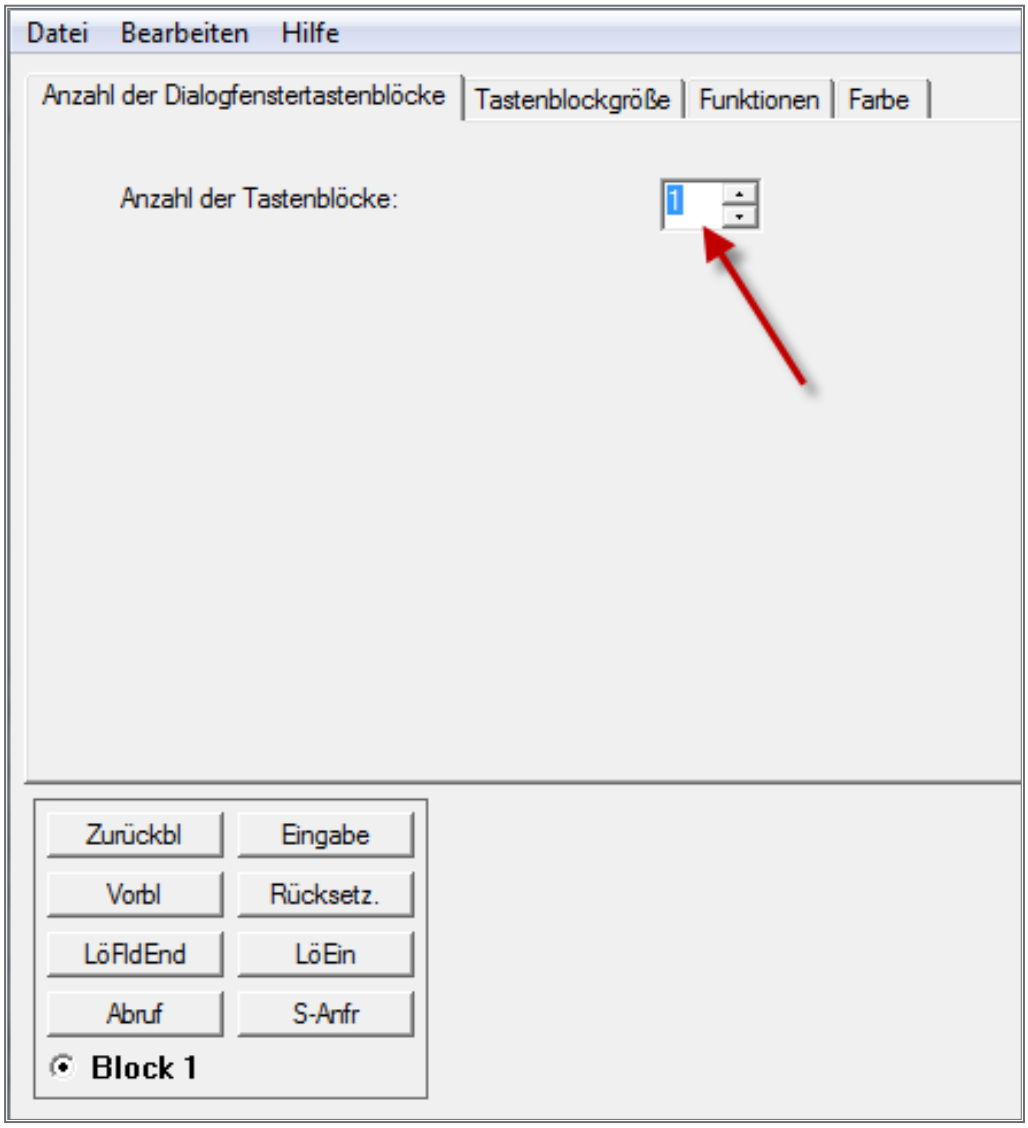


Der Tastenblockeditor

Sehen wir uns zuerst einmal die generellen Funktionsbereiche des Editors an.

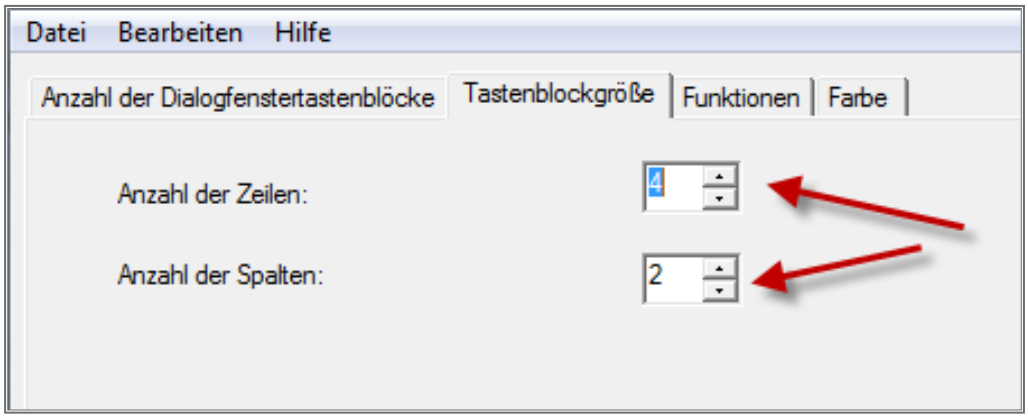
Im Hauptbild wählen wir die Anzahl der Tastenblöcke aus. Ich benötige nur einen Tastenblock.

7.3.6
Seite 4



Anzahl Tastenblöcke

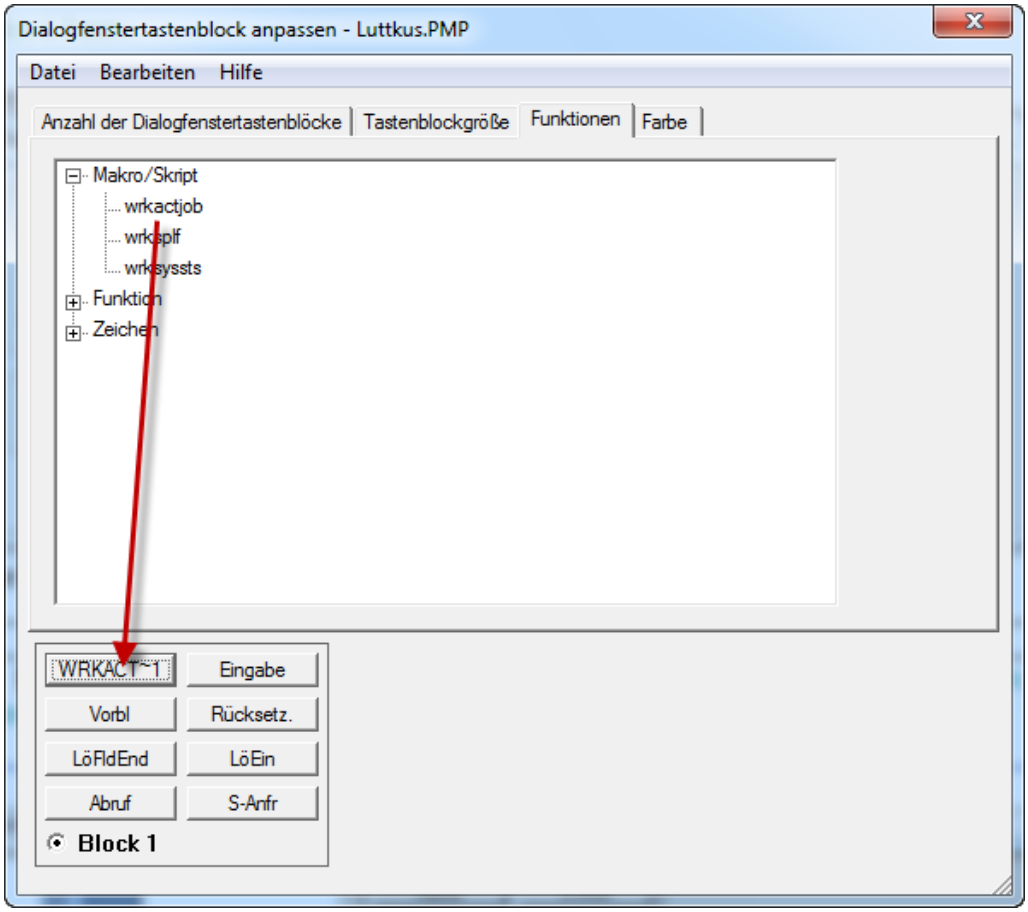
Rechts daneben finden wir den Dialog „Tastenblockgröße“.



Tastenblockgröße

Ich belasse es bei vier Zeilen mit zwei Spalten.

Jetzt wechsle ich in den Dialog „Funktionen“.



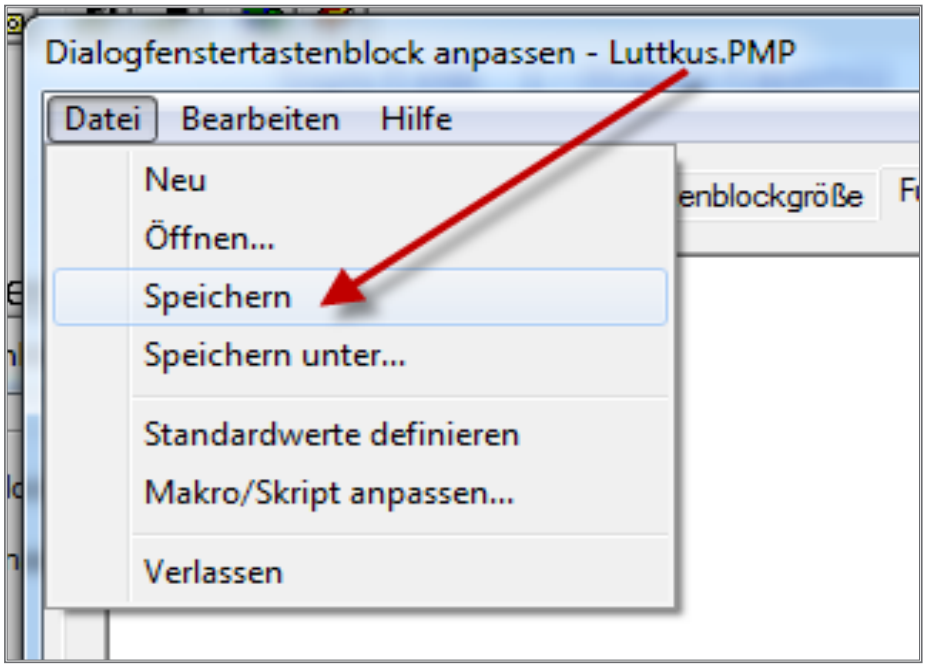
Funktionen wählen

Nachdem ich die Funktion „Makro/Script“ expandiert habe, ziehe ich – wie oben dargestellt – das Makro „wrkactjob“ mit der Maus auf den bezeichneten Knopf im Tastaturblock.

Gleichermaßen verfahre ich mit den anderen beiden Makros.

Diese Funktionen belasse ich so wie sie sind und speichere meinen benutzerdefinierten Tastenblock ab.

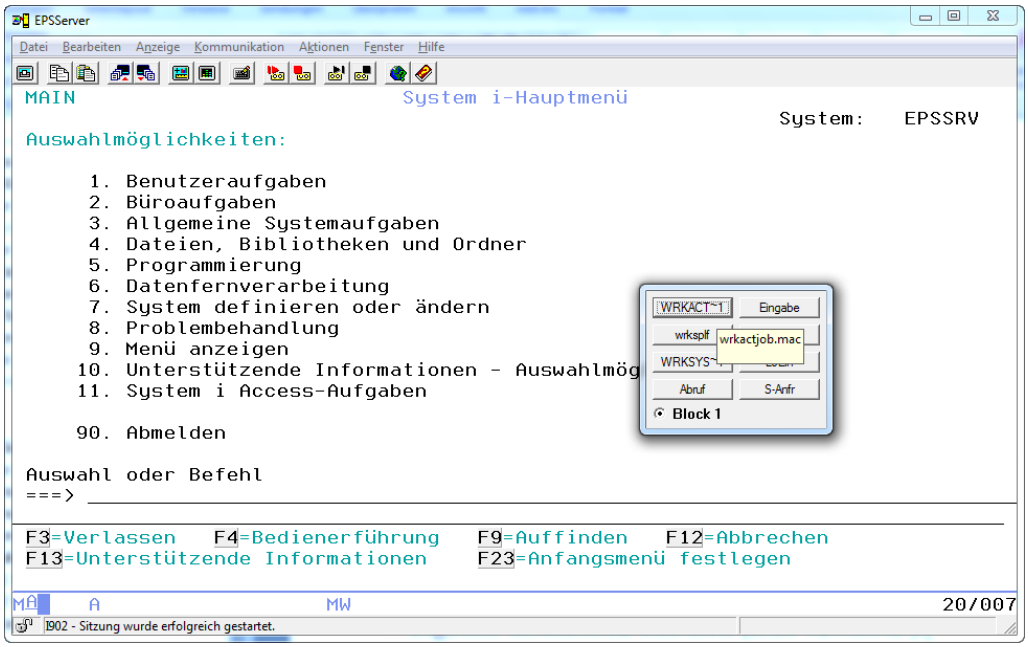
7.3.6
Seite 6



Tastenblock speichern

Im Menü „Datei“ wähle ich die Option „Sichern“. Jetzt wird mein Tastenblock in den Standardpfad „/Programme/IBM/System i Access/Emulator/private“ mit dem Namen „Luttkus.PMP“ gespeichert.

Zum Testen der Funktionen wechsle ich auf einen beliebigen System i-Bildschirm und drücke anschließend die rechte Maustaste.



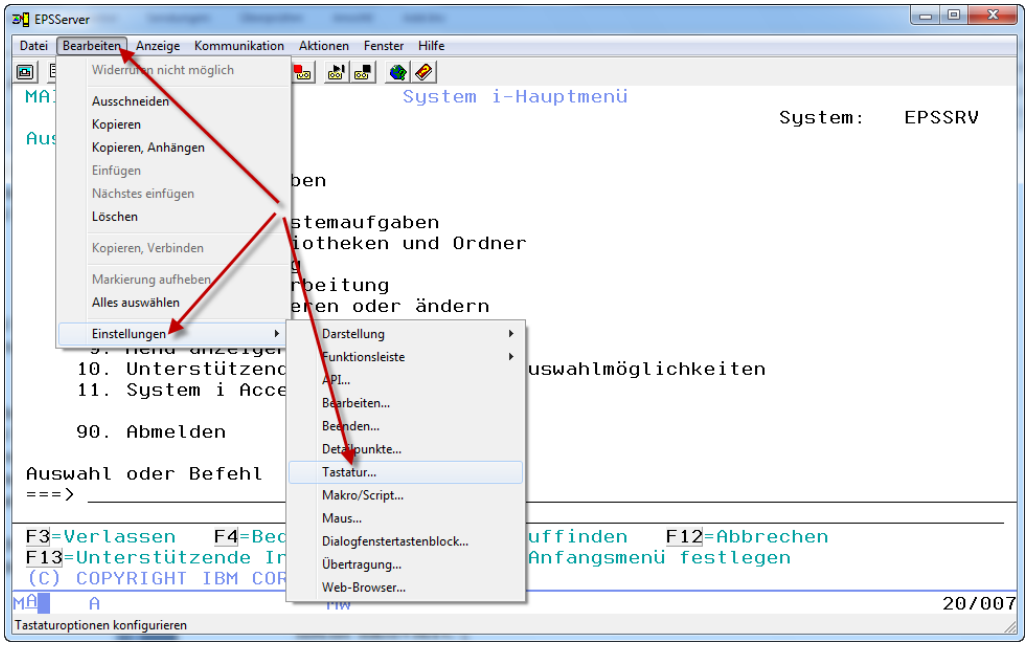
Eigenen Tastenblock aufrufen

Hier sehen Sie den soeben definierten Tastenblock.

Tastenbelegung ändern

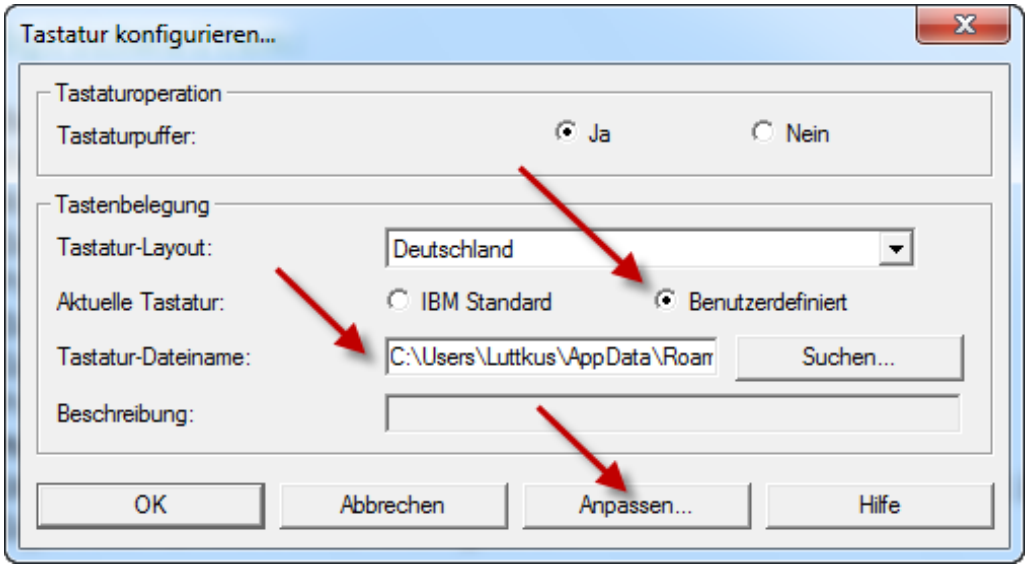
Die Tastaturbelegung für Ihre Bildschirmemulation wird in einer Datei mit dem Namen AS400.kmp verwaltet. In dieser Datei lassen sich Anpassungen vornehmen bzw. können auch eigene Tastaturprofile erstellt und dem Bildschirm zugeordnet werden.

Sie erreichen den Tastatur-Editor über den Pfad „Bearbeiten ➜ Einstellungen ➜ Tastatur…“.



Tastatur-Editor aufrufen

Jetzt öffnet sich der Dialog „Tastatur konfigurieren“.



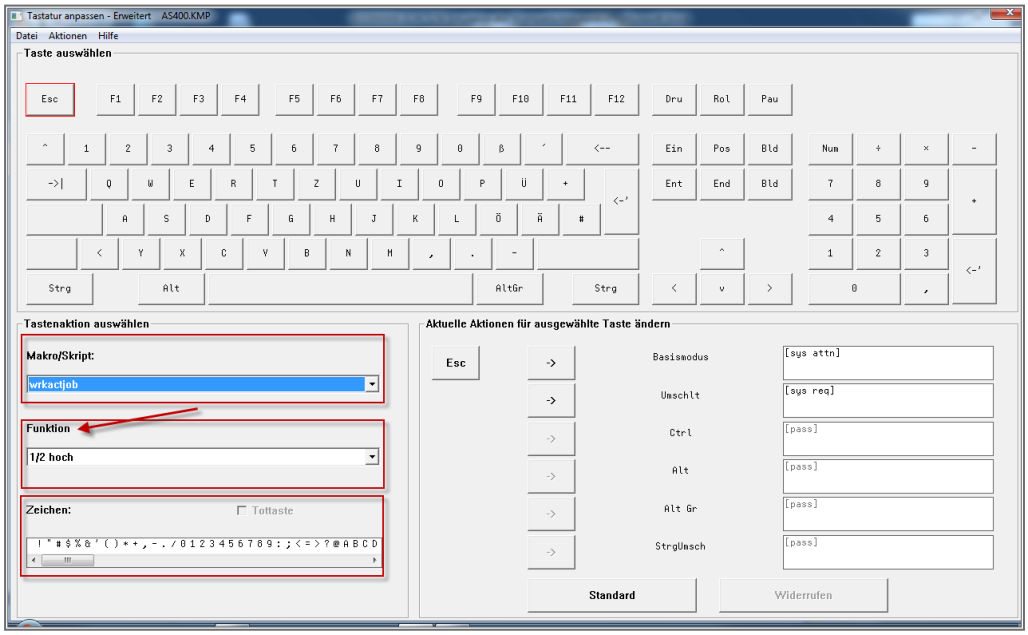
Tastatur konfigurieren…

7.3.7
Seite 2

Parameter „Aktuelle Tastatur“
Stellen Sie diesen Parameter auf „Benutzerdefiniert“.

Parameter „Tastatur-Dateiname“
Legen Sie hier den Namen des neuen Tastaturprofils fest. Der Speicherort der Tastaturprofildatei ist identisch mit dem des Bildschirmprofils und des Funktionsleistenprofils.

Verlassen Sie den gezeigten Dialog mit „Anpassen…“.



Tastaturprofil

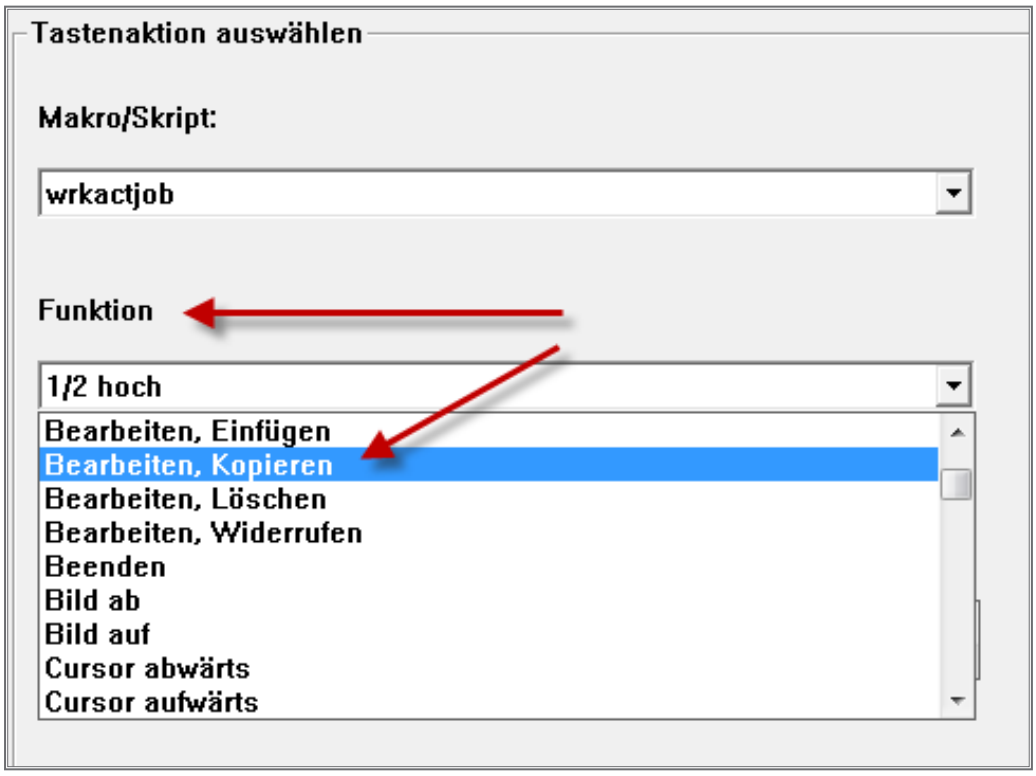
Grundsätzlich ist es möglich, drei verschiedene Objekte auf eine beliebige Taste zu legen:
– Makros/Script: Sie haben bereits Makros erstellt, die jetzt auf eine bestimmte Taste gelegt werden sollen.
– Zeichen: Sie beabsichtigen, das Zeichen auf einer Taste neu zu belegen.
– Funktion: Sie wollen eine neue Funktion auf eine bestimmte Taste bzw. eine Tastenkombination legen.

Wir werden in der Folge beispielhaft den Bereich „Funktion“ an einem konkreten Beispiel betrachten.

Will man Daten der Bildschirmoberfläche ins Clipboard stellen, so sind innerhalb der Bildschirmemulation die Windows Clipboard-Tasten
– Strg-C (markierten Inhalt ins Clipboard kopieren),
– Strg-V (Inhalt aus dem Clipboard in Bildschirm kopieren) und
– Strg-X (markierten Inhalt ausschneiden und ins Clipboard kopieren
nicht aktiviert.

Die genannten Funktionen werden wie folgt den entsprechenden Tastaturkürzeln zugeordnet.

Am konkreten Beispiel zeige ich Ihnen die Funktion Strg-C (markierten Inhalt ins Clipboard kopieren).



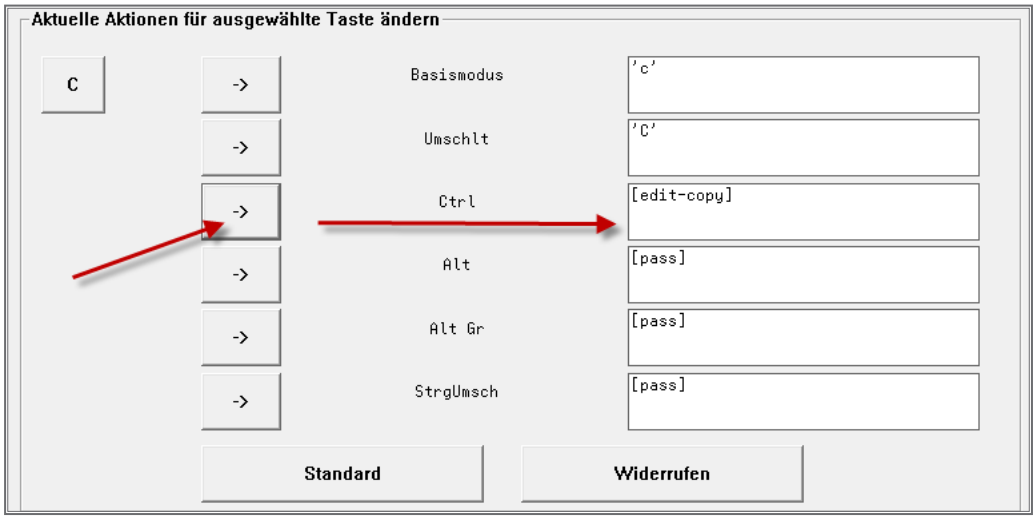
Windows Clipboard-Funktionen erstellen 1

Danach klicken Sie die Taste an, die mit dieser Funktion verbunden werden soll.

7.3.7
Seite 4

Windows Clipboard-Funktionen erstellen 2

Jetzt wechseln Sie in den nächsten Teilbereich des Tastatureditors.



Windows Clipboard-Funktionen erstellen 3

Im Dialog „Aktuelle Aktionen für ausgewählte Taste ändern“ ordnen Sie die angewählte Funktion der Funktionstaste zu.

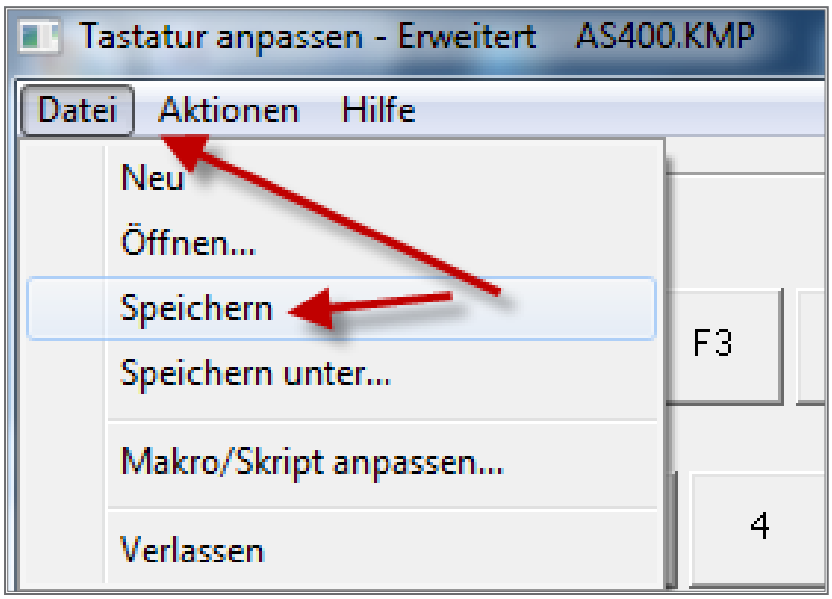
Sie sehen, dass eine Taste grundsätzlich sechsfach belegt werden kann.
– Basismodus: Grundsätzliche Funktion der Taste
– Umschlt: Funktion der Taste bei gedrückter Umschalt-Taste
– Ctrl: Funktion der Taste bei gedrückter Ctrl-Taste
– Alt: Funktion der Taste bei gedrückter Alt-Taste
– Alt Gr: Funktion der Taste bei gedrückter Alt Gr-Taste
– StrgUmsch: Funktion der Taste bei gedrückter Strg+Umschalt Taste

Ich lege die Kopierfunktion auf die Funktionsebene Ctrl.

Jetzt ist meine Clipboard-Kopierfunktion definiert.

Gleichermaßen verfahre ich bei den anderen Clipboard-Funktionen.

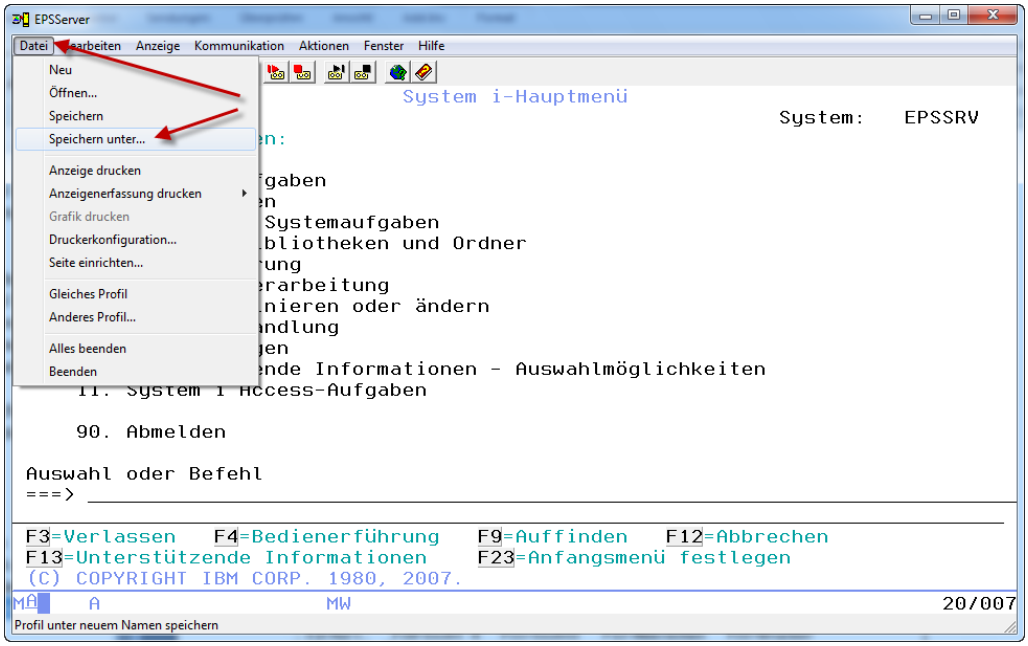
Nun wird das modifizierte Tastaturprofil mit „Datei ➜ Speichern“ weggeschrieben.



Windows Clipboard-Funktionen erstellen 4

7.3.7
Seite 6

Den Emulationsbildschirm sichern



Emulationsprofil sichern

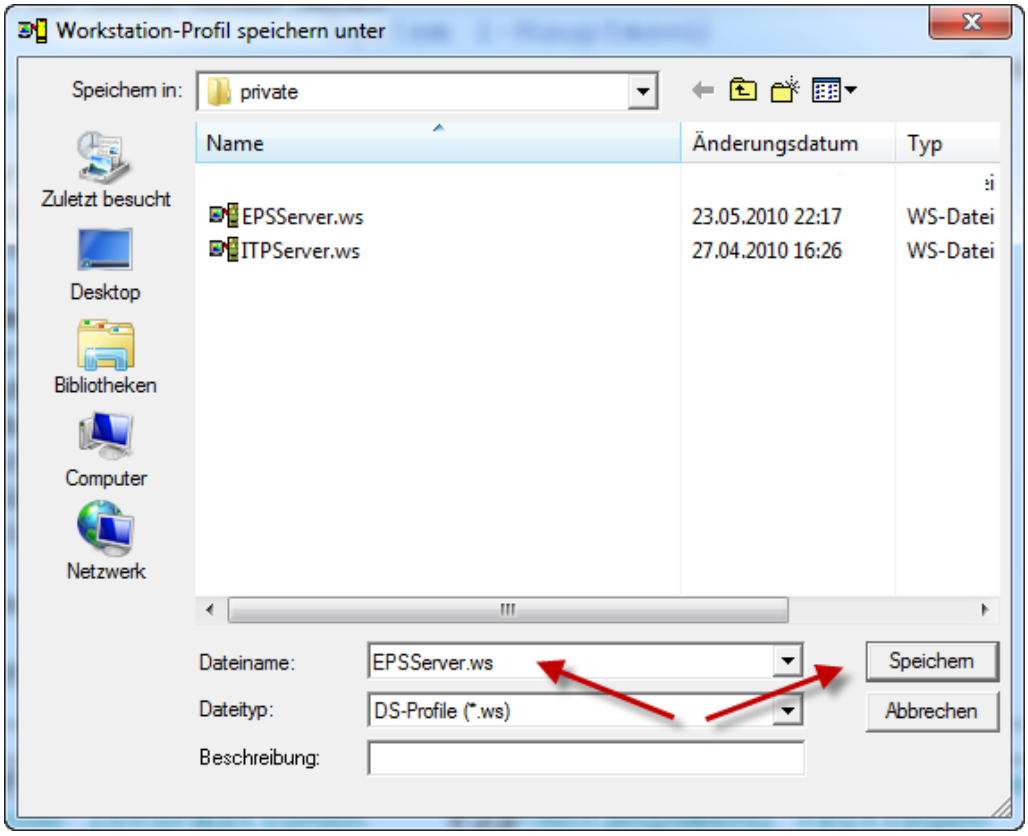
Nachdem die Konfiguration Ihren Wünschen entspricht, sollten die Einstellungen mittels „Datei ➜ Speichern unter...“ gespeichert werden. Für diese Sitzung werden zwei Konfigurationsdateien auf der lokalen Platte des PCs abgelegt:

1. Ein Bildschirmprofil, das die bildschirmbezogenen Konfigurationsdaten enthält: Verbindung zum System, Anzeigegröße, Windows-Darstellung etc.
2. Ein Funktionsleistenprofil, das die Darstellung der Funktionsleiste enthält. Dieses Profil kann dann in späteren Konfigurationen mit Hilfe des Menüs „Darstellung ➜ Funktionsleiste laden“ erneut verwendet werden.

Verlassen Sie den Emulationsbildschirm und Sie werden aufgefordert, die Konfiguration zu sichern.

Zuerst sichern Sie die Konfigurationsdaten für das Bildschirmprofil.

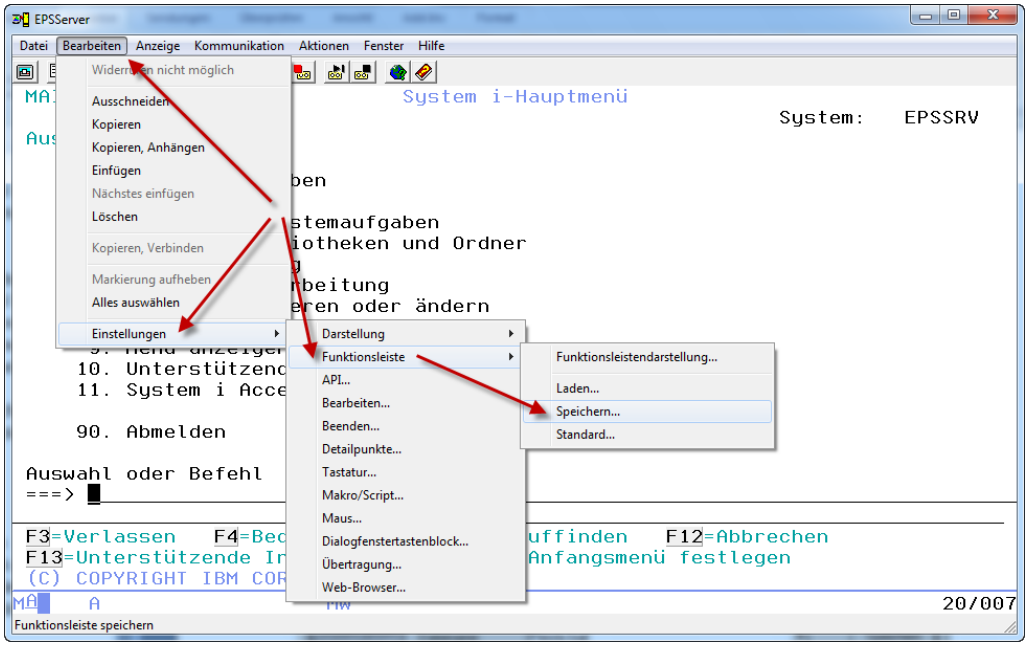
7.3.8
Seite 2



Bildschirmprofil sichern

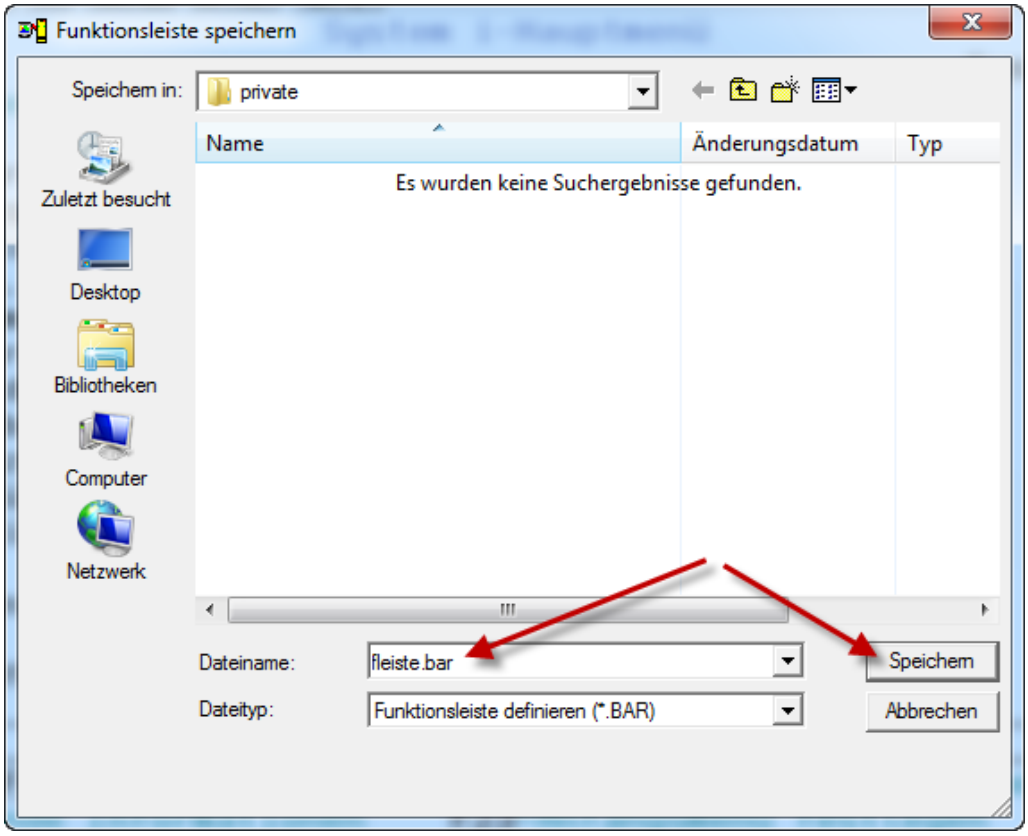
Bildschirmprofile werden per Default im Standardverzeichnispfad gespeichert. Die Konfiguration wird in eine Datei mit der Endung *.ws gespeichert. Sie sehen hier, dass ich als Profilname EPSServer.WS verwende.

Als Nächstes erfolgt das Speichern der Funktionsleiste über das Menü „Bearbeiten ➜ Einstellungen ➜ Funktionsleiste ➜ Speichern...“.



Funktionsleiste speichern

Ich vergebe als Namen „Fleiste.bar“.



Das Funktionsleistenprofil

7.3.8
Seite 4

Das Funktionsleistenprofil wird am gleichen Ort wie das Bildschirmprofil abgelegt. Die Konfiguration der Leiste wird in eine Datei mit der Endung *.bar gespeichert. Sie sehen hier, dass ich als Profilname FLEISTE.BAR verwende.

Damit ist die Bildschirmkonfiguration beendet.

Bildschirme konfigurieren – einmal anders

Wenn wir einmal davon ausgehen, dass Sie kein System i Access-Neuling sind, dann nehme ich an, dass die Erstellung von Bildschirmen in Rahmen von System i Access keine allzu schwierige Tätigkeit für Sie ist. Sollten Sie jedoch gerade in die System i Access-Materie eingestiegen sein, so empfehle ich Ihnen, sich zunächst einmal in den vorherigen Kapiteln damit vertraut zu machen, wie ein System i Access-basierter Bildschirm zu generieren ist.

7.3.9

Seite 2

Fallstudie – Massenauslieferung von Bildschirmen

Lassen Sie uns über folgenden Fall nachdenken: Wir haben ein neues IBM System i bekommen. Als Middleware zwischen den Windows-PCs und dem System i wird System i Access eingesetzt. Das System i ist netzwerktechnisch konfiguriert, d. h. TCP/IP läuft.

Unsere Aufgabe besteht nun darin, möglichst schnell für 80 an das Netzwerk angeschlossene PCs jeweils zwei System i-Bildschirmemulationen auszuliefern. Das heißt, wir haben letztendlich 160 Bildschirme zu erstellen.

Wenn wir uns vorstellen, dass die Generierung von Bildschirmen ein dialogorientierter Prozess ist, wird uns möglicherweise doch schwindelig. 160 Mal den gleichen Dialog an 80 Arbeitsplätzen ausführen? Das grenzt doch an Strafarbeit!

7.3.9.1

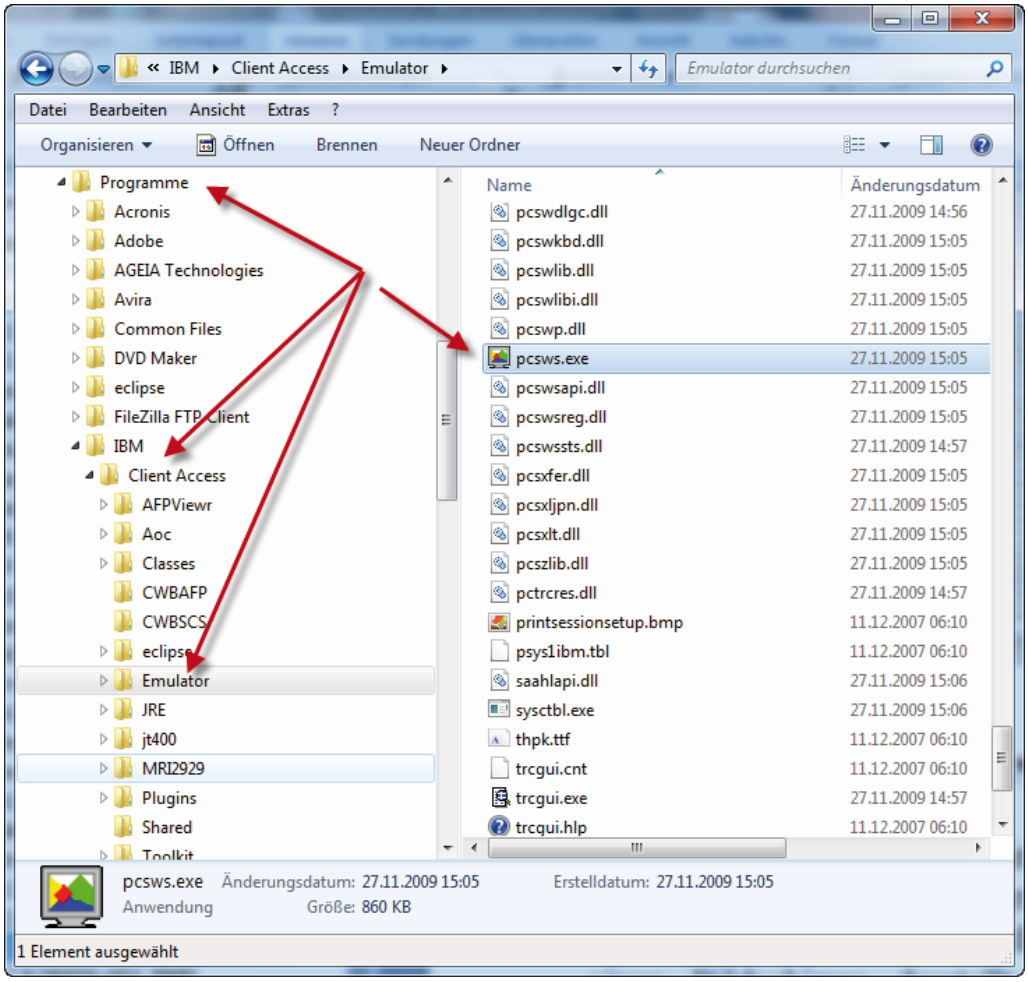
Seite 2

Bildschirmsitzung

Ich weiß nicht, wie es Ihnen geht, aber ich habe schlicht und einfach keine Lust, diese Arbeit in der oben beschriebenen Form durchzuführen. Wie können wir zu einer besseren Lösung kommen? Der Lösungsansatz liegt darin, dass wir ein Verständnis dafür entwickeln, welche Komponenten des System i Access auf dem PC tätig werden, um einen System i-Bildschirm emulieren zu können.

Die an einem Bildschirm aktiv beteiligten Objekte sind in Ihrer Zahl begrenzt:

1.) Die eigentliche Bildschirmfunktion verbirgt sich hinter dem Programm PCSWS.EXE. Dieses Programm kann jedoch allein nicht tätig werden. Das Programm PCSM.EXE befindet sich im Pfad „./Programme/IBM/Client Access/Emulator“.

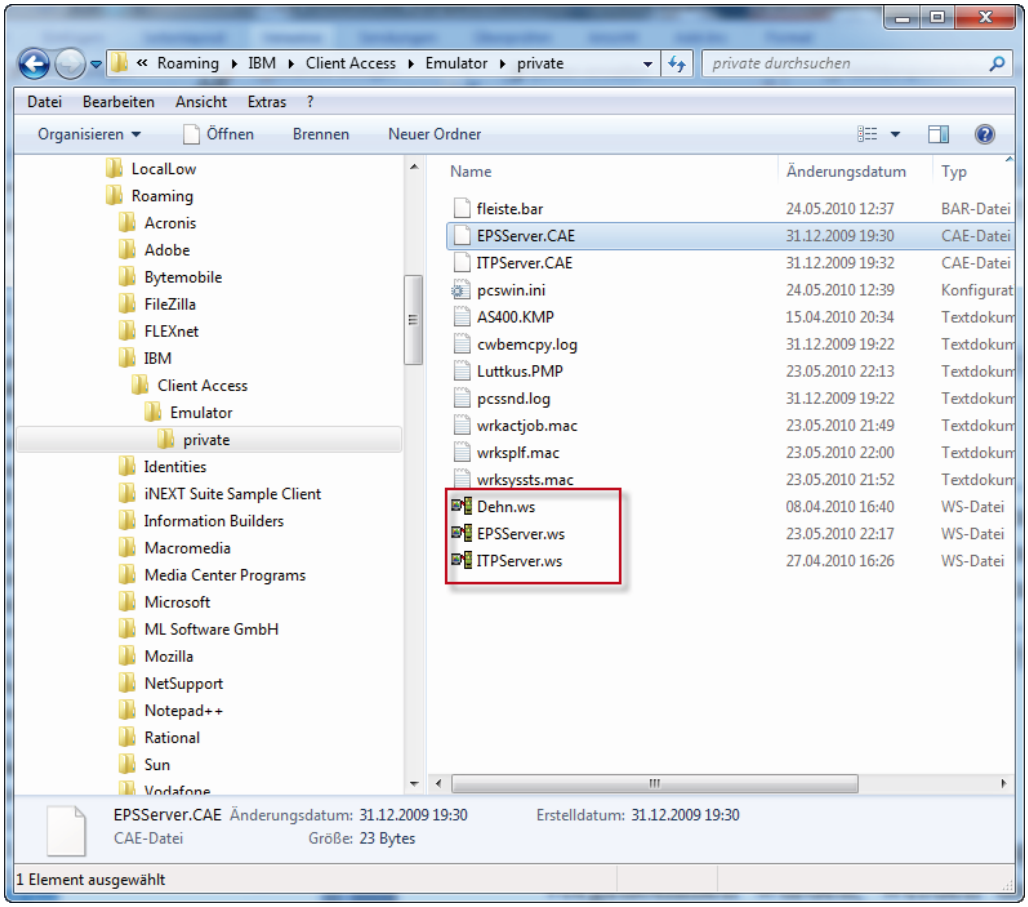


Das Programm PCSWS.EXE

2.) Da ja fast jeder Bildschirm individuelle Merkmale trägt, benötigt das Programm PCSWS.EXE eine Parameterdatei, die die individuellen Ausprägungen eines Bildschirms darstellt. Diese Datei wird als sogenanntes Bildschirmprofil bezeichnet und trägt die Namenserweiterung „*.WS“.

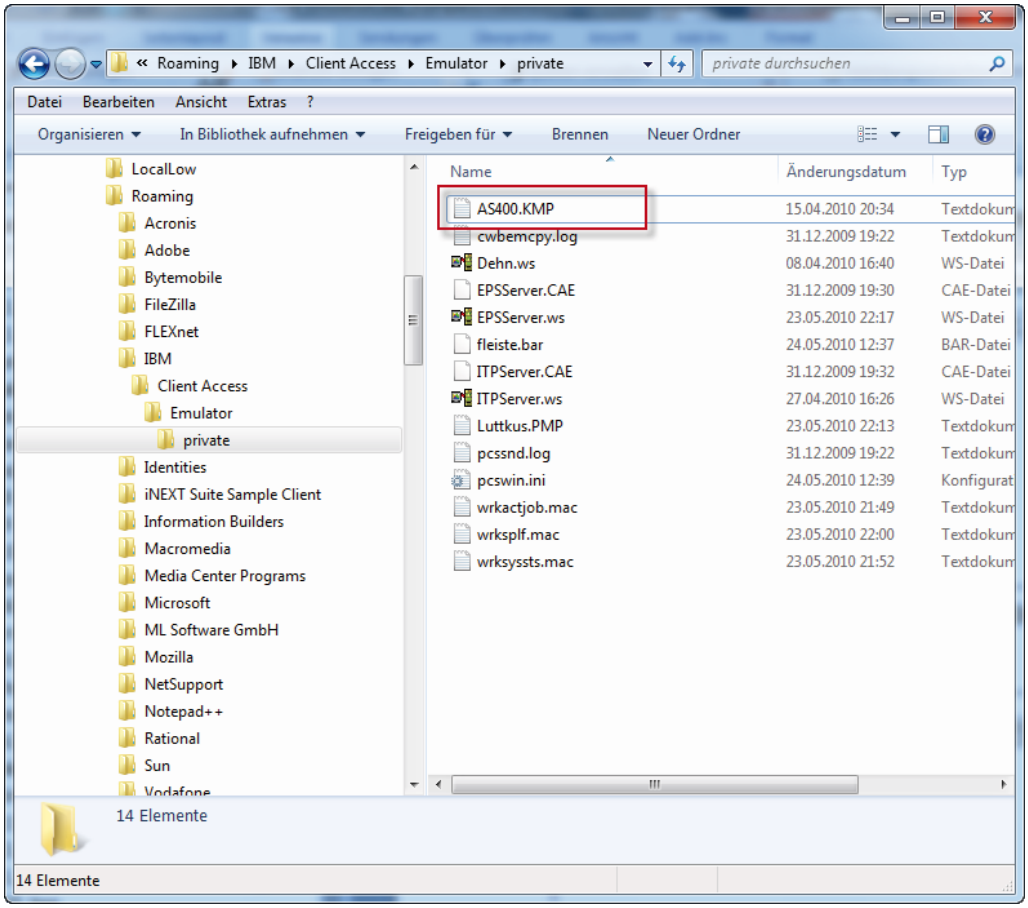
7.3.9.1.1
Seite 2

Bildschirmprofile werden im Standard-Pfad „.../*IBM/System i Access/Emulator/Private*“
gespeichert.



Bildschirmprofile

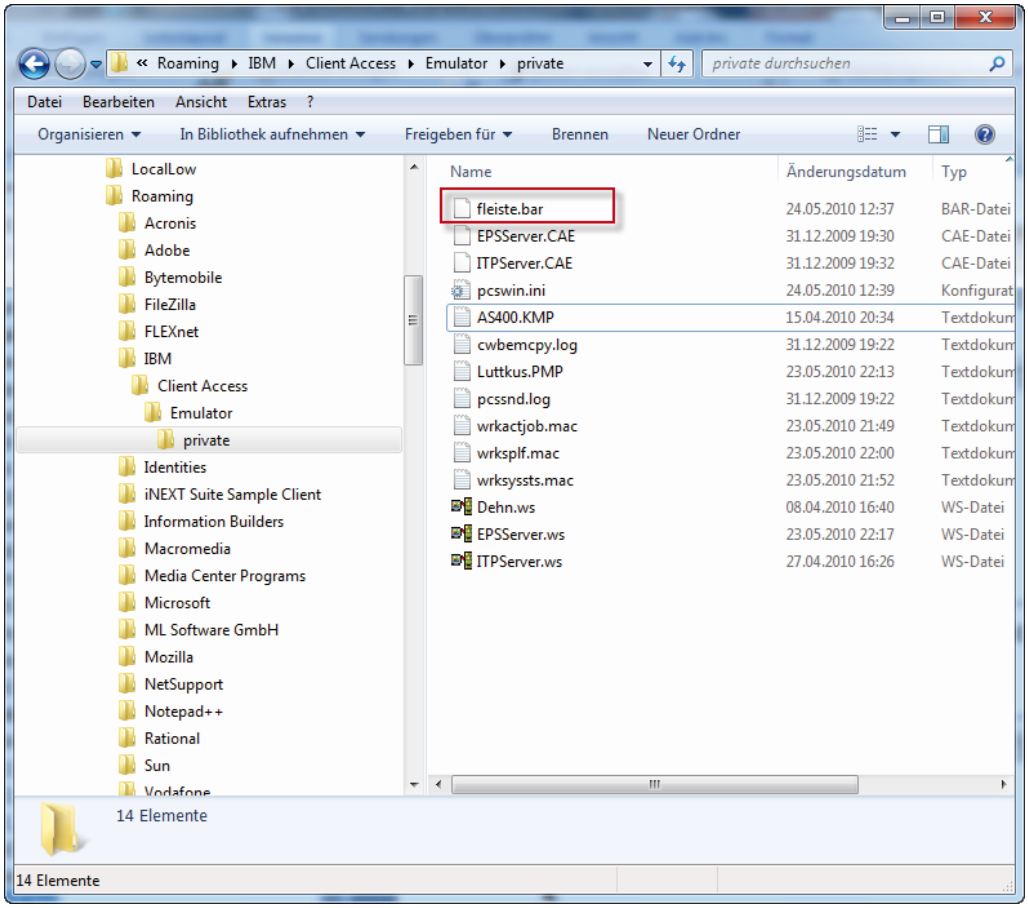
3.) Für den Fall, dass Anpassungen an der Tastatur vorgenommen wurden, werden diese Anpassungen in einem Tastaturprofil abgelegt. Tastaturprofile werden in Dateien mit der Endung *.KMP gespeichert. Das Tastaturprofil für das IBM-Tastenlayout, bei dem die Return-Taste des PCs in der Emulation die Feldausgangsfunktion wahrnimmt und die rechte STRG-Taste die Datenfreigabefunktion, ist eine Datei mit dem Namen „System i.KMP“. Diese Datei befindet sich ebenfalls im Pfad „.../*IBM/System i Access/Emulator/Private*“.



Tastaturprofile

4.) Weiterhin besteht die Möglichkeit, die Toolleiste einer Bildschirmemulation zu konfigurieren. Auch die Angaben zu einer modifizierten Toolleiste befinden sich in einer Profildatei. Diese Datei erkennen Sie an der Endung „*.BAR“. BAR-Dateien werden ebenfalls im Pfad „*.../IBM/System i Access/Emulator/Private*“ gespeichert.

7.3.9.1.1
Seite 4



Toolleistenprofile

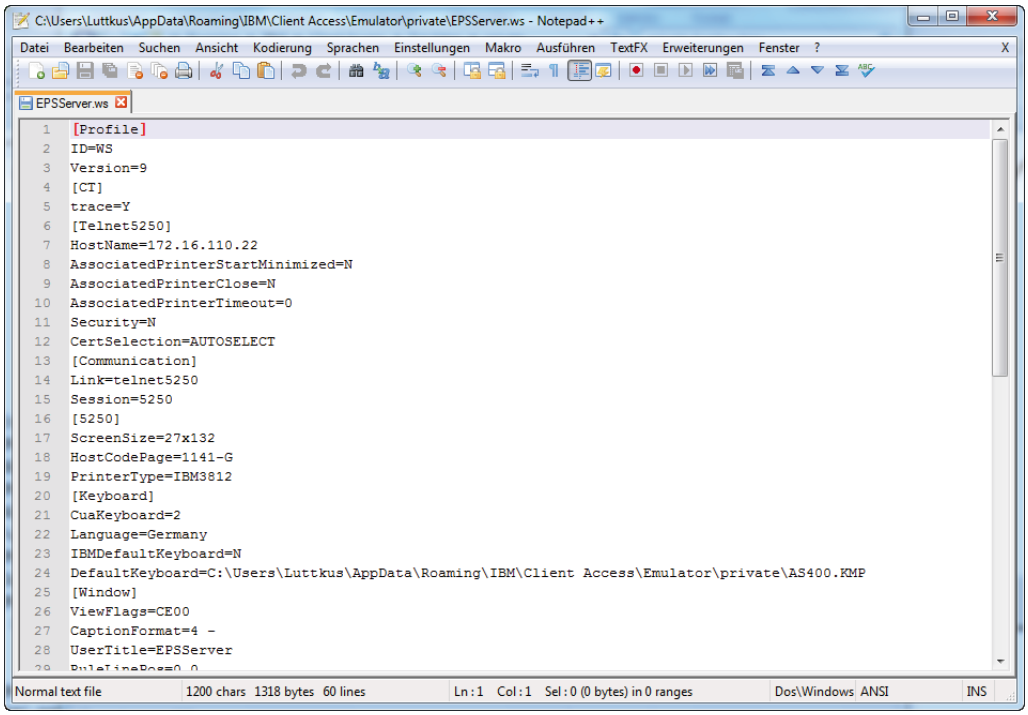
Der Aufruf einer Bildschirmemulation vom Desktop aus erfolgt über folgenden Pfad:

F:\programme\IBM\System i Access\Emulator>pcsws.exe
...\private\epsserver.ws

Sie sehen hier den Aufruf des Programms PCSWS.EXE und dass das Bildschirmprofil EPSSERVER.WS als Parameter mit kompletter Pfadangabe angegeben wird. Alle Einträge der Datei EPSSERVER.WS dienen als Eingabeparameter für das Programm PCSWS.EXE. Die oben genannten weiteren Profildateien werden nicht explizit aufgerufen, sondern über Parameter innerhalb des Bildschirmprofils an den Emulationsprozess angebunden.

Die für uns relevanten Fragen sind jetzt:
- Was steht in der Datei EPSSERVER.WS?
- In welchem Format wird diese Datei abgelegt?

Sehen wir uns doch einfach einmal die Datei EPSSERVER.WS mit einem Editor an:



Das Bildschirmprofil im Windows-Editor

Damit sind unsere beiden Fragen beantwortet. Eine WS-Datei ist nichts anderes als eine Windows-Initialisierungs-Datei (INI-Datei), die im Textformat Parameter enthält, die das Programm PCSWS.EXE steuern. Windows 3.1 lässt grüßen!

Die mit den Pfeilen bezeichneten Stellen werden als Paragraphen bezeichnet, die innerhalb der INI-Datei Blöcke bilden. Innerhalb eines Paragraphen finden wir dann Parameter, die formal nach folgendem Schema aufgebaut sind:

Parametername = Parameterwert

Ist der Parameterwert nicht eingetragen, so wird ein Defaultwert angenommen.

7.3.9.1.1
Seite 6

Zur Übersicht zeige ich Ihnen hier noch einmal die komplette Datei EPSSERVER.WS

```
[Profile]
ID=WS
Version=9
[CT]
trace=Y
[Telnet5250]
HostName=172.16.110.22
AssociatedPrinterStartMinimized=N
AssociatedPrinterClose=N
AssociatedPrinterTimeout=0
Security=N
CertSelection=AUTOSELECT
[Communication]
Link=telnet5250
Session=5250
[5250]
ScreenSize=27x132
HostCodePage=1141-G
PrinterType=IBM3812
[Keyboard]
CuaKeyboard=2
Language=Germany
IBMDefaultKeyboard=N
DefaultKeyboard=C:\Users\Luttkus\AppData\Roaming\IBM\Client Access\
Emulator\private\AS400.KMP
```

```
[Window]
ViewFlags=CE00
CaptionFormat=4 -
UserTitle=EPSServer
RuleLinePos=0 0
SessFlags=38862
[LastExitView]
A=3 113 36 976 609 3 12 20 400 0 IBM3270- 1141
[HotSpot]
Exists=Y
template1=PF PFnn
template2=PF FPnn
template3=PF Fnn
template4=URL
ShowItem=PF URL
[Colors]
ExtendedColorBlue=00A0A0 FFFFFF
ExtendedColorGreen=000000 FFFFFF
ExtendedColorPink=FF00FF FFFFFF
ExtendedColorRed=F01818 FFFFFF
ExtendedColorTurquoise=A000A0 FFFFFF
ExtendedColorWhite=7890F0 FFFFFF
ExtendedColorYellow=FFA200 FFFFFF
OIAColorInformationIndicators=000000
OIAColorBackground=FFFFFF
OtherScreenColor=FFFFFF
OtherRuleLine=A000A0
[Menu]
[Macro]
OpenMacro=wrksplf.MAC
[Poppad]
IBMDefaultPoppad=N
DefaultPoppad=C:\Users\Luttkus\AppData\Roaming\IBM\Client Access\
Emulator\private\Luttkus.PMP
```

7.3.9.1.1
Seite 8

```
PoppadLeft=720
PoppadTop=302
[Toolbar]
BarFile=C:\Users\Luttkus\AppData\Roaming\IBM\Client Access\Emula-
tor\private\fleiste.bar
```

Nehmen Sie sich einfach ein wenig Zeit, um diese WS-Datei durchzulesen. Viele der Parameter sind so benannt, dass Sie selbsterklärend sind.

Die für uns wichtigen Parameter habe ich in Fettdruck markiert.

Wie kommen wir jetzt konkret zur Lösung unseres Problems? Relativ einfach. Wir gehen davon aus, dass auf jedem in Frage kommenden PC System i Access installiert ist. Das bedeutet, dass das Programm PCSWS.EXE auf jedem PC vorhanden ist.

Wir werden uns also

1.) an unseren Arbeitsplatz begeben und dort einen oder mehrere Modell-Bildschirme konfigurieren.

2.) Dann kopieren wir die dort entstandenen WS-Dateien mehrfach.

3.) Zu guter Letzt ändern wir mit einem Texteditor nur noch die für unser Vorhaben relevanten Parameter in der WS-Datei ab und verteilen dann

4.) diese WS-Dateien über unser Netzwerk.

Modellbildschirme konfigurieren

Bevor wir uns an die Konfiguration machen noch einige Worte zu den Modellbildschirmen. Wir gehen von folgenden Voraussetzungen aus:

1.) Jeder Anwender bekommt zwei Bildschirmsitzungen. Diese haben jeweils einen fest vergebenen Namen. Bildschirmnamen werden nach folgender Logik gebildet:

Stelle 1 + 2: PC

Folgende Stellen: Namenskürzel, in meinem Fall KPL

Danach folgende zwei Stellen: Folgenummer

Daraus ergibt sich, dass die für mich konfigurierten zwei Bildschirme den Namen PCKPL01 bzw. PCKPL02 tragen müssen.

2.) Der erste Bildschirm weist immer die Standardfarbe auf. Der zweite Bildschirm ist monochrom und hat einen weißen Hintergrund mit schwarzer Schrift.

3.) Als Tastaturlayout wird die Standard-IBM-Belegung gewünscht, bei der die Datenfreigabe auf der rechten STRG-Taste der PC-Tastatur liegt.

4.) Die Toolleiste soll die dargestellten Funktionsknöpfe möglichst klein anzeigen.

Beginnen wir also mit der Beispielkonfiguration des ersten Bildschirms. Da dies ein Modellbildschirm ist, soll er den Namen MODEL01 tragen. Ich rufe also das Bildschirmkonfigurationsprogramm aus der Programmgruppe „System i Access“ und dem dort befindlichen Unterordner „Emulation“ auf.

Ich habe hier die für meine Umgebung wichtigen Werte eingetragen:

Systemname: EPSSERVER.EPOSDOM.LOCAL

Achtung:

Das ist nicht der Systemname, wie wir ihn in den Netzwerkattributen der System i (DSPNETA) finden, sondern der TCP/IP-Host-Name, der der IP-Adresse der System i 172.16.110.22 zugeordnet wurde. Die Verwendung von Hostnamen setzt voraus, das Sie mit einer Namensauflösung arbeiten, also entweder mit Hosttabellen oder DNS-Diensten . Es ist durchaus möglich, hier hart die IP-Adresse der System i einzutragen.

Datenstations-ID: MODEL01

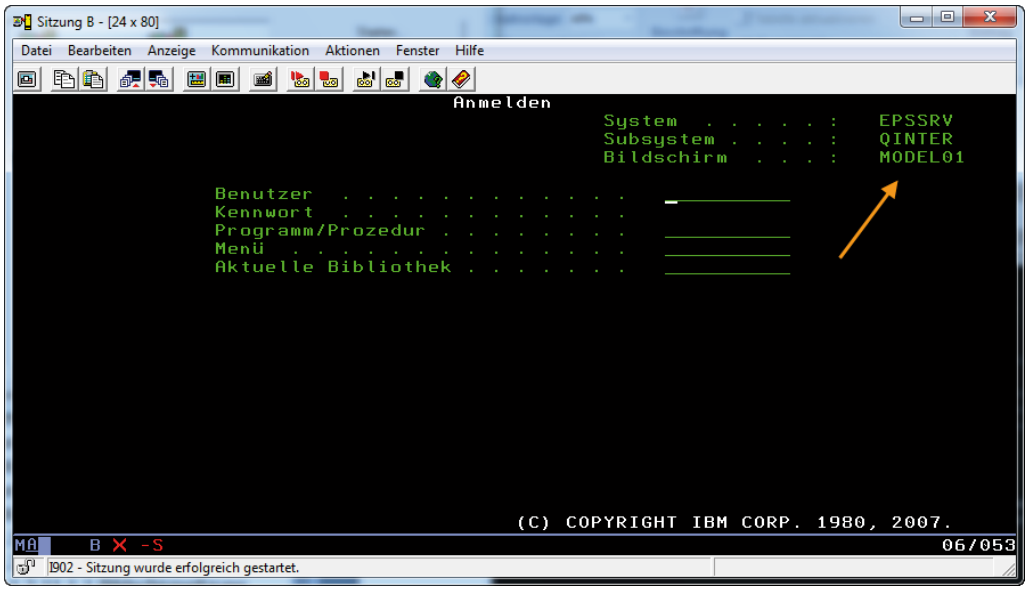
Dieser Eintrag bewirkt, dass bei der ersten Kontaktaufnahme mit dem System eine Bildschirmeinheit mit dem Namen MODELL01, *DEVD gebildet wird.

7.3.9.1.2

Seite 2

Die „Auflösung“ des Bildschirms habe ich dann noch auf 27x132 Zeichen gesetzt.

Ich bestätige den Dialog mit „OK“.



Die Verbindung konfigurieren

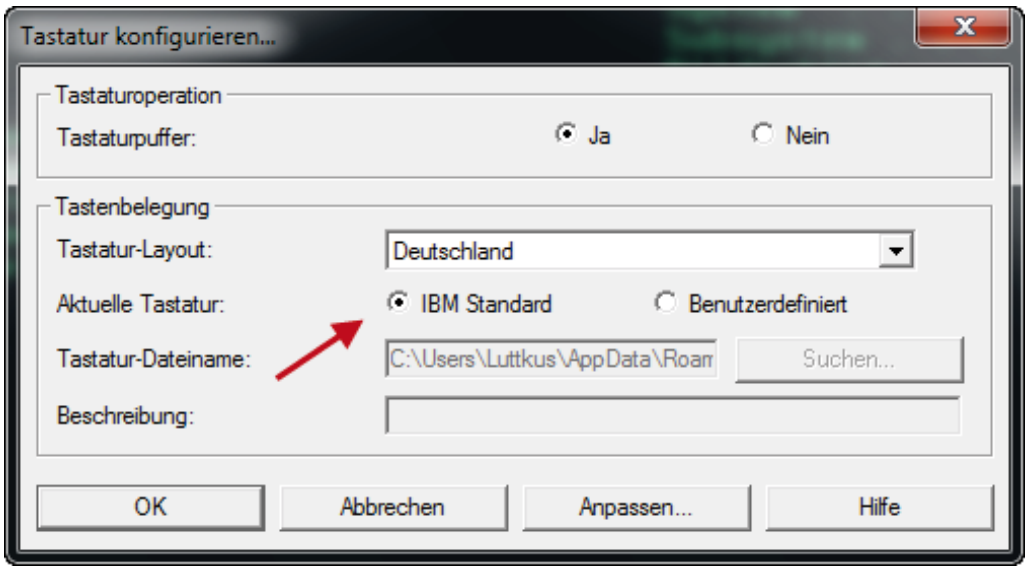
Waren alle Angaben korrekt, so wird die erste Verbindung zum System durchgeführt. Sie sehen, dass der Bildschirm den Namen MODEL01 trägt. So weit, so gut.

Wir könnten uns jetzt sicherlich um das kümmern, was ich immer die Kosmetik eines Bildschirms nenne, also z. B. um Cursordarstellung, Titelleistentext usw. usf. Wir werden jedoch die für uns wichtigen Punkte erledigen. Und der nächste wichtige Punkt besteht darin, die Tastaturbelegung auf die IBM-

Standardbelegung umzustellen.

2.) Dazu öffnen wir das „Menü Bearbeiten ➜ Einstellungen ➜ Tastatur...“

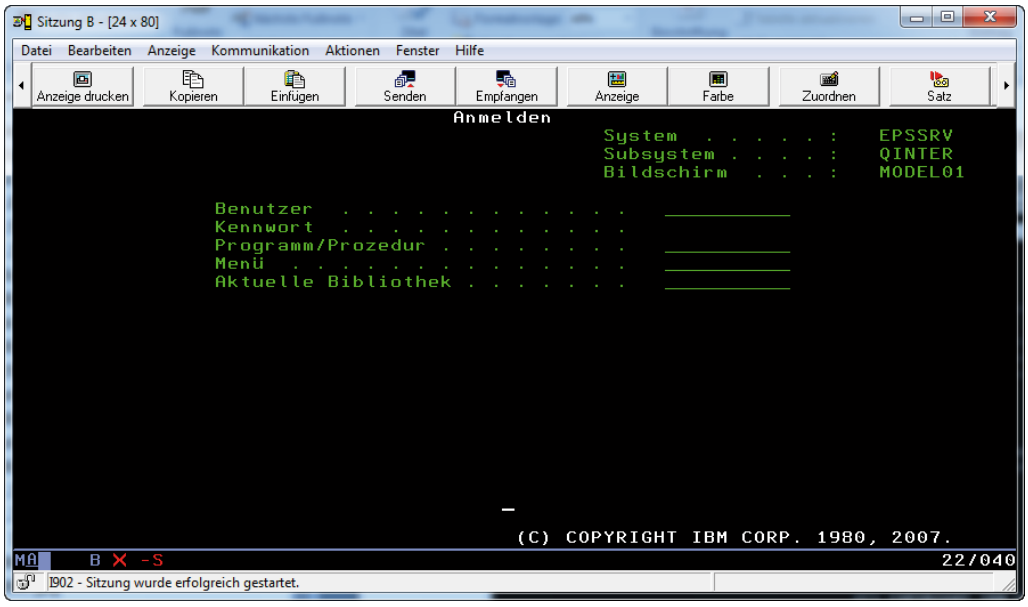
Hier gehen wir in die Auswahl „Tastatur konfigurieren“.



Die Tastatur konfigurieren

Hier habe ich den Parameter „Aktuelle Tastatur“ auf den Wert „IBM Standard“ gestellt. Dadurch wird das Tastaturprofil „AS400.KMP“ an diesen Bildschirm gebunden. Ich beende den Dialog mit „OK“.

3.) Nun kümmere ich mich um die Toolleiste im Emulationsbildschirm.



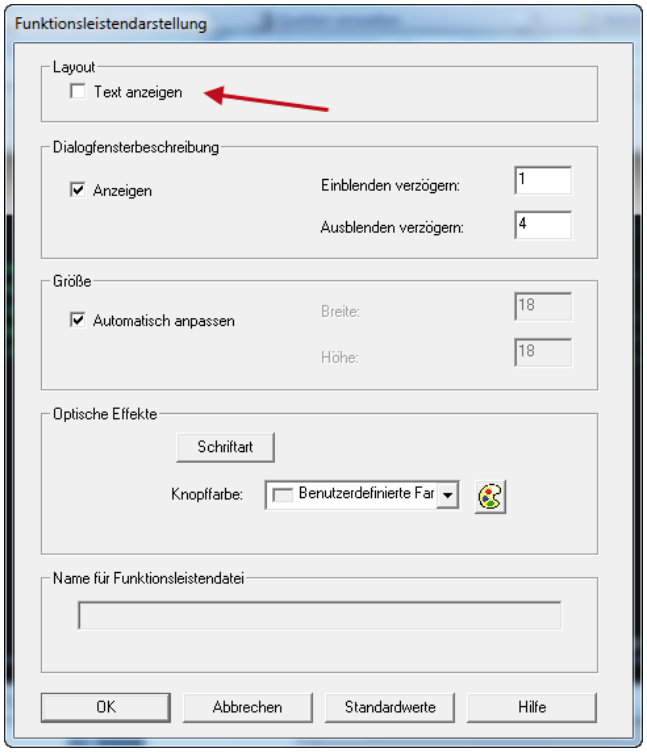
Die Toolleiste konfigurieren (1)

Diese Toolleiste verbraucht viel Platz in der Anzeige. Was die Knöpfe der

7.3.9.1.2
Seite 4

Toolleiste so groß erscheinen lässt, ist der erläuternde Text unterhalb des Bildchens.

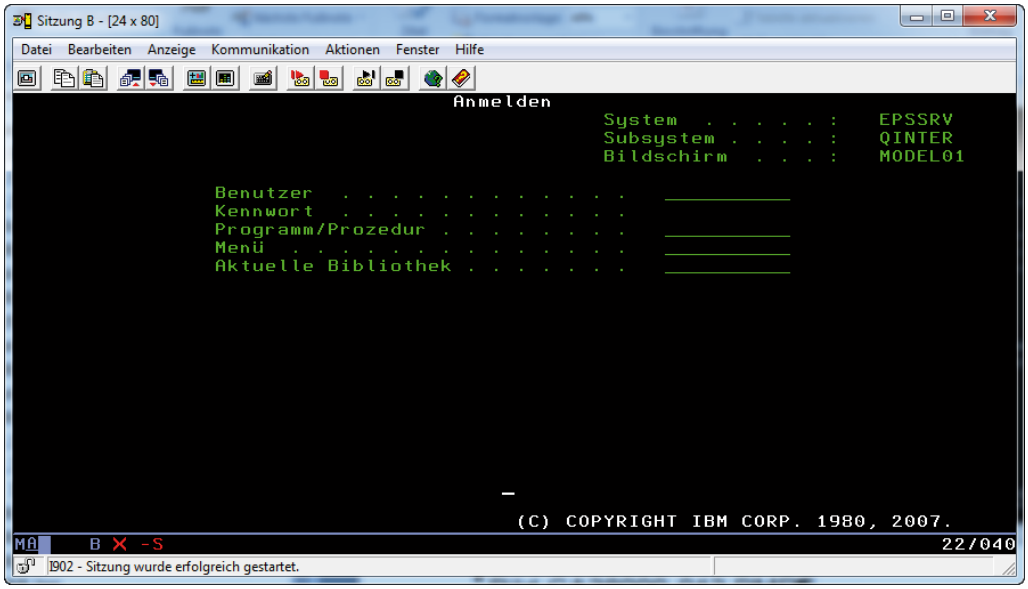
Ich werde diesen erläuternden Text wegkonfigurieren, indem ich in die Menüfolge „Funktionsleistendarstellung...“ wechsle.



Die Toolleiste konfigurieren (2)

Hier klicke ich einfach das Häkchen „Text anzeigen“ im Rahmen „Layout“ weg und beende den Dialog mit „OK“.

Daraufhin erscheint wieder der Emulationsdialog mit der veränderten Toolleiste.

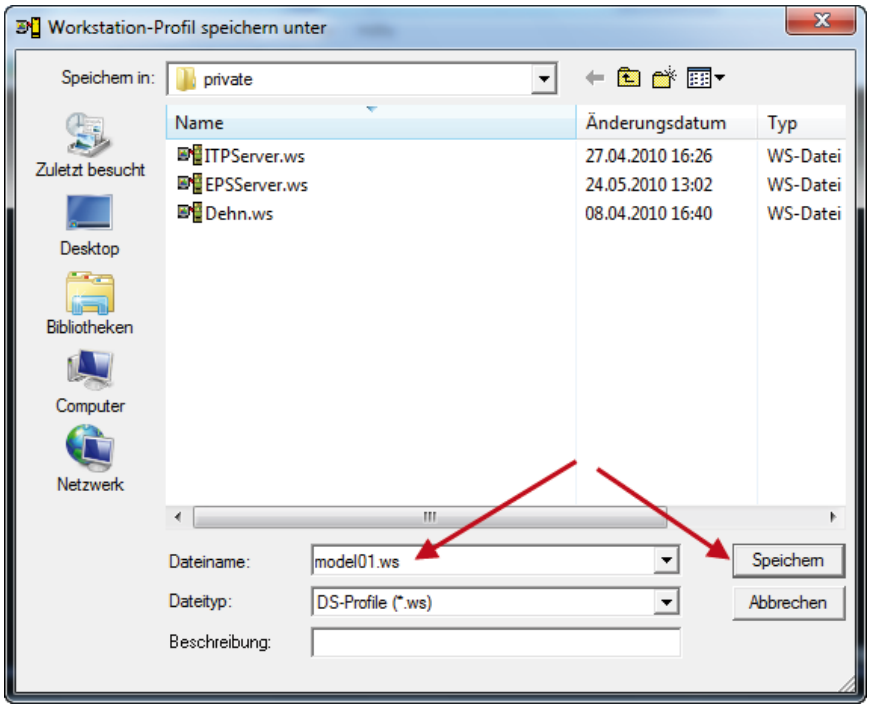


Die Toolleiste konfigurieren (3)

Sie sehen, dass die Knöpfe der Toolleiste jetzt deutlich kleiner geworden sind.

Nun geht es auf das Ende zu. Wir müssen die Konfiguration des Bildschirms speichern.

4.) Dazu gehen wir in die Menüfolge „Datei ➜ Sichern als...“



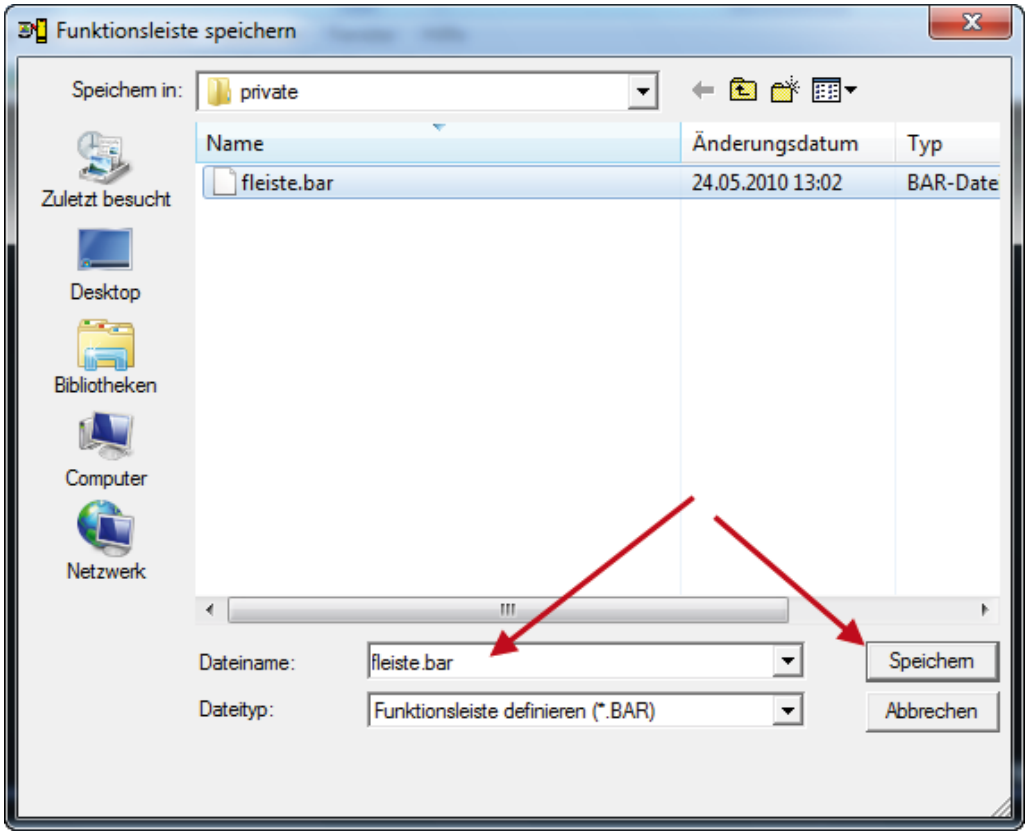
Bildschirmprofil sichern

In diesem Schritt wird das Bildschirmprofil – die WS-Datei – erstellt. Ich gebe

7.3.9.1.2
Seite 6

dem Bildschirmprofil den Namen MODEL01.WS.

Da ich das Toolleistenlayout verändert habe, erscheint nach dem Sichern des Bildschirmprofils automatisch der Sicherungsdialog zum Wegspeichern des Toolleistenlayouts.



Toolleistenprofil sichern

Ich sichere das Tooleistenlayout unter dem Namen FLEISTE.BAR.

Im Folgenden erstelle ich kein Symbol für den Bildschirm, sondern belasse es bei den gezeigten Sicherungsvorgängen.

Die Konfiguration des monochromen Bildschirms verläuft entsprechend. Er unterscheidet sich nur durch den Farbkonfigurationsdialog von dem hier gezeigten Bildschirm.

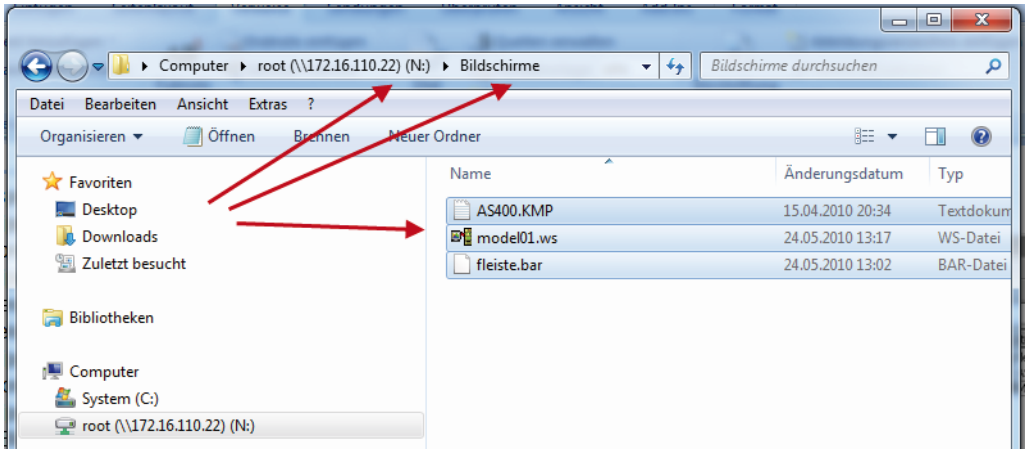
Damit ist das von uns benötigte „Rohmaterial“ beschafft. Wir sind jetzt im Besitz eines Bildschirmprofils (MODEL101.WS), eines Tastaturprofils (System i.KMP) und eines Toolleistenprofils (FLEISTE.BAR)

Aus diesem „Rohmaterial“ werden jetzt alle weiteren Bildschirme hergestellt.

Bildschirme klonen

Ich gehe nun davon aus, dass ich über einen Fileserver im Netzwerk verfüge. Das kann ein Windows Server, eine Workstation, aber auch eine System i mit konfiguriertem und laufendem Netserver sein.

Auf meinem System verfüge ich über ein Netzwerklaufwerk N:, und in diesem Laufwerk habe ich einen Ordner „Bildschirme“ angelegt. In diesen Ordner habe ich die Dateien MODELL0.WS, System i.KMP und FLEISTE.BAR hineinkopiert.



Profile für das Klonen von Bildschirmen

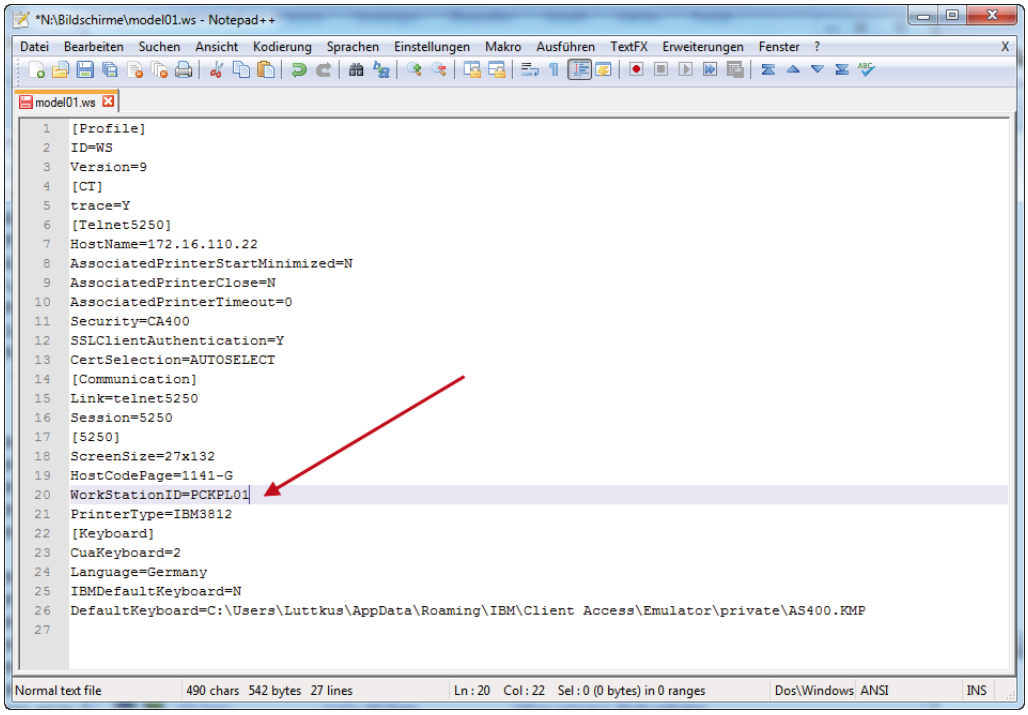
Um jetzt weitere Bildschirme herstellen zu können, müssen wir uns überlegen, welche Parameter des Bildschirmprofils MODEL01.WS zu verändern sind.

Folgende Parameter kommen primär in Betracht:
- HostName
- WorkStationID
- DefaultKeyboard
- BarFile

In unserem Fall kann der Parameter „HostName“ unverändert übernommen werden.
- Der Parameter „WorkStationID“ wird auf den jeweiligen Bildschirmnamen geändert.
- Für den Parameter „DefaultKeyboard“ muss der Zugriffspfad auf das Tastaturprofil auf den Netzwerkpfad eingestellt werden:
  - N:\Bildschirme\ AS400.KMP
- Analog muss der Parameterwert für den Parameter „BarFile“ geändert werden:
  - N:\Bildschirme\ fleiste.bar

Ich lade jetzt die Datei MODEL01.WS in den Editor meines Windows-PCs:

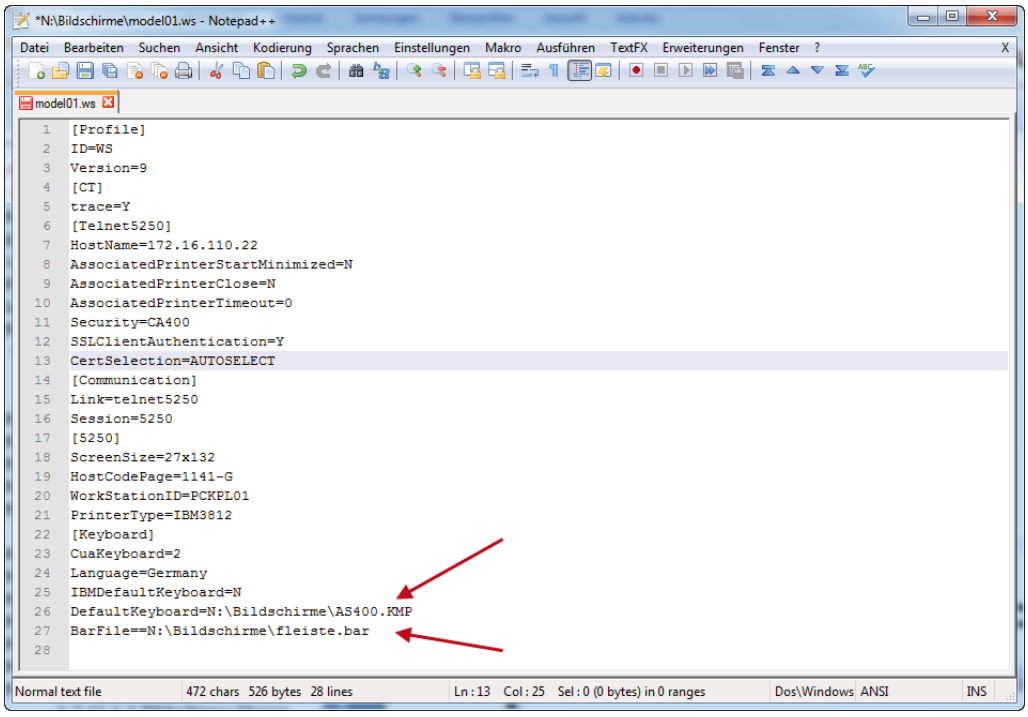
7.3.9.1.3
Seite 2



WorkStationID ändern

Hier ändere ich gemäß der von uns entwickelten Konvention den Parameterwert im Parameter „WorkStationID“ von MODELL01 in PCKPL01.

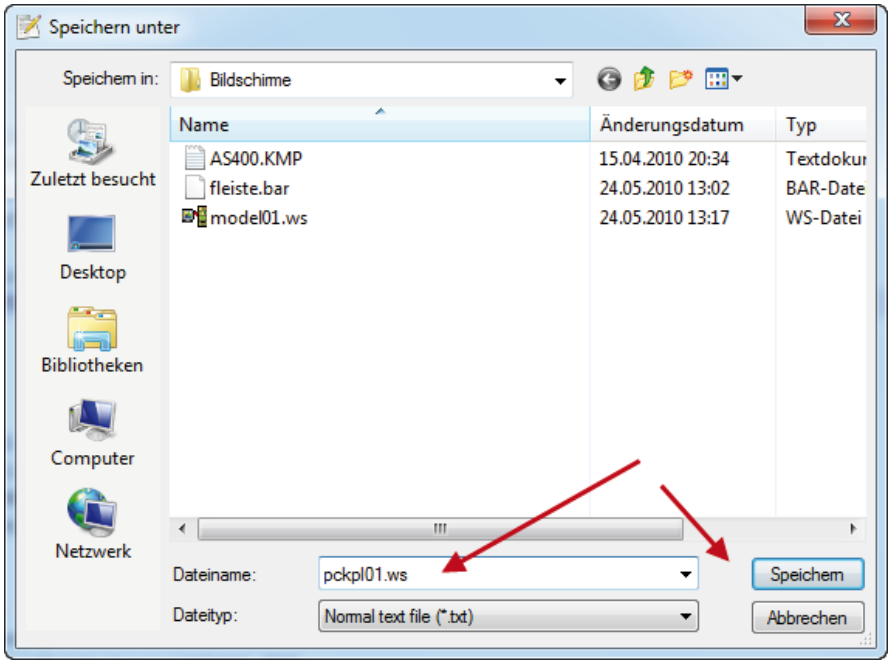
Danach ändere ich die Pfade für die restlichen beiden Parameter.



DefaultKeyboard und Barfile ändern

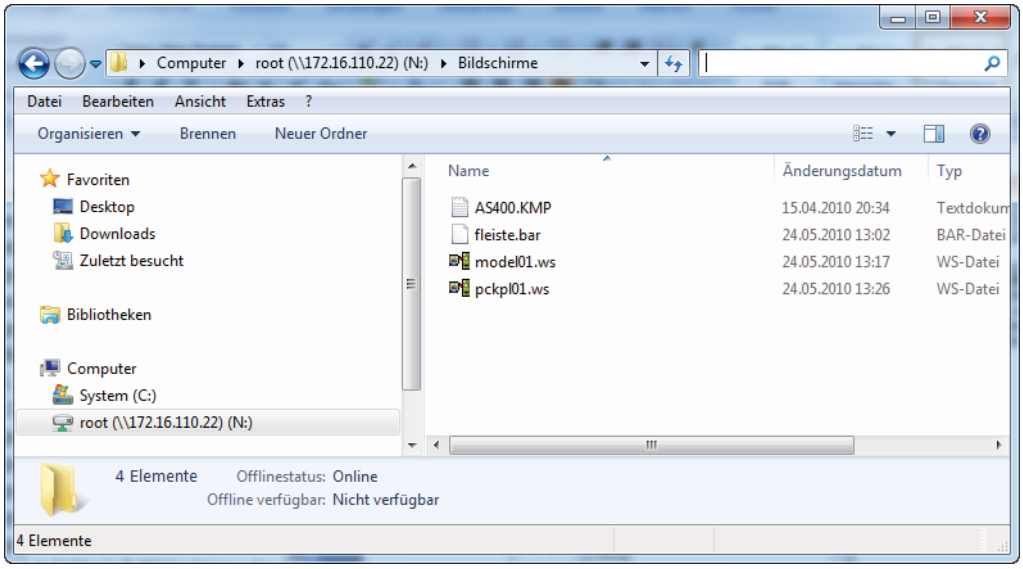
Ich trage nun für den Parameter „DefaultKeyboard“ den Pfad „n:\Bildschirme\AS400.KMP“ ein und für den Parameter „BarFile“ den Pfad „n:\Bildschirme\fleiste.bar“.

Dann wird das Profil mit dem Namen PCKPL01.WS gespeichert.



DefaultKeyboard und BarFile ändern

Sie sehen jetzt im Explorer, dass mir ein neues Bildschirmprofil zur Verfügung steht.



Der neue Bildschirm ist fertig!

Für jeden weiteren Bildschirm, der zu klonen ist, wiederhole ich den beschriebenen Vorgang.

7.3.9.1.3

Seite 4

Bildschirme testen

Jetzt teste ich die auf diese Weise neu erstellten Bildschirme, indem ich auf meinem Arbeitsplatz in das DOS-Fenster wechsle und händisch die Bildschirmfunktion ausführe.

Ich rufe den Bildschirm wie folgt auf:

E:\>pcsws n:\bildschirme\pckpl01.ws

Alternativ ist auch folgender Aufruf möglich:

E:\>pcsws /Q n:\bildschirme\pckpl01.ws

Der Parameter /Q bewirkt, dass der sogenannte Splash-Screen mit der Maske nicht mehr angezeigt wird, wenn der Bildschirm erstmalig startet.

7.3.9.1.4

Seite 2

Abwicklung des Bildschirm-Rollouts

Nachdem auf dem beschriebenen Weg alle Profile erzeugt worden sind, müssen im Windows-Umfeld der jeweiligen Arbeitsstation Verknüpfungen zu den Profilen eingerichtet werden, so dass jeder Anwender seine Bildschirme aufrufen kann.

Weitere Möglichkeit:

- Legen Sie auf dem System i eine Datei an, die die Bildschirmnamen, Pfadangaben für die zu erzeugenden Bildschirme und Profile enthält.
- Verknüpfen Sie via ODBC diese Datei mit Microsoft Access oder Excel.
- Schreiben Sie dann mit Visual Basic für Applications (VBA) eine kleine Anwendung, die die in der Datei gespeicherten Bildschirmnamen und sonstigen Daten automatisch in neue Bildschirmprofile einfügt.

Sie sehen, Ihre Phantasie ist gefordert! Krempeln Sie die Ärmel hoch, es gibt noch viel zu tun.

7.3.9.1.5

Seite 2

System i Access – emuliertes Drucken

Die Funktion des emulierten Druckens gehört zum Funktionsumfang des Programms PC5250. Emuliertes Drucken beschreibt ein Verfahren, bei dem ein lokaler PC-Drucker als System i-Systemdrucker verwendet wird. Definieren wir also einen PC-Drucker als emulierten Drucker für das System i, so wird auf der System i eine Ausgabewarteschlange für diesen Drucker eingerichtet, so dass ein System i-Prozess diesen Drucker verwenden kann.

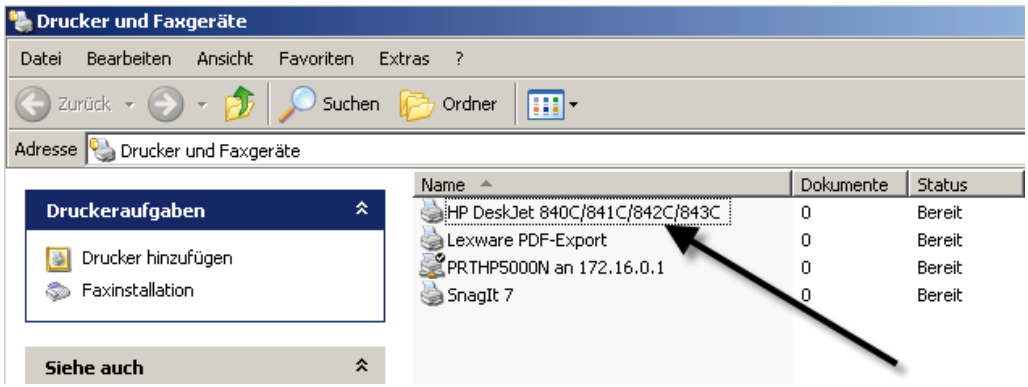
Wesentlich für eine erfolgreiche Installation ist der Sachverhalt, dass an dieser Stelle zwei Druckprozesse miteinander kommunizieren müssen. Einerseits der System i-Prozess, der das gesamte Spooling steuert und leitet, andererseits der Windows-Druckprozess, der dann das physikalische Drucken übernimmt.

Daraus ergibt sich die Voraussetzung, dass auf dem Windows PC bereits ein Drucker – entweder direkt oder als Netzwerkdrucker – installiert wurde, der dann mit dem System i-Druckprozess verbunden werden muss.

Die Konfiguration des emulierten Druckers entspricht vom allgemeinen Vorgehen her der Konfiguration eines Bildschirms.

Im ersten Konfigurationsschritt sehen wir uns den Windows-Arbeitsplatz an, um den als emulierten Drucker zu konfigurierenden lokalen Drucker zu identifizieren. Dazu wechsle ich in die Windows-Drucker-Anzeige:

Start ➜ Einstellungen ➜ Systemsteuerung ➜ Drucker und Faxgeräte



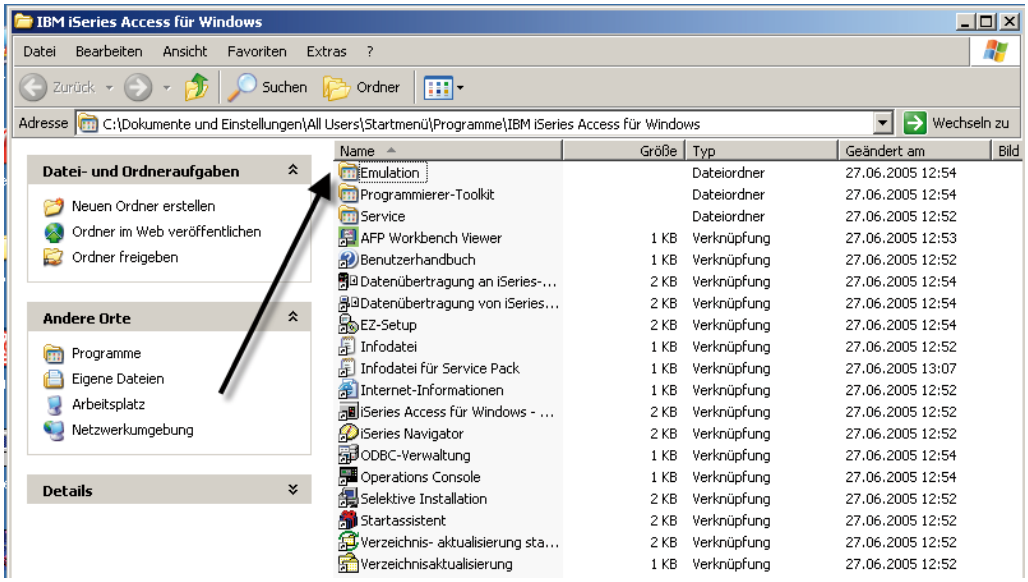
Drucker auswählen

Ich wähle aus den in Windows zur Verfügung stehenden Druckern das Gerät HP Deskjet 840C aus.

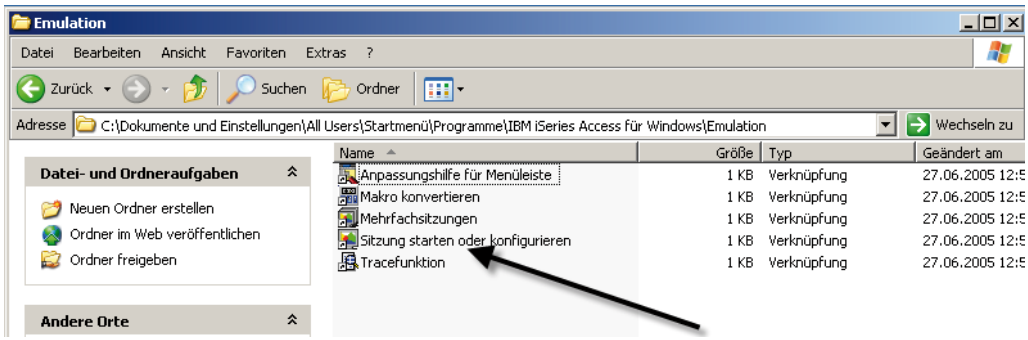
Ich beabsichtige also, die System i-Ausgaben lokal auf meinen HP Deskjet-Drucker umzuleiten.

Im nächsten Arbeitsschritt wird im System i Access der emulierte Drucker definiert. Dazu wechsle ich in die IBM System i Access für Windows-Programmgruppe, dort in das Verzeichnis „Emulation“.

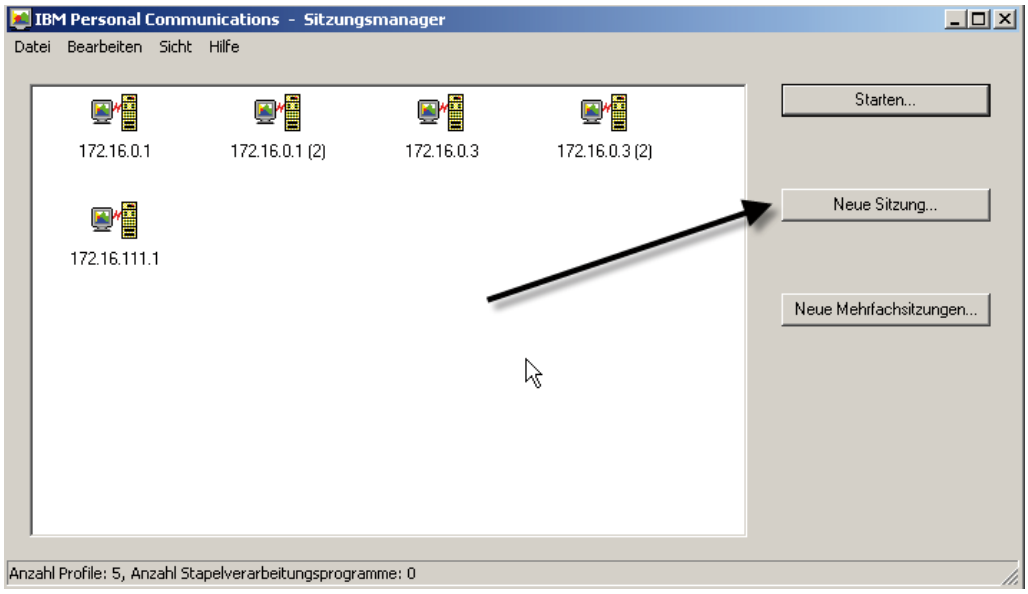
7.3.10
Seite 2



Ins Verzeichnis „Emulation“ wechseln

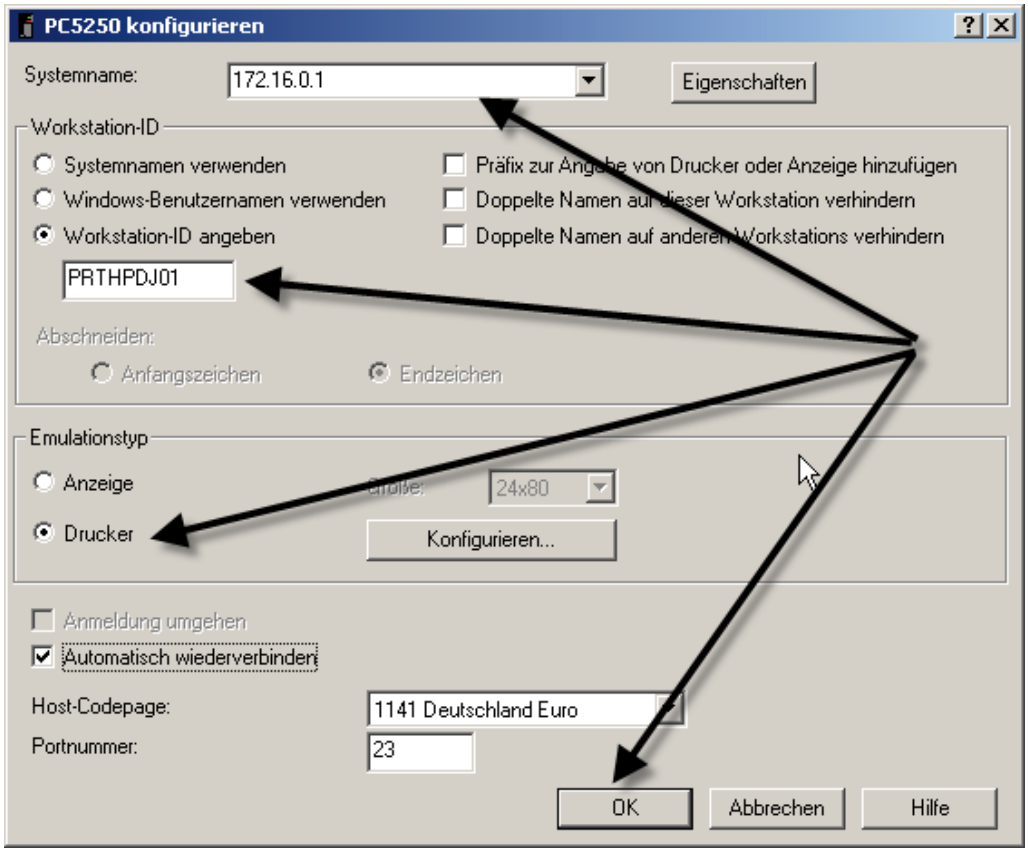


Emulierte Drucker: Beginn der Konfiguration



Neue Sitzung erstellen

Die Konfiguration der Emulation beginnt damit, dass ich eine neue Sitzung starte. Im folgenden Konfigurationsdialog sind dann die Verbindungsparameter zu konfigurieren.



Verbindungs- und Geräteparameter

Folgende Parameter sind zu versorgen:

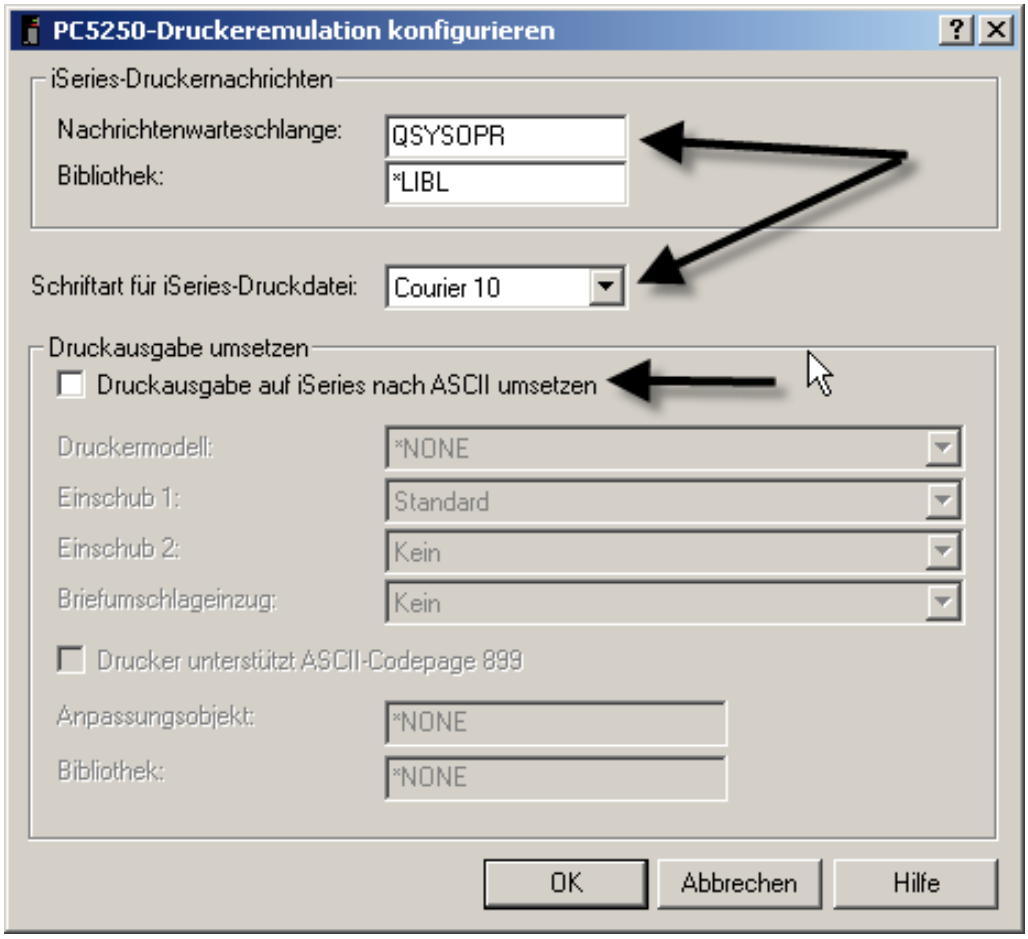
Systemname: Hier trage ich die IP-Adresse der System i ein, deren Ausgaben an diese Druckersitzung geleitet werden sollen.

Workstation ID angeben: Hier trage ich den Namen der auf der System i zu erstellenden Ausgabewarteschlange/des Druckers ein.

Drucker: Diesen Parameter markiere ich, weil ich einen emulierten Drucker erstellen will.

Drücken Sie dann den Knopf „Konfigurieren...“.

7.3.10
Seite 4



System i-Druckangaben einstellen

An dieser Stelle können Sie folgende Werte einstellen:

System i-Druckernachrichten

An welche Nachrichtenwarteschlange sollen Nachrichten, die von diesem Drucker abgehen, gesendet werden? Default-Einstellung ist hier die Nachrichtenwarteschlange des QSYSOPR der System i. Da es sich um einen arbeitsplatzbezogenen Drucker handeln kann, sollten Sie überlegen, ob Sie an dieser Stelle nicht die Nachrichtenwarteschlange dieses Bildschirms eintragen.

Schriftart für System i-Druckdatei

Sie Standardschriftart, die für diesen Druckprozess von Seiten der System i eingesetzt wird.

Druckausgabe umsetzen

Wenn Sie diese Option markieren, übernimmt die System i weiterführende Steuerungsfunktionen.

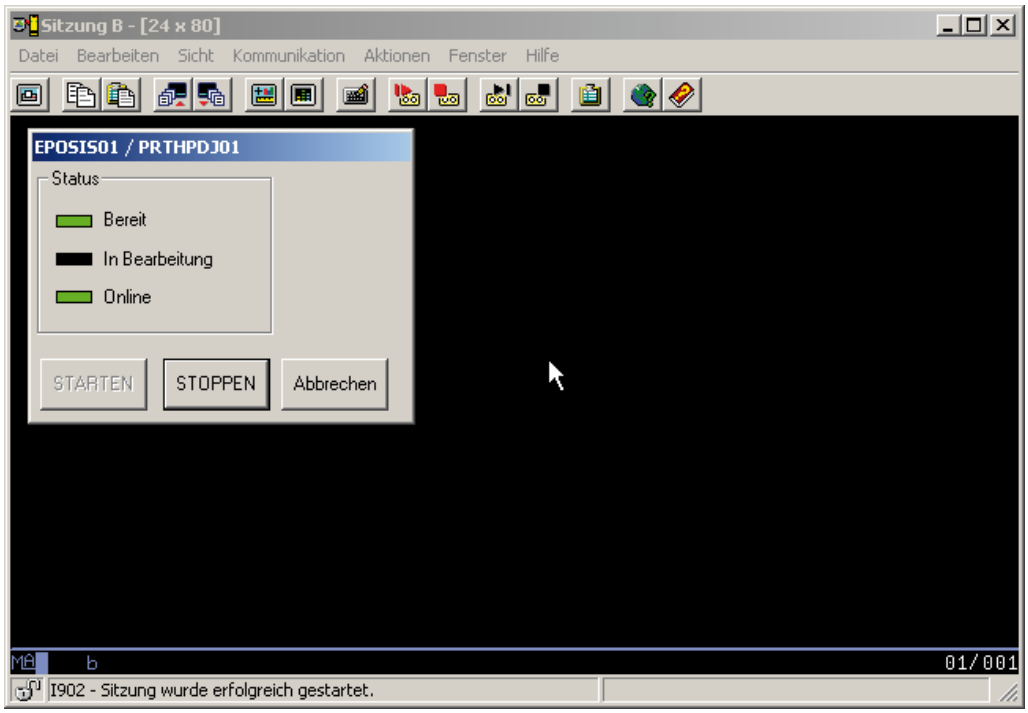
In diesem Bereich stellen Sie z. B. ein, wo die Zeichenumsetzung stattfinden soll. Auf dem PC oder auf der System i. Markieren Sie die Option „Druckaus-

gabe im System i umsetzen“, so übernimmt die System i für diesen Drucker die gesamte Zeichenumsetzung von EBCDIC nach ASCII. Aber Vorsicht, sollten Sie diese Option großflächig einstellen, also nicht nur für einen Drucker, so wird die Performance des System i stark davon beeinflusst. In einem Testlauf sollten Sie vorher die Auswirkung auf den restlichen Systembetrieb überprüfen.

Ansonsten lässt sich an dieser Stelle einstellen, welcher System i-Druckertyp emuliert wird und welche Einzüge an diesem Drucker zur Verfügung stehen.

Bestätigen Sie die hier vorgenommenen Einstellungen mit „OK“ und beenden Sie auch den Gesamtdialog mit „OK“.

Nach einiger Zeit erscheint dann folgendes Fenster:

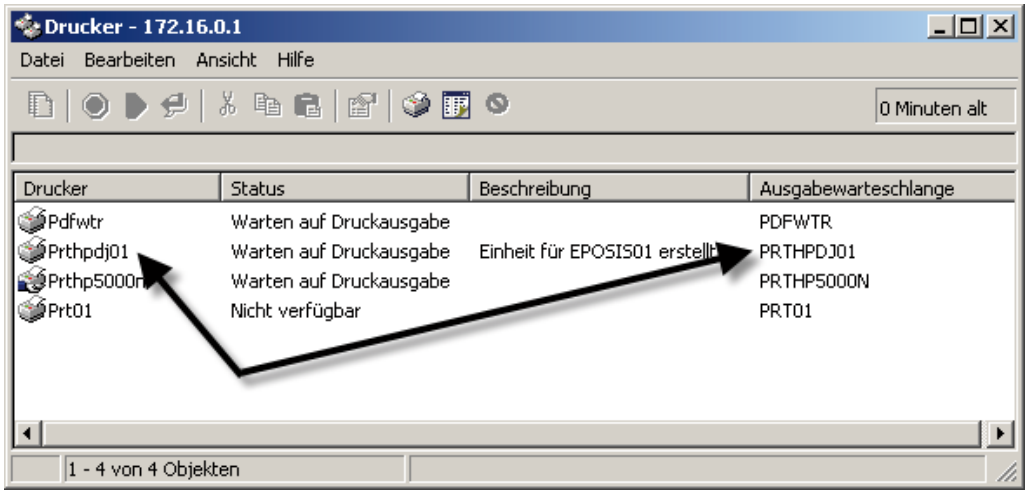


Das Bedienungsfeld des emulierten Druckers

Sie sehen hier das Bedienungsfeld des emulierten Druckers, das im Wesentlichen so aufgebaut ist wie das Bedienungspanel eines Druckers. Sie können den Druck starten bzw. anhalten und sehen auch, ob gerade ein Druckjob bearbeitet wird.

Für diesen emulierten Drucker ist auf dem System i eine Druckereinheit und eine Ausgabewarteschlange erzeugt worden.

7.3.10
Seite 6



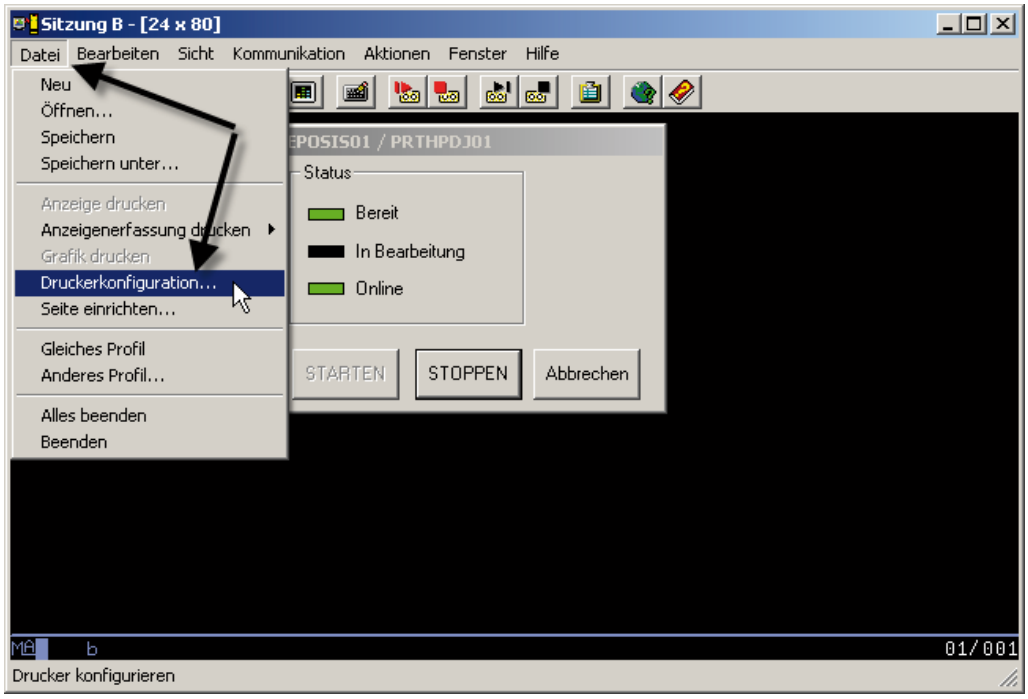
Anzeige der System i-Drucker

In der oben gezeigten Abbildung sehen Sie die auf dem System i neu erstellten Druckobjekte.

Damit ist die System i-seitige Konfiguration beendet.

Im nächsten Schritt wird die Verbindung zum Windows-Druckprozess hergestellt.

Hinter dem Bedienfeld des Druckers sehen Sie einen leeren System i-Bildschirm. Die in diesem Bereich dargestellten Menüs dienen der Konfiguration des emulierten Druckers. Sie gehen in das Menü „Datei“ und wählen den Menüpunkt „Druckerkonfiguration“ aus.



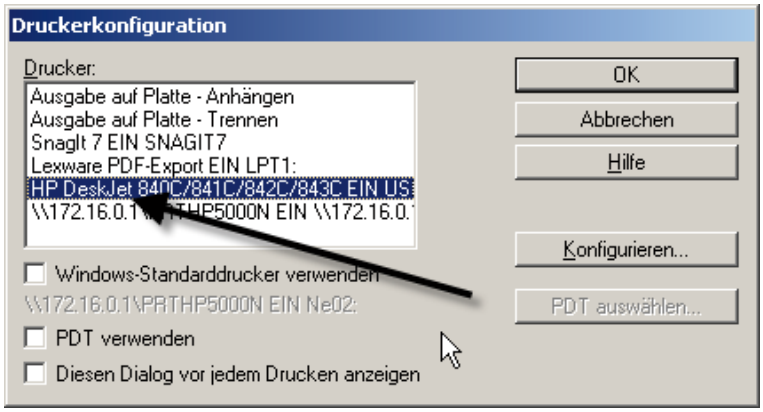
Verbindung zum Windows-Drucker definieren

Die Verbindung lässt sich auf zwei Arten konfigurieren.

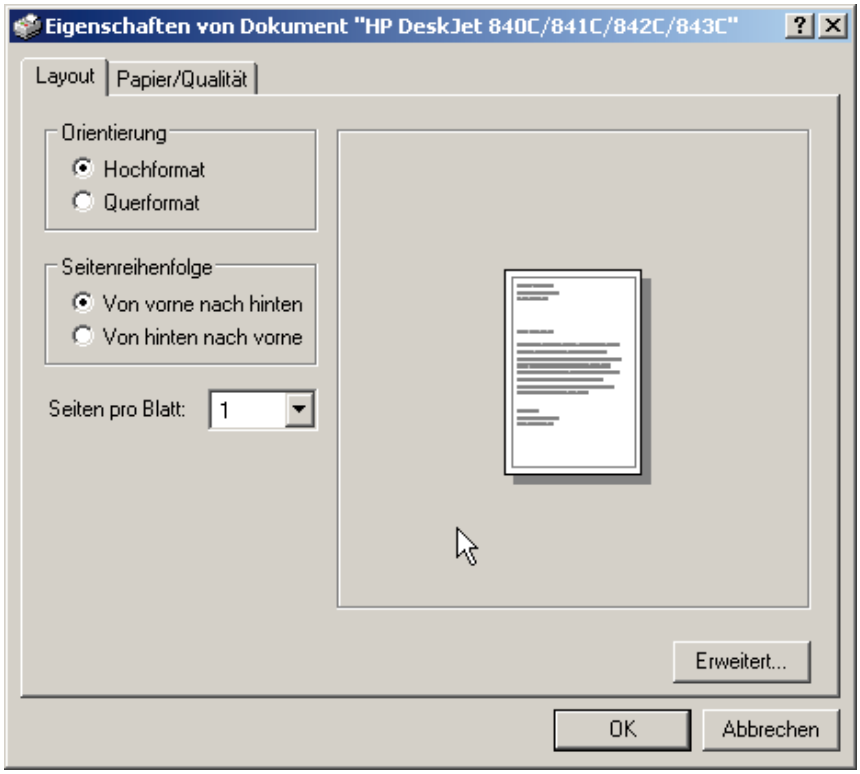
1.) Wählen Sie im folgenden Dialog den Windows-Drucker aus. In diesem Dialogfenster werden alle mit diesem Windows-PC verbundenen Drucker angezeigt:

- Lokale Drucker: erkennbar an der Zuordnung zur Hardwareschnittstelle – LPT1, LPT2 etc.
- Netzwerkdrucker: erkennbar an der Netzwerkressourcenzuordnung mittels des UNC-Namens – \\System\Ressource

In meinem Fall sollte der lokale Drucker „HP DeskJet 840C“ angeschlossen werden.



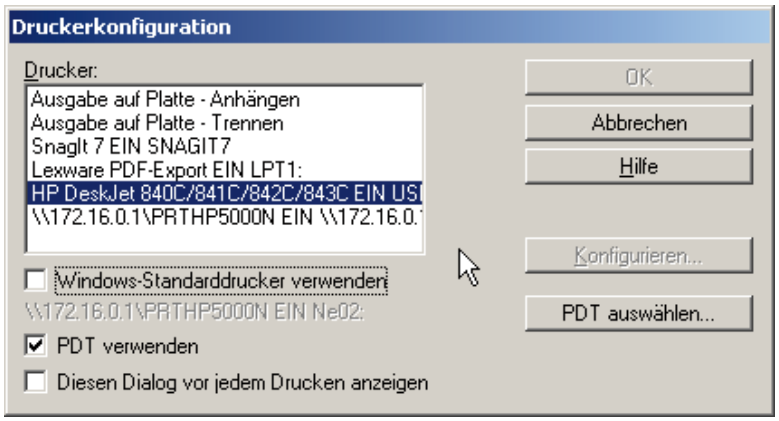
Windows-Drucker zuordnen



HP Deskjet konfigurieren

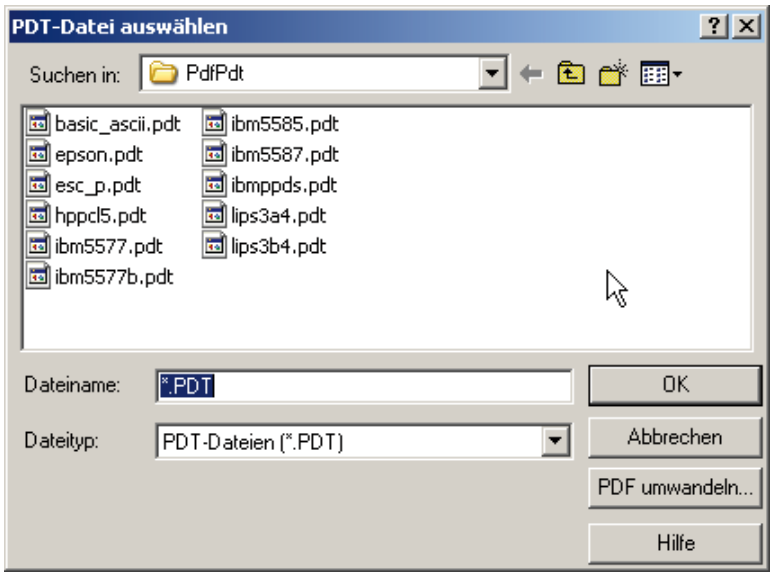
Mit Hilfe des Buttons „Konfigurieren“ lassen sich dann die Grundeinstellungen dieses lokalen Druckers einstellen. Der hier gezeigte Dialog ist typisch für den genannten HP-Drucker. Verwenden Sie den Drucker eines anderen Herstellers, so wird unter Umständen ein anderes Dialogfenster angezeigt.

2.) Wählen Sie im Dialog „Druckerkonfiguration“ den HP-DeskJet aus.



Mit Printer Definition Tables (PDT) arbeiten 1

Aktivieren Sie die Option „PDT verwenden“.



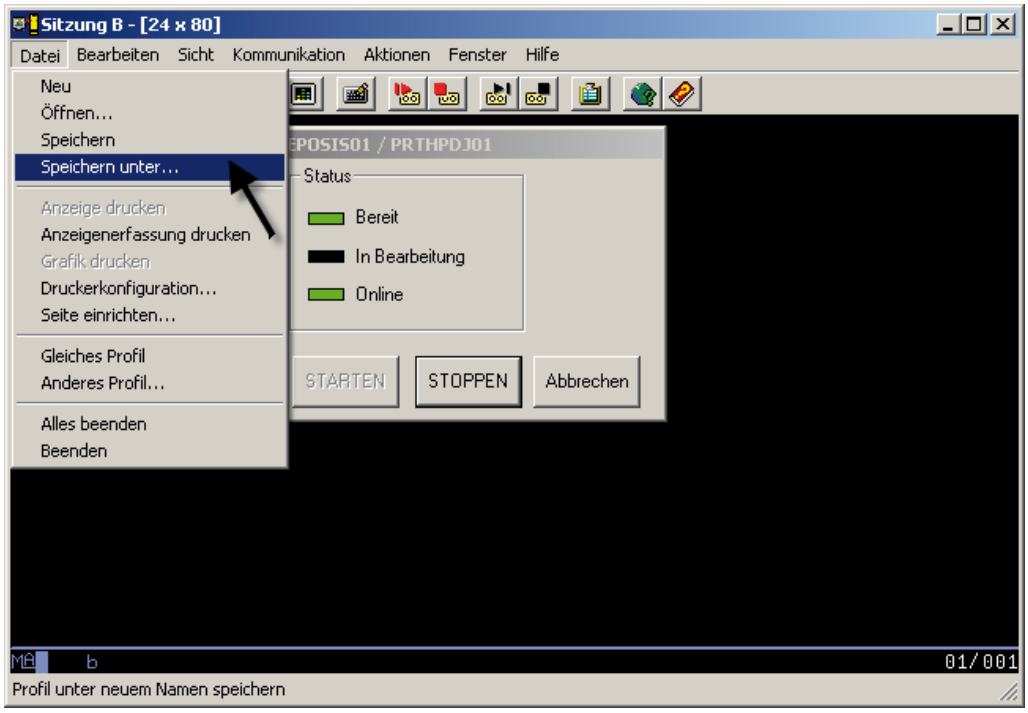
Anzeige der Printer Definition Tables

Was verbirgt sich hinter dieser Auswahl? Mit dem System i Access werden eine Reihe von sogenannten Printer Definition Tables (PDT) ausgeliefert, die für System i Access spezifische Druckerdefinitionstabellen enthalten. Diese Tabellen steuern unter Umständen einen emulierten Drucker effizienter als es die normale, lokale Druckerkonfiguration ermöglicht.

Drücken Sie danach den Knopf „PDT auswählen...“ und sie gehen in oben angezeigtem Dialog weiter. Im Fenster „PDT-Datei“ auswählen suchen Sie jetzt eine geeignete Printer Definition Table aus.

Bestätigen Sie die Auswahl mit „OK“.

Unabhängig davon, welchen Weg Sie gewählt haben, sollten Sie jetzt, wie schon bei den emulierten Bildschirmen, die Konfiguration in einem Emulationsprofil sichern.



Die Druckeremulation sichern

Dazu verwenden Sie den von der Bildschirmemulation bekannten Dialog „Speichern als...“.

Haben Sie für den Drucker ein Symbol erzeugt, so können Sie es auf dem Desktop oder in einem Ordner des Windows-PCs ablegen.

Windows-Desktop mit Druckeremulationssymbol

Sie sehen an dieser Stelle das für die Druckeremulation auf dem Windows-Desktop abgelegte Icon.

Danach sollten Sie einen Probedruck vornehmen, um die Installation/Konfiguration zu überprüfen.

Fazit: Die soeben erstellte Konfiguration ist lange Jahre eine Standardlösung für die oben beschriebene Problemstellung gewesen und wird sicherlich auch heute noch gut funktionieren. Jedoch muss ich anmerken, dass die Lösung mehrere Merkmale hat, die es zu bedenken gilt.

Auf dem Windows PC, auf dem die Funktion ausgeführt werden soll, muss System i Access für Windows installiert sein. Verwenden Sie andere Emulationsprodukte, wird eine Funktion für emuliertes Drucken nicht notwendigerweise zur Verfügung gestellt.

Die Funktion kann nur ausgeführt werden, wenn der Windows PC und eine Arbeitssitzung gestartet sind. Soll ein Windows Server diese Funktion als Printserver ausführen, so muss sichergestellt sein, dass der Windows PC gestartet wurde und weiterhin muss garantiert sein, dass eine Anmeldung an diesem Server erfolgt ist.

Ein positives Merkmal ist, dass diese Lösung der System i die Konvertierungslast von EBCDIC in den clientspezifischen Zeichensatz und die Druckersprache abnehmen kann, sofern Sie eine clientseitige Datenstromumsetzung eingestellt haben.

iSeries Access

Einleitung

Beginnend mit dem iSeries Betriebssystem V4R5 hat IBM eine neue Produktreihe aufgelegt, die unter dem Namen iAccess für Web im Rahmen der iSeries Access Familie aus einer Reihe von Diensten besteht, die wir aus der iSeries Access für Windows-Produktreihe kennen. Wir finden hier folgende Funktionen:

- Mit Ausgaben der iSeries arbeiten.
- Mit Druckern der iSeries arbeiten.
- Mit Nachrichten der iSeries arbeiten.
- Mit Jobs der iSeries arbeiten.
- Mit Datenbank- und anderen Objekten der iSeries arbeiten.
- SQL-Abfragen aufbauen und ausführen.
- Mit Integrated File System-Dateien arbeiten.
- Datenübertragungen ausführen.
- Befehle auf der iSeries ausführen.
- Plug-Ins herunterladen.
- Mit 5250-Dialogen arbeiten.
- Die Oberfläche anpassen.

Betrachten Sie diese Funktionsübersicht, so stellen Sie mit Sicherheit fest, dass die aufgeführten Funktionen ebenfalls im iSeries Access für Windows und iSeries Navigator zu finden sind. Warum also eine neue Produktreihe?

Im Unterschied zum iSeries Access für Windows und iSeries Navigator werden die iAccess for Web-Funktionen nicht lokal auf einem Windows Client installiert, sondern werden zentral auf einem Server installiert und mittels Web-basierter Techniken über einen Browser (Microsoft Internet Explorer, Netscape Navigator etc.) auf dem Client zur Verfügung gestellt.

Dieses Vorgehen bringt uns folgende Vorteile:

Die gesamte Intelligenz der oben angesprochenen Funktionen liegt auf dem Server, so dass nur eine zentrale Installation der iAccess für Web-Funktionen zu tätigen ist.

Sollten iAccess für Web-Funktionen zu Fehlern neigen, so kann der Fehler nur auf dem Server entstehen. Es gibt damit nur einen zentralen Punkt der Administration.

2. Ergänzung 03/2005

Da sämtliche Funktionen in einem Browser dargestellt werden, sind wir nicht wie bei iSeries Access für Windows und iSeries Navigator vom Windows-Betriebssystem abhängig, sondern können iAccess für Web-Funktionen auf jedem Betriebssystem zur Verfügung stellen auf dem ein Browser vorhanden ist.

iAccess für Web hat ein signifikantes Merkmal. Da es ausschließlich über Webtechniken ausführbar ist, läuft iAccess für Web nicht nativ auf dem Betriebssystem OS/400, sondern benötigt eine eigene Web-basierte Laufzeitumgebung, einen so genannten Applikations-Server. Bis OS/400 V5R1 standen dafür ausschließlich die Websphere Application Server der IBM zur Verfügung. Ab OS/400 V5R2 kann auch der Tomcat Application Server der Apache Foundation (ASF) zu diesem Zweck herangezogen werden. Vorherige Versionen des Tomcat ASF die vor OS/400 V5R2 ausgeliefert wurden, können nicht verwendet werden.

Zurück zu den Applikations-Servern. Es ist sicherlich leicht nachvollziehbar, dass das Haus IBM hier vornehmlich seine eigene Produktlinie die Websphere Application Server in den Vordergrund schiebt.

Vom Websphere Application Server existieren derzeit mehrere Versionen. Welche Websphere Application Server-Versionen sind derzeit am Markt erhältlich?

Websphere Application Server 4.0 Single Server: Diese Version ist kostenpflichtig und kann mit den oben genannten Tools eingesetzt werden ab der Version V5R1 des OS/400.

Websphere Application Server 4.0 Advanced Server: Diese Version ist kostenpflichtig und kann mit den oben genannten Tools eingesetzt werden ab der Version V5R1 des OS/400.

Websphere Application Server 5.0 Standard Server: Diese Version ist kostenpflichtig und kann mit den oben genannten Tools eingesetzt werden ab der Version V5R1 des OS/400.

Websphere Application Server 5.0 Net Deployment Server: Diese Version ist ebenfalls kostenpflichtig und kann mit den oben genannten Tools eingesetzt werden ab der Version V5R1 des OS/400.

Websphere Application Server 5.0 und 5.1 Express für iSeries. Diese Versionen des WAS sind ab OS/400 V5R1 einzusetzen. Beide Versionen sind unter OS/400 V5R1 und V5R2 kostenpflichtig. Unter V5R3 werden diese Versionen kostenfrei ausgeliefert.

Tipp:

Besitzen Sie OS/400 V5R2 oder V5R2 und verfügen Sie über eine Software-Subscription, so ordern Sie einfach OS/400 V5R3, dann kommen Sie in den Genuss der oben genannten kostenfreien Express Versionen des Websphere Application Server. Installieren Sie dann einfach die Software auf Ihrem bestehenden iSeries Betriebssystem.

Ich will Sie möglichst schnell und ohne Ballast auf den Weg bringen, damit Sie zum eigentlichen Ziel, nämlich der Anwendung der oben genannten Tools, vorstoßen können.

Folgende Themenbereiche erwarten Sie:

1.) Installation des Websphere Application Server. Hier erfahren Sie, wie Sie das Betriebssystem vorzubereiten haben, wie Sie den Websphere Application Server installieren und welche Nachbereitungstätigkeiten zu erledigen sind.

2.) Installation der Websphere Application Server Administrator-Konsole. Die Administration des iSeries Websphere Application Server hat nur ein relativ schwaches 5250-basiertes Administrations-Interface, deshalb ist die Konfiguration mittels des grafischen Administrator-Client leichter. Auch hier ist eine sorgfältige Vor- und Nachbereitung wesentlich für den Erfolg.

3.) Nach der Installation des Websphere Application Server, muss dieser konfiguriert werden. In diesem Bereich werden Sie mit so genannten Server-Instanzen des Websphere Application Server arbeiten.

4.) Nachdem Sie die Konfiguration gemeistert haben, müssen Sie einen Webserver aufsetzen. Die Konfiguration des Webservers muss auf die Erfordernisse des Websphere Application Server eingehen, so dass auch hier ganz bestimmte Schritte notwendig sind.

5.) Dann erfolgt der Test der gesamten Umgebung. Sollten die Tests ein positives Ergebnis ergeben, so verfügen Sie über eine arbeitsfähige Laufzeitumgebung für iAccess für Web.

6.) Installation des iAccess für Web-Lizenzprogramme mit nachfolgender Konfiguration.

Begriffe wie Subsystem, Bibliothek und Job sollten Ihnen geläufig sein. Auch die entsprechenden CL-Befehle, um mit diesen Objekten zu arbeiten, sollten Ihnen nicht unbekannt sein. Darüber hinaus sollten Sie Kenntnisse im Bereich des TCP/IP-Netzwerks mitbringen. Adressen-Regularien, Begriffe wie Domäne, DNS bzw. Host-Tabelle, sollten für Sie bekannt sein.

2. Ergänzung 03/2005

7.4
Seite 4

Sollten Ihnen diese Begriffe nicht bekannt sein, so möchte ich Sie an die entsprechenden Kapitel dieses Buches verweisen.

Voraussetzungen

Bevor wir damit beginnen, den Websphere Application Server zu installieren, muss erst geklärt werden, ob die nötigen Voraussetzungen gegeben sind, um den Websphere Application Server installieren zu können. Das ist wiederum die Voraussetzung für den Betrieb von iSeries Access für Web.

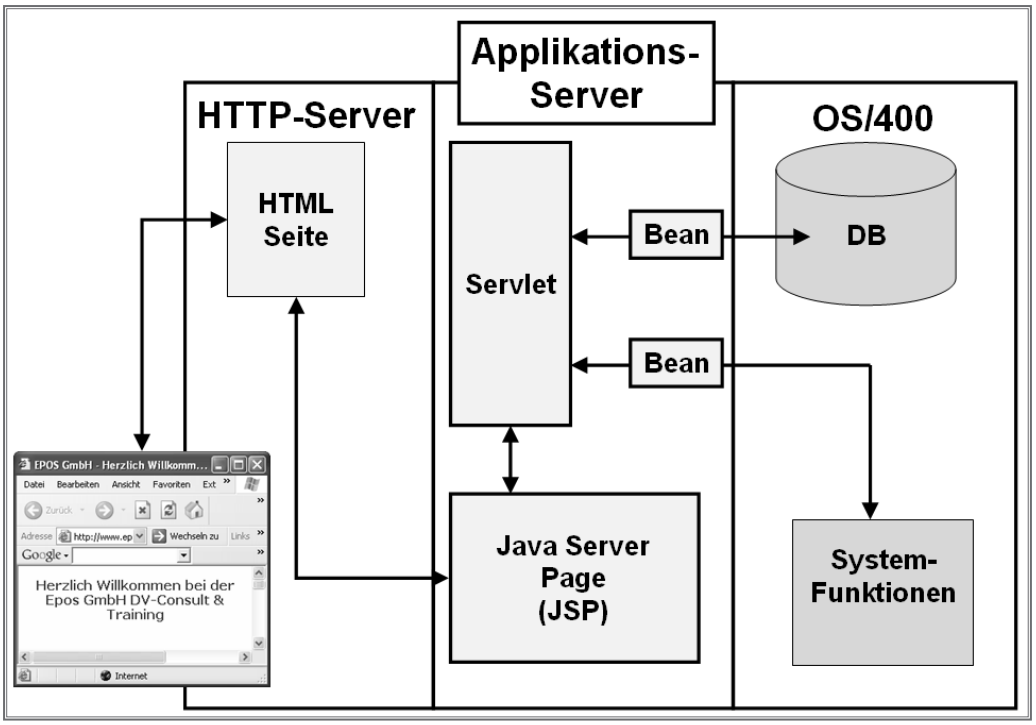
Folgende Bereiche sollten Sie dabei berücksichtigen:

- Umfeld: Was will ich eigentlich erreichen? Der vordergründige Wunsch, iAccess für Web zu nutzen, ist dabei nicht genug. Vielmehr stehen zentrale Fragen im Raum – wie z. B.: Mit welcher Technik umgebe ich mich? Was soll später auf meiner iSeries laufen und wie ist das Zusammenspiel der Objekte zu verstehen?
- OS/400: Welche Betriebssystem-Komponenten müssen installiert sein? Welche Lizenzprogramme müssen installiert werden. Welche PTFs benötige ich?
- Netzwerk: Wie sieht meine TCP/IP-Konfiguration aus?
- Websphere Application Server: Mit welcher Version werde ich arbeiten? Sind Programm-Aktualisierungen einzuspielen?

7.4.1
Seite 2

Das Umfeld

Applikations-Server eröffnen die Möglichkeit, Transaktionen zwischen schlanken HTML-Clients und robusten Server-Diensten einer iSeries aufzubauen.



Mit Applikations-Servern arbeiten

Sie sehen, dass zwischen Web-Server und OS/400 jetzt der Applikations-Server steht. Dieser Applikations-Server stellt die Laufzeitumgebung für das neue Anwendungskonstrukt zur Verfügung. Vereinfacht könnte man sagen, dass ein Applikations-Server ein Betriebssystem für transaktionsbasierte Web-fähige Anwendungen darstellt, die mit HTML-Dialogen arbeiten.

Das Konstrukt basiert darauf, dass die Dialoge der Web-Anwendung zum Teil dynamisch zur Laufzeit von Anwendungen erstellt werden; der andere Teil der Dialoge liegt in fester Form vor. Diese HTML-Informationen werden in so genannten JAVA Server Pages (JSPs) abgelegt. JSPs sind die Vermischung von HTML- und Java-Code. JSPs stellen die Verbindung zu JAVA-Servlets her. Servlets sind JAVA-Programme, die auf dem Server laufen. Servlets enthalten die Anwendungslogik der Programme. Mit Hilfe so genannter JAVA-Beans greifen dann die Servlets auf die Systemressourcen zu. JAVA-Beans sind eine besondere Form von JAVA-Programmen. Beans stellen eine definierte Schnittstelle zur Verfügung, die von Servlets angesprochen werden.

IBM Tools wie iSeries Access für Web basieren alle auf diesem Konstrukt. Ohne diese Basis sind sie nicht lauffähig.

7.4.1.1

Seite 2

Hardware

Für die Hardware gelten folgende Voraussetzungen, die als Mindestangaben gesehen werden sollten:

- AS/400e Server Modell 170 mit Prozessor 2292
- AS/400e Server Modell 720 mit Prozessor 2061
- AS/400e Server Modell 270 mit Prozessor 2248
- iSeries Server Modell 800 und höher
- iSeries i5 Server Modell 520 und höher

Es sollten 1 Gigabyte Hauptspeicher als Minimum zur Verfügung stehen. Systeme, die diese Mindestanforderungen nicht erfüllen, können in Umgebungen eingesetzt werden, die eine begrenzte Anzahl Benutzer unterstützen und für die längere Server-Initialisierungszeiten toleriert werden können.

Folgender Plattenspeicher sollte zur Verfügung stehen:

- Während der Installation: 500 Megabyte
- Nach der Installation: 250 Megabyte

Darüber hinaus sollte der DFV-Adapter, der TCP/IP unterstützt (Ethernet oder Token Ring), Bestandteil des Systems sein.

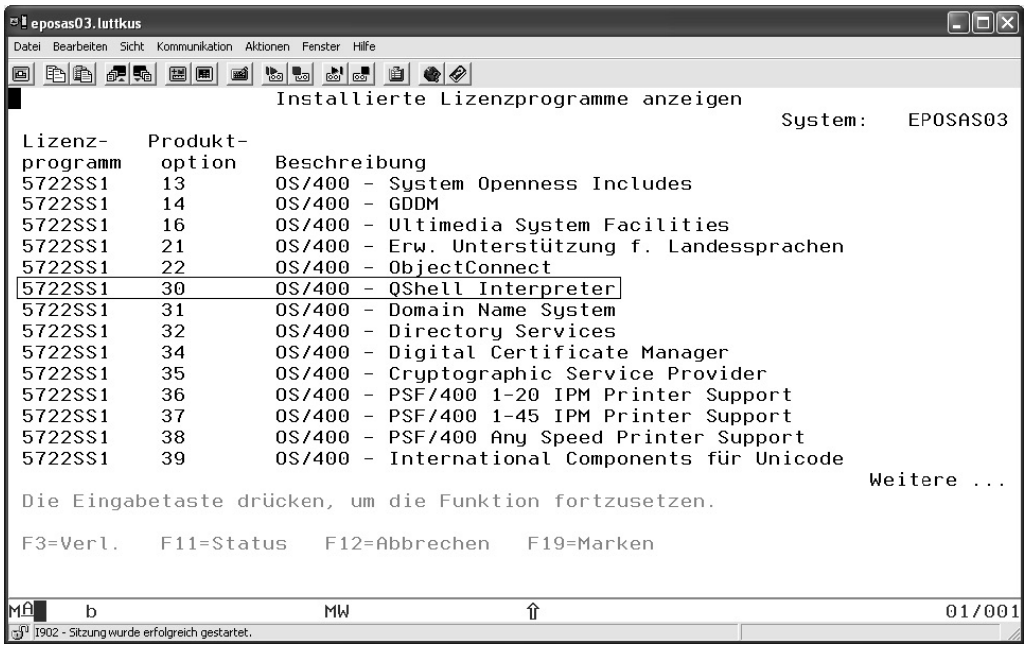
7.4.1.2

Seite 2

OS/400-Voraussetzungen

Bei der Prüfung der Voraussetzungen für die Installation des Websphere Application Server sollten Sie einen Blick auf die installierten Lizenzprogramme werfen.

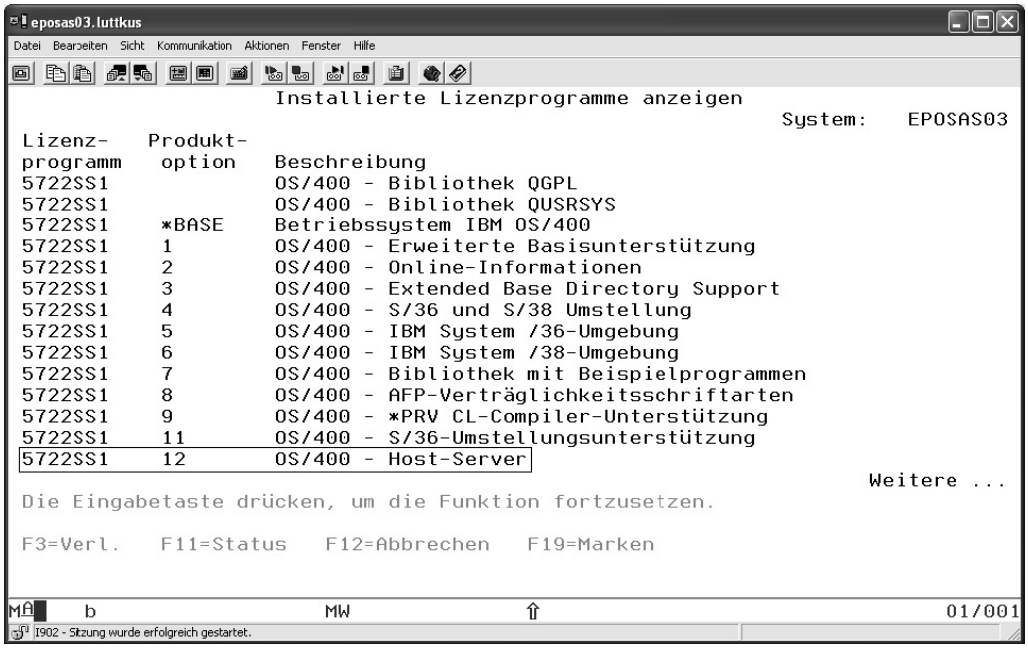
Um den Websphere Application Server 5.0 Express zu installieren, benötigen Sie mindestens OS/400 V5R1. Die in den folgenden Kapiteln durchgeführte Installation basiert auf OS/400 V5R3.



Qshell Interpreter

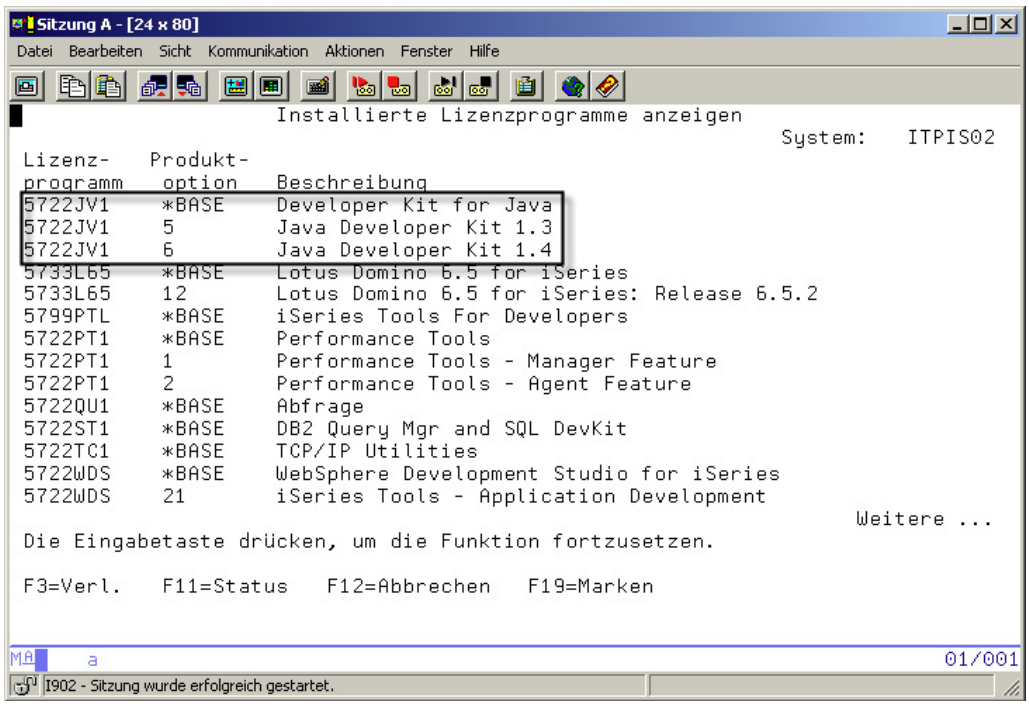
Soll die Installation vom CD-Laufwerk der iSeries durchgeführt werden, so muss die Komponente 30 des Betriebssystems – OS/400 Qshell Interpreter 5722SS1 – installiert sein.

7.4.1.3
Seite 2



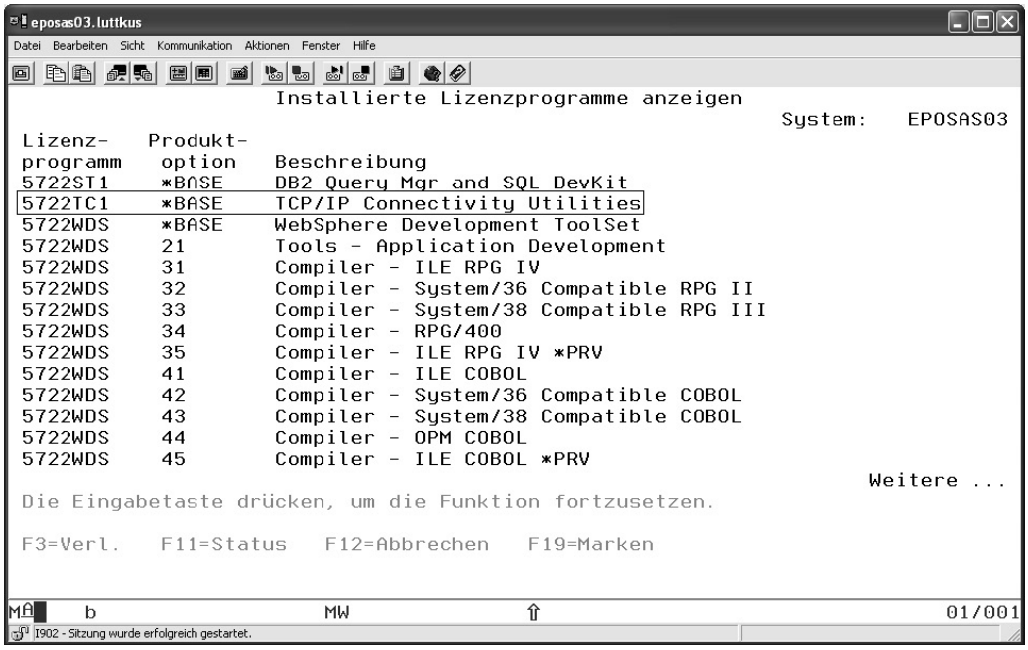
Host-Server

Soll die Installation vom CD-Rom-Laufwerk einer Arbeitsstation ausgeführt werden, so muss die Komponente 12 des Betriebssystems ‚Äì OS/400 Host-Servers 5722SS1 ‚Äì installiert sein. Sind Sie Client Access Express-Anwender, so ist diese Komponente bereits installiert.



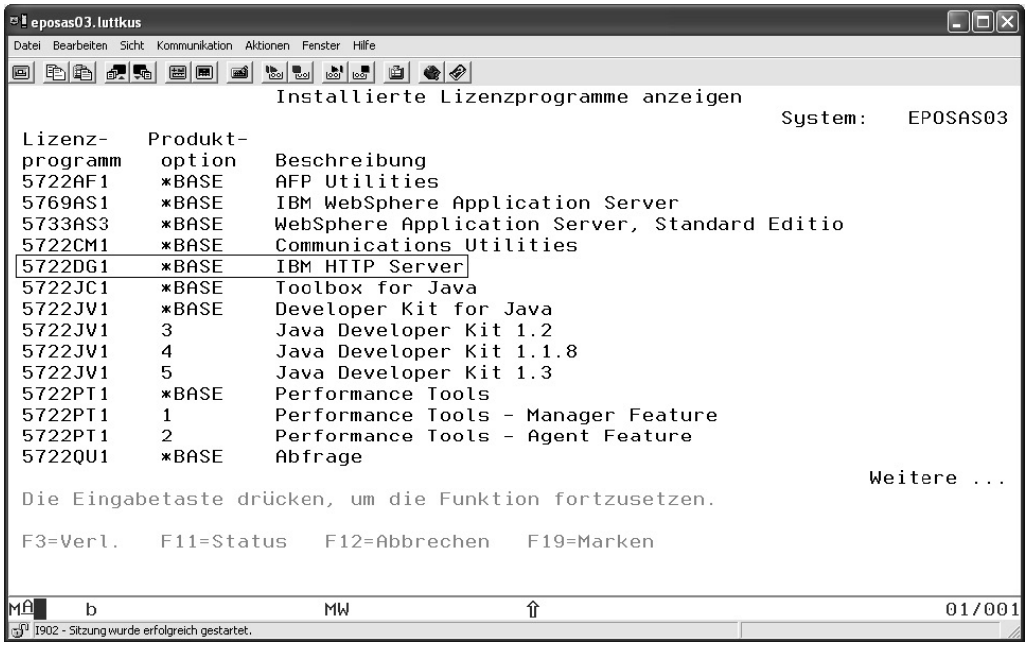
JAVA

Weiterhin benötigen Sie zwingend das AS/400 oder iSeries Developer Kit für JAVA Version 1.4 ‚Äì 5722JV1, Komponente 6.



TCP/IP

TCP/IP muss auf Ihrem System installiert, konfiguriert und betriebsbereit sein: TCP/IP Connectivity Utilities – 5722TC1.



HTTP-Server

Der IBM HTTP-Server muss ebenfalls auf Ihrem System installiert sein – 5722DG1.

Für die Installation benötigen Sie ein Benutzerprofil, das über das *ALLOBJ-Sonderrecht verfügt.

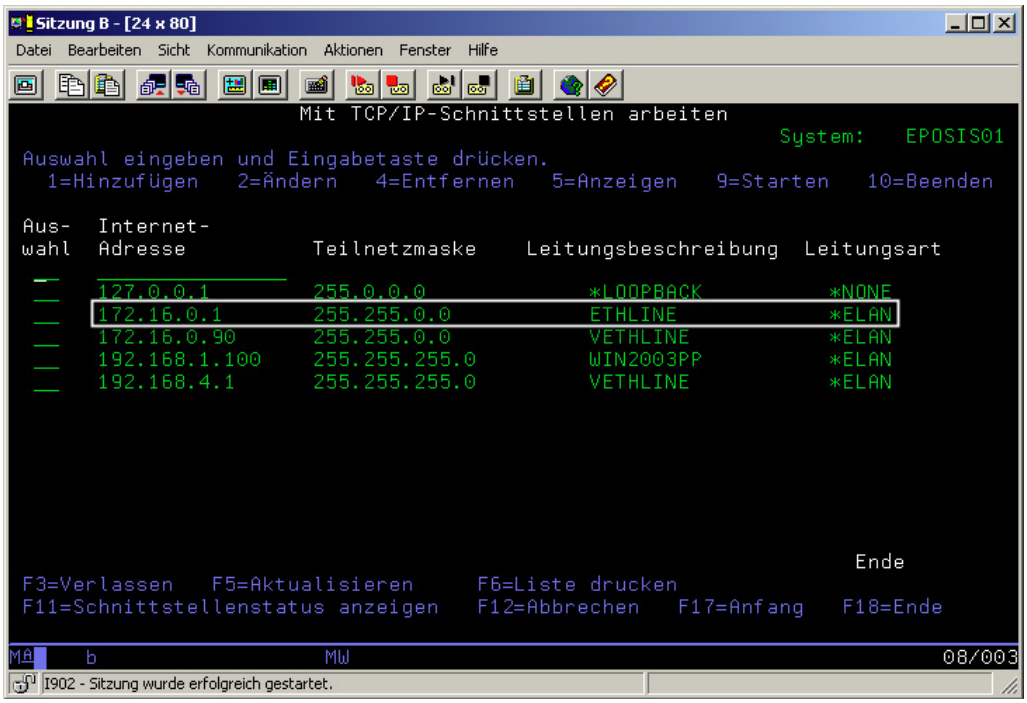
7.4.1.3
Seite 4

Für alle gezeigten Arbeitsschritte in diesem und im folgenden Kapitel verwende ich das QSECOFR-Profil meines Systems.

Netzwerk-Voraussetzungen

Bevor Sie mit der Installation des Websphere Application Server beginnen, sollte die TCP/IP-Konfiguration im OS/400 „stehen".

Dazu sollte auf Ihrem System entweder für eine Ethernet- oder für eine Token Ring-Schnittstelle eine IP-Adresse vergeben worden sein. Rufen Sie den Befehl CFGTCP auf und wählen Sie die Auswahl 1 – „Mit TCP/IP-Schnittstellen arbeiten" – auf.

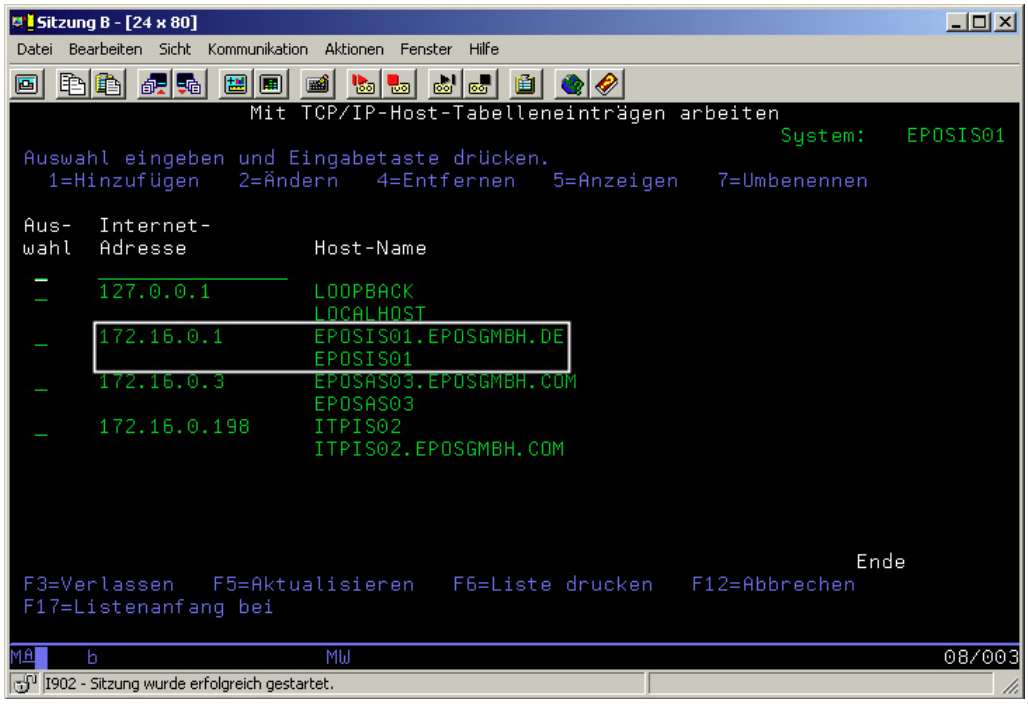


IP-Adressen prüfen

Für den Websphere Application Server werde ich die Adresse 172.16.0.1 verwenden.

Dann sollte entweder in der lokalen Host-Tabelle oder im Domain Name Service (DNS), auf den Ihre iSeries zugreifen kann, ein qualifizierter Host-Eintrag erfasst sein. Rufen Sie den Befehl CFGTCP auf und wählen Sie die Auswahl 10 – „Mit TCP/IP-Host-Tabellen arbeiten" – auf.

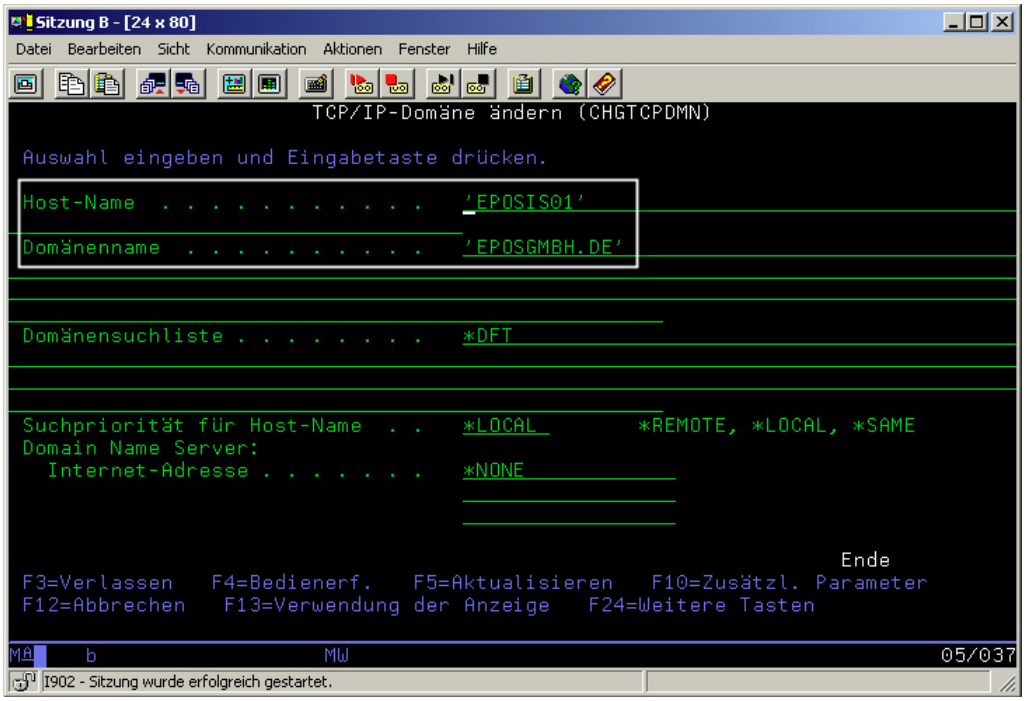
7.4.1.4
Seite 2



Host-Tabelle prüfen

Ich habe in der Host-Tabelle den Eintrag EPOSAS01.EPOSGMBH.DE hinterlegt.

Es muss in den TCP/IP-Domänen-Informationen ein Eintrag erzeugt worden sein, der entweder mit dem Host-Tabelleneintrag oder dem DNS-Eintrag übereinstimmt. Rufen Sie den Befehl CFGTCP auf und wählen Sie die Auswahl 12 – „Mit TCP/IP-Domänen-Informationen ändern“ – auf.



Domänen-Informationen prüfen

Ich habe in den Domänen-Attributen den Host-Namen EPOSIS01 und den Domänen-Namen EPOSGMBH.DE hinterlegt.

Sollten Sie nicht mit der lokalen Host-Tabelle der iSeries arbeiten wollen, so geben Sie im Parameter „Domain Name Server“ – falls nicht vorhanden – die IP-Adresse des in Ihrem Netzwerk befindlichen DNS ein. Arbeiten Sie mit mehr als einem DNS, so können Sie hier bis zu drei IP-Adressen angeben.

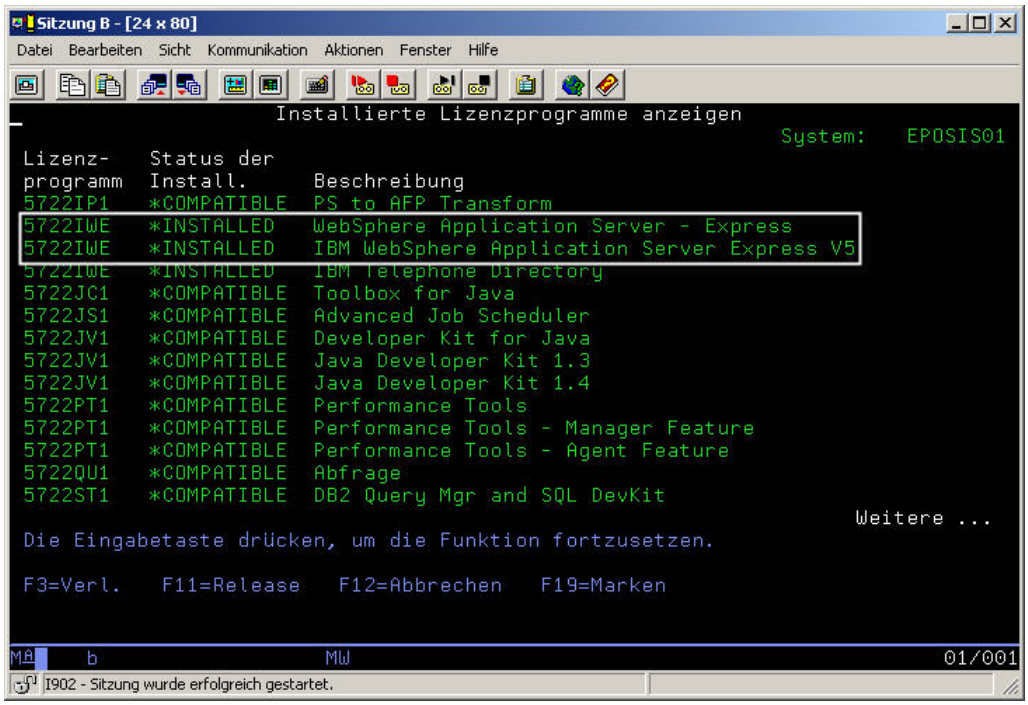
7.4.1.4

Seite 4

Websphere Application Server Version

Wie Sie bereits in der Einleitung gelesen haben, gibt es derzeit eine Reihe verschiedener aktueller Versionen des Websphere Application Server. Für die in diesem Buch zugrunde liegende Aufgabenstellung werde ich den Websphere Application Server 5.0 Express für iSeries verwenden.

Besorgen Sie sich also das IBM-Lizenzprogramm 5722IWE Websphere Application Server Express für iSeries.



Websphere Application Server Express für iSeries

7.4.1.5

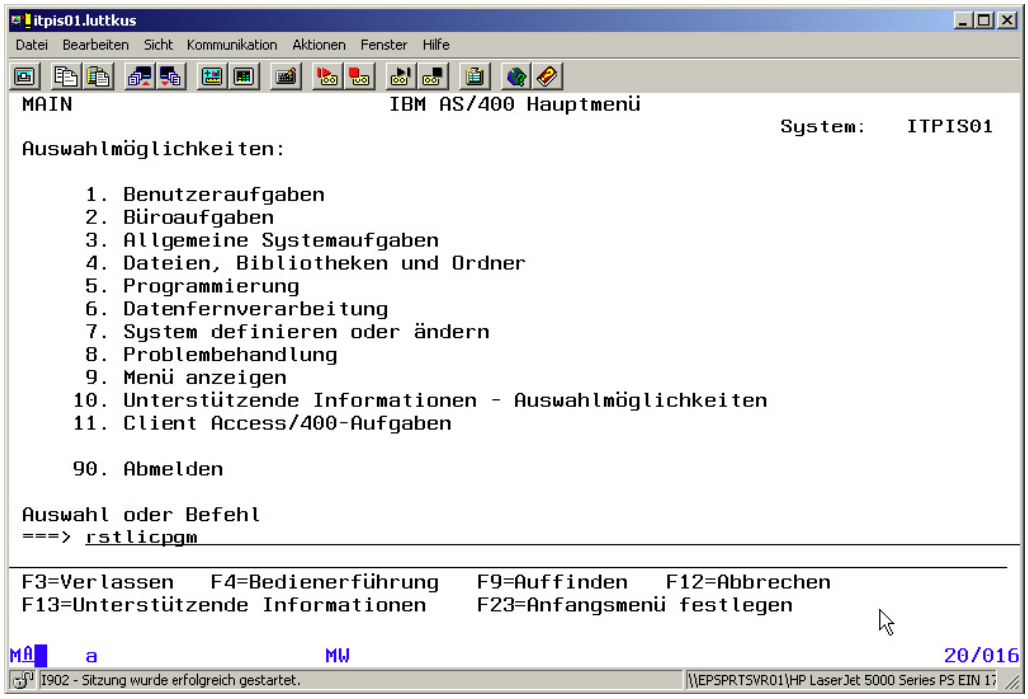
Seite 2

Installation und Konfiguration von iAccess für Web

Im Gegensatz zur Installation des Websphere Application Server kann die Installation und Konfiguration von iAccess für Web als geradezu einfach angesehen werden.

Wichtig ist jedoch, dass, wenn Websphere Application Server nicht korrekt zum Laufen gebracht wird, auch iAccess für Web nicht genutzt werden kann!

Das Produkt iAccess für Web (OS/400 V5R2 und V5R3) befindet sich auf der Lizenzprogramm-CD „Client Access Familie – 57nn-XW1“ und wird, wie wir es gewohnt sind, mit dem Befehl RstLicPgm (Restore License Program) installiert.



RstLicPgm aufrufen

Sollte sich iAccess für Web nicht auf dieser CD befinden, so müssen Sie es über den IBM-Vertrieb nachbestellen. iAccess für Web ist kostenlos!

Sie können jedoch auch iAccess für Web direkt aus dem Internet herunterladen von folgender URL:

http://www14.software.ibm.com/webapp/download/search.jsp?go=y&rs=iabeta

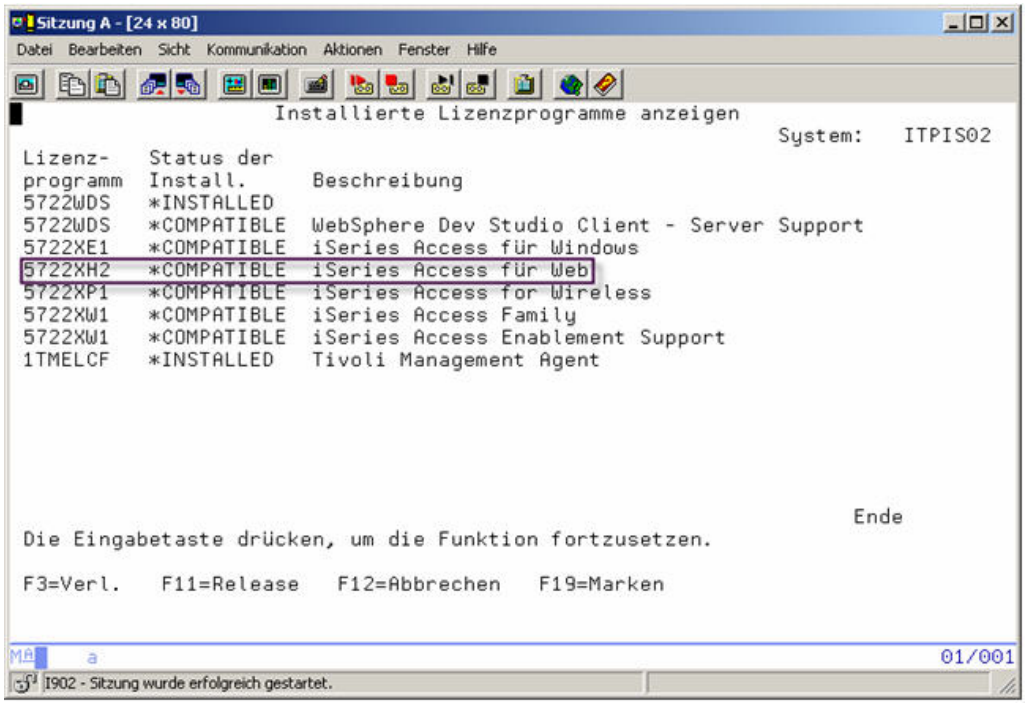
Eine weitere Voraussetzung für die Installation ist, dass der Websphere Application Server bei der Installation läuft.

7.4.2
Seite 2

Geben Sie also „RstLicPgm“ auf einer Befehlszeile des Betriebssystems ein, drücken Sie die F4-Bedienerhilfe und versorgen Sie die Parameter des Befehls wie folgt:

- Produkt: 5722XH2
- Einheit: OPT01
- Wahlweise zurückgespeicherter Teil: *Base

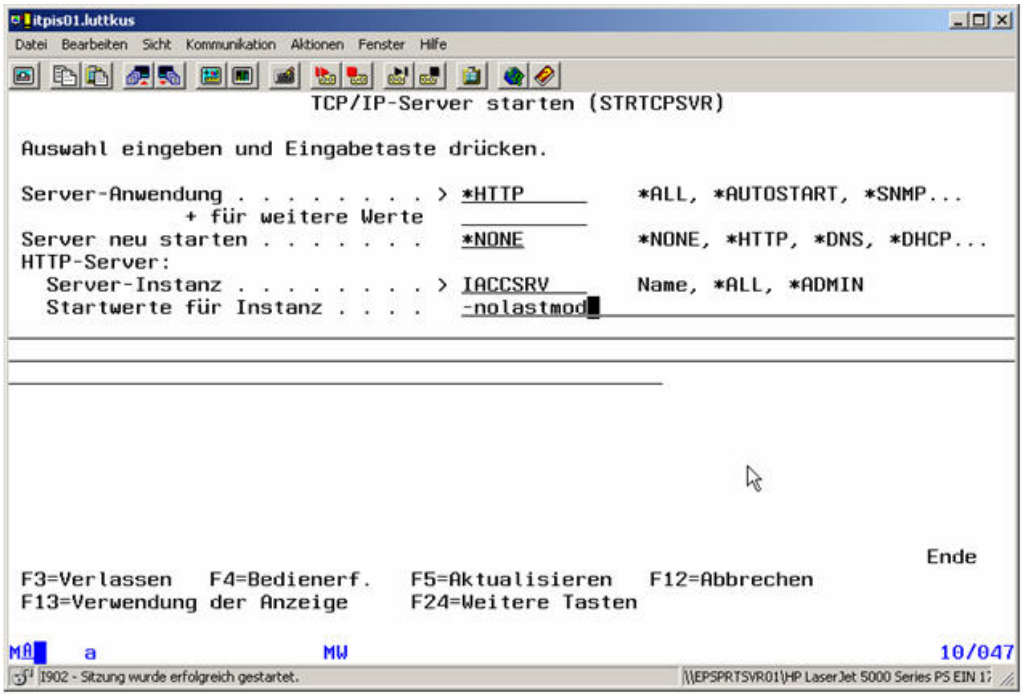
Nach ungefähr fünf Minuten (kann bei älteren Systemen länger dauern) ist die Installation beendet.



Lizenzprogramme anzeigen

Nach der erfolgreichen Installation werden Sie dann das Produkt „5722XH2 iAccess für Web“ in der Lizenzprogramm-Anzeige des OS/400 sehen (Go LicPgm ➜ Auswahl 10).

Damit iAccess für Web optimal arbeiten kann, sollten Sie den HTTP-Server beenden und dann ‚Äì wie im folgenden Bild gezeigt ‚Äì starten.

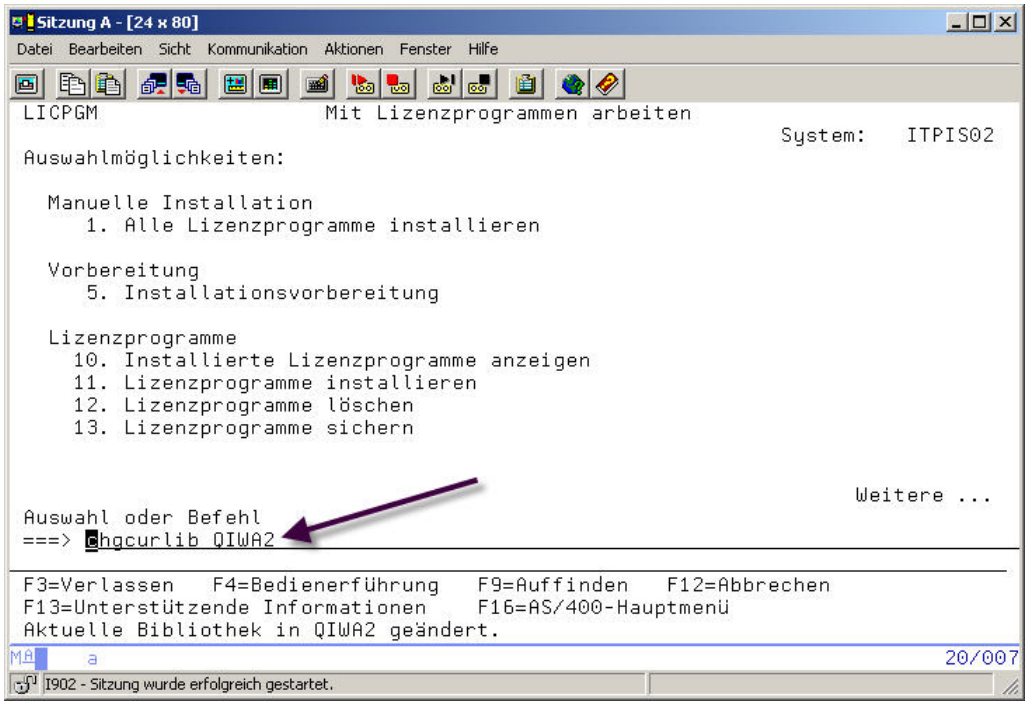


Start des HTTP-Servers

Wichtig ist, dass Sie im Parameter „Startwerte für Instanz" den Wert „-nolastmod" setzen. Dieser Parameter beeinflusst das Caching-Verhalten des Web-Servers in der Art, dass unmodifizierte Seiten im Speicher gecached werden. So kommt es insgesamt zu einer Performance-Verbesserung beim Zugriff auf iAccess für Web-Funktionen.

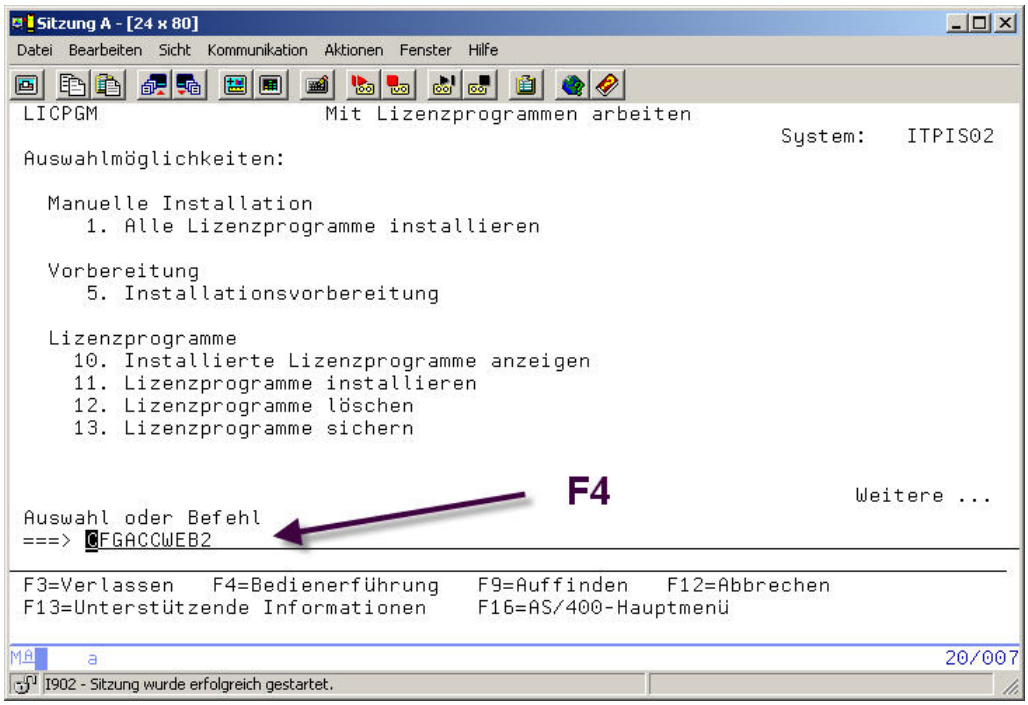
Da alle OS/400-seitigen Objekte des iAccess für Web in der Bibliothek QIWA2 gespeichert sind, sollten Sie diese Bibliothek für die Konfiguration mittels CHGCURLIB zur aktuellen Bibliothek machen.

7.4.2
Seite 4



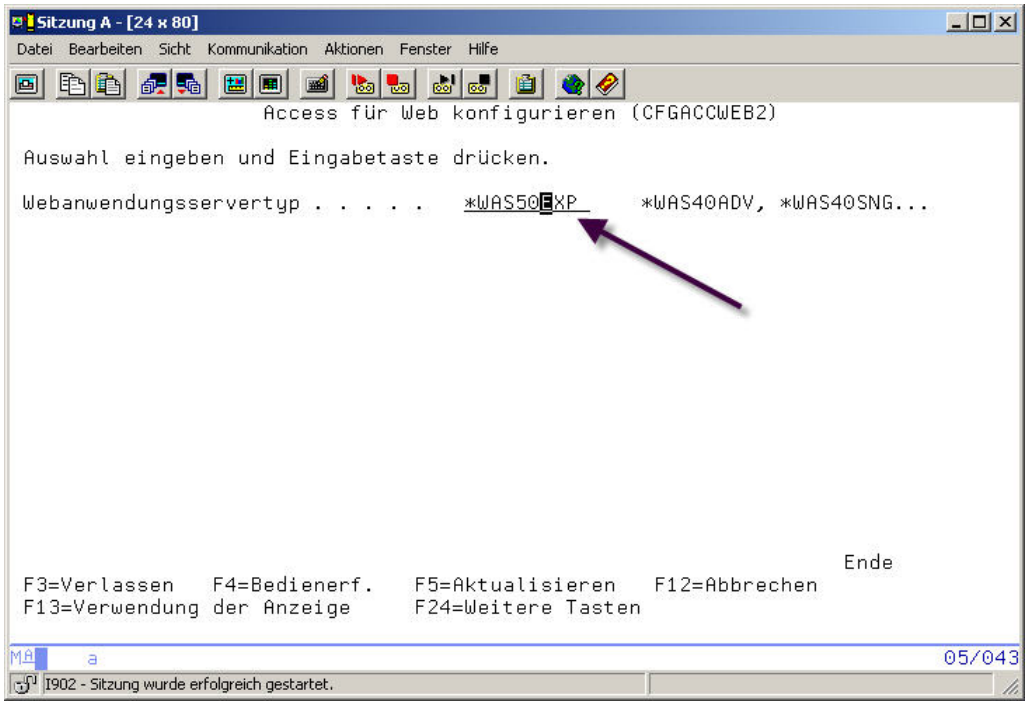
CHGCURLIB QIWA2

Danach beginnt die Konfiguration von iAccess für Web. Geben Sie dazu den Befehl CfgAccWeb2 (Konfigurieren iAccess for Web) ein.



CfgAccWe2b – Konfigurieren iAccess für Web

Prompten Sie den Befehl mit der F4-Bedienerhilfe.



Konfiguration von iAccess für Web

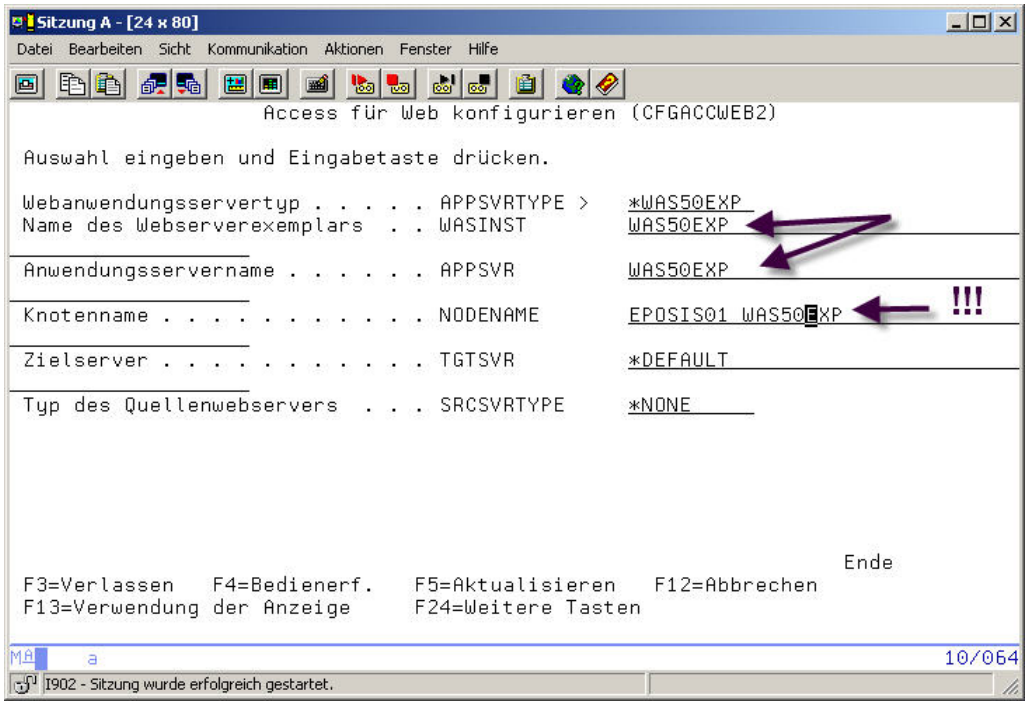
In unserem Fall ist ein erster Parameter einzutragen.

Hierbei handelt es sich um die Angabe des verwendeten Application Server. Folgende Werte sind möglich:

- *WAS40ADV: Websphere Application Server 4.0 Advanced Server.
- *WAS40SNG: Websphere Application Server 4.0 Single Server.
- *WAS50: Websphere Application Server 5.0 Standard/Net Deployment.
- *WAS50EXP: Websphere Application Server 5.0 Express – die von mir getroffene Auswahl.
- *WP41: Websphere Protal Server.
- *ASFTOMCAT: Tomcat Application Server.

Nach Bestätigung dieses ersten Parameters öffnen sich folgende weitere Eingabefelder:

7.4.2
Seite 6



Befehl CFGWEB2

Parameter:

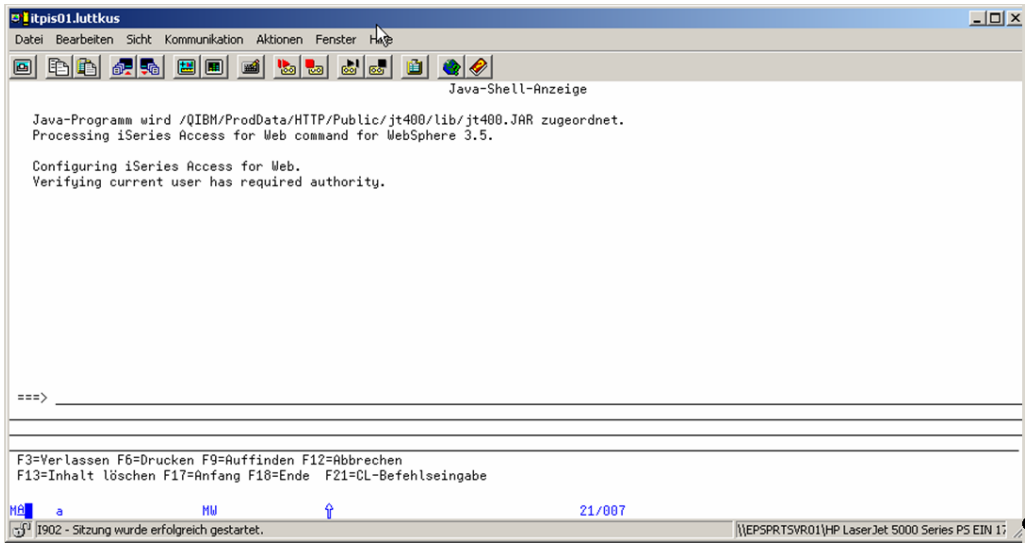
WASINST: Hier trage ich den Namen meines Websphere Application Server ein – hier: WAS50EXP.

APPSVR: Auch hier trage ich den Namen meines Websphere Application Server ein – hier: WAS50EXP.

NODENAME: Dies ist ein äußerst wichtiger Parameter! Der so genannte Knotenname bildet sich aus dem Host-Namen der iSeries, so wie er in den Domänen-Attributen (CFGTCP, Option 12) hinterlegt ist, dem Trennzeichen „_“ und dem Namen des Application Servers – hier: EPOSIS01_WAS50EXP.

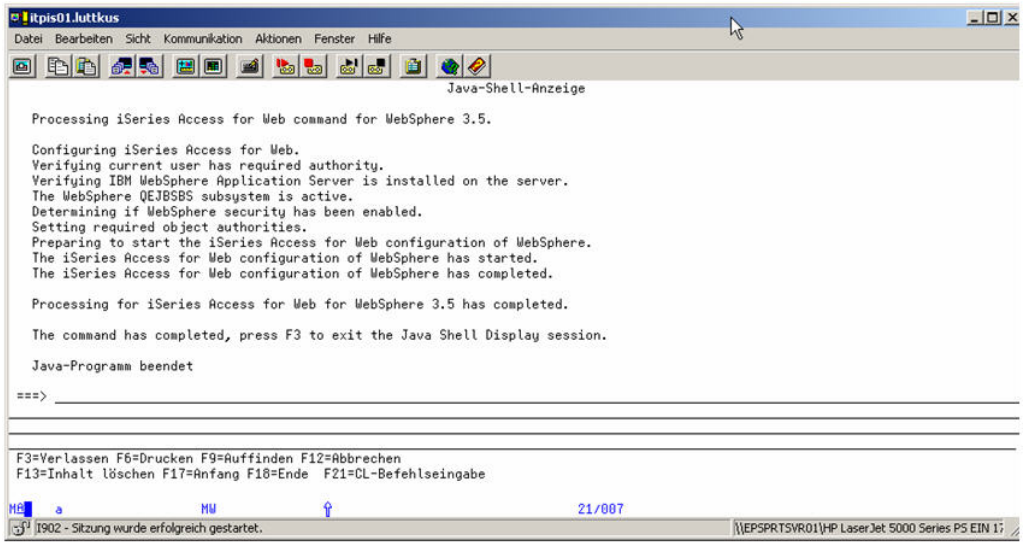
Der Eintrag unterscheidet Groß- und Kleinschreibung!

Nachdem Sie den Befehl CfgAccWeb2 mit Datenfreigabe bestätigt haben, öffnet sich die Java-Konsole und zeigt Statusmeldungen zur Konfiguration an.



Konfiguration läuft

Vorsicht! Beenden Sie diesen Dialog nicht, auch wenn die Sitzung nicht, wie Sie es im OS/400 gewohnt sind, gegen Eingaben gesperrt ist.



Konfiguration läuft

Erst wenn die Meldung „The command has completed, press F3 to exit...“ erscheint, verlassen Sie die Java Shell mit F3.

Danach öffnen Sie den Administrator-Client des Websphere Application Server; dort sehen Sie im Bereich „Installierte Anwendungen verwalten“ eine neue Web-Anwendung: iSeries Access für Web.

7.4.2
Seite 8

Start von iSeries Access für Web

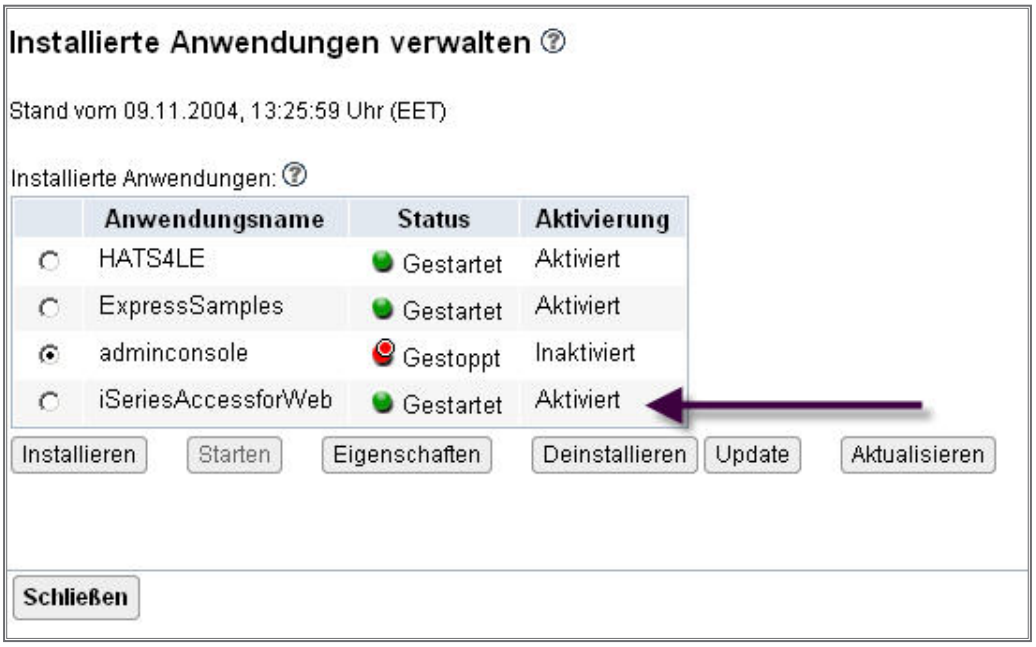
Diese Applikation hat den Status „Gestoppt“ und „Inaktiviert“.

Anwendung iSeries Access für Web aktivieren

Mittels des Eigenschaften-Dialogs aktivieren Sie die „Applikation iSeries Access for Web“.

Danach kann die Applikation gestartet werden.

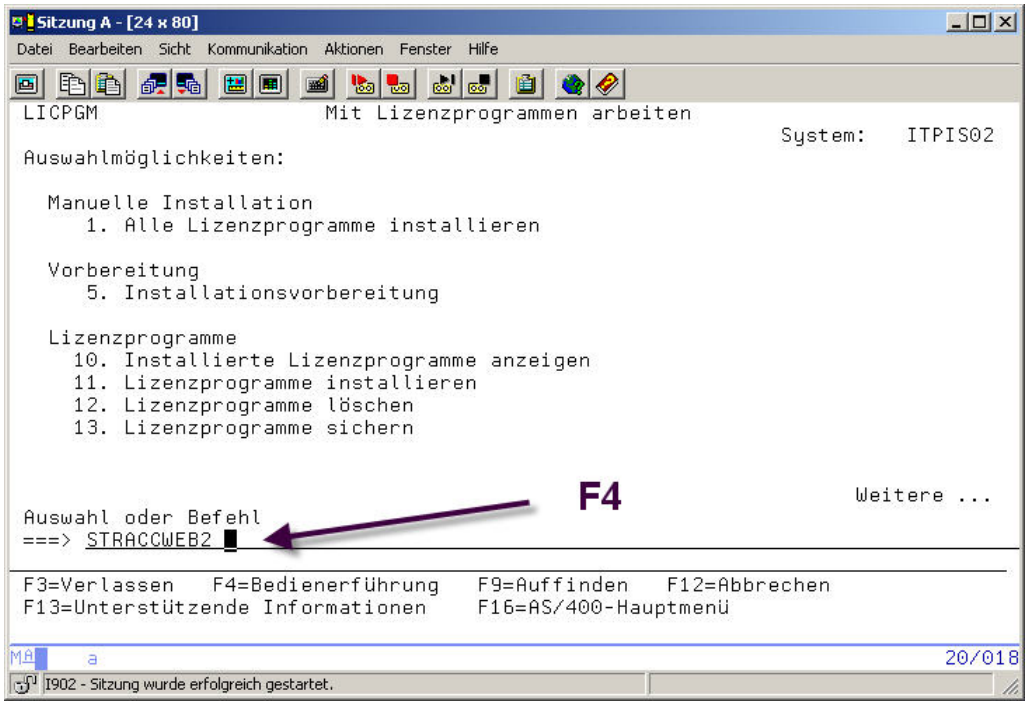
Ist der Start erfolgreich verlaufen, so sollte sich folgender Dialog zeigen.



iSeries Access für Web gestartet

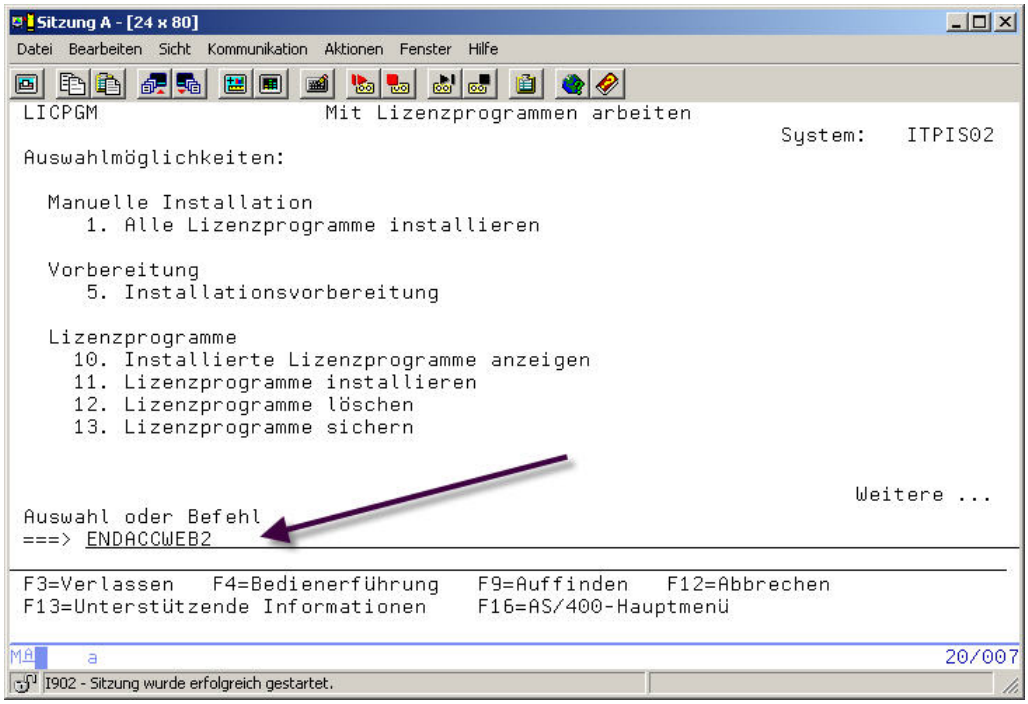
Einige Anwender haben mir die Frage gestellt, ob der Start von iAccess für Web nur aus der Administrator-Konsole des Websphere Application Server erfolgen kann. Zum Glück nicht!

Haben Sie iAccess für Web auf Ihrer iSeries installiert, so ist bei der Installation die Bibliothek QIAW2 erstellt worden. Diese Bibliothek enthält alle CL-Befehle, die für die Konfiguration, den Start und das Beenden des iAccess für Web-Servers notwendig sind. Benötigen Sie diese Befehle ständig oder beabsichtigen Sie den iAccess für Web-Server im Startprogramm der iSeries zu starten, so sollten Sie diese Bibliothek in die Standard-Bibliothekssuchliste des Systems aufnehmen.



StrAccWeb2

Haben Sie diese Vorarbeiten erledigt, so wird der iAccess für Web-Server einfach mit dem Befehl StrAccWeb2 gestartet.



iSeries Access für Web beenden

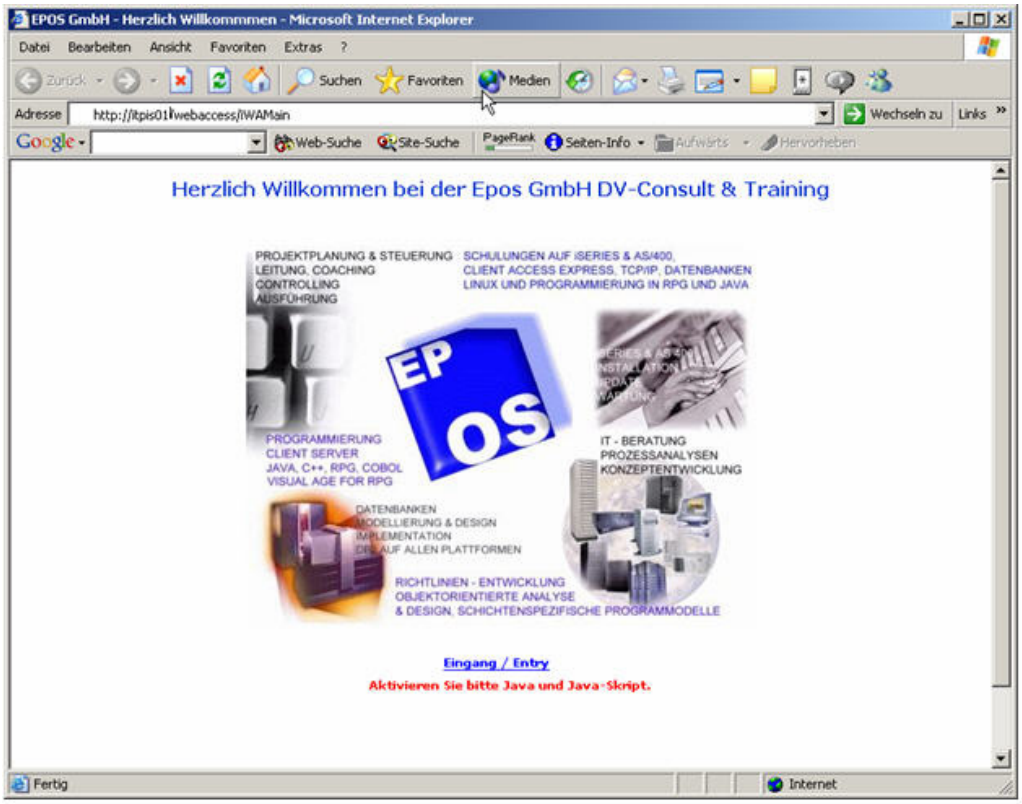
Die Beendigung der Applikation erreichen Sie durch Eingabe des Befehls ENDACCWEB2.

Nach dem erfolgreichen Start des Servers öffnen Sie Ihren Browser und geben folgende URL ein:

http://<<hostname oder adresse>>/webaccess/iWAMain

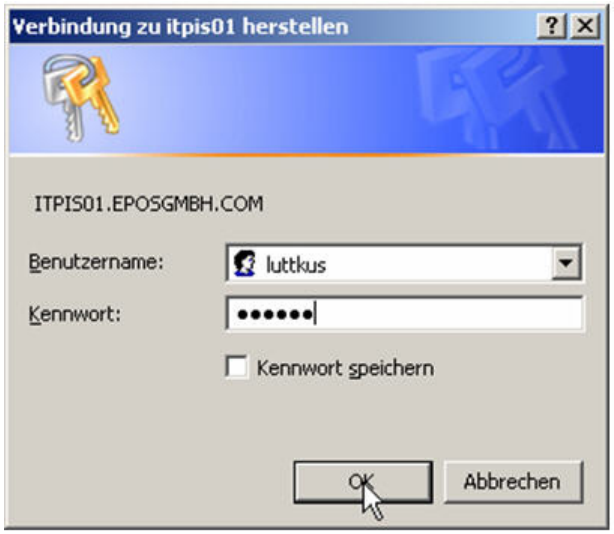
Ich habe hier für meine Umgebung folgende Adresse eingegeben:

http://<<itpis01/webaccess/iWAMain



iAw-URL eingeben

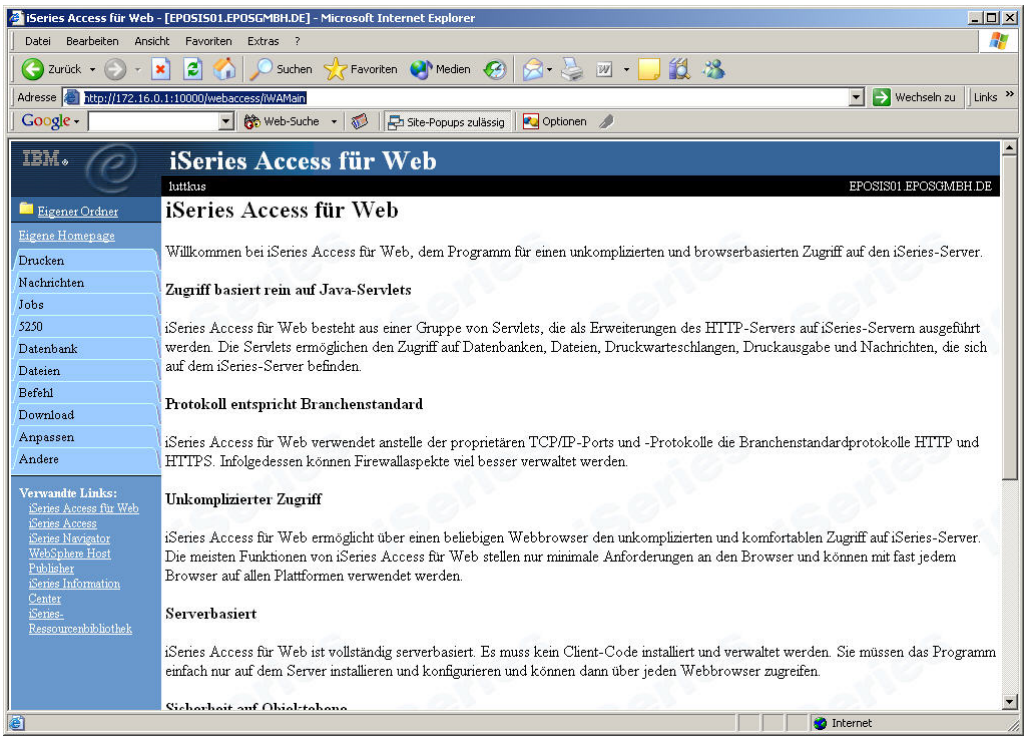
Danach wird ein Anmeldungsdialog gesendet.



Anmeldung für iAccess für Web

7.4.2
Seite 12

Melden Sie sich hier als gültiger OS/400-Benutzer mit dem entsprechenden Kennwort an.



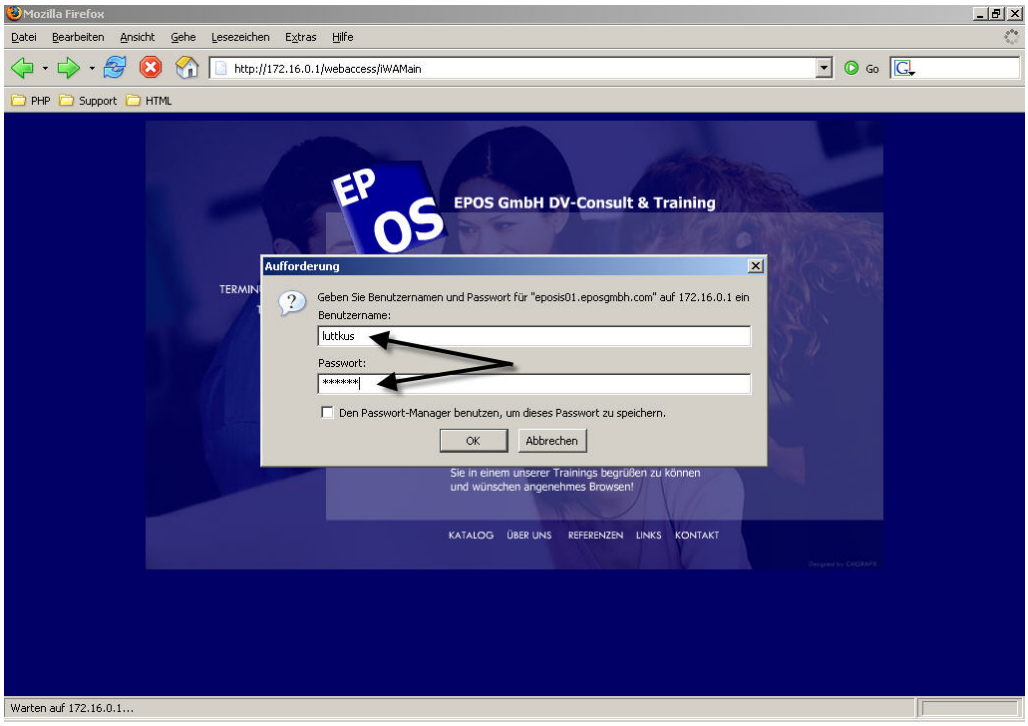
Start-Seite von iAW

Jetzt sehen Sie die Startseite von iAccess für Web.

iSeries Access für Web-Dienste

Nach der Installation und dem ersten Verbindungstest von iSeries Access für Web (IAW), werde ich in diesem Kapitel die einzelnen IAW-Dienste, ihre Konfiguration und die Arbeit mit diesen Diensten beschreiben.

Zuerst melde ich mich am IAW an.



IAW-Anmeldung

Dazu gebe ich im URL-Feld meines Browsers folgende Adresse ein:

http://172.16.0.1/webaccess/iWAMain

Ich gelange daraufhin auf die Startseite des IAW und konzentriere mich auf die linke Seite mit der Menü-Leiste und den zur Verfügung stehenden IAW-Funktionen.

Anmerkung:

Nach IBMs Dafürhalten könnte so eine Startseite aussehen! Haben Sie eigene Vorstellungen bezüglich des Layouts, so können Sie – HTML- und Web-Kenntnisse vorausgesetzt – eigene Startseiten-Layouts erstellen. Dieses ist jedoch eine Tätigkeit, die Sie erst umsetzen sollten, wenn Sie mit IAW und seinen Funktionen vertraut sind.

7.4.3
Seite 2

Folgende Dienste stehen Ihnen zur Verfügung:

Drucken – Mit Druckausgaben arbeiten

- Sie können die Druckausgabe auf dem iSeries-Server auflisten, verwalten und eine Vorschau anzeigen.
- Sie können eine Spooldatei anhalten, freigeben oder als nächste drucken lassen.
- Sie können die Druckausgabe auf einen anderen Drucker oder in eine andere Ausgabewarteschlange versetzen.
- Sie können die Druckausgabe an einen anderen iSeries-Server senden.
- Sie können die Druckausgabe im PDF-Format anzeigen, als Mail versenden oder senden.
- Sie können eine Vorschau von SCS- und AFP-Druckausgabe in den Formaten PNG, TIFF, PCL und AFP anzeigen.
- Sie können ASCII-Druckausgabe in Ihren Browser herunterladen.

Drucken – Mit PDF-Druckausgaben arbeiten

- Sie können die PDF-Druckausgabe auf dem iSeries-Server auflisten, verwalten und eine Vorschau anzeigen.
- Außerdem können Sie die PDF-Druckausgabe als Mail senden und speichern.

Drucken – Mit Druckern

- Sie können Drucker auf dem iSeries-Server auflisten und verwalten.
- Drucker- und Statusinformationen können angezeigt werden.
- Sie können die Druckereinheit anhängen und abhängen.
- Über die Druckerliste können Sie die Druckausgabe für einen bestimmten Drucker auflisten, verwalten und eine Vorschau anzeigen.
- Sie können Nachrichten, die für den Ausgabeprogrammjob anstehen, anzeigen und beantworten.
- Über die Funktionen „Anhalten“, „Freigeben“, „Starten“ und „Stoppen“ können Sie den Ausgabeprogrammjob verwalten, der dem Drucker zugeordnet ist.

Drucken – Mit PDF-Druckern arbeiten

- Sie können Ihre PDF-Drucker verwalten.
- Außerdem können Sie mit Ihrer Spooldatei als PDF-Datei arbeiten.

Drucken – Mit Internet-Druckern (IPP) arbeiten

- Hiermit können Sie Internet-Drucker auf dem iSeries-Server auflisten.
- Sie können mit Internet-Druckern auf dem iSeries-Server arbeiten.

Drucken – Mit Internet-Druckerfreigaben arbeiten

- Hiermit können Sie Internet-Druckerfreigaben auf dem iSeries-Server auflisten.
- Über die Liste der Internet-Druckerfreigaben können Sie Konfigurationsdaten und Statusangaben anzeigen.
- Außerdem können Sie über die Liste der Internet-Druckerfreigaben auf die Druckausgabe zugreifen und diese verwalten. Dies gilt auch für den iSeries-Drucker, der durch die Internet-Druckerfreigabe definiert ist.

6. Ergänzung 03/2006

Drucken – Mit Windows-Druckerfreigaben arbeiten

- Hiermit können Sie die Druckerfreigaben auf dem iSeries-Server inklusive Informationen und Status der Druckerfreigabe auflisten.
- Über die Liste der Druckerfreigaben können Sie auf Druckausgabe zugreifen und sie verwalten.

Drucken – Ausgabewarteschlangen

- Hiermit können Sie die Ausgabewarteschlangen auf dem iSeries-Server auflisten, verwalten und ihren Status anzeigen.
- Über die Liste der Ausgabewarteschlangen können Sie eine Ausgabewarteschlange anhalten und freigeben.
- Außerdem können Sie über die Liste der Ausgabewarteschlangen Druckausgaben auflisten, verwalten und eine Vorschau anzeigen.

Nachrichten – Nachrichten anzeigen

- Sie können die Nachrichten in der Nachrichtenwarteschlange für den angemeldeten Benutzer anzeigen, beantworten und verwalten.

Nachrichten – Nachricht senden

- Hiermit können Sie eine Nachricht an Benutzer und Nachrichtenwarteschlangen senden.

Nachrichten – Durchbruchnachricht senden

- Sie können eine „Sametime"-Ankündigung an „Sametime"-Benutzer von iSeries Access für Web senden.
- Sie können eine Durchbruchnachricht an iSeries-Workstations senden.

Nachrichten – Bedienernachrichten (QSYSOPR) anzeigen

- Sie können Nachrichten in der Nachrichtenwarteschlange für Systembediener anzeigen, beantworten und verwalten.

Nachrichten – Mit Nachrichtenwarteschlangen arbeiten

- Sie können die Nachrichtenwarteschlangen auf dem iSeries-Server auflisten.
- Nachrichten in Nachrichtenwarteschlangen können Sie auswählen, anzeigen und verwalten.

Jobs – Mit Benutzerjobs arbeiten (WRKUSRJOB)

- Sie können Benutzerjobs auf dem iSeries-Server auflisten und verwalten.
- Sie können Jobprotokolle für aktive Jobs anzeigen.
- Für beendete Jobs kann die Druckausgabe angezeigt werden.

Jobs – Mit Server-Jobs arbeiten

- Sie können Server-Jobs auf dem iSeries-Server auflisten und verwalten.
- Für Server-Jobs können Jobprotokolle angezeigt werden.

5250 – Mit „Aktive Sitzungen" arbeiten

- Aktive 5250-Benutzerschnittstellensitzungen auflisten.

5250 – Mit konfigurierten Sitzungen arbeiten

- Konfigurierte 5250-Benutzerschnittstellensitzungen auflisten.

5250 – Sitzung starten

- 5250-Benutzerschnittstellensitzung starten.

Datenbank – Mit Tabellen (Dateien)

- Sie können eine Liste der Datenbanktabellen auf dem iSeries-Server aufrufen.
- Für diese Tabellen können Sie Aktionen ausführen, ohne dass Kenntnisse über SQL und seine Syntax erforderlich sind.
- Über die Aktion „Schnellansicht" können Sie den Inhalt einer Tabelle in einer Liste aus Seiten anzeigen.
- Mit den Aktionen „Einfügen" und „Aktualisieren" können Sie Sätze in einer Tabelle hinzufügen und aktualisieren.
- Über die Aktion „SQL ausführen" können Sie eine eigene angepasste SQL-Anforderung für eine Tabelle erstellen.
- Über die Aktion „Daten in Tabelle kopieren" können Sie eine eigene angepasste Anforderung für das Kopieren von Daten erstellen.

Datenbank – Eigene SQL-Abfragen verwalten

- Sie können eine Liste der zuvor gespeicherten Anforderungen anzeigen.
- Über diese Liste können Sie Anforderungen ausführen oder bearbeiten.
- Mit den Aktionen „Kopieren", „Löschen" und „Umbenennen" können Sie Listen verwalten.
- Außerdem können Sie Verknüpfungen mit Anforderungen erstellen und verwalten.

6. Ergänzung 03/2006

7.4.3
Seite 6

Datenbank – SQL-Abfragen ausführen

- Hiermit können Sie SQL-Anweisungen dynamisch ausführen.
- Die Ausgabe wird als Liste aus Seiten oder in einem gängigen Dateiformat (z. B. Microsoft Excel oder Lotus 1-2-3) angezeigt. (Das Ausgabeformat richtet sich danach, wie der verwendete Browser konfiguriert ist und ob der Browser ein Plug-in für den ausgewählten Ausgabetyp findet.)
- Die Rückgabe der Daten können Sie anpassen, indem Sie spezifische Optionen für den Ausgabetyp festlegen.
- Mit einem SQL-Assistenten können Sie „SELECT“ -Anweisungen erstellen.
- Anforderungen können zur erneuten Verwendung gespeichert werden.

Datenbank – Daten in Tabelle kopieren

- Sie können vorhandene Datendateien vom PC in eine Datenbanktabelle auf dem iSeries-Server kopieren.
- Diese Datendateien können viele unterschiedliche Formate (z. B. Microsoft Excel und Lotus 1-2-3) haben.
- Sie können den Inhalt einer Tabelle ersetzen oder Daten zu einer vorhandenen Tabelle hinzufügen.
- Sie können eine neue Datenbanktabelle erstellen, die auf dem Inhalt einer Workstationdatei basiert.
- Anforderungen können zur erneuten Verwendung gespeichert werden.

Datenbank – Datentransfers importieren

- Hiermit können Sie Upload- und Downloadanforderungen, die aus der Client Access-Komponente „Datenübertragung“ stammen, in iSeries Access für Web-Anforderungen zum Kopieren von Daten und SQL-Anforderungen importieren.
- Importierte Anforderungen werden automatisch in das Format von iSeries Access für Web konvertiert.
- Die konvertierten Anforderungen können Sie auf dem iSeries-Server genauso wie andere Anforderungen zum Kopieren von Daten oder SQL-Anforderungen ausführen und bearbeiten.

Datenbank – Serverdaten extrahieren

- Sie können Serverobjektinformationen in eine Datenbanktabelle extrahieren.

Dateien – Dateien durchsuchen

- Sie können in den Verzeichnissen des Integrated File System auf dem iSeries-Server navigieren und sie durchsuchen.
- Dateien anzeigen, hochladen und herunterladen.
- Dateien kopieren und als Mail senden.
- Dateien und Verzeichnisse umbenennen und löschen.
- Dateien und Verzeichnisse komprimieren und dekomprimieren.
- Berechtigungen bearbeiten und das Besitzverhältnis von Objekten im Integrated File System ändern.

Dateien – Dateifreigaben

- Sie können eine Liste der Dateifreigaben von iSeries NetServer anzeigen.
- NetServer-Dateifreigaben durchsuchen und über die Liste der Dateifreigaben in den Verzeichnissen navigieren.
- Dateien aus einer NetServer-Dateifreigabe anzeigen und herunterladen.
- Dateien als Mail senden.
- Dateien in eine nicht schreibgeschützte Dateifreigabe hochladen.
- Dateien und Verzeichnisse in einer nicht schreibgeschützten Dateifreigabe kopieren, umbenennen und löschen.
- Dateien und Verzeichnisse in einer nicht schreibgeschützten Dateifreigabe komprimieren und dekomprimieren.
- Berechtigungen bearbeiten und das Besitzverhältnis von Objekten in einer nicht schreibgeschützten Dateifreigabe ändern.

Befehl – Mit selbst erstellten Befehlen arbeiten

- Über diese Option können Sie zuvor gespeicherte Befehle auflisten und verwenden.

Befehl – Selbst erstellte Befehle ausführen

- Hiermit können Sie auf dem iSeries-Server einen CL-Befehl ausführen.

Befehl – Nach Befehlen suchen

- Hiermit können Sie auf dem iSeries-Server nach CL-Befehlen suchen.

7.4.3
Seite 8

Downloads

Ein Download ist eine Datei, die auf dem iSeries-Server gespeichert ist und zur Verteilung über iSeries Access für Web verfügbar gemacht wird. Manche IBM Produkte sind als Downloads für iSeries Access für Web verfügbar. Sie haben auch die Möglichkeit, eigene Downloads zu erstellen. Für die Aktualisierung von Downloads und die Steuerung des Zugriffs auf die Downloads stehen eine Reihe von Verwaltungsfunktionen zur Verfügung.

- Hiermit können Sie verfügbare Downloads auflisten.
- Und die Downloads verwalten, auf die Sie zugreifen dürfen.

Download erstellen

- Hiermit können Sie neue Downloads erstellen.

Anpassen

Mit iSeries Access für Web können Sie Vorgaben, Richtlinien und Einstellungen für die Nutzung und Konfiguration von IAW-Diensten erstellen.

Anpassen – Vorgaben

- Sie können die Vorgabeneinstellungen aktualisieren.
- Spalten für das Anzeigen von Ausgabe in Listenform auswählen.
- Navigationselemente anzeigen oder ausblenden.
- Die Anzahl der Einträge/Objekte anpassen, die pro Seite angezeigt werden sollen.

Anpassen – Richtlinien

- Sie können Richtlinieneinstellungen für andere Benutzer- und Gruppenprofile anpassen.
- Richtlinieneinstellungen für Profile kopieren oder zurücksetzen.
- Standardrichtlinieneinstellungen für alle Benutzer verwalten.

Anpassen – Einstellungen

- Sie können Einstellungen für dieses Exemplar von iSeries Access für Web konfigurieren.

5250-Emulation

Die 5250-Benutzerschnittstelle von iSeries Access für Web stellt eine Webschnittstelle zur Verfügung, über die Sie 5250-Sitzungen konfigurieren, starten und verwenden können. Außerdem werden Makros für die Automatisierung wiederholt benötigter Tasks, anpassbare Tastenblöcke für den schnellen Zugriff auf häufig verwendete Tasten und Makros sowie eine Webansicht und eine herkömmliche Ansicht unterstützt.

Aufruf des 5250-Dienstes

Um mit 5250-Sitzungen zu arbeiten, klicke ich auf das Register 5250 in der Dienstleiste des IAW.

Jetzt öffnet sich folgender Dialog.

Neue Sitzung starten

7.4.3.1
Seite 2

Mit dem Befehl „Sitzung starten“ können Sie Sitzungen mit der 5250-Benutzerschnittstelle starten. Einstellungen, die während der aktiven Sitzung nicht geändert werden können, werden beim Starten der Sitzung angegeben. Für Einstellungen, die auch bei einer aktiven Sitzung geändert werden können, werden Standardwerte verwendet.

Da ich bisher noch keine 5250-Sitzungen im IAW konfiguriert habe, klicke ich auf „Sitzung starten“.

Folgende Startparameter sind einzustellen.

5250-Startparameter

Server

Der iSeries-Server, auf dem iSeries Access für Web ausgeführt wird, wird als Standard-Server für die Sitzung verwendet. Sie können Sitzungen für jeden beliebigen iSeries-Server starten, selbst wenn auf diesem Server iSeries Access für Web nicht ausgeführt wird. Die Servereinstellungen definieren, wie die 5250-Benutzerschnittstelle mit dem angegebenen iSeries-Server gestartet wird.

Port

Der Wert „Port“ gibt die Port-Nummer an, die vom iSeries-Telnet-Server (TN5250) verwendet wird, um eingehende Anforderungen für 5250-Sitzungen anzunehmen. Dieser Wert ist standardmäßig 23.

Codepage

Die Codepage hat Auswirkungen auf die Zeichendatenkonvertierung zwischen der EBCDIC-Codepage des Server und der Zeichencodierung des Browser. Der Standardwert basiert auf der CCSID (Coded Character Set ID – ID des codierten Zeichensatzes), die für die authentifizierte Benutzer-ID von iSeries Access für Web angegeben ist. Die folgenden Werte für Codepages werden unterstützt:

37	Brasilianisches Portugiesisch, Niederländisch, Englisch, Portugiesisch
273	Deutsch
277	Dänisch, Norwegisch
278	Finnisch, Schwedisch
280	Italienisch
284	Spanisch
285	Englisch (Großbritannien)
290	Japanisch Katakana Einzelbyte
297	Französisch
420	Arabisch
424	Hebräisch
500	Albanisch, Niederländisch (Belgien), Englisch (Belgien), Französisch (Belgien), Französisch (Kanada), Westeuropäisch mehrsprachig
838	Thailändisch
870	Kroatisch, Tschechisch, Ungarisch, Polnisch, Rumänisch, Slowenisch, Slowakisch, Mitteleuropäisch mehrsprachig, Sprachen mit lateinischem Zeichensatz 2
871	Isländisch
875	Griechisch
924	Europäisch mehrsprachig mit Euro (wie ISO 8895-15)
930	Japanisch Katakana Kanji Mischbyte
933	Koreanisch Mischbyte
935	Vereinfachtes Chinesisch Mischbyte
937	Traditionelles Chinesisch Mischbyte
939	Japanisch Lateinisch Kanji Mischbyte
1025	Bulgarisch, Mazedonisch, Russisch, Serbisch, Kyrillisch mehrsprachig
1026	Türkisch
1112	Lettisch, Litauisch
1122	Estnisch
1123	Ukrainisch
1137	Devanagari
1140	Brasilianisches Portugiesisch, Englisch, Portugiesisch mit Euro

1141	Deutsch mit Euro
1142	Dänisch, Norwegisch mit Euro
1143	Finnisch, Schwedisch mit Euro
1144	Italienisch mit Euro
1145	Spanisch mit Euro
1146	Englisch (Großbritannien) mit Euro
1147	Französisch mit Euro
1148	Westeuropäisch mehrsprachig mit Euro
1149	Isländisch mit Euro
1153	Tschechisch, Mitteleuropäisch mehrsprachig, Sprachen mit lateinischem Zeichensatz 2 mit Euro
1154	Kyrillisch mehrsprachig mit Euro
1155	Türkisch mit Euro
1156	Lettisch, Litauisch mit Euro
1157	Estnisch mit Euro
1158	Ukrainisch mit Euro
1160	Thailändisch mit Euro
1364	Koreanisch Erweitert Mischbyte
1388	Vereinfachtes Chinesisch (GBK) Mischbyte
1399	Japanisch lateinischer Zeichensatz Kanji (JIS-99) Mischbyte mit Euro

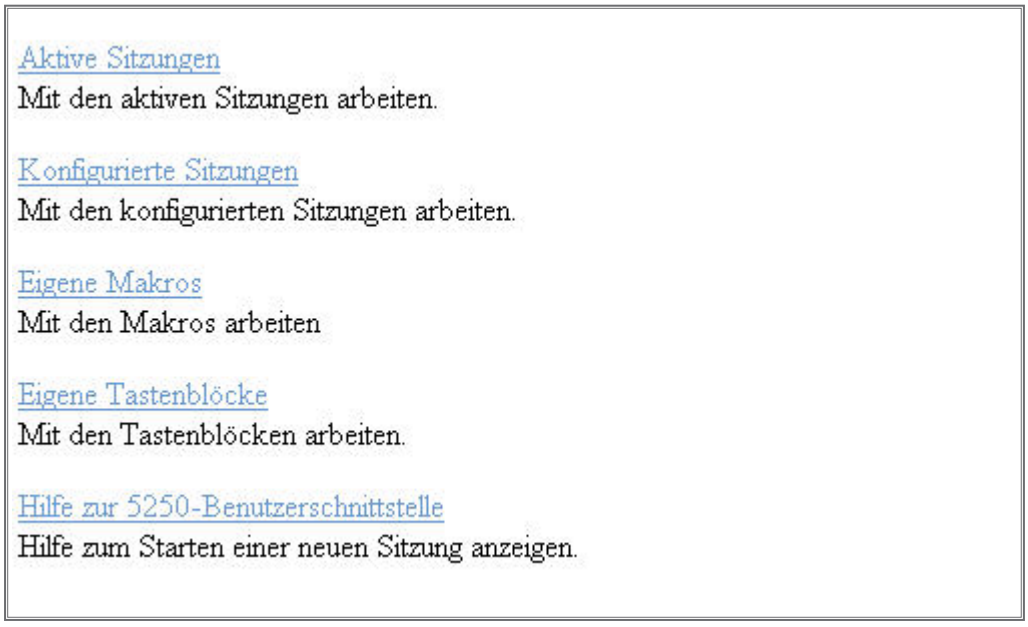
Ich gebe hier den Hostnamen meiner iSeries an. Es ist hier ebenfalls möglich, die IP-Adresse hart einzugeben.

Workstation-ID

Die Workstation-ID gibt den Einheitennamen (DEVNAME) an, der für die 5250-Sitzung verwendet wird.

Ich gebe hier den Namen WEB ein und erzwinge, falls der Bildschirm mehrfach gestartet wird, die Eindeutigkeit des Bildschirmnamen durch markieren der Parameter „Gleiche Werte…“.

Wenn Sie den Dialog ein wenig nach unten scrollen, so finden Sie weiteren Funktionen, die die Arbeitsweise, das Aussehen und die Funktionen Ihres 5250-Dialogs steuern.



Weitere Funktionen

Aktive Sitzungen – Ändert Parameter in aktiven Bildschirmsitzungen. Nicht alle möglichen Parameter stehen zur Verfügung.

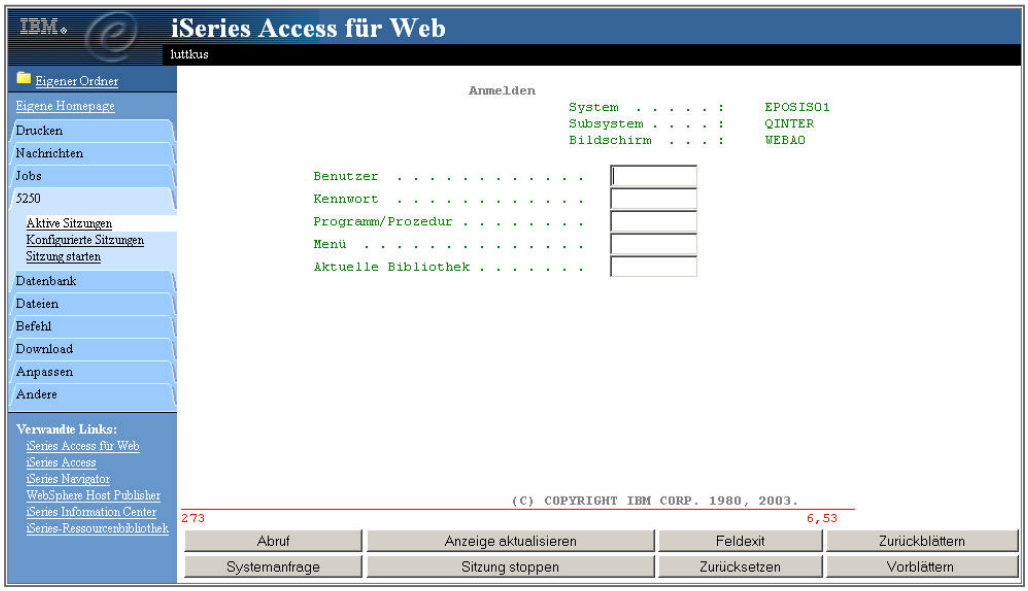
Konfigurierte Sitzungen – Ändert Parameter in nicht-aktiven Bildschirmsitzungen. Alle veränderbaren Parameter stehen zur Verfügung.

Eigene Makros – Haben Sie eigene Makros erstellt, so können diese verwaltet und bearbeitet werden.

Eigene Tastenblöcke – Sie können definieren, welche Funktionstasten im Dialog zugeordnet werden, welche Funktionstasten angezeigt werden und Funktionstasten Makros zuweisen.

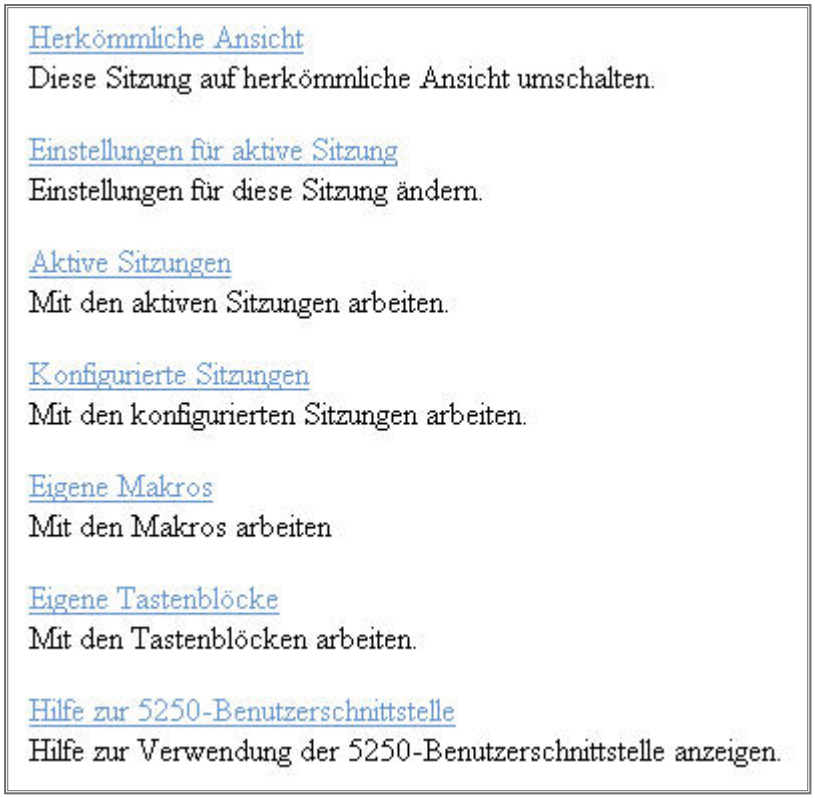
Nachdem alle Parameter versorgt sind, klicke ich auf „Sitzung starten“.

7.4.3.1
Seite 6



iSeries-Signon

Auch in diesem Dialog besteht die Möglichkeit, weiter nach unten zu scrollen.



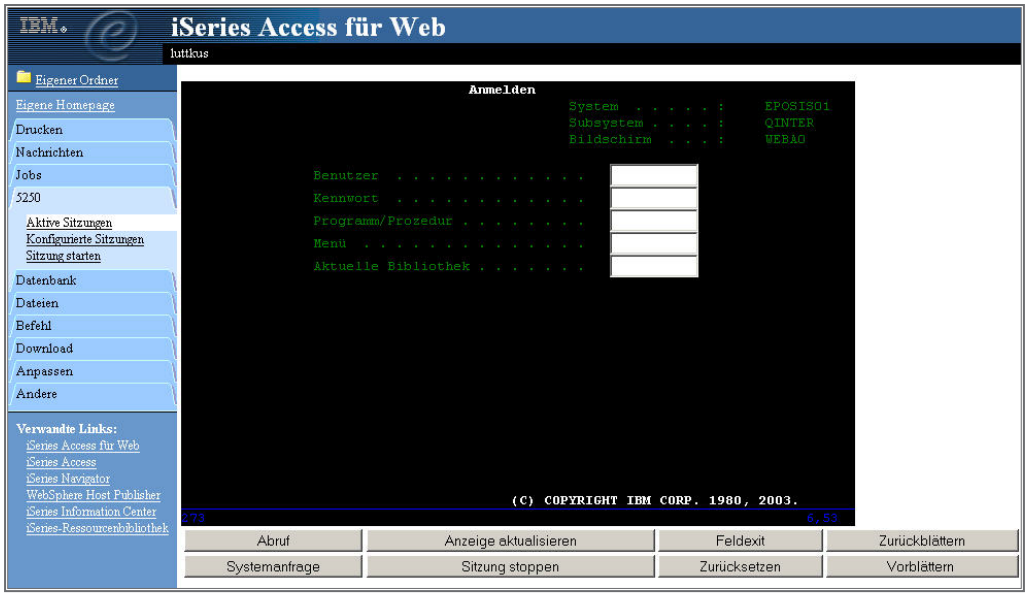
Bildschirmbearbeitungsmöglichkeiten

Auch hier sehen Sie wieder eine Reihe von Bearbeitungsmöglichkeiten, die im Wesentlichen ein Duplikat der bereits oben angesprochenen Funktionen sind.

Herkömmliche Sicht

Wenn wir uns jetzt unseren Bildschirmdialog ansehen, so befindet der sich in der Web-Ansicht. Das bedeutet, dass das Aussehen eines 5250-Dialogs an eine Web-Umgebung angepasst wurde: Heller Hintergrund, Proportionalschrift etc.

Sollte Ihnen dieses Layout generell nicht gefallen, weil Sie besser mit der Standard-Grün/Schwarz-Darstellung der 5250-Emulationen zurechtkommen, so wählen Sie einfach diese Option an.



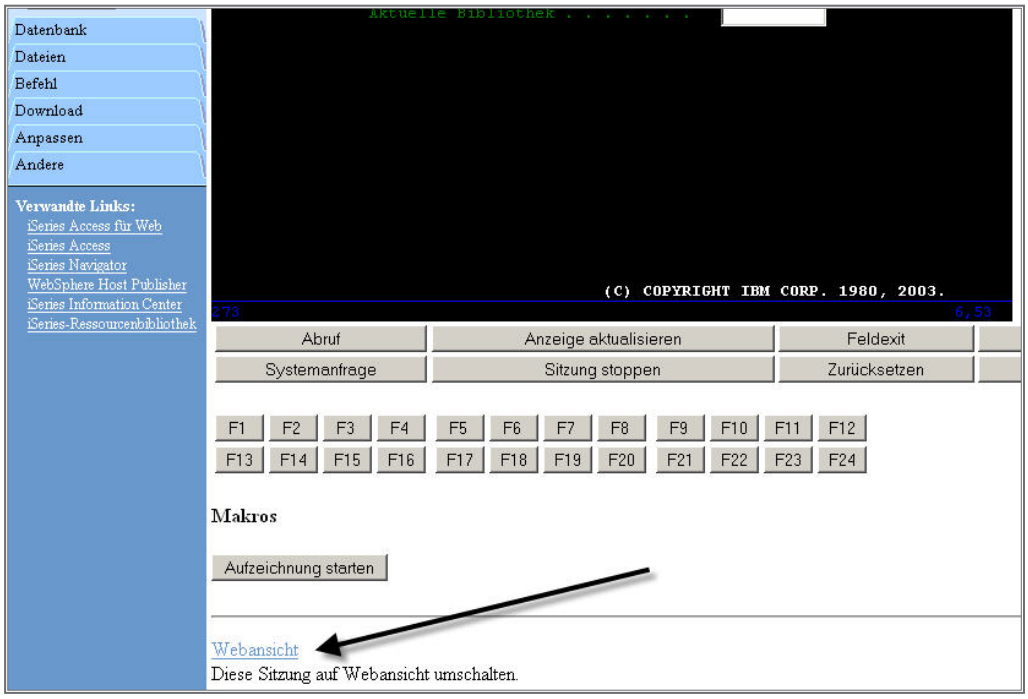
Grün/Schwarz-Darstellung

Mir persönlich gefällt die Web-Ansicht besser.

Scrollen Sie die Seite ein wenig nach oben.

7.4.3.1.1

Seite 2



Zurück zur Web-Ansicht

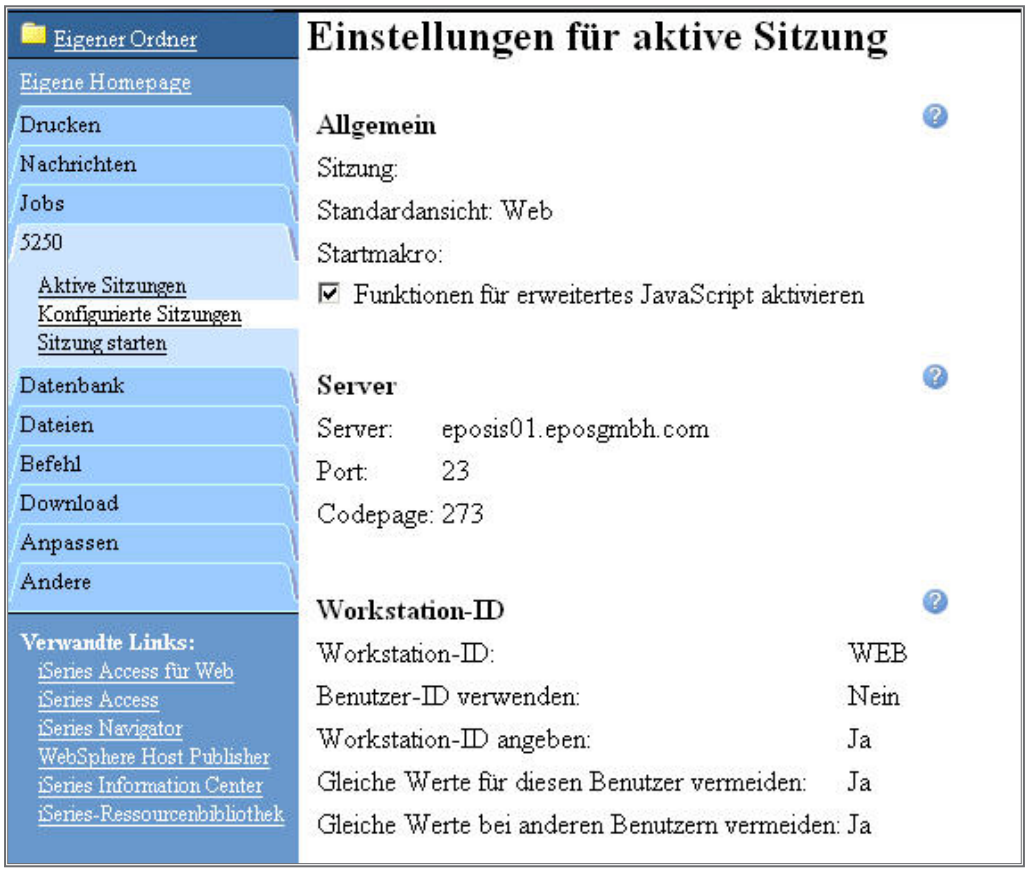
Dann finden Sie unterhalb der Funktionstasten den Link Web-Ansicht und der Dialog wird entsprechend angezeigt.

Achtung! Die IAW-5250-Dialoge sind grundsätzlich mittels Funktionstasten zu bedienen. Drücken Sie auf Ihrer Tastatur F3, so wird auf der Seite die 5250-Funktion F3=verlassen ausgelöst.

Achtung! Achtung! Eine Funktionstaste hat jedoch eine andere Funktionsweise: PC-Enter entspricht der Datenfreigabetaste. Das lässt sich auch nicht umkonfigurieren. Will ein Anwender von Feld zu Feld springen, muss er das mit Hilfe der PC-Tabulator-Taste tun!

Einstellungen für aktive Sitzungen

Beabsichtigen Sie die Konfiguration für eine aktive Sitzung zu ändern, so wählen Sie den Link „Einstellungen für aktive Sitzungen“ an.



Einstellungen für aktive Sitzungen

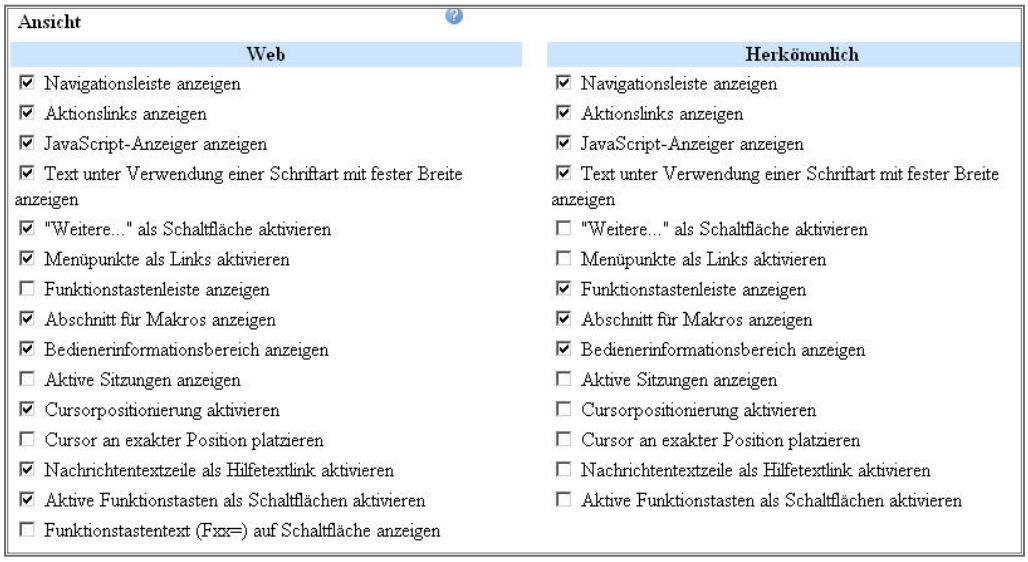
Es ist unschwer zu erkennen, dass die meisten Parameter nicht verändert werden können.

Deaktivieren Sie den Parameter „Funktionen für erweitertes JavaScripts aktivieren“, so können Funktionstasten nur über Mausklicks ausgelöst werden.

Auch diese Seite lässt sich nach unten durchscrollen.

Sie sehen dann Parameter, mit denen Sie das Aussehen der Bildschirmsitzung für die Web-Ansicht und die herkömmliche Sicht verändern können.

7.4.3.1.2
Seite 2



Aussehen verändern

Navigationsleiste anzeigen

Verwenden Sie diese Einstellung um anzugeben, ob die Navigationsleiste von iSeries Access für Web angezeigt wird, wenn eine 5250-Benutzerschnittstellensitzung aktiv ist.

Aktionslinks anzeigen

Diese Einstellung steuert, ob Aktionslinks wie beispielsweise "Einstellungen für aktive Sitzung", "Eigene Makros" und "Eigene Tastenblöcke" in der Benutzerschnittstelle für die 5250-Sitzung angezeigt werden.

JavaScript-Anzeiger anzeigen

Mit dieser Einstellung geben Sie an, ob der JavaScript-Anzeiger unten auf der Seite mit der 5250-Benutzerschnittstelle angezeigt werden soll. Der JavaScript-Anzeiger erfordert einen JavaScript-fähigen Browser mit aktiviertem JavaScript.

Text unter Verwendung einer Schriftart mit fester Breite anzeigen

Diese Einstellung steuert, ob die Sitzung mit der 5250-Benutzerschnittstelle eine Schriftart mit fester Breite oder eine Proportionalschriftart verwendet.

„Weitere...“ als Schaltfläche aktivieren

Geben Sie mit dieser Einstellung an, ob Felder, die „Weitere...“ enthalten, als Schaltflächen aktiviert oder als Text angezeigt werden. Wenn Felder mit „Weitere...“ als Text angezeigt werden, können zusätzliche Seiten mit der Taste zum Vorblättern auf einem Tastenblock oder mit dieser Taste auf der Tastatur abgerufen werden

Menüpunkte als Links aktivieren

Diese Einstellung gibt an, ob Auswahlnummern in Menüs (zwischen 1 und 99) als Links angezeigt werden. Wenn diese Einstellung nicht aktiviert ist, muss zur Auswahl eines Menüpunkts die Nummer des Menüpunkts in der Befehlszeile eingegeben werden.

Funktionstastenleiste anzeigen

Mit dieser Einstellung wird die Funktionstastenleiste unter der 5250-Anzeige angezeigt. Die Funktionstastenleiste enthält alle Funktionstasten (F1 bis F24) als Schaltflächen. Das Layout und der Inhalt der Funktionstastenleiste kann nicht angepasst werden. Mit der Option „Eigene Tastenblöcke“ können Sie einen angepassten Tastenblock über, unter oder neben der 5250-Anzeige anzeigen.

Abschnitt für Makros anzeigen

Diese Einstellung gibt an, ob Makros aufgezeichnet und wiedergegeben werden können. Der Zugriff auf die Liste „Eigene Makros“ wird durch diese Einstellung nicht eingeschränkt. Weitere Informationen zu Makros (einschließlich Angaben zur ihrer Erstellung und Verwendung) finden Sie unter „Makros“

Bedienerinformationsbereich anzeigen

Mit dieser Einstellung geben Sie an, ob der Bedienerinformationsbereich für die 5250-Sitzung angezeigt wird. Der Bedienerinformationsbereich ist der Bereich, in dem Informationen wie beispielsweise der Anzeiger für das Sperren der Eingabe oder die Cursorposition angezeigt werden.

Aktive Sitzungen anzeigen

Diese Einstellung gibt an, ob die Liste der aktiven Sitzungen auf jeder Seite der 5250-Sitzung angezeigt wird. Durch das Anzeigen der Liste mit den aktiven Sitzungen auf jeder 5250-Seite können Sie schnell zwischen mehreren aktiven Sitzungen wechseln.

Cursorpositionierung aktivieren

Diese Einstellung gibt an, ob Nicht-Eingabefelder als Links oder als normaler Text implementiert werden. Wenn Nicht-Eingabefelder als Links implementiert werden, wird bei Auswahl eines Links die aktuelle Cursorposition für die aktive Sitzung mit der ersten Position des Feldes aktualisiert. Dies ist bei Funktionen hilfreich, die die aktuelle Cursorposition verwenden. Wenn „Nachrichtentextzeile als Hilfetextlink aktivieren“ ebenfalls aktiviert ist, wird die Einstellung für Cursorpositionierung von der Nachrichtentexteinstellung überschrieben.

7.4.3.1.2
Seite 4

Cursor an exakter Position platzieren

Mit dieser Einstellung können Sie die Möglichkeit aktivieren, dass der Cursor an einer exakten Position in einem Feld platziert wird. Dies ist hilfreich, wenn Sie eine Anwendung verwenden, in der sich der Cursor bei kontextbezogenen Operationen an einer exakten Position in einem Feld befinden muss. Diese Einstellung ist nur dann wirksam, wenn die Einstellung „Cursorpositionierung aktivieren“ ebenfalls aktiviert ist.

Nachrichtentextzeile als Hilfetextlink aktivieren

Mit dieser Einstellung können Sie die Nachrichtentextzeile als Link zur Hilfe für die Nachricht aktivieren. Wenn diese Einstellung nicht aktiviert ist, kann die Einstellung „Cursorpositionierung aktivieren“ verwendet werden, um die Hilfe für die Nachricht abzurufen.

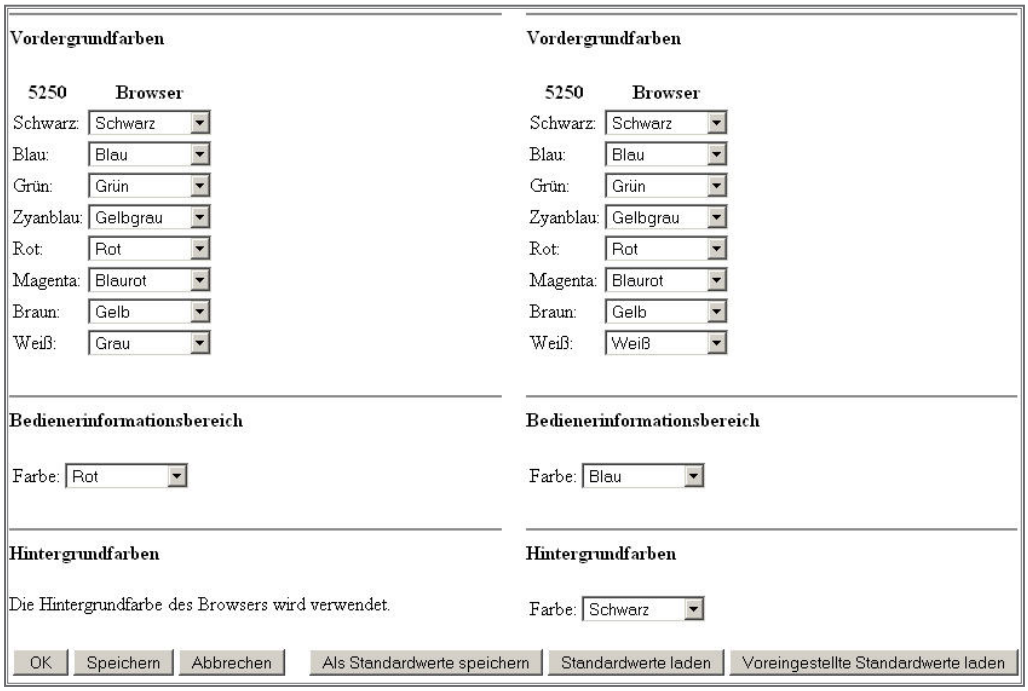
Aktive Funktionstasten als Schaltflächen aktivieren

Diese Einstellung gibt an, ob die aktiven Funktionstasten unten in der Anzeige als Schaltflächen aktiviert oder als Text angezeigt werden. Wenn die Funktionstasten als Text angezeigt werden, können die Funktionen unter Verwendung der Funktionstastenleiste eines konfigurierten Tastenblocks oder der Funktionstasten auf der Tastatur abgerufen werden. Zu jeder dieser Optionen gibt es Überlegungen bezüglich der Konfiguration.

Funktionstastentext (Fxx=) auf Schaltfläche anzeigen

Diese Einstellung gibt an, ob die Angabe „Fxx=“ im Schaltflächentext enthalten ist, wenn die Option „Aktive Funktionstasten als Schaltflächen aktivieren“ ausgewählt wurde. Werden aktive Funktionen nicht als Schaltflächen angezeigt, wird diese Einstellung ignoriert. Diese Einstellung ist nur für die Webansicht verfügbar. Bei der herkömmlichen Ansicht wird „Fxx=“ immer angezeigt, wenn aktive Funktionen als Schaltflächen auf der Anzeige erscheinen.

Wie in jeder guten 5250-Emulation, lässt sich auch im IAW-5250 Umfeld, die Farbgebung beeinflussen.



Farbeinstellungen

Links sehen Sie die Einstellungen für die Web-Ansicht, rechts die für die herkömmliche Sicht.

Die Logik der Konfiguration besteht darin, die acht Twinax-Farben (Spalte 5250) durch individuell wählbare Farben (Spalte Browser) zu ersetzen.

7.4.3.1.2

Seite 6

Download

Die Funktion Download des iSeries Access für Web gibt dem iSeries Access für Web-Administrator die Möglichkeit, Anwendern Dateien bereitzustellen, die er für seine tägliche Arbeit braucht.

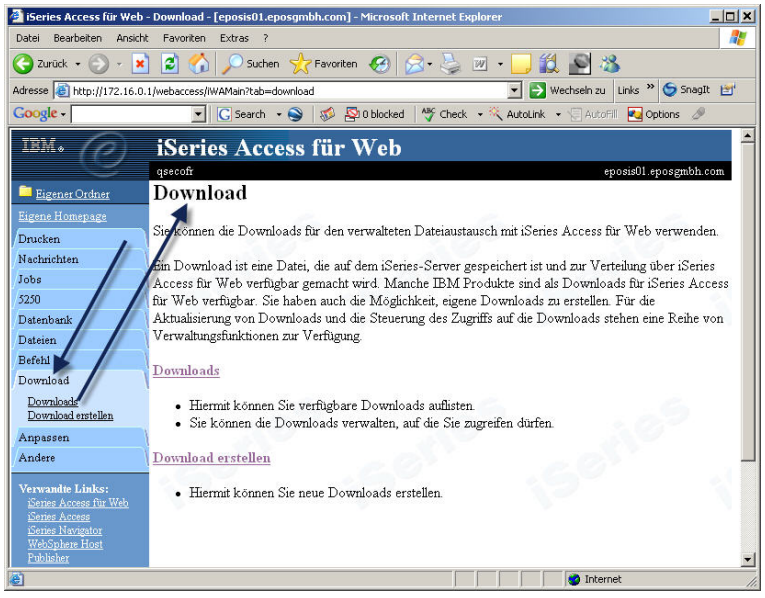
Auch IBM selbst macht von dieser Funktion Gebrauch, indem dem Anwender hier Dateien bereitgestellt werden, die er im iSeries Access-Umfeld unter Umständen gut einsetzen kann.

Dazu gehören zwei prozessorabhängige Versionen des iSeries Access für Linux und – für uns unmittelbar wichtiger – ein Browser-Plugin des AFP-Workbench Viewers, mit dem man sich Spooldateien optimal anzeigen lassen kann.

Ich kann Ihnen nur dringend empfehlen dieses Plug-In herunterzuladen, wenn Sie später optimal mit der Funktion „Drucken“ im iSeries Access für Web arbeiten wollen.

Deshalb werden wir gemeinsam das AFP-Workbench Plug-In herunterladen.

Dazu markiere ich im Funktionsmenü auf der linken Seite die Funktion „Download“.



Download Menü

In diesem Menü sehen Sie zwei Einträge:

- Download
- Download erstellen

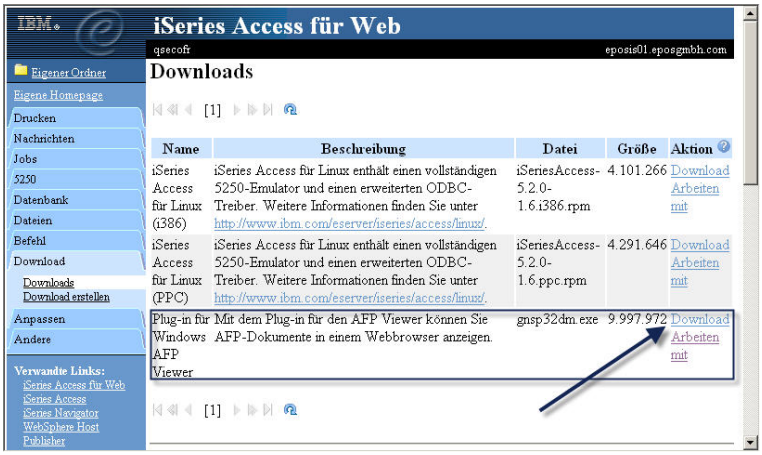
7.4.3.2

Seite 2

Bestehenden Download nutzen

Zuerst rufe ich die Funktion „Download“ auf, indem ich auf den entsprechenden Link klicke. Die Funktion „Download“ stellt Dateien, die zum Herunterladen bereit sind zur Verfügung.

Sie sehen dann folgenden Dialog:



AFP Workbench Viewer

Sie sehen hier folgende Download-Einträge:
- iSeries Access für LINUX (i386).
- iSeries Access für LINUX (ppc).

Diese beiden Einträge interessieren uns derzeit nicht.

Der nächste Eintrag:
- Plug-in für Windows AFP Viewer,

ist für uns jedoch von Interesse. Wir wollen nämlich diese Datei herunterladen und den Viewer sogleich installieren.

Dazu klicken wir in der Tabellenspalte „Aktion“ auf „Download“. Der Viewer wird in einer sich selbst extrahierenden Datei mit dem Namen gnsp32dm.exe angeboten.

Download beginnen

7.4.3.2.1
Seite 2

Je nach Sicherheitseinstellungen an Ihrem Arbeitsplatz beginnt der Download sofort oder oben gezeigtes Fenster öffnet sich.

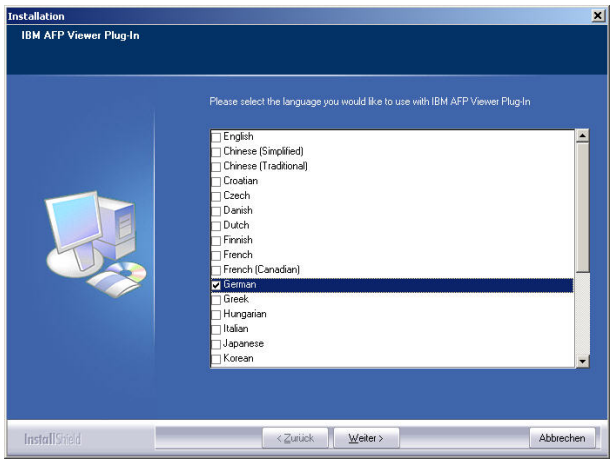
Ich schließe diesen Dialog mit „Ausführen".

Eventuell folgt dann noch eine weitere Sicherheitsmeldung der Windowsumgebung.

Sicherheitswarnung

Auch diesen Dialog quittiere ich mit „Ausführen".

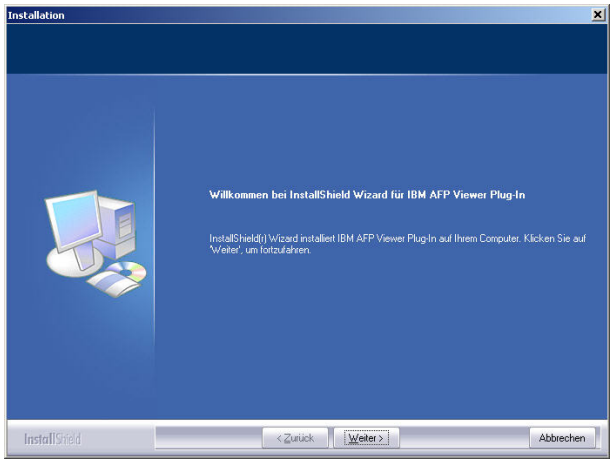
Danach beginnt die eigentliche Installation des Viewer Plug-ins.



Sprachauswahl

Ich wähle hier German als Sprache und klicke auf „Weiter".

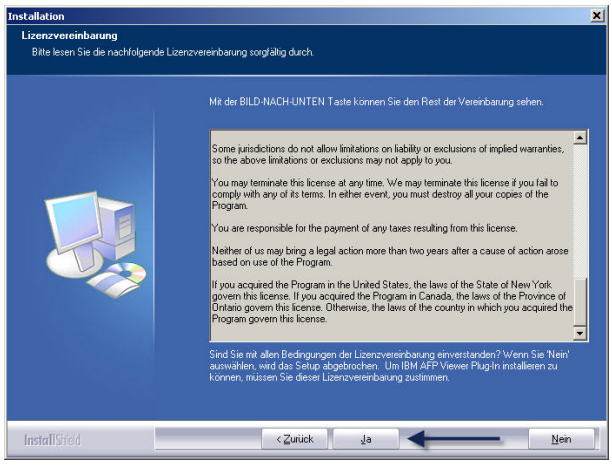
Ich werde jetzt auf Deutsch Willkommen geheißen:



Willkommen

Diesen Dialog beende ich mit „Weiter“.

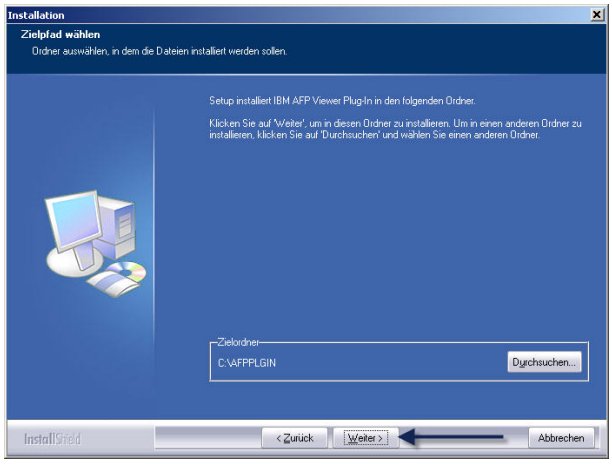
Danach werden lizenzrechtliche Informationen angezeigt.



Lizenzbedingungen

Ich akzeptiere die Lizenzbedingungen mit „Ja“.

Danach wird nach dem Installationsverzeichnis auf meinem Arbeitsplatz gefragt.

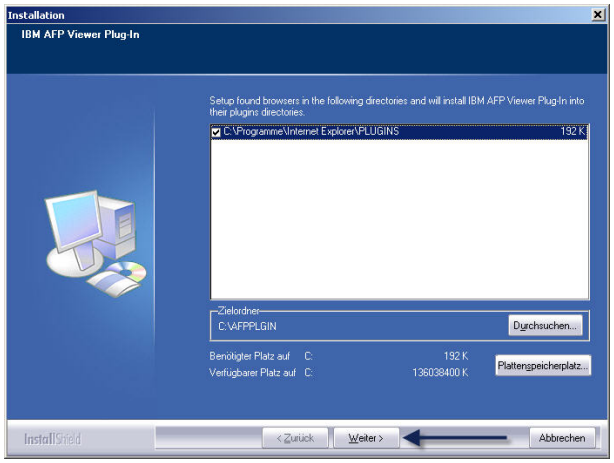


Installationsverzeichnis

Ich belasse es bei den Standardeinstellungen und beende den Dialog mit „Weiter“.

Im nächsten Fenster wird angezeigt, ob und welcher Plug-in-fähige Browser auf meinem Arbeitsplatz installiert wird.

7.4.3.2.1
Seite 4



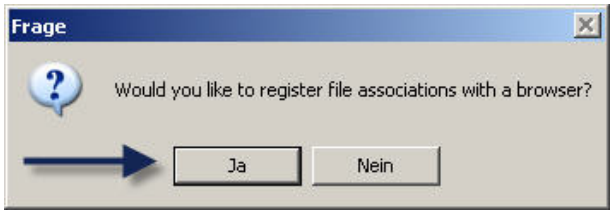
Browser-Erkennung

Sie sehen, dass auf meinem Arbeitsplatz der Internet Explorer von Microsoft gefunden wurde.

Langsam nähert sich der Installations-Dialog dem Ende.

Es kommen lediglich drei Fenster.

1.) Sollen AFP-Dateien mit dem Viewer assoziiert werden.



Assoziationen erlauben

Ich bestätige die Angabe mit „Ja“.

2.) Ist diese Arbeit korrekt beendet worden?

Assoziationen sind registriert

Ich bestätige diese Ausgabe mit „OK“.

3.) Die Installation des Plug-ins ist beendet.

Auch diesen Dialog bestätige ich mit „OK“.

Einen neuen Download erstellen

Nachdem wir im vorherigen Abschnitt einen bestehenden Download für uns genutzt haben, werden wir jetzt einen neuen Download erstellen, der von unseren Anwendern verwendet werden kann.

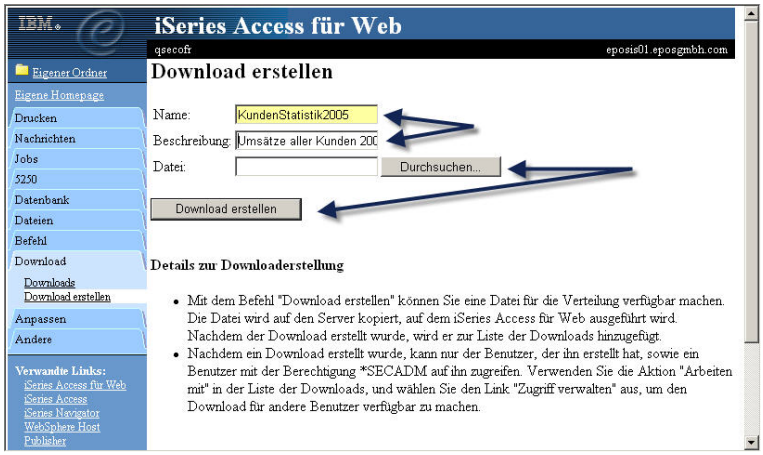
Folgende Situation ist gegeben. Im Unternehmen ist eine Kunden-bezogene Umsatzstatistik für das Jahr 2005 in Form eines Excel-Arbeitsblattes erstellt worden.

Alle Mitarbeiter der Buchhaltung, die den Gruppenprofilen FIBU01 und FIBU02 zugeordnet sind, sollen diese Kunden-Umsatz-Statistik mit Hilfe ihrer Download-Funktion an ihren Arbeitsplatz herunterladen können.

Dazu klicke ich im Funktionsmenü „Download“ auf den Eintrag „Download erstellen“.

Download erstellen

Folgender Dialog öffnet sich.



Download erstellen

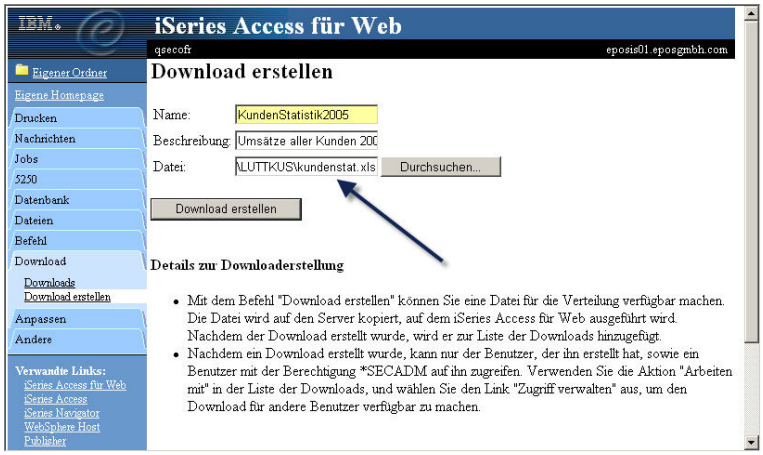
Im Parameter „Name“ tragen ich einen beschreibenden Begriff ein – hier: KundenStatistik2005. Dieses ist der Anzeigename der später in der Download-Funktion angezeigt wird.

Im Parameter „Beschreibung“ trage ich eine zusätzliche Erläuterung ein – hier: Umsätze aller Kunden 2005. Dieser Begriff dient ausschließlich der Anzeige und kann auch weggelassen werden.

Mittels „Durchsuchen“ suche ich auf meinem Dateiserver nach der Datei kundenstat.xls.

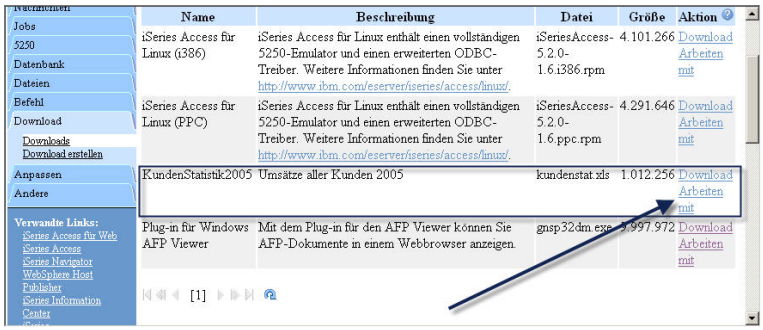
7.4.3.2.2

Seite 2



Download fertig stellen

Jetzt beende ich den Vorgang mit Download erstellen und wechsele in die Anzeige bereits definierter Downloads.



Anzeige der Download KundenStatistik2005

Als dritten Eintrag sehen Sie den von mir erstellten „Download Kunden Statistik2005“.

Zum jetzigen Zeitpunkt kann nur der Download-Ersteller auf diesen Download zugreifen. Um jetzt den beiden Benutzergruppen FUBU01 und FIBU02 den Zugriff zu gewähren, klicke ich auf den Link „Arbeiten mit“.

Mit Download arbeiten

Folgende Funktionen stehen hier bereit:

Name und Beschreibung bearbeiten
Mit der Aktion „Name und Beschreibung bearbeiten“ können Sie den Namen und die Beschreibung des Downloads ändern.

Datei aktualisieren
Mit der Aktion „Datei aktualisieren“ können Sie die herunterzuladende Datei ändern. Sie können sie beispielsweise verwenden, um die herunterzuladende Datei durch eine neuere Version zu ersetzen.

Löschen
Mit der Aktion „Löschen“ können Sie die Datei vom Server und aus der Liste der verfügbaren Downloads entfernen.

Zugriff verwalten
Mit der Aktion „Zugriff verwalten“ können Sie den Zugriff auf den Download steuern. Nachdem ein Download erstellt wurde, kann nur der Benutzer, der ihn erstellt hat, sowie ein Benutzer mit der Berechtigung *SECADM auf ihn zugreifen.

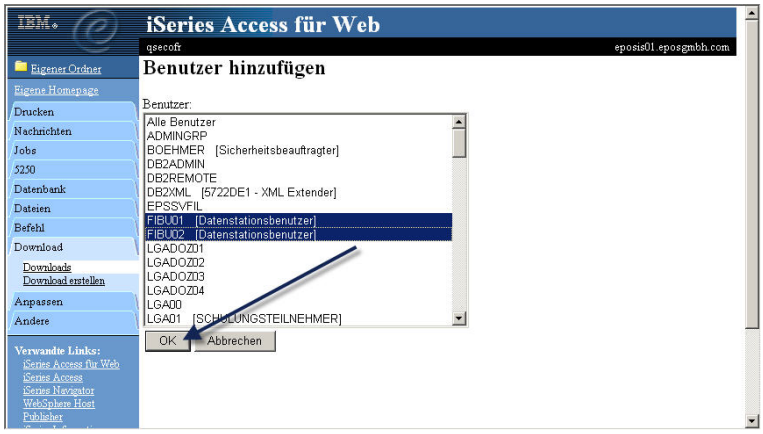
Über die Aktion „Zugriff verwalten“ können Sie den Download für andere Benutzer verfügbar machen.

Zugriff verwalten

Sie sehen hier, dass ausschließlich der QSECOFR derzeit einen Zugriff auf den Download hat, und dass nur er das Recht hat, den Download zu administrieren (Verwalten). Alle anderen Anwender sind von der Nutzung ausgeschlossen.

Um weiteren Benutzerkreisen die Anwendung der Downloads zu ermöglichen, klicke ich in der Spalte „Zugriff zulassen“ auf „Hinzufügen“.

7.4.3.2.2
Seite 4



Benutzer auswählen

In der Benutzerliste wähle ich die beiden Gruppenprofile FIBU01 und FIBU02 aus und bestätige die Übernahme der markierten Profile mit „OK“.

Veränderte Zugriffe

Wie Sie sehen ist die Zugriffsanzeige verändert worden.

- Download-Nutzer sind: QSECOFR (Eigner), die Mitglieder von FIBU01 und die Mitglieder von FIBU02
- Download-Administrator ist: QSECOFR
- Alle restlichen Benutzer sind ausgeschlossen.

Drucken

Die Steuerung von Output und Druckern in diversen Varianten ist ein weiteres Funktionsmerkmal des iSeries Access für Web.

Nur hier finden Sie die Möglichkeit sämtliche Druckfunktionen des System i zu administrieren. Der iSeries Access für Web geht über das hinaus, was der iSeries Navigator und das CL-Interface des i5/OS gemeinsam (!) bereitstellen.

Ich werde mit Ihnen in diesem Kapitel zuerst die Steuerung der Basis-Druckfunktionen des System i besprechen:

Einem erfahrenen Administrator des System i sind die meisten dieser Funktionen vertraut, da sie die direkte Übersetzung von CL-Befehlen darstellen.

11. Ergänzung 06/2007

7.4.3.3

Seite 2

Auswahl „Druckausgabe“

In der Auswahl „Druckausgabe“ stehen Ihnen folgende Funktionen zur Verfügung:

- Druckausgabe auf dem iSeries-Server auflisten, verwalten und eine Vorschau anzeigen.
- Spooldatei anhalten, freigeben oder als nächste drucken lassen.
- Druckausgaben auf einen anderen Drucker oder in eine andere Ausgabewarteschlange versetzen.
- Druckausgaben an einen anderen iSeries-Server senden.
- Druckausgaben im PDF-Format anzeigen, als Mail versenden oder senden.
- Eine Vorschau von SCS- und AFP-Druckausgaben in den Formaten PNG, TIFF, PCL und AFP anzeigen.
- ASCII-Druckausgabe in Ihren Browser herunterladen.

Basis Funktionen des Drucker-Menüs

Ich werde zuerst folgende Funktionen mit Ihnen besprechen:

- Druckausgabe – Implementation des WRKSPLF-Befehls.
- Drucker – Implementation des WRKWTR-Befehls.
- Druckerfreigaben – Mit für das Windows-Netzwerk freigegebenen Druckern arbeiten.
- Ausgabewarteschlangen – Implementation des WRKOUTQ-Befehls.

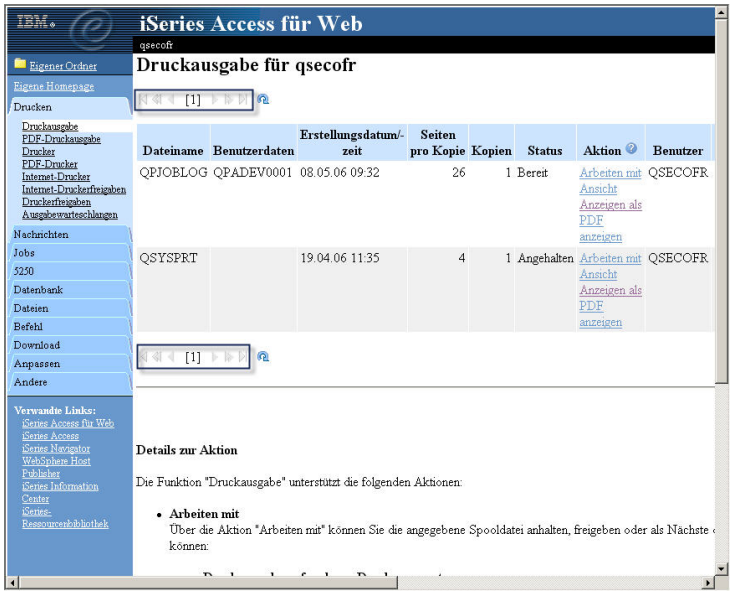
7.4.3.3.1

Seite 2

Navigation

Ich öffne das Menü Drucken => Druckausgabe. Diese Anzeige entspricht weitgehend der Ausgabe des Befehls WRKSPLF.

Im Bild erkennen Sie die typischen Kenndaten für Druckausgaben des System i.



Blättern

Gibt es mehr Sätze als gleichzeitig im Dialog angezeigt werden, so aktiviert sich eine Navigationsleiste, in der Sie durch Anklicken direkt die entsprechende Seite anzeigen.

Da im Browser Anzeigen ähnlich wie im 5250-Bereich seitenorientiert erfolgen, wird eine Änderung im anzuzeigenden Datenbestand nicht automatisch erfolgen.

Hierzu gibt es eine Aktualisierungsfunktion.

Aktualisieren

7.4.3.3.1.1
Seite 2

Aktualisieren Sie die Anzeige, indem Sie auf das kleine runde Pfeil-Symbol klicken.

Um mit Druckausgaben zu arbeiten, finden Sie mehrere Einträge in der Spalte „Aktion“.

Mit einer Druckausgabe arbeiten

Folgende Optionen finden Sie hier:

- Arbeiten mit – Dieses ist ein Konglomerat von unterschiedlichen Bearbeitungsmöglichkeiten, deren Dialoge wir uns im Folgenden gemeinsam ansehen werden.
- Ansicht – Ausgabe als Grafik anzeigen, eine Art DSPSPLF-Funktion. Die Qualität dieser Anzeige ist fragwürdig.
- Anzeigen als – Ausgabe anzeigen in folgenden Formaten: PNG, TIFF, PCL, AFP. Bis auf das PNG-Format benötigen Sie spezielle Viewer-Plug-ins.
- PDF anzeigen – Ausgabe im PDF-Format. Haben Sie in Ihrem Browser das Adobe PDF-Plug.in installiert, so übernimmt dieses Plug-In die Konvertierung der System i Druckausgabe in PDF. Haben Sie den IBM Infoprint Server auf Ihrem System i installiert, so übernimmt dieses Lizenzprogramm die Konvertierung.

Mit Druckausgaben arbeiten

Rufen Sie Drucken => Druckausgaben => Arbeiten mit für eine bestimmte Ausgabe auf, so wird folgender Dialog angezeigt.

Mit Druckausgaben arbeiten

Hier werden im oberen Bereich die Attribute der Druckausgabe angezeigt.

Auch hier finden Sie wieder Aktionen, die Sie ausführen können.

Die meisten dieser Verarbeitungs-Optionen sprechen für sich selbst, deshalb belasse ich es an dieser Stelle bei der Anzeige der Dialoge und benenne gegebenenfalls den CL-Befehl, der hinter einer Funktion verborgen ist.

Druckausgabe auf einen anderen Drucker versetzen

Druckausgabe auf einen anderen Drucker versetzen

Hierbei handelt es sich um eine reduzierte Form des CHGSPLF-Befehls, mit dem Sie eine Ausgabe auf einen im Pulldown auszuwählenden Drucker senden können.

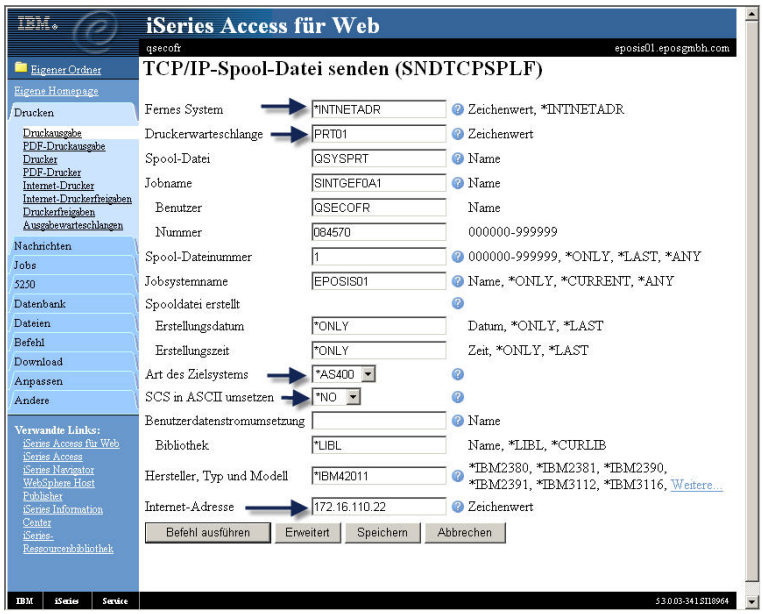
7.4.3.3.1.2
Seite 2

Druckausgabe in eine andere Ausgabewarteschlange versetzen.

Druckausgabe in eine andere Ausgabewarteschlange versetzen

Ebenfalls ein reduzierter CHGSPLF-Befehl.

TCP/IP-Spooldatei senden

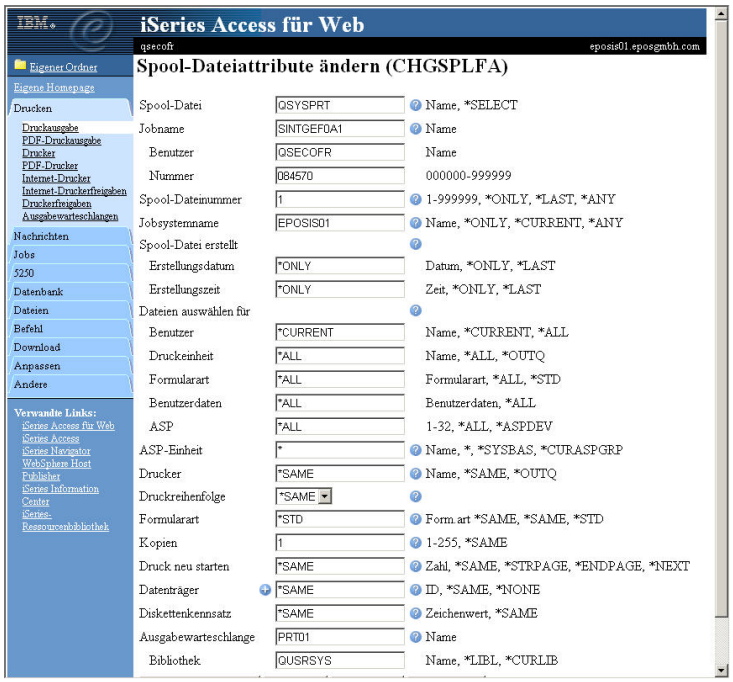


TCP/IP-Spooldatei senden

Diese Seite entspricht dem CL-Befehl SNTCPSPLF, mit dem Sie Druckausgaben von einem System i auf einen fernen Druckerserver senden.

Ich habe hier die notwendigen Parameter eingetragen, um meine Ausgabe auf ein zweites System i zu senden, wobei die Druckausgabe nicht in ASCII konvertiert werden soll.

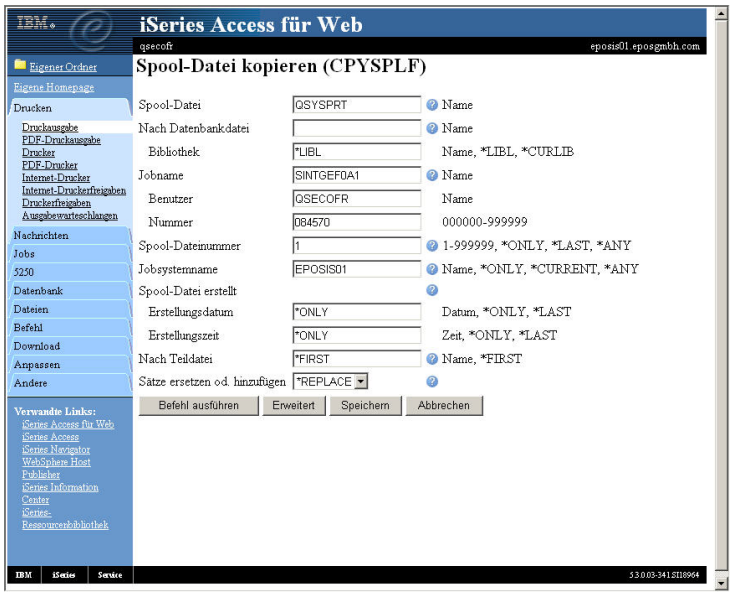
Spool-Dateiattribute ändern



Spool-Dateiattribute ändern

Bei dieser Seite handelt es sich um die komplette Implementation des Befehls CHGSPLF, mit dem Sie alle veränderbaren Attribute einer Druckausgabe des System i bearbeiten können.

Spool-Datei in Datenbank-Tabelle kopieren

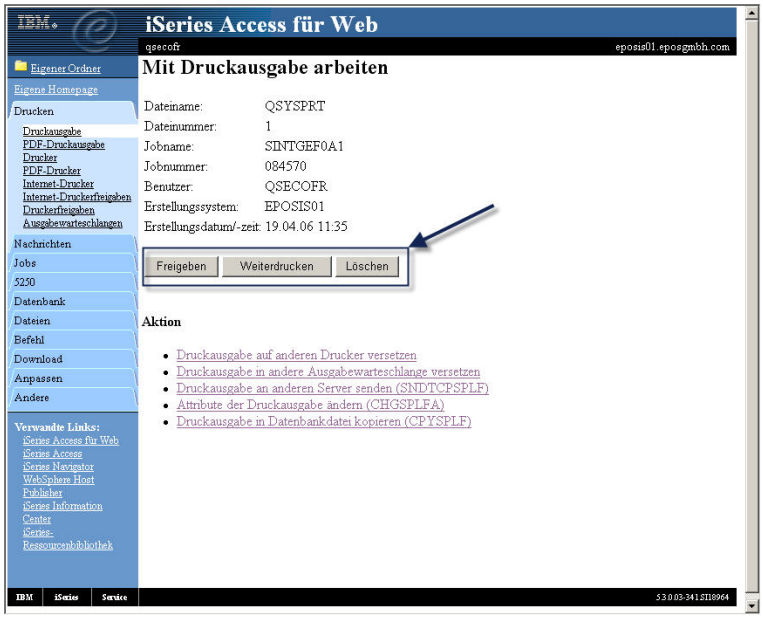


Spool-Datei in Datenbank-Tabelle kopieren

7.4.3.3.1.2
Seite 4

Dieser Dialog stellt die Implementation des CPYSPLF-Befehls dar, mit dem Sie eine System-Druckausgabe in eine Datenbank-Tabelle (Datei) kopieren können.

Als Letztes sehen Sie dann auf der Seite „Mit Druckausgabe arbeiten“ einige direkte Verarbeitungstasten.



Direkte Bearbeitung der Druckausgabe

- Freigeben – Eine angehaltene Druckdatei freigeben (RLSSPLF).
- Weiterdrucken – Den angehaltenen Drucker wieder starten (RLSOUTQ).
- Löschen – Druckausgabe löschen.

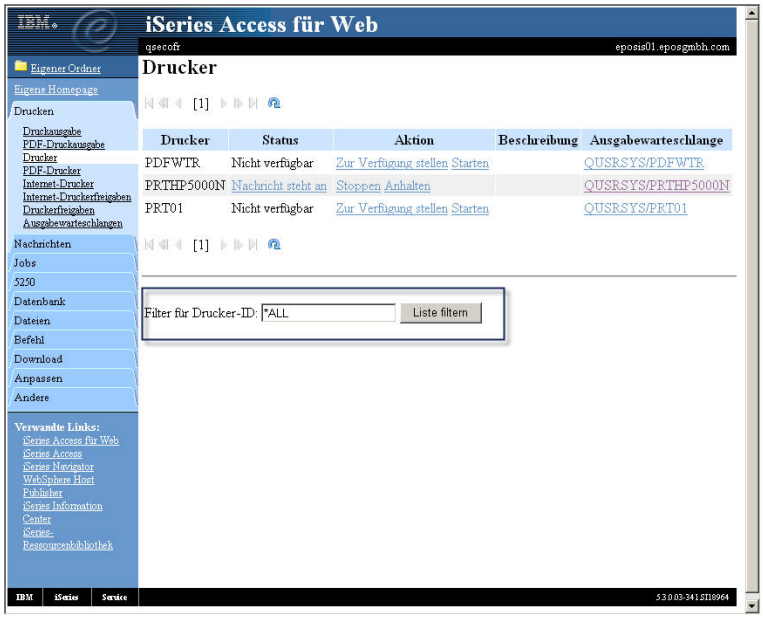
Auswahl „Drucker“

In der Auswahl Drucker stehen Ihnen folgende Funktionen zur Verfügung:

- Drucker auf dem System i auflisten und verwalten.
- Drucker- und Statusinformationen anzeigen.
- Druckereinheiten anhängen und abhängen.
- Über die Druckerliste die Druckausgabe für einen bestimmten Drucker auflisten, verwalten und eine Vorschau anzeigen.
- Nachrichten, die für den Ausgabeprogrammjob anstehen, anzeigen und beantworten.
- Mit Hilfe der Funktionen „Anhalten“, „Freigeben“, „Starten“ und „Stoppen“ können Sie den Ausgabeprogrammjob verwalten, der einem Drucker zugeordnet ist.

Bei dem Menü Drucken => Drucker handelt es sich weitgehend um die Implementation des Befehls WRKWTR.

Wenn Sie diese Funktion aufrufen, so sehen Sie alle Drucker, für die Sie das Anzeigerecht besitzen.



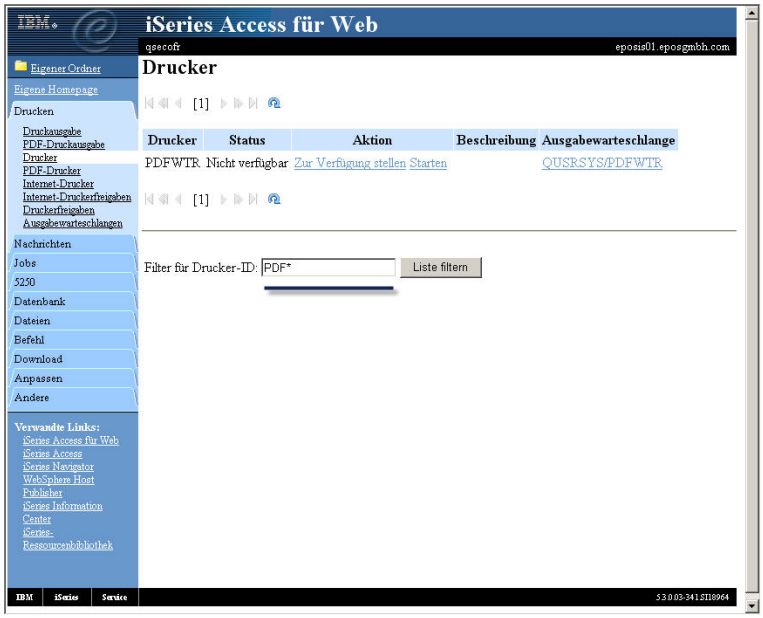
Anzeige Filtern 1

Innerhalb dieser Gruppe von Druckern können Sie jedoch Auswahlmöglichkeiten bezüglich der anzuzeigenden Drucker treffen, die hier als Filter bezeichnet werden.

Der Anfangsfilter steht auf *ALL, so dass alle Drucker angezeigt werden.

7.4.3.3.2
Seite 2

Beabsichtigen Sie die Filterbedingung zu verändern, so gehen Sie folgendermaßen vor:



Anzeige Filtern 2

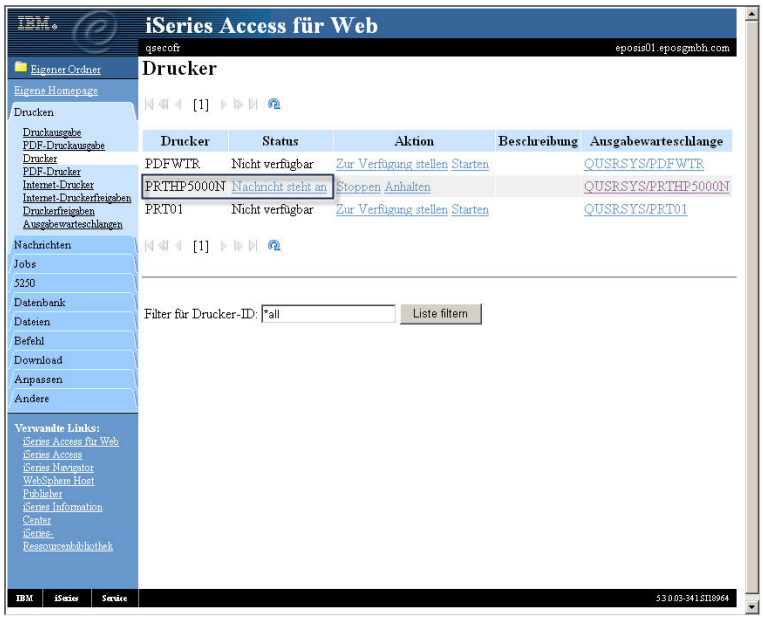
Tragen Sie einfach eine Anfangszeichenfolge im Feld „Filter für Drucker-ID“ ein:

Hier – PDF*

Jetzt werden alle Drucker deren Name mit der Zeichenfolge PDF beginnt angezeigt.

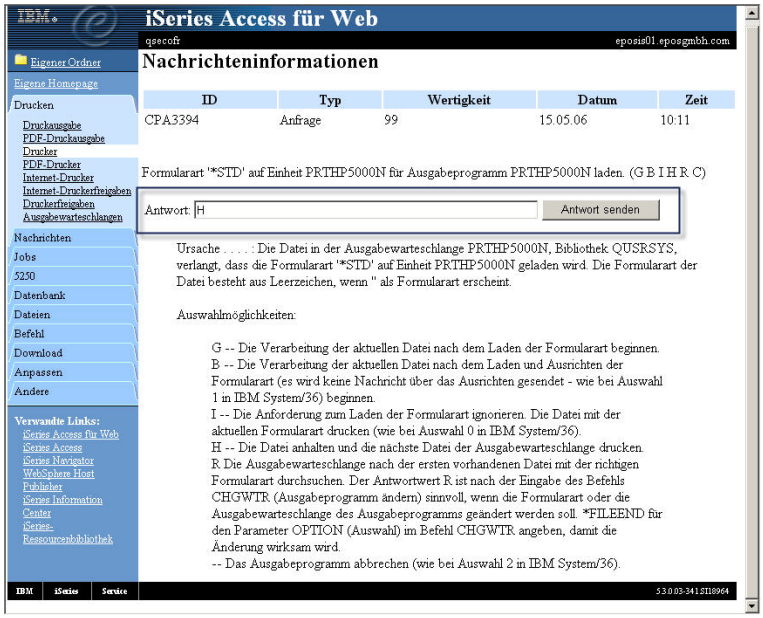
Nachrichten für Drucker bearbeiten

Sehen Sie sich einmal den Drucker PRTHP5000N an. In der Spalte „Status“ erkennen Sie, dass für diesen Drucker eine Nachricht ansteht.



Nachrichten bearbeiten

Um die anstehende Nachricht zu lesen und gegebenenfalls zu beantworten, klicken Sie auf den Link „Nachricht steht an“ und folgende Seite öffnet sich:



Nachricht beantworten

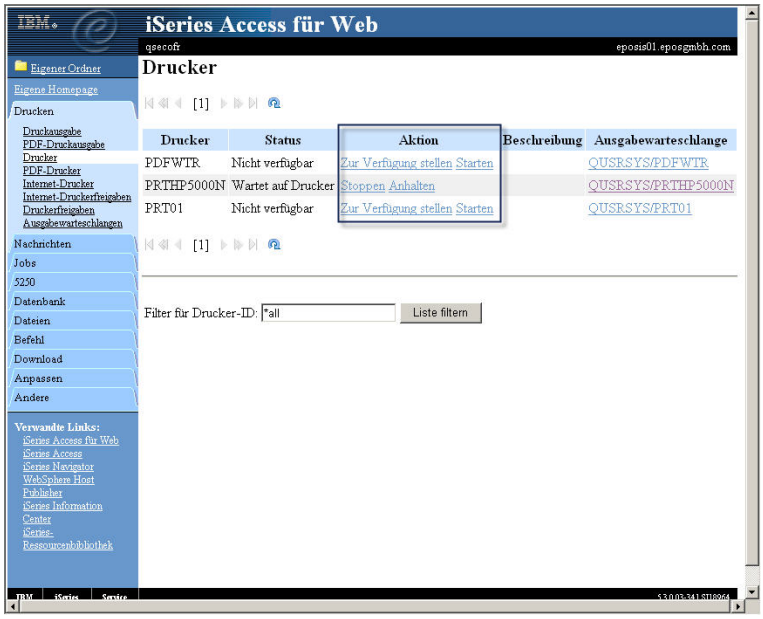
Sie sehen hier die typische Anzeige einer System i-Nachricht, ähnlich der, die ein DSPMSG erzeugt.

7.4.3.3.2.1

Seite 2

Hier geben Sie die gewünschte Antwort im Feld „Antwort“ ein und klicken dann auf „Antwort senden“.

Um mit einem Drucker zu arbeiten, finden Sie im Menü Drucken => Drucker wieder eine Spalte „Aktion“.

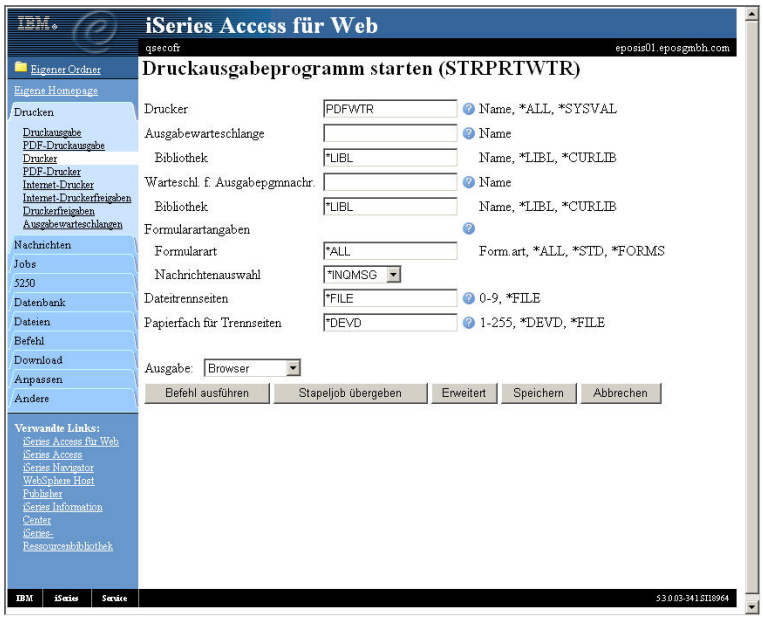


Aktionen für Drucker

Drucker starten

7.4.3.3.2.2
Seite 1

Der Link „Starten“ bringt Sie in folgenden Dialog:



Druckausgabeprogramm starten

Die oben gezeigte Seite entspricht dem Befehl STRPRTWTR, mit dem Sie einen beendeten Drucker starten.

Darüber hinaus finden Sie noch Funktionen wie:

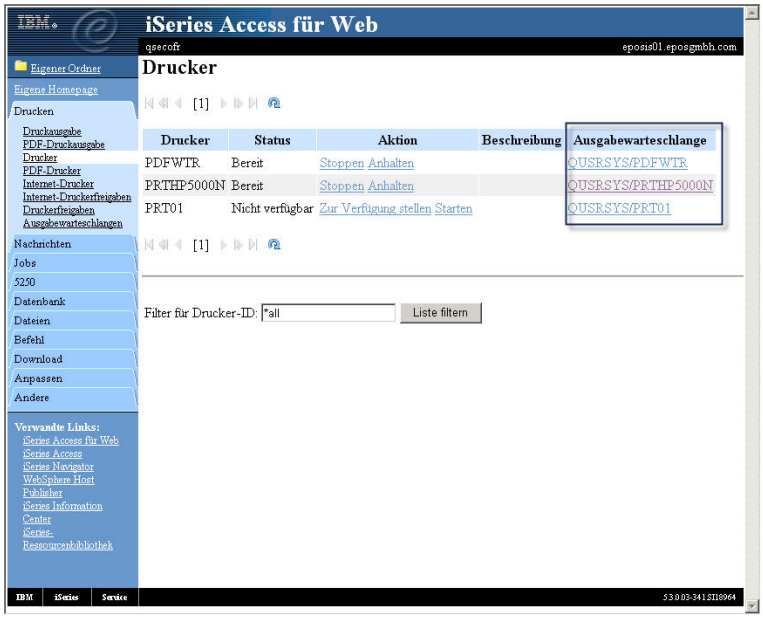
- Zur Verfügung stellen – Einen angehaltenen Drucker wieder starten.
- Anhalten – Einen laufenden Drucker anhalten.
- Stoppen – Einen laufenden Drucker beenden.

7.4.3.3.2.2

Seite 2

Mit Ausgabewarteschlange eines Drucker arbeiten

Über die Links in der Spalte „Ausgabewarteschlange“ gelangen Sie in die einem Drucker zugeordnete Ausgabewarteschlange.



Mit Ausgabewarteschlange eines Druckers arbeiten

Druckausgabe in…

Die Verarbeitungsoptionen entsprechen den in Kapitel 7.4.3.3.1 beschriebenen Möglichkeiten.

7.4.3.3.2.3

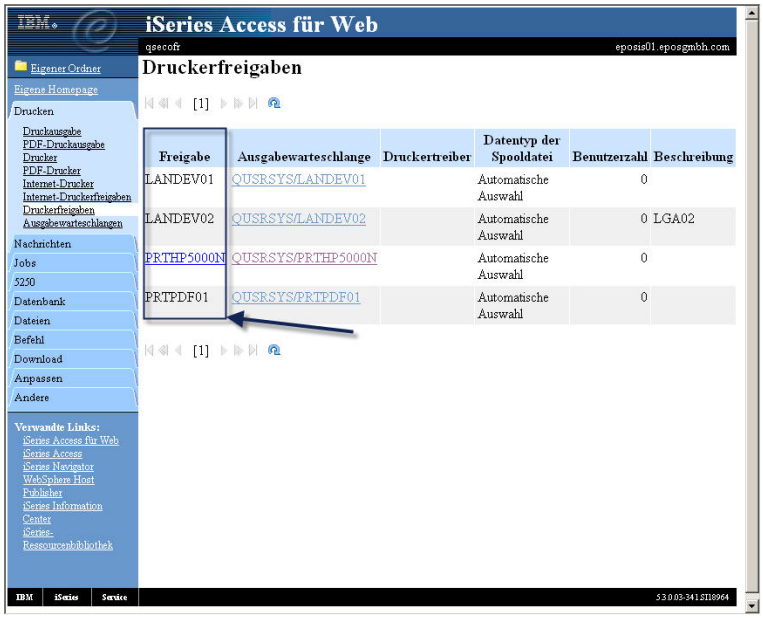
Seite 2

Auswahl „Druckerfreigaben“

In der Auswahl „Druckerfreigaben“ sehen folgende Funktionen zur Verfügung:

- Die Druckerfreigaben auf dem iSeries-Server inklusive Informationen und Status der Druckerfreigabe auflisten.
- Mittels der Druckerfreigaben können Sie auf Druckausgabe zugreifen und sie verwalten.

Wenn Sie den Menüpunkt Drucken => Druckfreigaben aufrufen, so erhalten Sie in der Anzeige eine Liste der für das Windows-Netzwerk freigegebenen System i Drucker.

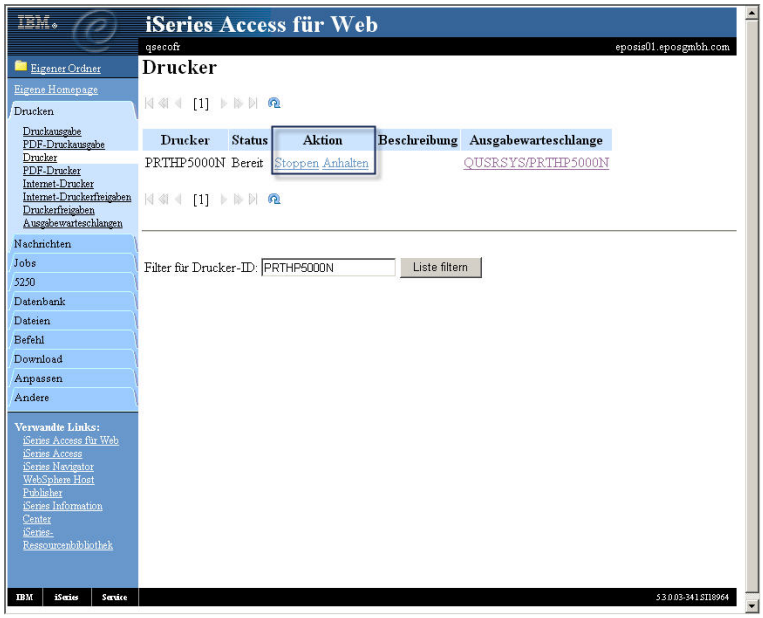


Druckerfreigaben

Ein Klick auf einen aktiven Link in der Spalte Freigabe bringt Sie in die Bearbeitung des einer Freigabe zugeordneten Druckers.

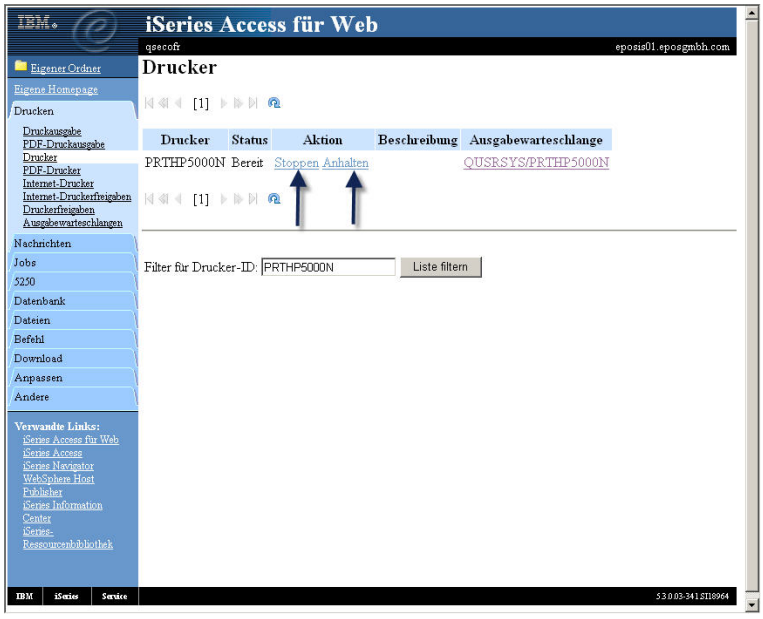
7.4.3.3.3

Seite 2



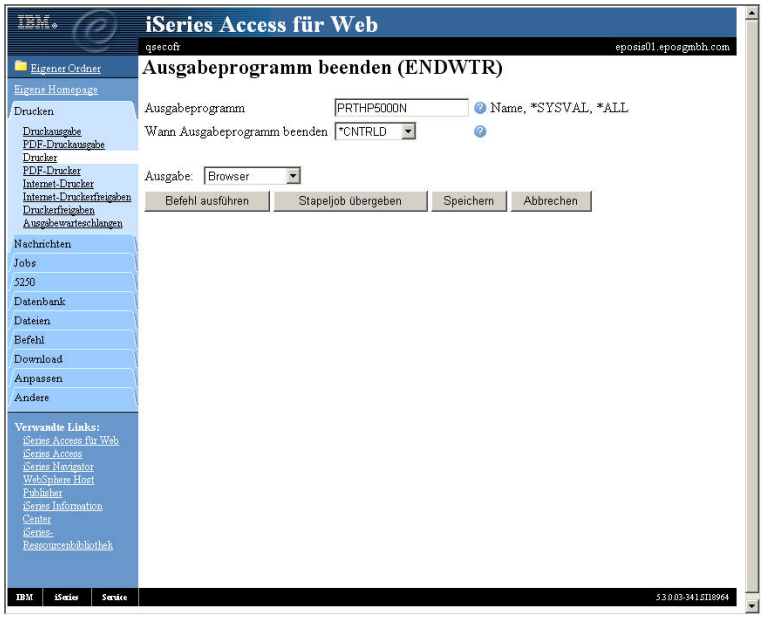
Drucker

Die Verarbeitungsmöglichkeiten entsprechen denen, die im Abschnitt 7.4.3.3.2 bereits beschrieben wurden.



Drucker stoppen und anhalten

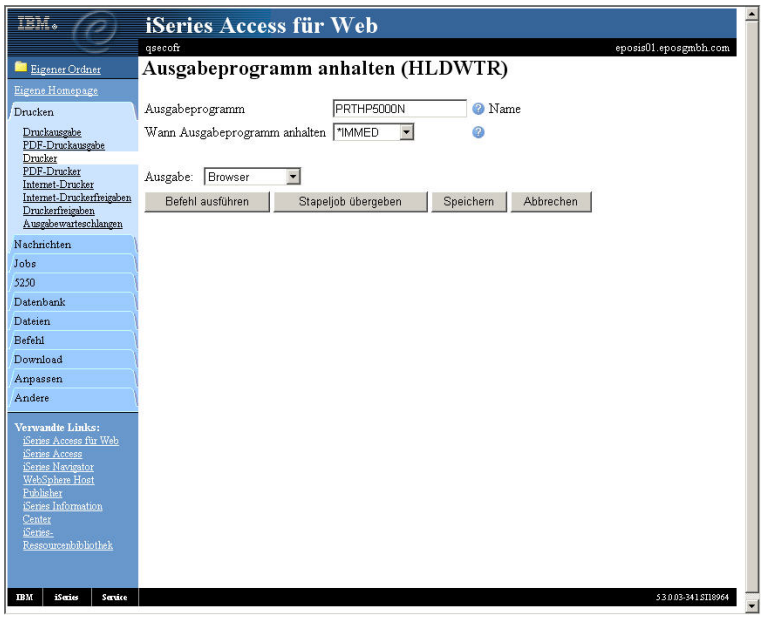
Klicken Sie auf den Link „Stoppen“, so wird der freigegebene Drucker beendet.



Ausgabeprogramm beenden

Sie sehen in dem oben gezeigten Dialog, dass es sich hierbei um die ENDWTR-Implementation des i5/OS handelt.

Wählen Sie die Option „Anhalten“, so wird der Befehl HLDWTR bereitgestellt.

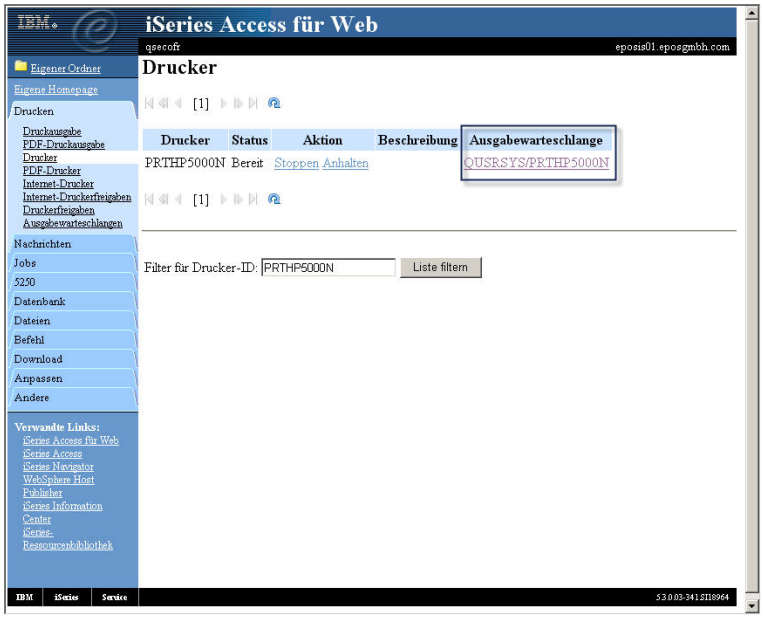


Ausgabeprogramm anhalten

Wenn Sie beabsichtigen mit der Ausgabewarteschlange eines freigegebenen Druckers zu arbeiten, so klicken Sie in die entsprechenden Links in der Spalte „Ausgabe“.

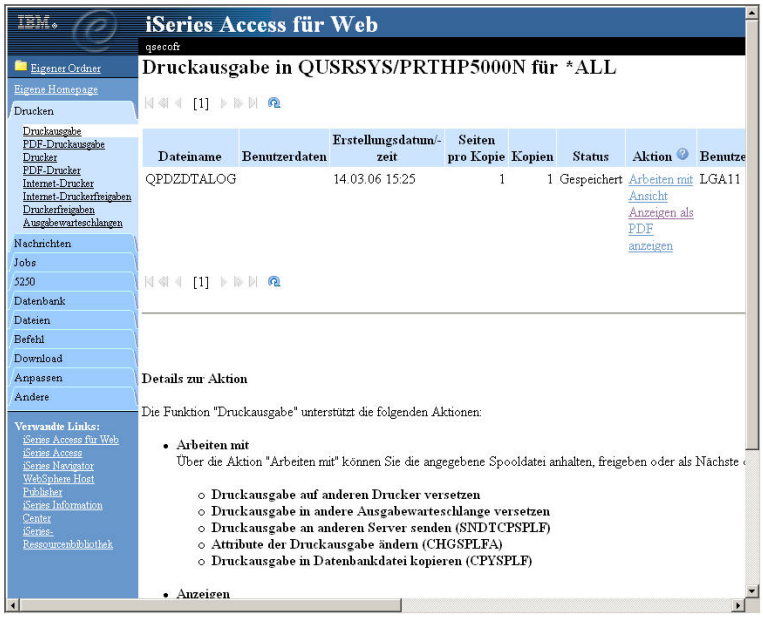
7.4.3.3.3

Seite 4



Mit Ausgabewarteschlangen arbeiten

Sie gelangen sodann in die Ausgabewarteschlange eines freigegebenen Druckers.



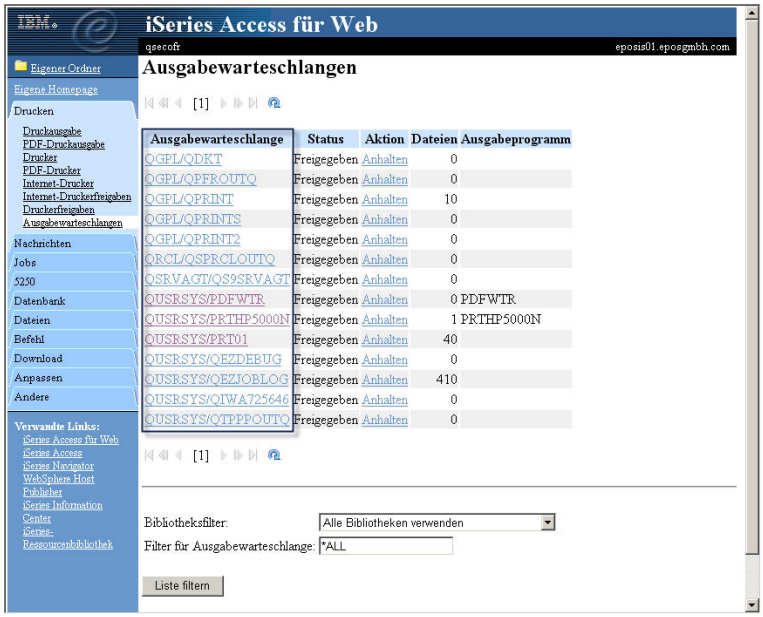
Mit Druckausgaben in … arbeiten

Auswahl „Ausgabewarteschlangen“

In der Auswahl Ausgabewarteschlangen stehen Ihnen folgende Funktionen zur Verfügung:

- Ausgabewarteschlangen auf dem System i auflisten, verwalten und ihren Status anzeigen.
- Mit Hilfe der Liste der Ausgabewarteschlangen können Sie eine Ausgabewarteschlange anhalten und freigeben.
- Außerdem können Sie über die Liste der Ausgabewarteschlangen die Druckausgabe auflisten, verwalten und eine Vorschau anzeigen.

Im Menü Drucken => Ausgabewarteschlangen können Sie direkt mit den Ausgabewarteschlangen Ihres System i arbeiten.



Ausgabewarteschlangen

Diese Anzeige entspricht fast 1:1 der Ausgabe des Befehls WRKOUT in der 5250-Sitzung.

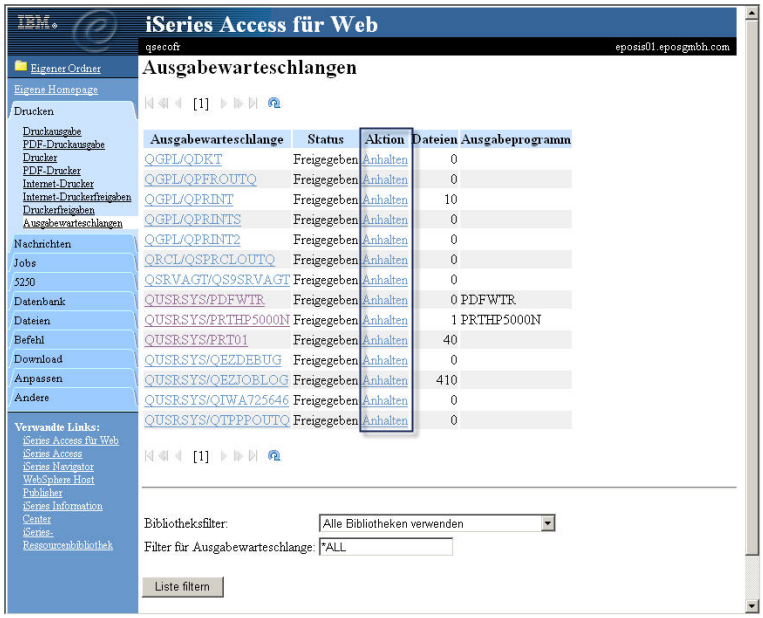
Klicken Sie auf einen Link in der Spalte Ausgabewarteschlange, so arbeiten Sie direkt mit den Druckausgaben dieser Warteschlange.

Die Verarbeitungsmöglichkeiten innerhalb einer Warteschlange wurden in Kapitel 7.4.3.3.1 beschrieben.

Die Spalte „Aktion“ stellt Ihnen, je nach aktuellen Status der Ausgabewarteschlange die Möglichkeit bereit, eine Ausgabewarteschlange anzuhalten (HLDOUT) bzw. zu starten (RLSOUTQ).

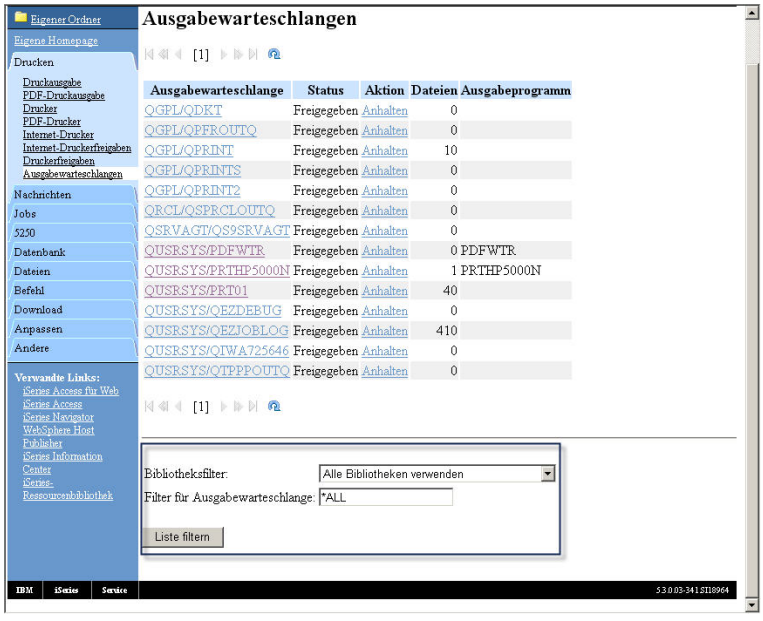
7.4.3.3.4

Seite 2



Ausgabewarteschlange starten und anhalten

Auch hier stehen Ihnen wieder Filterfunktionen zur Verfügung, um die Menge der angezeigten Ausgabewarteschlangen zu reduzieren und somit die Anzeige Ihren Wünschen anzupassen.



Anzeige filtern

Mit Nachrichten arbeiten

Die Nachrichtenfunktion von iSeries Access für Web bietet dem erfahrenen Administrator des System i einige Funktionen, die er direkt mit den entsprechenden Funktionen des 5250-CL-Umfelds assoziieren kann.

iSeries Access-Nachrichtenfunktion

- **Nachrichten anzeigen**
  Sie können die Nachrichten in der Nachrichtenwarteschlange für den angemeldeten Benutzer anzeigen, beantworten und verwalten.
- **Nachricht senden**
  Damit können Sie eine Nachricht an Benutzer und Nachrichtenwarteschlange senden.
- **Durchbruchnachricht senden**
  Sie können eine Durchbruchnachricht an Sametime-Benutzer der iSeries Access für Web senden.

  Sie können eine Durchbruchnachricht an iSeries-Workstations senden.
- **Bedienernachrichten**
  Sie können Nachrichten in der Nachrichtenwarteschlange für den Systembediener anzeigen, beantworten und verwalten.
- **Nachrichtenwarteschlangen**
  Sie können die Nachrichtenwarteschlangen auf dem iSeries-Server auflisten.

  Nachrichten in Nachrichtenwarteschlangen können Sie auswählen, anzeigen und verwalten.

7.4.3.4

Seite 2

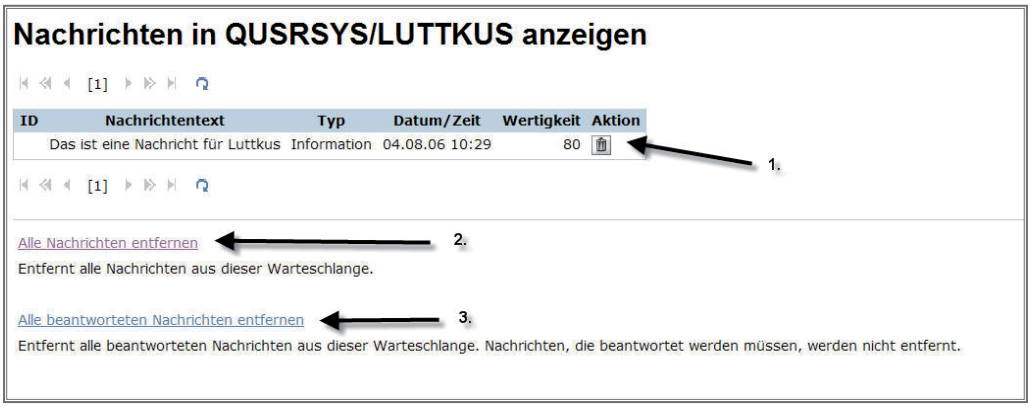
Nachrichten anzeigen

Die Funktion „Nachrichten anzeigen“ entspricht weitestgehend der Funktion DSPMSG (Nachrichten anzeigen) des 5250-Umfelds. Deshalb werde ich meine Erläuterungen in diesem Bereich so knapp wie möglich halten.

Sie erreichen diese Funktion über das Menü „Nachrichten ➜ Nachrichten anzeigen“.

Aufruf der Nachrichtenfunktion

Im folgenden Bild sehen Sie den von iSeries Access für Web erzeugten Dialog.



Eigene Nachrichten anzeigen (DSPMSG)

Sie sehen hier die Nachrichtenanzeige für den Benutzer „Luttkus“. Da ich mich beim Aufruf von iSeries Access für Web mit dem Benutzer „Luttkus“ angemeldet habe, werden die entsprechenden Nachrichten angezeigt. Sie merken: Das ist genauso wie bei einem 5250-Bildschirm.

7.4.3.4.1
Seite 2

Der Unterschied zur 5250-Nachrichtenanzeige besteht jedoch darin, dass hier ausschließlich die Nachrichten angezeigt werden, die an die Nachrichtenwarteschlange des Benutzerprofils gesendet wurden.

In der 5250-Nachrichtenanzeige werden, wie Ihnen vielleicht bekannt ist, zuerst die Nachrichten des Benutzers angezeigt, danach die Nachrichten, die an die Datenstation gesendet wurden.

Da es sich bei dieser iSeries-Access-für-Web-Funktion nicht um eine emulierte 5250-Einheit handelt, ist hier eine Anzeige der Datenstationsnachrichten eigentlich auch nicht zu erwarten.

Nachdem ich meine Nachrichten zur Kenntnis genommen habe, werde ich jetzt Nachrichten löschen.

Dabei stehen mir drei Funktionen zur Verfügung:

1. Eine einzelne Nachricht löschen.
   Klicken Sie einfach mit der Maus auf das entsprechende Symbol/Link und die betreffende Nachricht wird gelöscht.

   Achtung! Hier gibt es keine Sicherheitsabfrage!

2. Alle meine Nachrichten löschen.

Alle Nachrichten entfernen

   Sie sehen, dass hier eine Sicherheitsabfrage angezeigt wird, bevor all Ihre Nachrichten gelöscht werden. Um den Löschvorgang zu beenden, klicken Sie auf den entsprechenden Druckknopf.

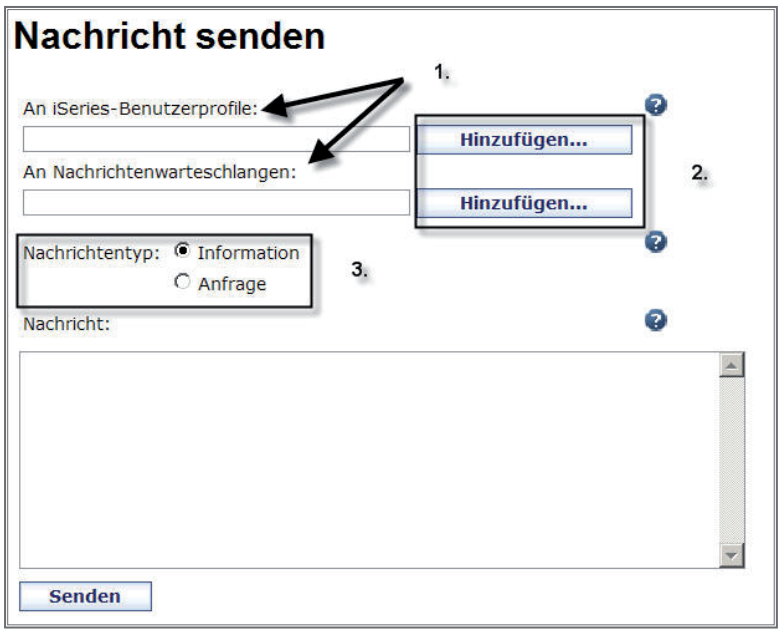
3. Alle von mir beantworteten Nachrichten löschen.

Alle beantworteten Nachrichten löschen

   Sie sehen, dass auch hier eine Sicherheitsabfrage angezeigt wird, bevor all Ihre beantworteten Nachrichten gelöscht werden. Um den Löschvorgang zu beenden, klicken Sie auf den entsprechenden Druckknopf.

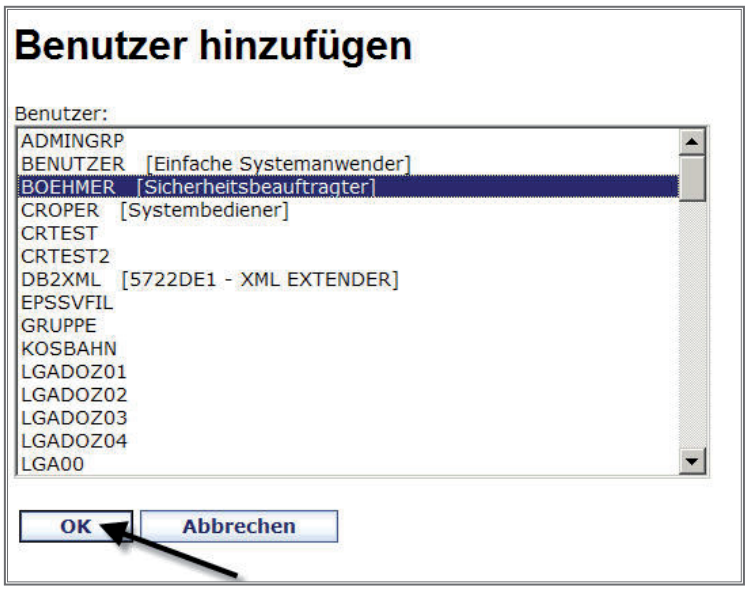
Nachrichten senden

Bei der iSeries-Access-für-Web-Funktion „Nachrichten senden“ handelt es sich um die grafische Umsetzung des CL-Befehls SNDMSG (Nachrichten senden).



Nachrichten senden (SNDMSG)

1. Mit dieser Funktion werden Nachrichten an Benutzer und an sonstige Nachrichtenwarteschlangen gesendet.
2. Sowohl die Benutzer als auch die sonstigen Warteschlangen, an die Sie Nachrichten senden möchten, lassen sich „erprompten“. Dazu klicken Sie mit der Maus auf die jeweiligen Buttons „Hinzufügen…“.

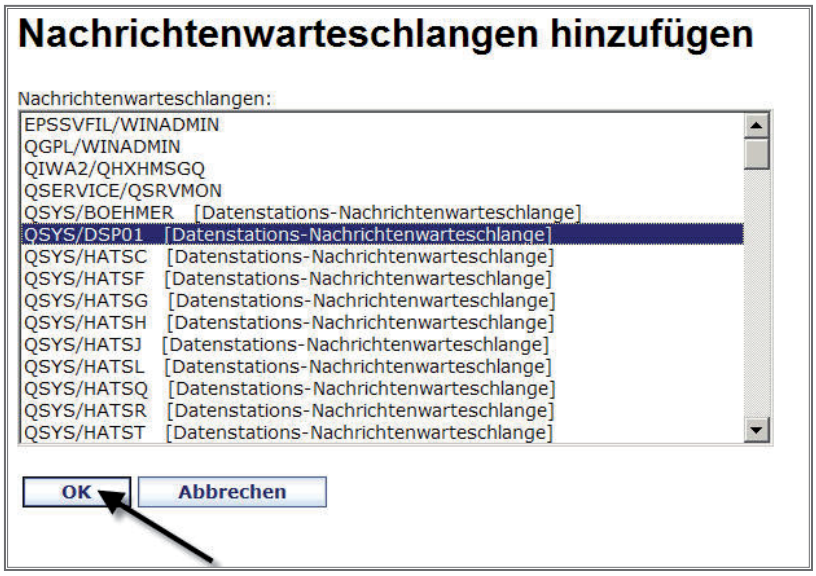


Benutzer hinzufügen

7.4.3.4.2
Seite 2

Markieren Sie den Benutzer, an den Sie eine Nachricht senden wollen. Danach erfolgt ein weiter Klick auf „OK“ und der Benutzername wird in den Hauptdialog übernommen.

Analog gehen Sie vor, wenn Sie die Nachricht beispielsweise an eine Datenstation senden möchten.



Nachrichtenwarteschlange hinzufügen

Im oben gezeigten Dialog markiere ich die Datenwarteschlange DSP01, um eine Nachricht an die Systemkonsole zu senden.

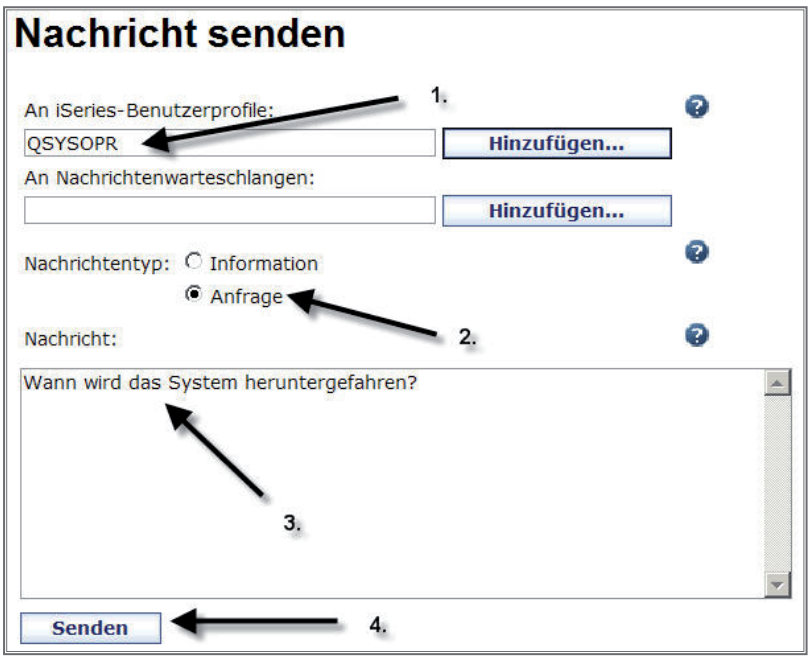
3. Im Nachrichtentyp wird festgelegt, ob Ihre Nachricht als reine Information gesendet wird, die der Empfänger zur Kenntnis nehmen kann, oder ob vom Empfänger eine Antwort erwartet wird (Anfrage).

Fallstudie: Nachricht an Systembediener senden

In dieser kleinen Fallstudie werden wir gemeinsam die Anfragenachricht „Wann wird das System heruntergefahren?“ an den Benutzer QSYSOPR absetzen.

Folgende Arbeitsschritte sind notwendig:

Zuerst öffnen wir die iSeries-Access-für-Web-Funktion „Nachrichten senden“.



Nachrichten an QSYSOPR

1. Eintragen des Namens QSYSOPR in das Feld iSeries-Benutzerprofile.
2. Markieren des Nachrichtentyps „Anfrage“.
3. Eingeben der Textnachricht in den Bereich „Nachricht:“.
4. Bestätigen des Dialogs mit „Senden“.

Sendebestätigung

In der oberen Abbildung sehen Sie, dass das Senden der Nachricht bestätigt wurde.

7.4.3.4.2.1

Seite 2

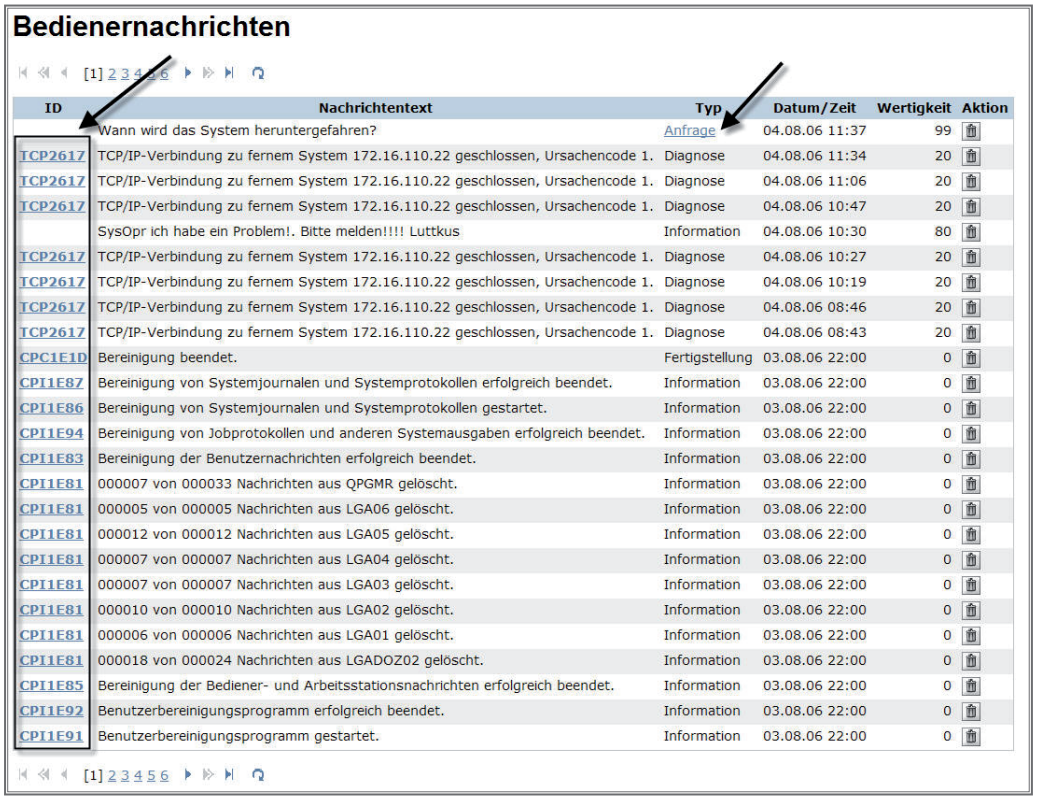
Um weitere Nachrichten zu senden, klicken Sie den entsprechenden Button an.

Wie Sie bis hierhin feststellen konnten, ist das Senden von Nachrichten eine sehr einfach zu bedienende Funktion.

Da wir jetzt eine Nachricht an QSYSOPR – also an den Systembediener des System i – gesendet haben, werden wir mit der Funktion „Bedienernachrichten“ des iSeries Access für Web fortfahren.

Bedienernachrichten

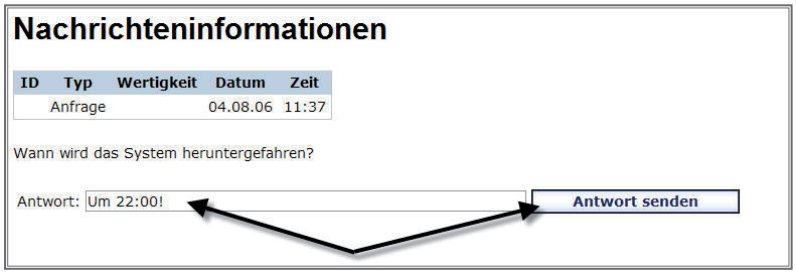
Die Funktion „Bedienernachrichten“ entspricht – von der Bedienung aus gesehen – der Funktion „Nachrichten anzeigen“.



Anzeige Bedienernachrichten (QSYSOPR)

Bezüglich des Löschens von Nachrichten finden Sie hier die gleichen Bedienelemente vor wie in der Funktion „Nachrichten anzeigen“.

Sie erinnern sich bestimmt, dass wir dem Systembediener eine Anfragenachricht gesendet haben. Als erste Nachricht erkennen Sie im oben gezeigten Dialog genau diese Nachricht. Als Nachrichtentyp wird hier „Anfrage“ angezeigt. Der Nachrichtentyp „Anfrage“ ist als Hyperlink realisiert. Also werden wir diesem Link durch einen Mausklick folgen.

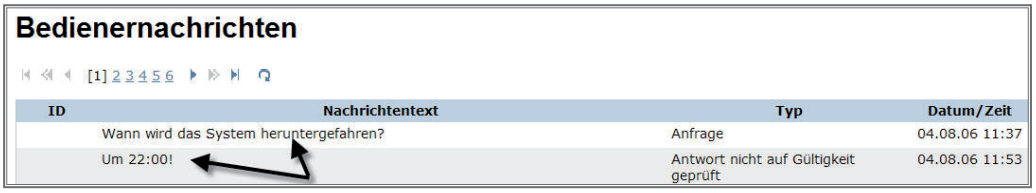


Anfragenachricht beantworten

7.4.3.4.3
Seite 2

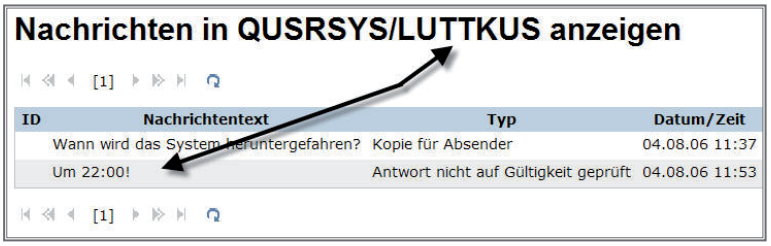
Als Beantwortung auf diese Anfrage klicken wir auf „Antwort senden“.

Jetzt sieht die Nachrichtenanzeige des QSYSOPR folgendermaßen aus:



Anzeige der Systembedienernachrichten

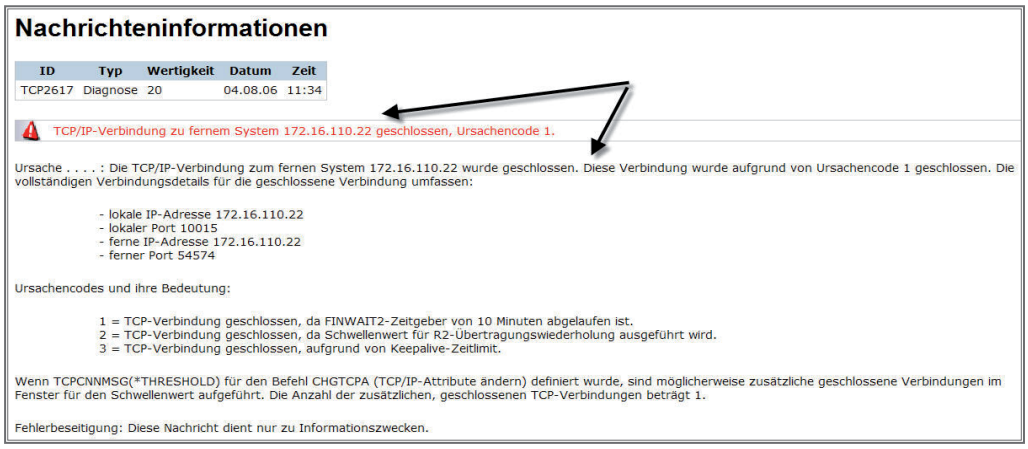
Der Systembediener hat nun also die Nachricht beantwortet und die Nachricht ist beim Absender eingetroffen!



Empfangene Antwort

In der Nachrichtenanzeige sehen Sie – neben der eben besprochenen Individualnachricht – noch eine ganze Reihe an Systemnachrichten, die an den Systembediener gegangen sind.

Erkennbar sind Systemnachrichten durch die vorangestellte Nachrichten-ID. Diese ID ist als Hyperlink abgebildet. Um eine Systemnachricht zu lesen, klicken wir auf den entsprechenden Hyperlink.

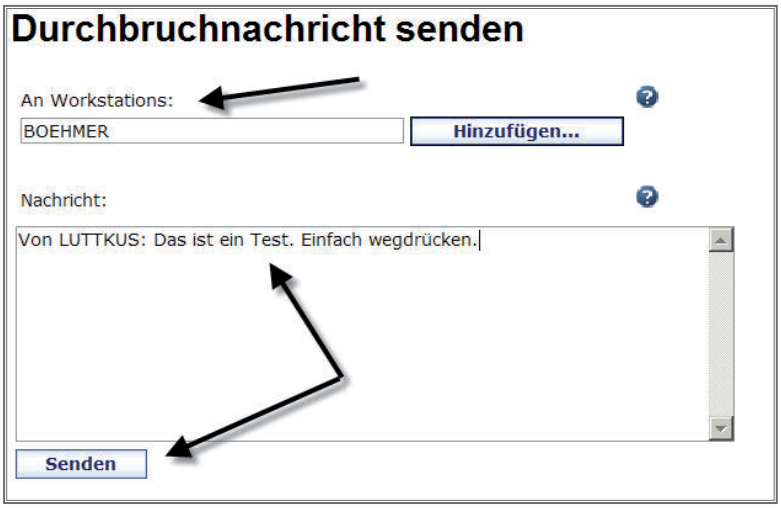


Nachrichtentext einer Systemnachricht

Sie sehen, das die Nachrichtentexte systemkonform angezeigt werden: Durch ein Symbol gekennzeichnet sehen Sie den Nachrichtentext der ersten Ebene. Darunter finden Sie den ausführlicheren Text der zweiten Ebene.

Durchbruchnachrichten senden

Die Funktion „Durchbruchnachrichten senden“ ist fast identisch mit der Funktion „Nachrichten senden“. Sie unterscheidet sich lediglich dadurch, dass sie noch einfacher gestaltet ist.



Nachrichtentext einer Systemnachricht

Es stellt sich hier nun die Frage: Was macht Durchbruchnachrichten so einfach?

Durchbruchnachrichten können nur an Datenstationen gesendet werden. Deshalb bietet dieser Dialog auch nur diese Möglichkeit an. Über den Button „Hinzufügen…“ lassen sich die Bildschirmeinheiten auch hier wieder „erprompten“.

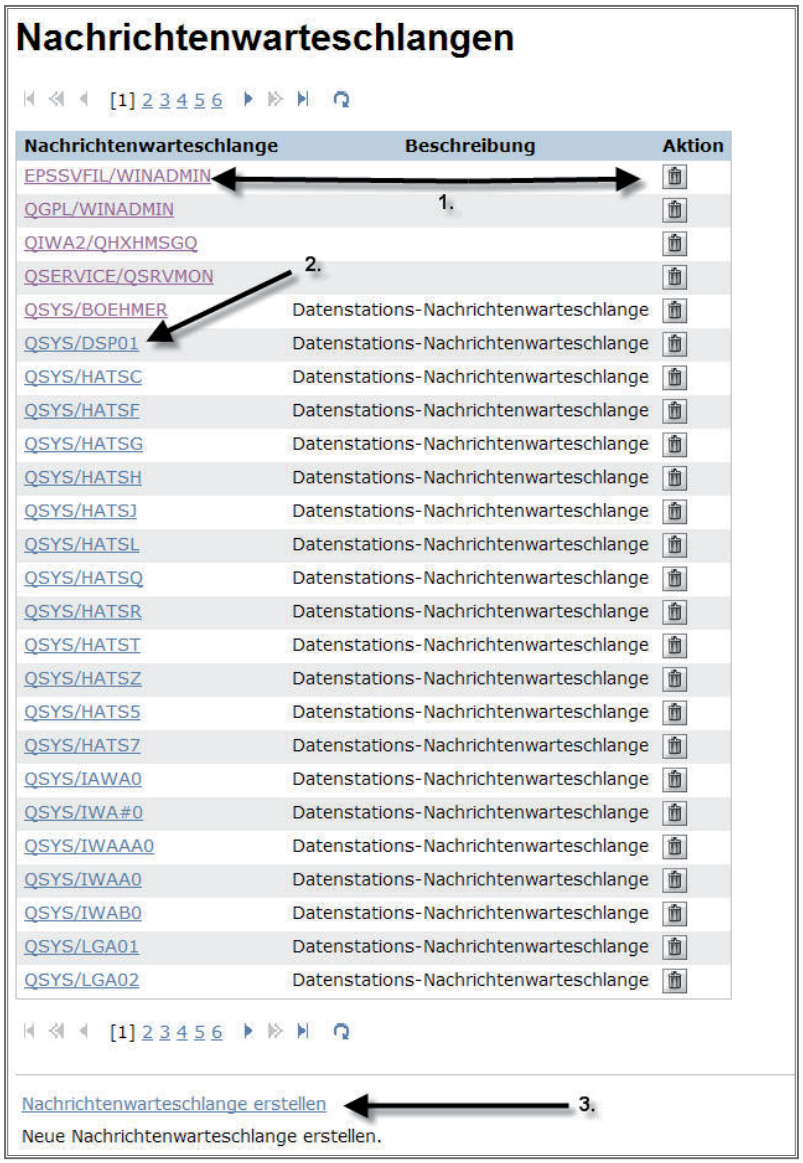
Ansonsten bleibt Ihnen nur die Möglichkeit, die Nachricht einzugeben und dann abzusenden.

7.4.3.4.4

Seite 2

Nachrichtenwarteschlangen

Wenn Sie als System-i-Benutzer entsprechende Rechte besitzen, dann können Sie mit der Funktion „Nachrichtenwarteschlangen“ in diesem Umfeld solche auch verwalten.



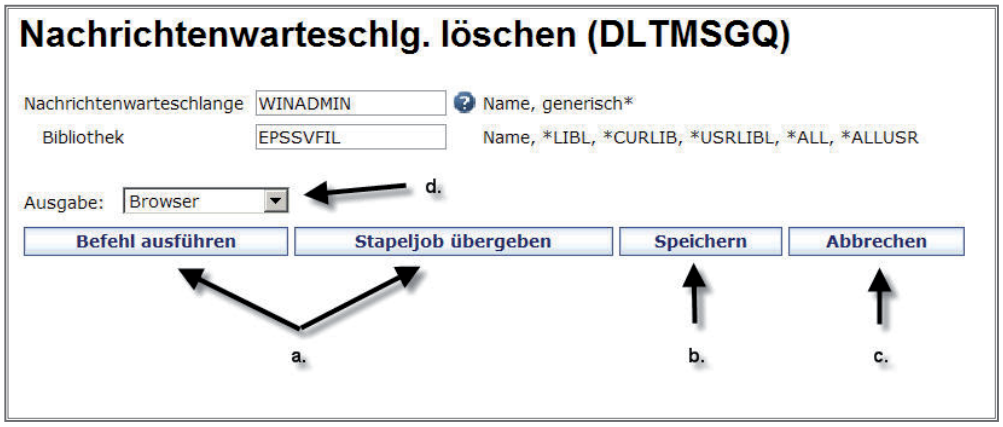
Mit Nachrichtenwarteschlangen arbeiten

Folgende Verarbeitungsmöglichkeiten bestehen in dieser Funktion:

1. Nachrichtenwarteschlangen löschen:

   Da ich mit dem verwendeten System i vertraut bin, weiß ich, dass die Nachrichtenwareschlange WINADMIN in der Bibliothek EPSSVFIL aufgrund einer Fehlkonfiguration im System entstanden ist. Um diese Warteschlange zu löschen, klicke ich einfach auf das Mülleimer-Symbol.

7.4.3.4.5
Seite 2



Nachrichtenwarteschlange löschen (DLTMSGQ)

a.) Das Löschen der Nachrichtenwarteschlange kann direkt oder als Stapeljob ausgeführt werden.

b.) Das Löschen der Nachrichtenwarteschlange soll als wieder ausführbarer Befehl gespeichert werden.

c.) Die Löschaktion soll abgebrochen werden.

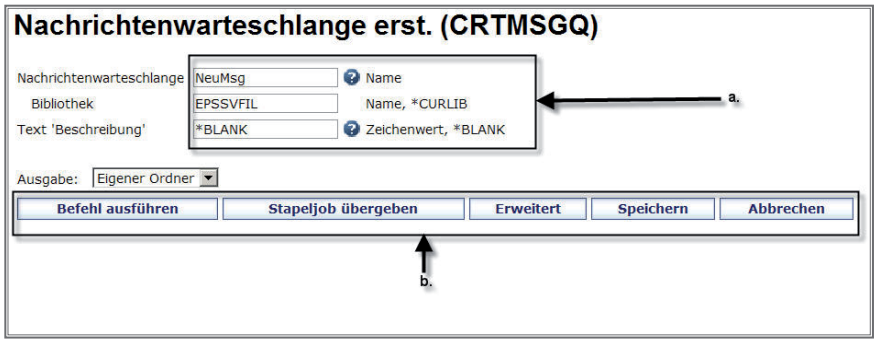
d.) Wenn der Löschvorgang eine Ausgabe produziert, so kann diese entweder direkt im Browser angezeigt oder als Textdatei im Menü „Eigene Ordner“ gespeichert werden.

2. Um mit den Nachrichten in einer Nachrichtenwarteschlange zu arbeiten, folgen Sie einfach dem jeweiligen Hyperlink.

Nachrichtenwarteschlange DSP01 anzeigen

Auch hier handelt es sich wiederum um die bereits besprochene Funktion „Nachrichten anzeigen“.

3. Um eine neue Nachrichtenwarteschlange zu erzeugen, klicken Sie auf den Hyperlink „Nachrichtenwarteschlange erstellen“.



Neue Nachrichtenwarteschlange erstellen (CRTMSGQ)

a.) Tragen Sie in die angezeigten Felder den Namen, die Bibliothek und gegebenenfalls einen beschreibenden Text für die neue Nachrichtenwarteschlange ein.

b.) Wählen Sie eine der aufgezeigten Bearbeitungsmöglichkeiten.

7.4.3.4.5

Seite 4

Mit der Datenbank arbeiten

Für den System i-Administrator gibt es einen weiteren, großen Funktionsbereich von iSeries Access für Web – und zwar die Funktion „Datenbank“.

Datenbankfunktion des Access für Web

Die Funktion „Datenbank“ des iSeries Access für Web stellt Ihnen folgende Funktionsbereiche zur Verfügung:

- Tabellen
  - Sie können eine Liste der Datenbanktabellen auf dem iSeries Server aufrufen.
  - Für diese Tabellen können Sie Aktionen ausführen, ohne dass Kenntnisse über SQL und deren Syntax erforderlich sind.
  - Über die Aktion „Schnellansicht“ können Sie den Inhalt einer Tabelle in einer Liste aus Seiten anzeigen.
  - Mit den Aktionen „Einfügen“ und „Aktualisieren“ können Sie Sätze in einer Tabelle hinzufügen und aktualisieren.
  - Über die Aktion „SQL ausführen“ können Sie eine eigene, angepasste SQL-Anforderung für eine Tabelle erstellen.
  - Über die Aktion „Daten in Tabelle kopieren“ können Sie eine eigene, angepasste Anforderung für das Kopieren von Daten erstellen.
- Eigene Anforderungen
  - Sie können eine Liste der zuvor gespeicherten Anforderungen anzeigen.
  - Über diese Liste können Sie Anforderungen ausführen oder bearbeiten.
  - Mit den Aktionen „Kopieren“, „Löschen“ und „Umbenennen“ können Sie Listen verwalten.

7.4.3.5
Seite 2

  - Außerdem können Sie Verknüpfungen mit Anforderungen erstellen und verwalten.
- SQL ausführen
  - Hiermit können Sie SQL-Anweisungen dynamisch ausführen.
  - Die Ausgabe wird als Liste aus Seiten oder in einem gängigen Dateiformat (zum Beispiel: Microsoft Excel oder Lotus 1-2-3) angezeigt. Das Ausgabeformat richtet sich danach, wie der verwendete Browser konfiguriert ist und ob der Browser ein Plug-in für den ausgewählten Ausgabetyp findet.
  - Die Rückgabe der Daten können Sie anpassen, indem Sie spezifische Optionen für den Ausgabetyp festlegen.
  - Mit einem SQL-Assistenten können Sie die Anweisungen über SELECT erstellen.
  - Anforderungen können zur erneuten Verwendung gespeichert werden.
- Daten in Tabelle kopieren
  - Sie können vorhandene Datendateien vom PC in eine Datenbanktabelle auf dem iSeries Server kopieren.
  - Diese Datendateien können viele unterschiedliche Formate (zum Beispiel: Microsoft Excel und Lotus 1-2-3) haben.
  - Sie können den Inhalt einer Tabelle ersetzen oder Daten zu einer vorhandenen Tabelle hinzufügen.
  - Sie können eine neue Datenbanktabelle erstellen, die auf den Inhalt einer Workstation-Datei aufsetzt.
  - Anforderungen können zur erneuten Verwendung gespeichert werden.
- Anforderung importieren
  - Hier können Sie Upload- und Download-Anforderungen, die aus der Client-Access-Komponente „Datenübertragung“ stammen, in iSeries Access für Web-Anforderungen zum Kopieren von Daten und SQL-Anforderungen importieren.
  - Importierte Anforderungen werden automatisch in das Format von iSeries Access für Web konvertiert.
  - Die konvertierten Anforderungen können Sie auf dem iSeries Server genauso ausführen und bearbeiten wie bei anderen Anforderungen zum Kopieren von Daten oder SQL-Anforderungen.
- Abfrage importieren
  - Hiermit können Sie Abfragen importieren, die mit Query für iSeries und DB2 UDB for iSeries Query Manager generiert wurden.

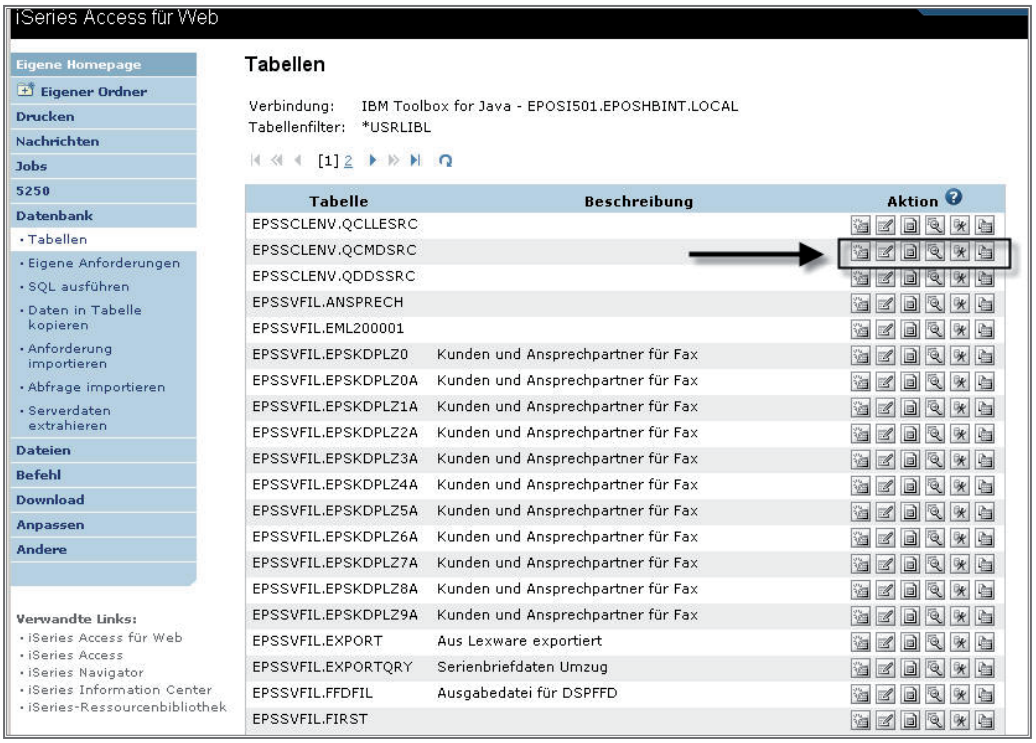
  - Importierte Abfragen können in Datenbankanforderungen von iSeries Access für Web gespeichert werden.
  - Die konvertierten Abfragen können Sie auf dem iSeries Server genauso wie bei SQL-Anforderungen von iSeries Access für Web ausführen und bearbeiten.
- Serverdaten extrahieren
  - Sie können Server-Objektinformationen in eine Datenbanktabelle extrahieren.

7.4.3.5

Seite 4

Tabellen

Wenden wir uns zunächst der Funktion „Tabellen“ zu. Ein Klick auf diese Funktion öffnet folgende Seite:



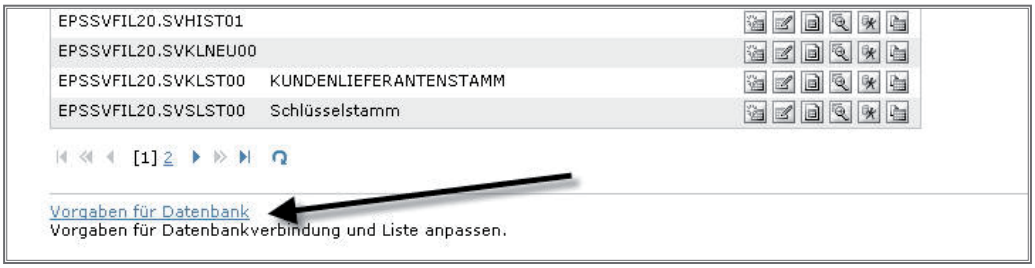
Tabellen anzeigen

In dieser Anzeige sehen Sie eine Reihe von Datenbanktabellen. Die Grundeinstellung hier ist, dass alle Bibliotheken der *USRLIBL angezeigt werden. Die Arbeit mit den einzelnen Tabellen wird durch die oben markierte Icon-Leiste ermöglicht. Die folgende Aufzählung beinhaltet die dahinterliegenden Funktionen (von links nach rechts gelesen):

- Satz einfügen (INSERT)
- Satz aktualisieren (UPDATE/DELETE)
- Schnellansicht
- Satz suchen
- SQL ausführen
- Daten in Tabelle kopieren (IMPORT)

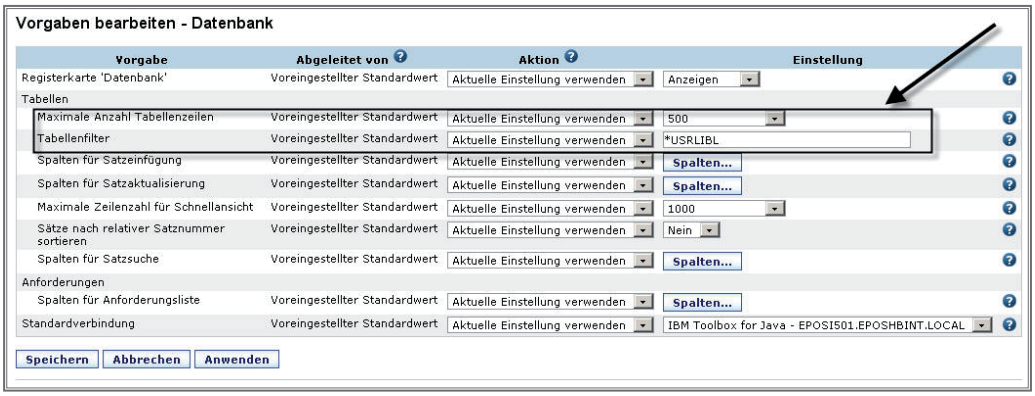
Diese Funktionen werden wir später betrachten. Zunächst möchte ich jedoch die Anzahl der angezeigten Tabellen reduzieren. Dazu blättere ich an den unteren Rand der Seite.

7.4.3.5.1
Seite 2



Vorgaben ändern

Ein Klick auf „Vorgaben für Datenbank“ öffnet die entsprechende Seite:



Vorgaben bearbeiten

In dieser Seite lassen sich eine Reihe von Voreinstellungen für den Bereich „Datenbank“ festlegen. Mich interessiert momentan lediglich der markierte Bereich, da ich die Anzeige auf wichtige Objekte reduzieren möchte.

Maximale Anzahl Tabellenzeilen

Beschreibung: Diese Vorgabe gibt an, wie viele Tabellen höchstens zurückgegeben werden, wenn die Liste „Datenbanktabellen“ angezeigt wird.

Gültige Werte: „50“ bis „10.000“, Kein Höchstwert = Standardwert „500“

Ich ändere diesen Wert auf Standartwert „500“. Achtung! Wenn Sie sich zu viele Bibliotheken mit einer großen Anzahl von Tabellen anzeigen lassen, dann wird die Datenanzeige träge.

Tabellenfilter

Beschreibung: Diese Vorgabe gibt die Tabellen an, die in der Liste „Datenbanktabellen“ angezeigt werden sollen.

Gültige Werte: Eine durch Kommata getrennte Liste mit Schemata, Schemafiltern, Tabellen und Tabellenfiltern. Das Prozentzeichen (%) wird als Filterzeichen verwendet.

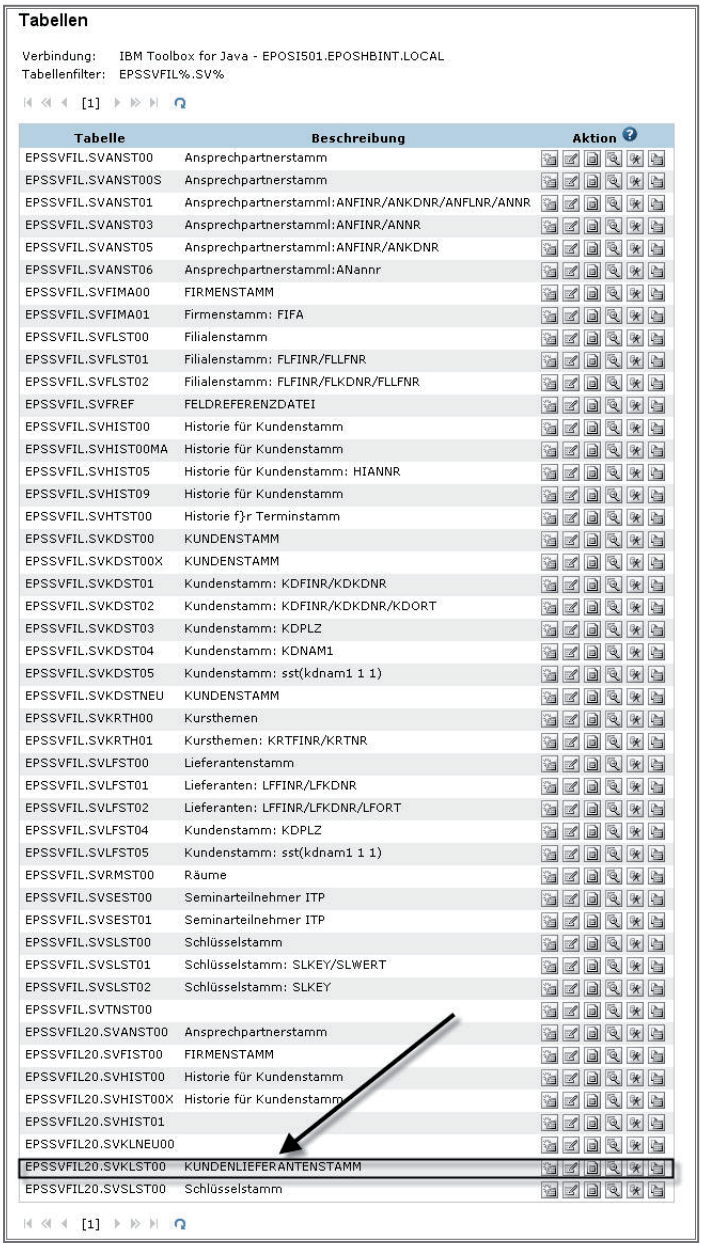
Mit dem Sonderwert *USRLIBL werden alle Tabellen im Benutzerteil der

Bibliotheksliste angegeben. Die Angabe „QIWS,MYSCHEMA. MYTABLE,T%,%.R%“ ruft beispielsweise alle Tabellen in QIWS ab; die Tabelle „MYSCHEMA.MYTABLE“ ruft alle Tabellen in Schemata, die mit „T“ beginnen, sowie alle Tabellen, die mit dem Buchstaben „R“ beginnen, ab.

Standardwert: *USRLIBL

Diesen Wert ändere ich auf „EPSSVFIL%.SV%“. Er zeigt mir alle Tabellen mit der Startzeichenfolge „SV“ in Bibliotheken an, deren Name mit der Zeichenfolge „EPSSVFIL“ beginnt.

Ich beende die Seite mit „Speichern“.



Tabellen

Verbindung: IBM Toolbox for Java - EPOSI501.EPOSHBINT.LOCAL
Tabellenfilter: EPSSVFIL%.SV%

[1]

Tabelle	Beschreibung	Aktion
EPSSVFIL.SVANST00	Ansprechpartnerstamm	
EPSSVFIL.SVANST00S	Ansprechpartnerstamm	
EPSSVFIL.SVANST01	Ansprechpartnerstamml:ANFINR/ANKDNR/ANFLNR/ANNR	
EPSSVFIL.SVANST03	Ansprechpartnerstamml:ANFINR/ANNR	
EPSSVFIL.SVANST05	Ansprechpartnerstamml:ANFINR/ANKDNR	
EPSSVFIL.SVANST06	Ansprechpartnerstamml:ANannr	
EPSSVFIL.SVFIMA00	FIRMENSTAMM	
EPSSVFIL.SVFIMA01	Firmenstamm: FIFA	
EPSSVFIL.SVFLST00	Filialenstamm	
EPSSVFIL.SVFLST01	Filialenstamm: FLFINR/FLLFNR	
EPSSVFIL.SVFLST02	Filialenstamm: FLFINR/FLKDNR/FLLFNR	
EPSSVFIL.SVFREF	FELDREFERENZDATEI	
EPSSVFIL.SVHIST00	Historie für Kundenstamm	
EPSSVFIL.SVHIST00MA	Historie für Kundenstamm	
EPSSVFIL.SVHIST05	Historie für Kundenstamm: HIANNR	
EPSSVFIL.SVHIST09	Historie für Kundenstamm	
EPSSVFIL.SVHTST00	Historie f}r Terminstamm	
EPSSVFIL.SVKDST00	KUNDENSTAMM	
EPSSVFIL.SVKDST00X	KUNDENSTAMM	
EPSSVFIL.SVKDST01	Kundenstamm: KDFINR/KDKDNR	
EPSSVFIL.SVKDST02	Kundenstamm: KDFINR/KDKDNR/KDORT	
EPSSVFIL.SVKDST03	Kundenstamm: KDPLZ	
EPSSVFIL.SVKDST04	Kundenstamm: KDNAM1	
EPSSVFIL.SVKDST05	Kundenstamm: sst(kdnam1 1 1)	
EPSSVFIL.SVKDSTNEU	KUNDENSTAMM	
EPSSVFIL.SVKRTH00	Kursthemen	
EPSSVFIL.SVKRTH01	Kursthemen: KRTFINR/KRTNR	
EPSSVFIL.SVLFST00	Lieferantenstamm	
EPSSVFIL.SVLFST01	Lieferanten: LFFINR/LFKDNR	
EPSSVFIL.SVLFST02	Lieferanten: LFFINR/LFKDNR/LFORT	
EPSSVFIL.SVLFST04	Kundenstamm: KDPLZ	
EPSSVFIL.SVLFST05	Kundenstamm: sst(kdnam1 1 1)	
EPSSVFIL.SVRMST00	Räume	
EPSSVFIL.SVSEST00	Seminarteilnehmer ITP	
EPSSVFIL.SVSEST01	Seminarteilnehmer ITP	
EPSSVFIL.SVSLST00	Schlüsselstamm	
EPSSVFIL.SVSLST01	Schlüsselstamm: SLKEY/SLWERT	
EPSSVFIL.SVSLST02	Schlüsselstamm: SLKEY	
EPSSVFIL.SVTNST00		
EPSSVFIL20.SVANST00	Ansprechpartnerstamm	
EPSSVFIL20.SVFIST00	FIRMENSTAMM	
EPSSVFIL20.SVHIST00	Historie für Kundenstamm	
EPSSVFIL20.SVHIST00X	Historie für Kundenstamm	
EPSSVFIL20.SVHIST01		
EPSSVFIL20.SVKLNEU00		
EPSSVFIL20.SVKLST00	KUNDENLIEFERANTENSTAMM	
EPSSVFIL20.SVSLST00	Schlüsselstamm	

[1]

7.4.3.5.1
Seite 4

Mit einer Tabelle arbeiten

Im Folgenden werde ich mit der Tabelle „SVKLST00“ in der Bibliothek „EPS-VFIL20“ arbeiten.

Satz einfügen (INSERT)

Dieser Dialog bedarf kaum einer Erläuterung.

Satz einfügen

Um einen Satz einzufügen, geben Sie Spaltenwerte an, und klicken Sie auf 'Satz einfügen'.

Spalte	Typ	Wert	Beschreibung
KLFINR	DECIMAL(2,0)		
KLIDNR	DECIMAL(7,0)	NULL	
KLANKZ	DECIMAL(2,0)	NULL	
KLNAM1	CHAR(32)	NULL	
KLNAM2	CHAR(32)	NULL	
KLNAM3	CHAR(32)	NULL	
KLSTR	CHAR(32)	NULL	
KLPSTF	CHAR(10)	NULL	
KLPLZP	CHAR(7)	NULL	
KLORTP	CHAR(25)	NULL	
KLLAND	CHAR(3)	NUL	
KLPLZ	CHAR(7)	NULL	
KLORT	CHAR(25)	NULL	
KLTELK	CHAR(3)	NUL	
KLTEL	CHAR(20)	NULL	
KLFAXK	CHAR(3)	NUL	
KLFAX	CHAR(20)	NULL	
KLFTFK	CHAR(3)	NUL	
KLFTF	CHAR(20)	NULL	
KLTLXK	CHAR(3)	NUL	
KLULFNR	CHAR(20)	NULL	
KLEML	CHAR(60)	NULL	
KLSBE1	CHAR(7)	NULL	
KLSBE2	CHAR(7)	NULL	
KLBANK	CHAR(30)	NULL	
KLBLZ	CHAR(8)	NULL	
KLKTO	CHAR(15)	NULL	
KLDATN	DATE(10)		
KLUSRN	CHAR(10)	NULL	
KLDATC	DATE(10)		
KLUSRC	CHAR(10)	NULL	
KLINTER	CHAR(60)	NULL	

Satz einfügen

Satz einfügen 1

Folgende Details sind zu beachten:

- Felder, die nicht versorgt werden, verlangen den Standarddatentyp. Sehen Sie „NULL“, dann handelt es sich hier um Felder die NULL-fähig sind.
- Das Pluszeichen („+“) macht ein Feld mit einem generierten Standardwert kenntlich.
- Das Sternzeichen („*“) besagt, dass ein Feld erforderlich ist.
- Datums- und Zeitwerte müssen in einem Format eingegeben werden, das der aktuellen Konfiguration des JDBC-Treibers entspricht.
- Die folgenden SQL-Registerwerte für Datums-, Zeit- und Zeitmarkenfelder werden unterstützt:
  - CURRENT_DATE
  - CURRENT_TIME
  - CURRENT_TIMESTAMP

Ich füge folgenden Satz ein:

KDFINR	1	Firma
KDIDNR	3000000	Idennummer
KDNAM1	EPOS GmbH	Name
KDDATN	Aktuelles Datum	Datum der Erfassung
KDUSRN	LUTTKUS	Verantwortlich für Erfassung
KDDATC	Aktuelles Datum	Datum der letzten Änderung
KDUSRC	LUTTKUS	Verantwortlich für letzte Änderung

7.4.3.5.1

Seite 6

Satz einfügen

Um einen Satz einzufügen, geben Sie Spaltenwerte an, und klicken Sie auf 'Satz einfügen'.

Spalte	Typ	Wert	Beschreibung
KLFINR	DECIMAL(2,0)	1	
KLIDNR	DECIMAL(7,0)	3000000	
KLANKZ	DECIMAL(2,0)	NULL	
KLNAM1	CHAR(32)	EPOS GmbH	
KLNAM2	CHAR(32)	NULL	
KLNAM3	CHAR(32)	NULL	
KLSTR	CHAR(32)	NULL	
KLPSTF	CHAR(10)	NULL	
KLPLZP	CHAR(7)	NULL	
KLORTP	CHAR(25)	NULL	
KLLAND	CHAR(3)	NUL	
KLPLZ	CHAR(7)	NULL	
KLORT	CHAR(25)	NULL	
KLTELK	CHAR(3)	NUL	
KLTEL	CHAR(20)	NULL	
KLFAXK	CHAR(3)	NUL	
KLFAX	CHAR(20)	NULL	
KLFTFK	CHAR(3)	NUL	
KLFTF	CHAR(20)	NULL	
KLTLXK	CHAR(3)	NUL	
KLULFNR	CHAR(20)	NULL	
KLEML	CHAR(60)	NULL	
KLSBE1	CHAR(7)	NULL	
KLSBE2	CHAR(7)	NULL	
KLBANK	CHAR(30)	NULL	
KLBLZ	CHAR(8)	NULL	
KLKTO	CHAR(15)	NULL	
KLDATN	DATE(10)	URRENT_DATE	
KLUSRN	CHAR(10)	LUTTKUS	
KLDATC	DATE(10)	URRENT_DATE	
KLUSRC	CHAR(10)	LUTTKUS	
KLINTER	CHAR(60)	NULL	

Satz einfügen

Satz einfügen 2

Die Eingabe bestätige ich mit „Satz einfügen“.

Sofern keine Eingabefehler vorliegen, wird folgende Seite angezeigt:

Satz einfügen

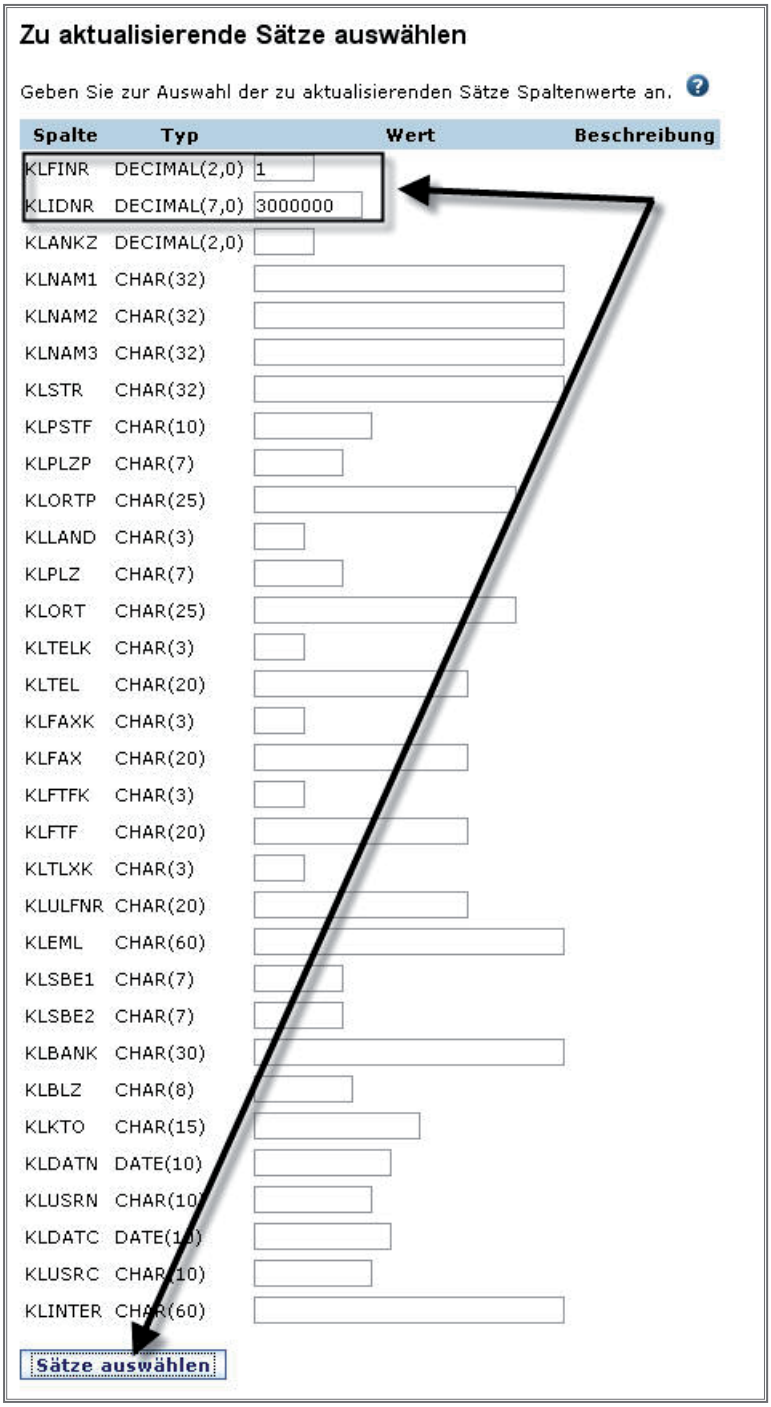
Der Satz wurde erfolgreich eingefügt.

Weiteren Satz einfügen
Zurück

Satz einfügen 3

Satz aktualisieren (UPDATE/DELETE)

Als Nächstes werde ich den soeben erfassten Satz ändern. In der Liste der Tabellenanzeige klicke ich auf das entsprechende Icon.



Schlüsselwerte eingeben

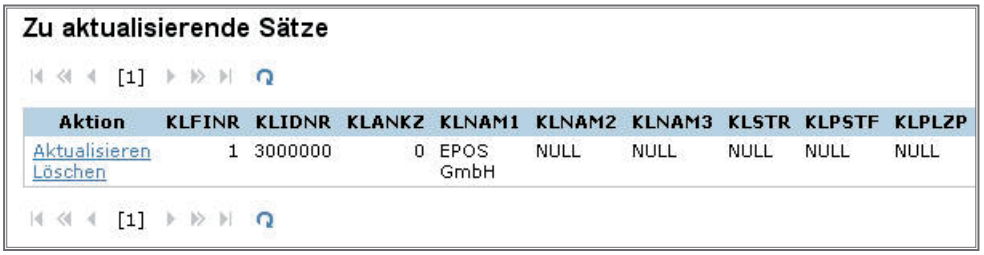
7.4.3.5.1
Seite 8

Details

Um alle Sätze auszuwählen, geben Sie keine Werte an.

Bei Zeichenfeldern kann das Prozentzeichen („%“) als Platzhalterzeichen verwendet werden. Wird beispielsweise der Wert „A%“ in einer Spalte angegeben, werden alle Sätze ausgewählt, deren Wert mit A beginnt.

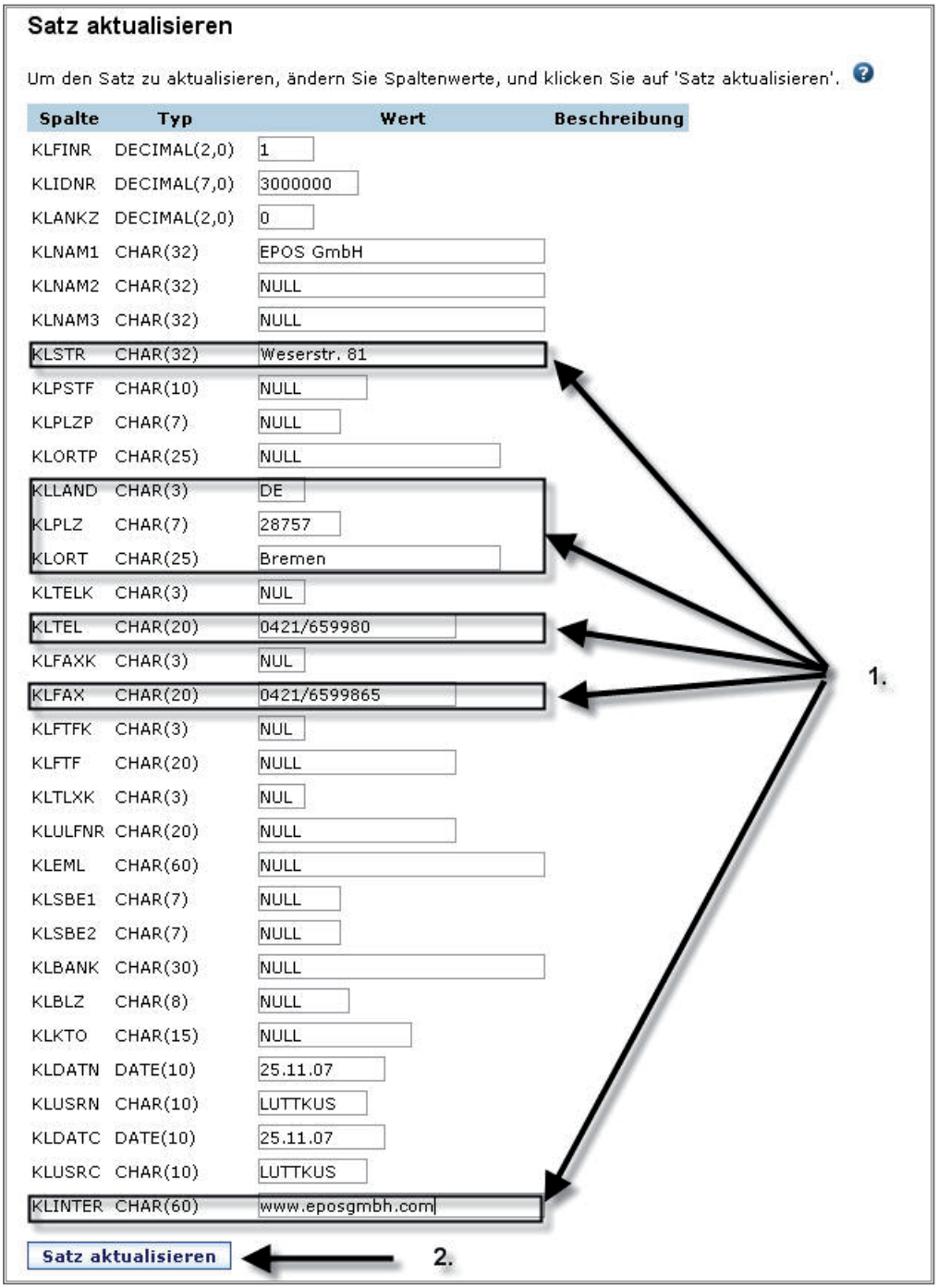
In der angezeigten Seite gebe ich die Schlüsselwerte des Satzes ein, die im vorherigen Arbeitsschritt erfasst wurden; anschließend klicke ich auf „Sätze auswählen“.



Zu aktualisierenden Satz auswählen

Sie sehen, dass es hier die Möglichkeit gibt, Werte eines bestehenden Satzes zu ändern bzw. diesen aus der Tabelle zu löschen.

Ich klicke auf „Aktualisieren“.



Werte eintragen

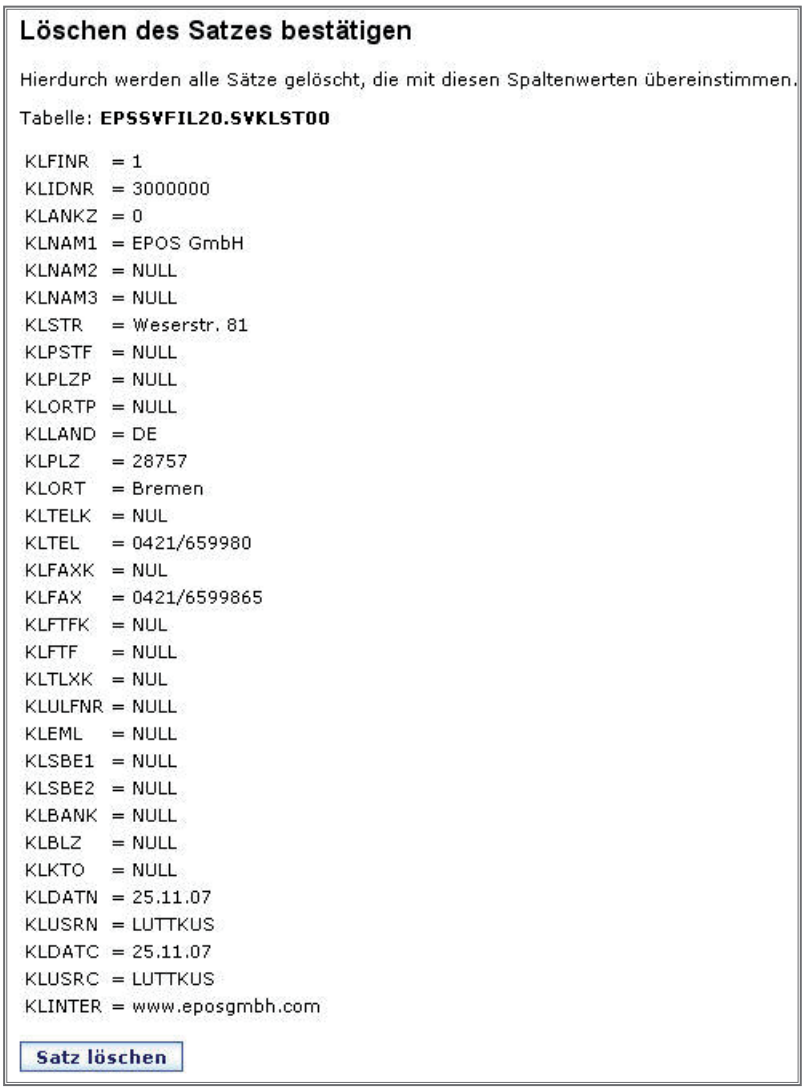
Im oben angezeigten Dialog trage ich die markierten Werte ein und klicke auf „Satz aktualisieren“.

7.4.3.5.1
Seite 10

Als Bestätigung erscheint folgende Seite:

Satz wurde aktualisiert

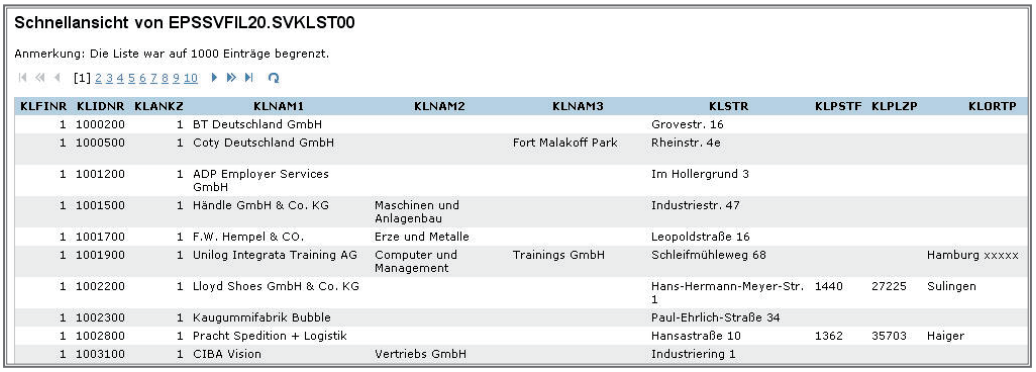
Um einen Satz zu löschen, klicken Sie auf den entsprechenden Link (siehe „Löschen“ in der Abbildung „Zu aktualisierenden Satz auswählen „).



Satz löschen

Schnellansicht

Die Funktion der Schnellansicht stellt den gesamten Dateiinhalt dar.



Schnellansicht von EPSSVFIL20.SVKLST00

Anmerkung: Die Liste war auf 1000 Einträge begrenzt.

[1] 2 3 4 5 6 7 8 9 10

KLFINR	KLIDNR	KLANKZ	KLNAM1	KLNAM2	KLNAM3	KLSTR	KLPSTF	KLPLZP	KLORTP
1	1000200	1	BT Deutschland GmbH			Grovestr. 16			
1	1000500	1	Coty Deutschland GmbH		Fort Malakoff Park	Rheinstr. 4e			
1	1001200	1	ADP Employer Services GmbH			Im Hollergrund 3			
1	1001500	1	Händle GmbH & Co. KG	Maschinen und Anlagenbau		Industriestr. 47			
1	1001700	1	F.W. Hempel & CO.	Erze und Metalle		Leopoldstraße 16			
1	1001900	1	Unilog Integrata Training AG	Computer und Management	Trainings GmbH	Schleifmühleweg 68			Hamburg xxxxx
1	1002200	1	Lloyd Shoes GmbH & Co. KG			Hans-Hermann-Meyer-Str. 1	1440	27225	Sulingen
1	1002300	1	Kaugummifabrik Bubble			Paul-Ehrlich-Straße 34			
1	1002800	1	Pracht Spedition + Logistik			Hansastraße 10	1362	35703	Haiger
1	1003100	1	CIBA Vision	Vertriebs GmbH		Industriering 1			

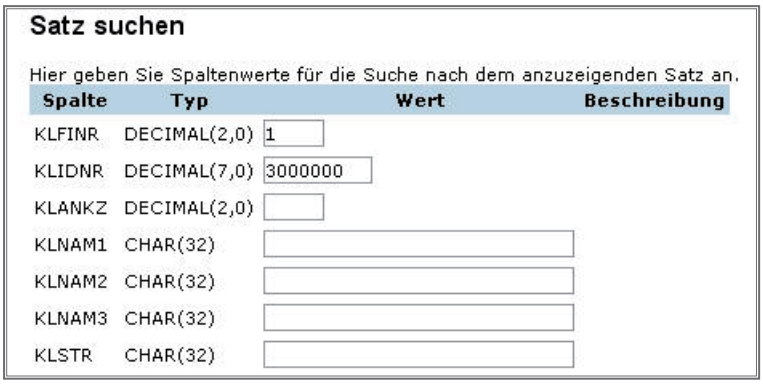
Schnellansicht

Die Schnellansicht entspricht in ihrer Funktionalität einem unqualifizierten RUNQRY; sie benötigt daher keine weitere Erläuterung.

Satz suchen

Um die Funktion „Satz suchen“ zu testen, werde ich den vorher erfassten und dann geänderten Satz suchen.

Ein Klick auf das entsprechende Symbol öffnet die nachfolgend angezeigte Seite:



Satz suchen

Hier geben Sie Spaltenwerte für die Suche nach dem anzuzeigenden Satz an.

Spalte	Typ	Wert	Beschreibung
KLFINR	DECIMAL(2,0)	1	
KLIDNR	DECIMAL(7,0)	3000000	
KLANKZ	DECIMAL(2,0)		
KLNAM1	CHAR(32)		
KLNAM2	CHAR(32)		
KLNAM3	CHAR(32)		
KLSTR	CHAR(32)		

Satz suchen

Folgendes Ergebnis wird angezeigt:

7.4.3.5.1
Seite 12

Satz anzeigen

Spalte	Typ	Wert	Beschreibung
KLFINR	DECIMAL(2,0)	1	
KLIDNR	DECIMAL(7,0)	3000000	
KLANKZ	DECIMAL(2,0)	0	
KLNAM1	CHAR(32)	EPOS GmbH	
KLNAM2	CHAR(32)	NULL	
KLNAM3	CHAR(32)	NULL	
KLSTR	CHAR(32)	Weserstr. 81	
KLPSTF	CHAR(10)	NULL	
KLPLZP	CHAR(7)	NULL	
KLORTP	CHAR(25)	NULL	
KLLAND	CHAR(3)	DE	
KLPLZ	CHAR(7)	28757	
KLORT	CHAR(25)	Bremen	
KLTELK	CHAR(3)	NUL	
KLTEL	CHAR(20)	0421/659980	
KLFAXK	CHAR(3)	NUL	
KLFAX	CHAR(20)	0421/6599865	
KLFTFK	CHAR(3)	NUL	
KLFTF	CHAR(20)	NULL	
KLTLXK	CHAR(3)	NUL	
KLULFNR	CHAR(20)	NULL	
KLEML	CHAR(60)	NULL	
KLSBE1	CHAR(7)	NULL	
KLSBE2	CHAR(7)	NULL	
KLBANK	CHAR(30)	NULL	
KLBLZ	CHAR(8)	NULL	
KLKTO	CHAR(15)	NULL	
KLDATN	DATE(10)	25.11.07	
KLUSRN	CHAR(10)	LUTTKUS	
KLDATC	DATE(10)	25.11.07	
KLUSRC	CHAR(10)	LUTTKUS	
KLINTER	CHAR(60)	www.eposgmbh.com	

Suchergebnis

Hier ist es nicht nur möglich, einen einzelnen Satz zu suchen, es können auch Satzgruppen gesucht und gefunden werden.

Satz suchen

Hier geben Sie Spaltenwerte für die Suche nach dem anzuzeigenden Satz an.

Spalte	Typ	Wert	Beschreibung
KLFINR	DECIMAL(2,0)		
KLIDNR	DECIMAL(7,0)		
KLANKZ	DECIMAL(2,0)		
KLNAM1	CHAR(32)	%bau%	
KLNAM2	CHAR(32)		

Wildcard-Suche

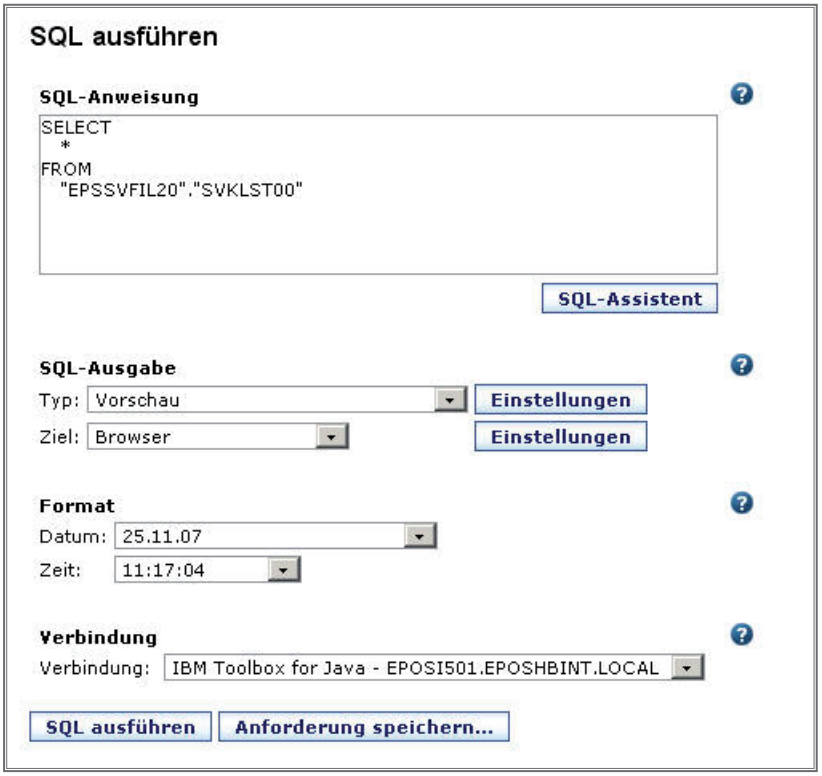
Um alle Sätze anzuzeigen, in denen die Zeichenfolge „bau“ vorkommt, wird ein SQL-Platzhalter (Wildcards – %) verwendet.

Suchergebnis

SQL ausführen

Mit der Funktion „SQL ausführen“ lassen sich komplexere Aktionen ausführen.

Im Folgenden rufe ich diese Funktion auf:



SQL ausführen

Im Parameter „SQL-Anweisung“ finden Sie einen einfachen Texteditor, mit dem SQL-Select-Befehle ausgeführt werden können. Hier ist bereits ein Standard-Select-Befehl eingetragen, der alle Sätze der abgefragten Datei anzeigt.

Die SQL-Anweisung kann jede Anweisung sein, die durch den zu Grunde lie-

7.4.3.5.1
Seite 14

genden JDBC-Treiber unterstützt wird.

Tabellennamen müssen in der SQL-Schreibweise mit Trennzeichen angegeben werden. Der Wert „MYSCHEMA.MYTABLE“ ist ein Beispiel für einen gültigen Tabellennamen.

Der SQL-Assistent unterstützt Sie bei der Erstellung von der SQL-Anweisung SELECT. Die resultierende Anweisung wird auf der Hauptseite „SQL ausführen“ angezeigt; sie kann dort wie gewünscht geändert werden.

Details zur SQL-Ausgabe

Bei der Ausführung einer SQL-Anweisung kann die Ausgabe an den Browser, an das Integrated-File-System des System i, als Mail-Anlage oder an einen persönlichen Ordner auf dem System i gesendet werden.

Wenn die SQL-Ausgabe an den Browser gesendet wird, kann sie entweder im Browser angezeigt oder mit einer anderen Anwendung geladen beziehungsweise als Datei gespeichert werden. Das Ausgabeverhalten richtet sich nach dem ausgewählten SQL-Ausgabetyp und den aktuellen Browser-Einstellungen.

Wenn ein anderes Ausgabeziel als ein Browser verwendet wird, wird die SQL-Anweisung im Hintergrund ausgeführt, und die Steuerung wird sofort an den Browser zurückgegeben. Der Status über die Fertigstellung der Anweisung wird entweder als E-Mail-Nachricht gesendet oder als Element in einem persönlichen Ordner gespeichert.

Die meisten Zieloptionen unterstützen Einstellungen, mit denen die Ausgabe weiter angepasst werden kann.

Die folgenden Ausgabetypen werden unterstützt:

- Vorschau
  Diese Option verwendet das Format einer Liste aus Seiten, die im Browser angezeigt wird. Die maximale Zeilenzahl, die in der Liste zurückgegeben wird, kann über die Anpassungsfunktion zum Festlegen einer Datenbankvorgabe eingeschränkt werden.
- Comma Separated Value (CSV)
  Ein Textformat, bei dem Felder durch Kommata voneinander abgegrenzt werden.
- Data Interchange Format (DIF)
  Ein Format, bei dem Daten in Zeilen und Spalten dargestellt werden. Das Format DIF wird für den Datenaustausch zwischen Tabellenkalkulationsprogrammen und anderen Anwendungsprogrammen verwendet.
- Extensible Markup Language (XML)
  Ein Universalformat für gegliederte Dokumente und Daten im Web.

- HTML
  Das Format, das allgemein von Internet-Browsern verwendet wird. Die Einstellung „Zeilen pro Tabelle“ steuert, ob die Ausgabe als Liste aus Seiten – ähnlich wie beim Typ „Vorschau“ – oder in einer einzigen HTML-Seite zurückgegeben wird.
- Lotus 1-2-3 Version 1
  Das Format, das durch Lotus 1-2-3, Version 1 verwendet wird. Die generierte Ausgabe kann auch mit neueren Versionen von Lotus 1-2-3 verwendet werden. Die Ausgabe wird auf 8192 Zeilen abgeschnitten, da das Format von Lotus 1-2-3, Version 1 maximal 8192 Zeilen pro Arbeitsblatt unterstützt. Die Ausgabe wird auf 256 Spalten abgeschnitten, da dieses Format maximal 256 Spalten unterstützt.
- Microsoft Excel 3
  Das Format, das durch Microsoft Excel 3 verwendet wird. Die generierte Ausgabe kann auch von neueren Versionen der Anwendung „Microsoft Excel“ verwendet werden. Die Ausgabe wird bei 16384 Zeilen abgeschnitten, da das Format von Microsoft Excel 3 maximal 16384 Zeilen unterstützt. Die Ausgabe wird auf 256 Spalten abgeschnitten, da dieses Format maximal 256 Spalten unterstützt.
- Microsoft Excel 4
  Das Format, das durch Microsoft Excel 4 verwendet wird. Die generierte Ausgabe kann auch von neueren Versionen der Anwendung „Microsoft Excel“ verwendet werden. Die Ausgabe wird bei 16384 Zeilen abgeschnitten, da das Format von Microsoft Excel 4 maximal 16384 Zeilen unterstützt. Die Ausgabe wird auf 256 Spalten abgeschnitten, da dieses Format maximal 256 Spalten unterstützt.
- Microsoft Excel XML
  Das XML-Tabellenformat, das von Microsoft Excel 2002 verwendet wird. Die generierte Ausgabe kann auch von neueren Versionen der Anwendung „Microsoft Excel“ verwendet werden. Die Ausgabe wird in Arbeitsblätter mit maximal 65536 Zeilen pro Arbeitsblatt gestellt. Es werden solange zusätzliche Arbeitsblätter in der Excel-Arbeitsmappe erstellt, bis alle Zeilen hinzugefügt wurden. Die Ausgabe wird auf 256 Spalten abgeschnitten, da dieses Format maximal 256 Spalten unterstützt.
- OpenDocument-Arbeitsblatt
  Das XML-Tabellenformat, das von Büroanwendungen wie beispielsweise OpenOffice.org verwendet wird. Die Ausgabe wird in Arbeitsblätter mit maximal 65536 Zeilen pro Arbeitsblatt gestellt. Es werden solange zusätzliche Arbeitsblätter erstellt, bis alle Zeilen hinzugefügt wurden.
- Portable Document Format (PDF)
  Ein Dateiformat, das alle Schriftarten, Formatierungen, Grafikelemente

7.4.3.5.1
Seite 16

und Farbeinstellungen ungeachtet der Anwendung und der Plattform beibehält, die bei der Erstellung der Datei verwendet wurden. Dieses Dateiformat wird durch Adobe Acrobat unterstützt.

- Unformatierter Text
  Das Format für unformatierten Text, das zum Bearbeiten, Anzeigen und Drucken verwendet wird. Zwischen den Datenfeldern werden keine Trennzeichen verwendet.
- Text mit Tabulatorfeldern
  Ein Textformat, bei dem Felder durch Tabulatorzeichen voneinander abgegrenzt werden.

Details zum Format

Die Formateinstellungen steuern, wie Werte in Datums- und Zeitspalten zurückgegeben werden.

Verfügbare Formate sind das kurze, das mittlere, das lange und das vollständige Format der aktuellen Sprache. Außerdem stehen die ISO-Standardformate für Datum und Zeit zur Verfügung.

Manche Sprachen unterstützen nicht vier unterschiedliche Formate für Datum und Zeit. Daher sind einige Einträge in den Formatlisten unter Umständen identisch.

Details zur Verbindung

Die Verbindungseinstellungen steuern die Datenbankverbindung, mit der die SQL-Anweisung ausgeführt wird.

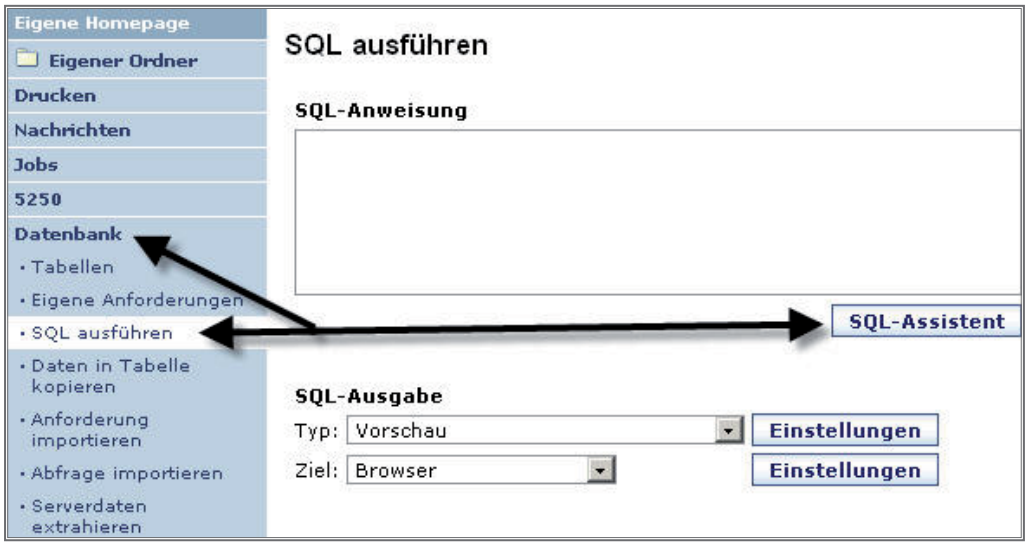
Datenbankverbindungen können durch den Systemadministrator über die Datenbankanpassung definiert werden.

Der SQL-Assistent

Im vorherigen Beitrag haben Sie die Funktion „SQL Ausführen“ kennengelernt; Sie wissen jetzt, dass Sie direkt in das Feld „SQL-Anweisungen“ einen SQL-Befehl eintragen können. Allerdings setzt das jedoch voraus, dass Sie sich in der SQL-Syntax gut auskennen.

Wenn Sie jedoch SQL-syntaktisch noch nicht so sattelfest sind, dann hilft Ihnen bei der Abfrageerstellung der SQL-Assistent.

Im Funktionsmenü klicken Sie links auf „Datenbank → SQL ausführen“.



SQL-Assistent

Sie sehen jetzt links unterhalb des Felds „SQL-Anweisung“ den Button „SQL-Assistent“. Dieser Knopf wird angeklickt.

Vorher möchte ich jedoch mit Ihnen besprechen, welcher Aufgabe wir uns zuwenden werden.

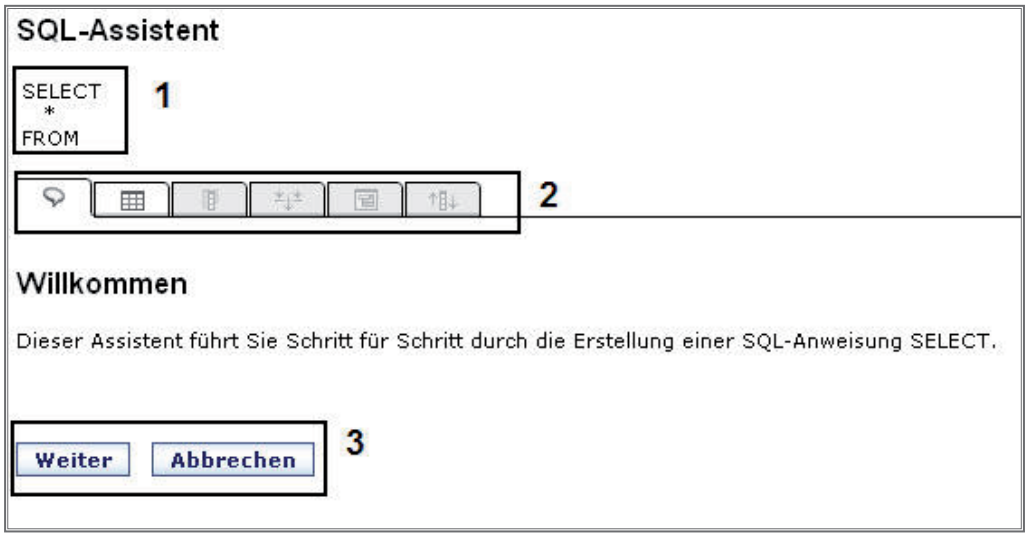
1. Arbeitsschritt:
Zuerst wird eine Abfrage mit dem SQL-Assistenten erstellt, mit der die Datei „SVKLST00“ in der Bibliothek „EPSSVFIL20“ abgefragt wird. Diese Datei ist mandantenfähig. Mit Hilfe des Feldes „KLFINR“ (Firmennummer) sollen gezielt alle Kunden der Firma „1, 2 etc.“ abgefragt werden.

2. Arbeitsschritt:
Wir werden für den Anwender, der die Abfrage ausführen soll, einen Dialog erstellen, mit dessen Hilfe er nur die Firmen auswählen kann, deren Sätze in der Datei „SVKLST00“ gespeichert sind.

7.4.3.5.2
Seite 2

3. Arbeitsschritt:
Wir werden die Abfrage publizieren, so dass alle Systemanwender auf diese Abfrage zugreifen können.

Jetzt klicke ich auf den Knopf „SQL-Assistent“.

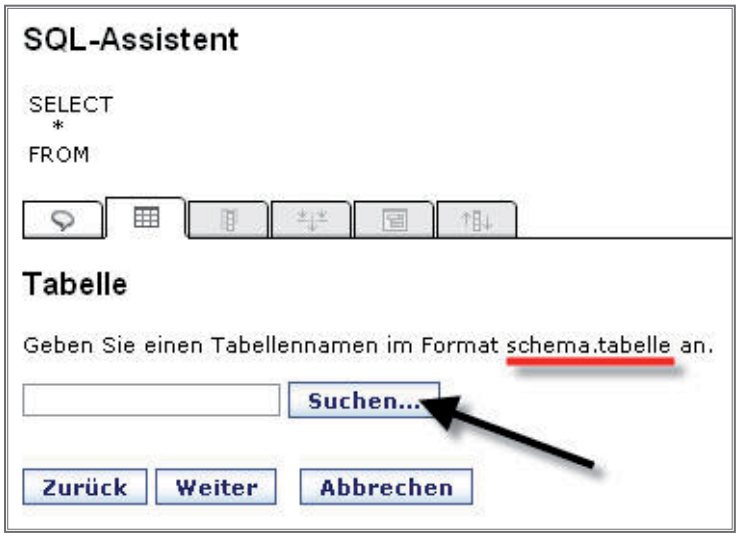


SQL-Assistent-Dialog

Die Startseite des SQL-Assistenten besteht aus drei, für uns wichtigen Elementen:

1. Hier baut sich der von uns via Assistent erstellte SQL-Befehl auf.
2. Mit Hilfe der Reiterleiste wählen wir die jeweiligen Funktionsbereiche eines SQL-Befehls aus – wie z. B. „Felder auswählen“, „Auswahlbedingungen setzen“, „Sortierfolgen bestimmen“ etc. Die Reiter sind kontextsensitiv. Je nach Fortschritt der SQL-Generierung sind verschiedene Reiter deaktiviert. Sie erkennen dies an der leicht grauen Farbe der nicht aktiven Reiter. Am Anfang ist nur der aktuelle Reiter aktiv, dessen Dialog wir in der Anzeige sehen, sowie der logisch nächste Reiter „Tabelle“.
3. Knopfleiste: Alternativ zu den Reitern können wir mit dem Knopf „Weiter“ in den nächsten logischen Reiter springen.

Mit „Weiter“ gehen wir zum nächsten Reiter.



Reiter „Tabelle“

Jetzt wird angegeben, mit welcher Datei der SQL-Befehl abgefragt werden soll. Dazu können wir im gezeigten Eingabefeld direkt die Datei angeben. Die Angabe muss in der syntaktischen Konvention des SQL erfolgen: „bibliothek. dateiname“ oder – wie im Dialog gezeigt – „schema.tabelle“.

Es besteht auch die Möglichkeit, nach der Datei zu suchen. Dazu klicken wir auf „Suchen…“.

Daraufhin öffnet sich folgende Seite:

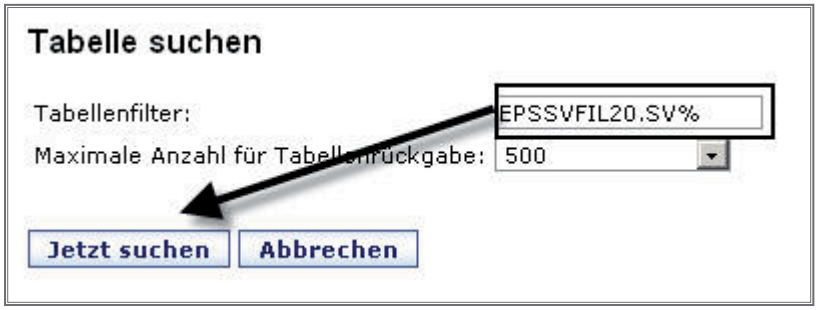


Tabelle suchen

Im Feld „Tabellenfilter“ werden wir jetzt nach allen Dateien in der Bibliothek „EPSSVFIL20 suchen, deren Name mit der Zeichenfolge „SV“ beginnt.

7.4.3.5.2
Seite 4

Wenn man die bereits beschriebenen syntaktischen Konventionen beachtet, dann ist die korrekte Filterzeichenfolge:

```
EPSSVFIL20.SV%
```

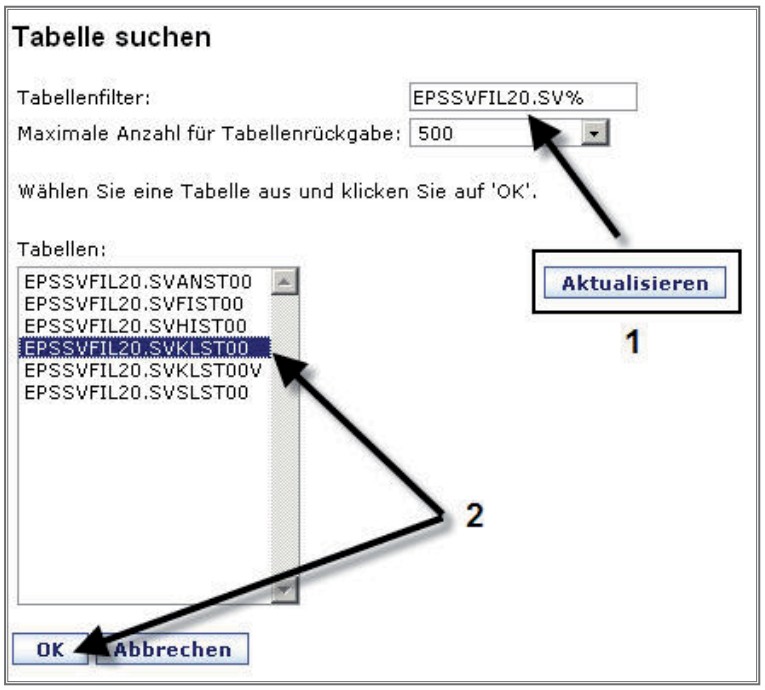
Achtung!

Verwenden Sie hier nicht die Syntax des Betriebssystems: z. B. „SV*“!

Auch bei Bibliotheksnamen können Sie mit der Wildcard „%“ arbeiten. Die Filterzeichenfolge „EPS%.SV%“ wird Ihnen alle Dateien anzeigen, deren Namen mit „SV“ beginnen. Außerdem werden in den Bibliotheken die Namen angezeigt, die mit „EPS“ beginnen.

Ich rate Ihnen, die Suche im System eng zu fassen, da es sonst an dieser Stelle zu relativ langen Wartezeiten kommen kann.

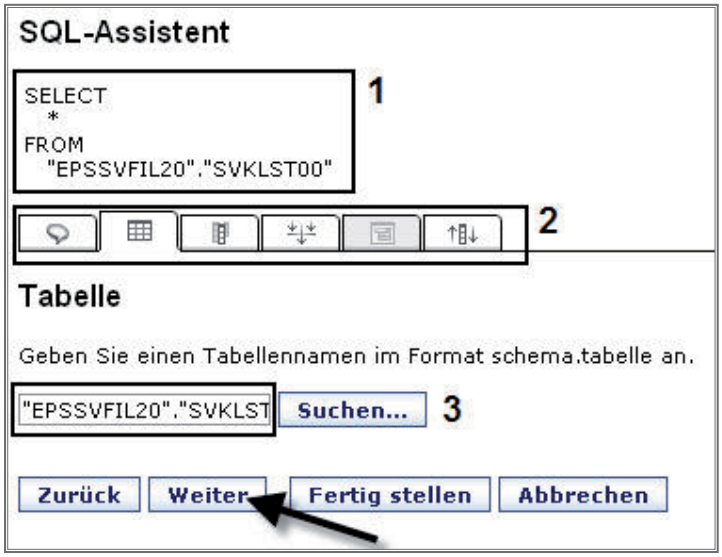
Mit „Jetzt suchen“ gehen wir weiter.



Datei auswählen.

Wenn Sie beim gewählten Filter nicht die gewünschten Dateien (Feld Tabellen) erhalten, dann können Sie („1“) jederzeit den Tabellenfilter ändern und die Anzeige aktualisieren.

Wir wählen im Feld „Tabellen“ die Datei „SVKLST00“ aus und klicken auf „OK“. Dadurch kehren wir automatisch zur Anzeige „Tabellen“ zurück.



Dialog „SQL-Assistent“

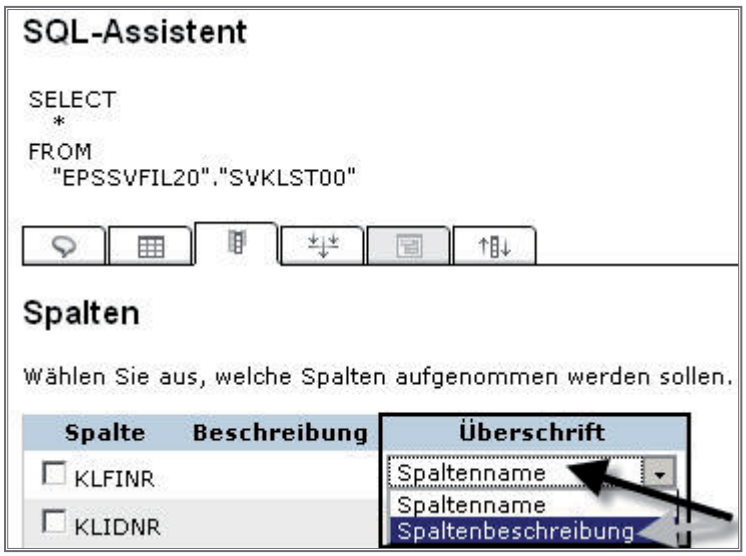
Sie sehen, dass sich die Anzeige des SQL-Assistenten verändert hat:

1. Der SQL-Befehl ist entsprechend unseren Angaben aktualisiert worden.

2.Weitere Reiter sind aktiviert worden.

3. Im Feld „Suchen“ ist der Pfad „EPSSVFIL20.SVKLST00“ eingetragen worden.

Der Button „Weiter“ aktiviert den Reiter „Spalten“.

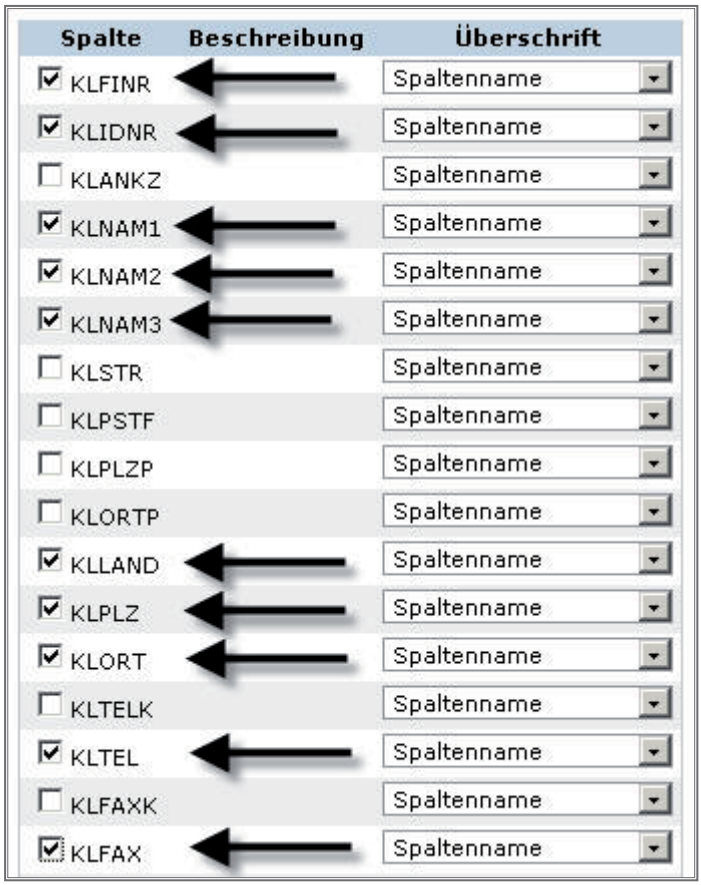


Reiter „Spalten“

7.4.3.5.2
Seite 6

Im Reiter „Spalten“ erhalten Sie eine Liste aller in der Datei „SVKLST00“ befindlichen Felder. Hier entscheiden Sie zum einen, welche Felder in die Auswahl übernommen werden, und zum anderen, ob der Feldname oder die Spaltenüberschrift in die Auswertung mit aufgenommen werden soll.

Da in der Datei „SVKLST00“ Feldnamen und Spaltenüberschriften identisch sind, belasse ich es bei den Grundeinstellungen.



Feldliste

Durch Anklicken der entsprechenden Checkboxen übernehmen wir die gezeigten Felder in die Abfrage.

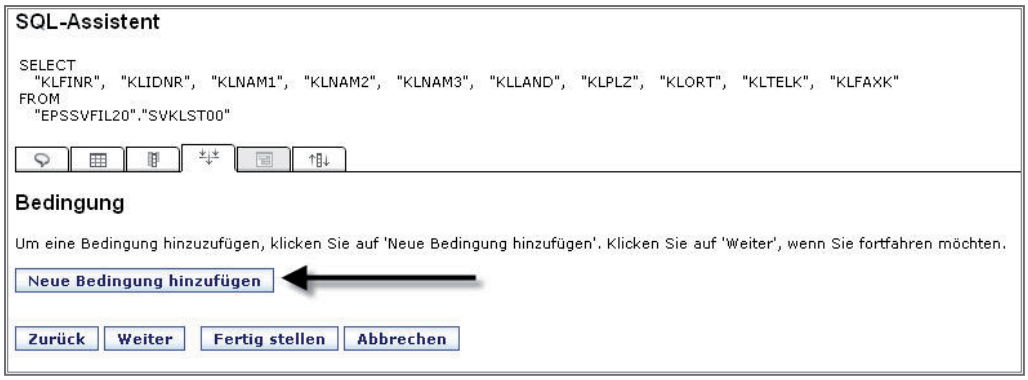
Spaltenreihenfolge

Um die Reihenfolge der Felder, die in die Auswahl übernommen wurden, zu modifizieren, klicken wir auf „Spaltenreihenfolge“.

Feldreihenfolge verändern

Jetzt sehen Sie die übernommenen Felder – versehen mit einer Folgenummer. Durch die Änderung der Folgenummer modifizieren Sie anschließend die Felderreihenfolge.

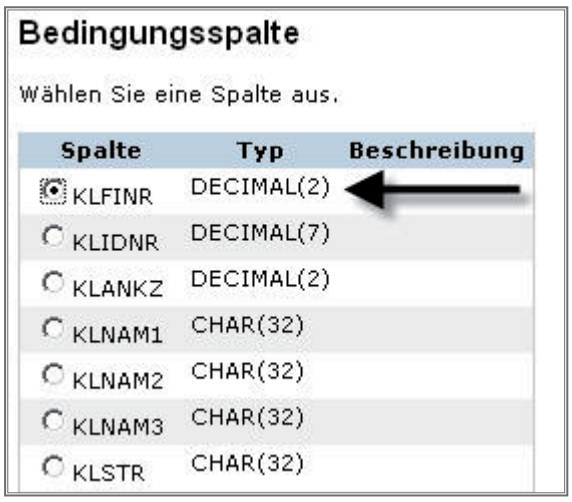
Ich nehme keine Änderung vor und klicke auf „OK“. Dann klicke ich auf „Weiter“ und gelange auf die Seite „Bedingung“.



Reiter „Bedingung“

7.4.3.5.2
Seite 8

Im Reiter „Bedingung“ klicken wir auf „Neue Bedingung hinzufügen“.



Bedingungsfeld wählen

Im Dialog „Bedingungsspalte“ werden alle Felder angezeigt, die Bestandteil der Abfrage sind.

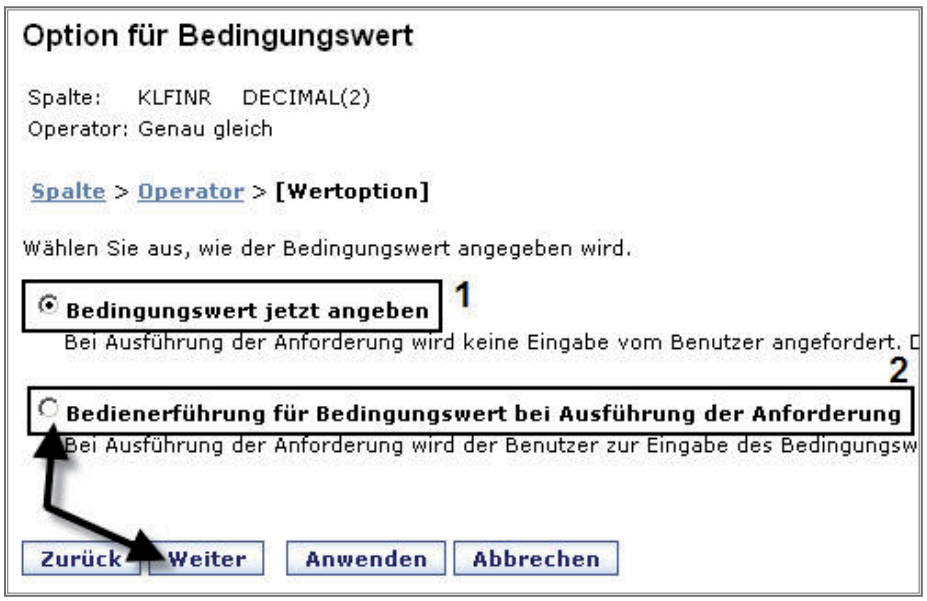
Wir aktivieren das Feld „KLFINR“ als Selektionsfeld und gelangen in den Dialog „Bedingungsoperator“.

Bedingungsoperator

Zur Erinnerung: Wir beabsichtigen eine Abfrage aufzubauen, in der alle Kunden der Firma 1, alle Kunden der Firma 2 etc. angezeigt werden. Aus diesem Grund wählen wir hier den Operator „Genau gleich“ und klicken anschließend auf „Weiter“.

7.4.3.5.2
Seite 9

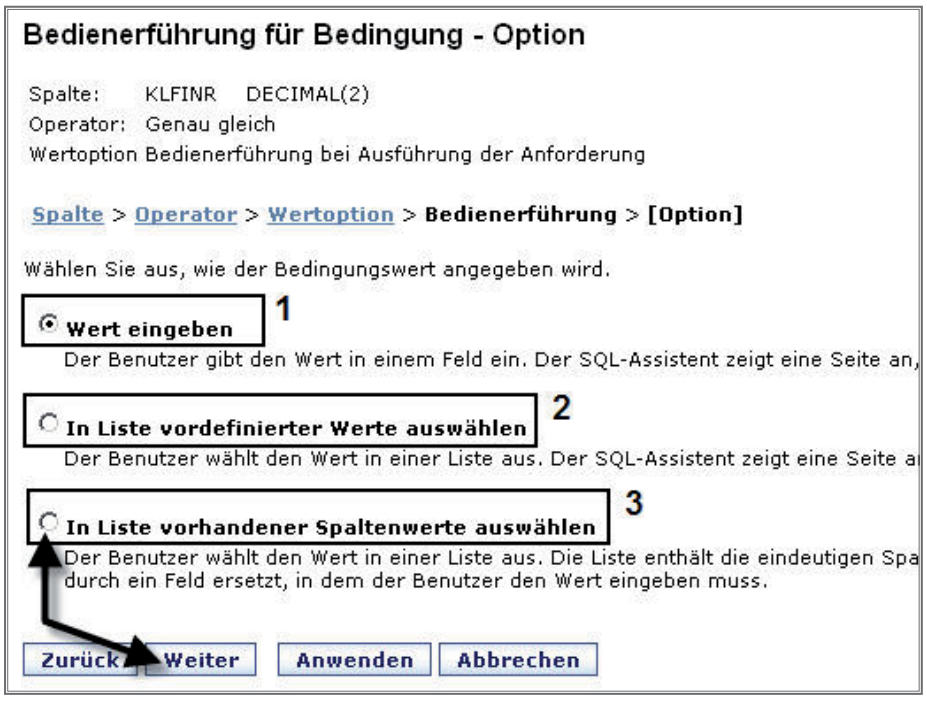
Wir gelangen dann auf die Seite „Option für Bedingungswert“.



Option für Bedingungswert

Wählen Sie die mit „1“ markierte Option an, dann werden Sie eine feste Abfrage für eine Firma erstellen. Da wir jedoch beabsichtigen, die abgefragte Firma durch den Anwender bestimmen zu lassen, klicken wir auf die mit „2“ markierte Auswahl – und anschließend auf „Weiter“.

Wir gelangen in den Dialog „Bedienerführung“.



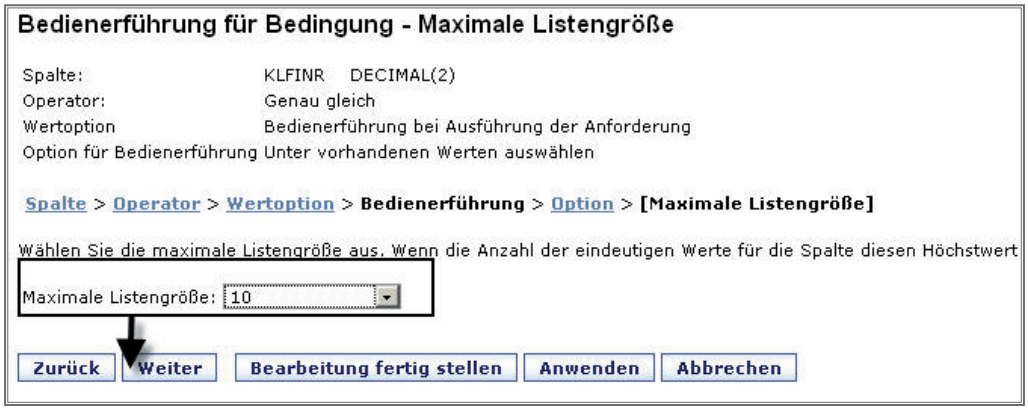
Abfrage „Vergleichswerte wählen“

7.4.3.5.2
Seite 10

Auf dieser Seite haben wir drei Möglichkeiten:

1. Der Anwender erhält ein Eingabefeld. In dieses Feld trägt er die Firmennummer der abzufragenden Firma ein. Diese Dialogform birgt die Gefahr in sich, dass der Anwender eine nicht gültige Firmennummer einträgt und damit eine Null-Ausgabe erhalten kann.
2. Wir definieren für den Anwender eine Firmennummer-Auswahlliste, aus der er den für seine Auswertung passenden Vergleichswert auswählt. Nachteil dieser Methode: Wenn Kunden einer weiteren Firma in die Datei aufgenommen werden, dann kann der Anwender nicht sofort auf die neuen Datensätze zugreifen. Zuerst muss die Vergleichswerte-Liste aktualisiert werden.
3. In einer Auswahlliste wird nur die Firmennummer angezeigt, deren Sätze sich in der Datei befinden. Diese Option wählen wir, denn sie verhält sich in jedem Fall dynamisch.

Jetzt klicken wir auf „Weiter“.



Auswahlliste dimensionieren

Auf der nächsten Seite begrenzen wir die Anzahl der angezeigten Firmennummern auf „10“. Wenn diese Anzahl von Firmen überschritten wird, dann wird statt der Liste wieder ein Eingabefeld ausgegeben.

Auch dieser Dialog wird mit „Weiter“ bestätigt.

Beispielanzeige

Eine Beispielanzeige zeigt, wie der Auswahldialog für Firmennummern später dargestellt wird. Mit „Weiter“ geht es weiter.

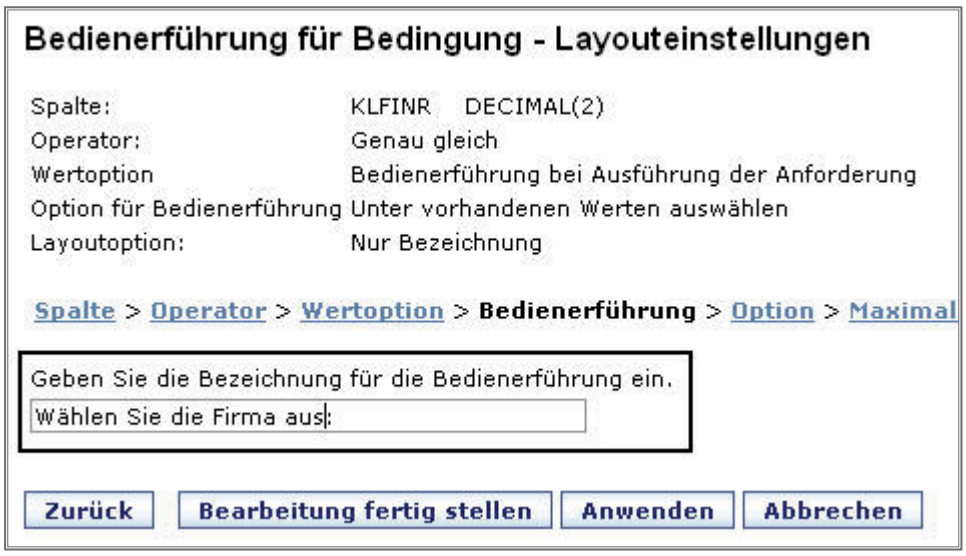
In der Folge geht es um die optische Ausgestaltung des Firmenauswahldialogs.

Bezeichner darstellen

In der gezeigten Combo-Box haben Sie die Möglichkeit, die Anzeige eines Bezeichnungstextes in Relation zur Auswahlliste anzugeben. Die Darstellungsvarianten erklären sich im eingerahmten Bereich selbst.

7.4.3.5.2
Seite 12

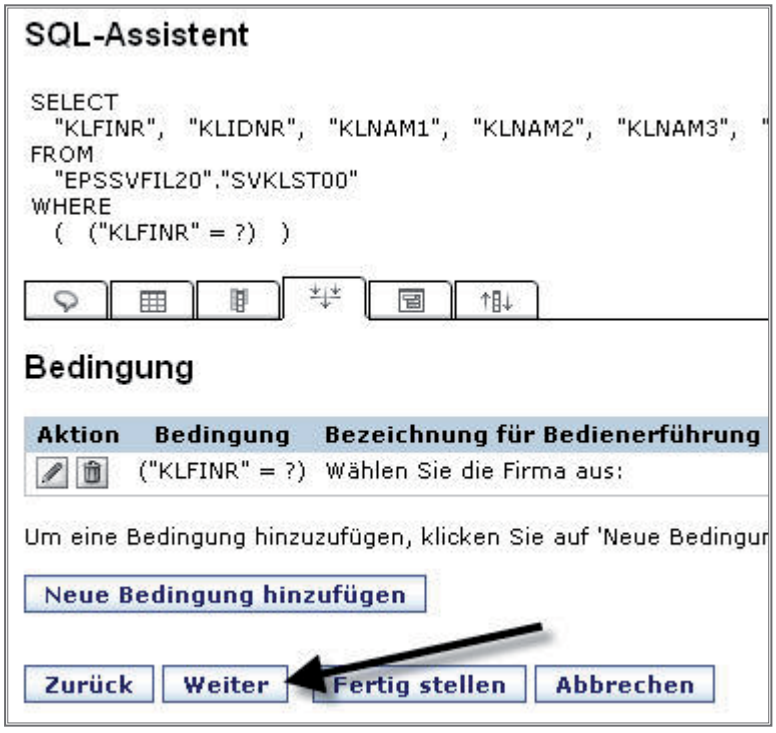
Wir wählen hier „Nur Bezeichnung“ aus und gehen weiter.



Bezeichnertext

Dann muss ein Bezeichnertext vergeben und der Dialog mit „Bearbeitung fertig stellen“ beendet werden.

Jetzt sehen wir die ausformulierte „SQL-Bedingung“.

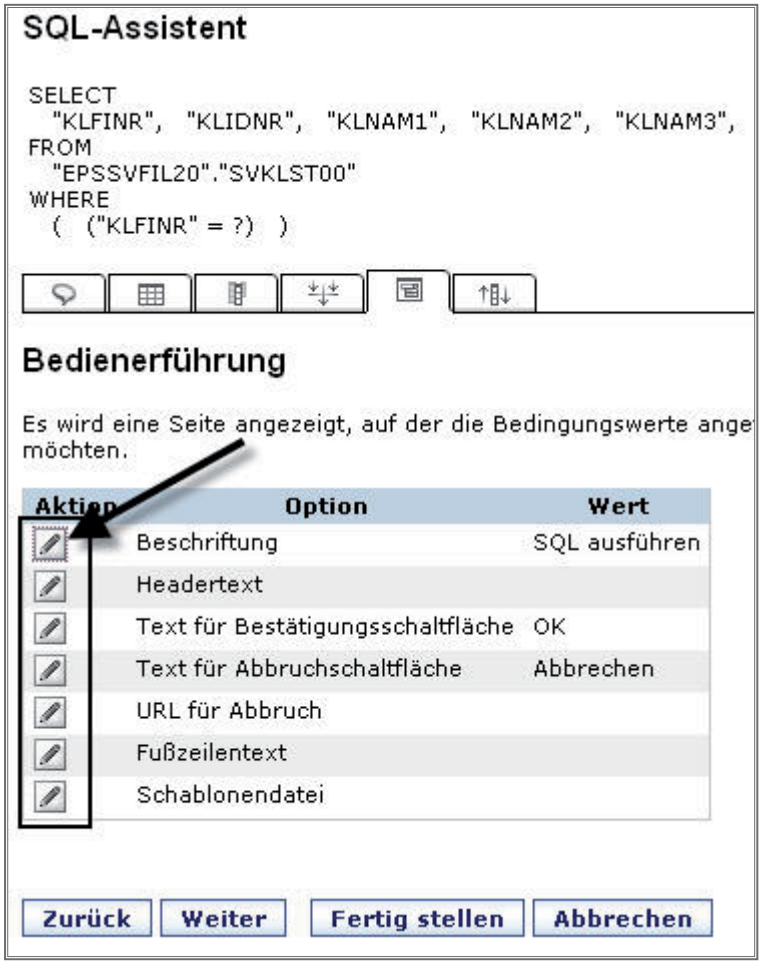


Bedingung komplett

Sollten Sie die Suchbedingung verändern wollen, dann klicken Sie auf das Stift-Symbol. Die Bedingung kann durch einen Klick auf das Mülleimer-Symbol gelöscht werden.

Weitere Bedingungen fügen Sie der Abfrage durch einen Klick auf „Neue Bedingung hinzufügen“ hinzu.

Wir klicken auf „Weiter“ und gelangen auf die Seite „Bedienerführung“.



Reiter „Bedienerführung“

Auf dieser Seite geht es darum, den gesamten Dialog „Firmenauswahl“ und die „Kundenanzeige“ zu formatieren und auszugestalten.

7.4.3.5.2
Seite 14

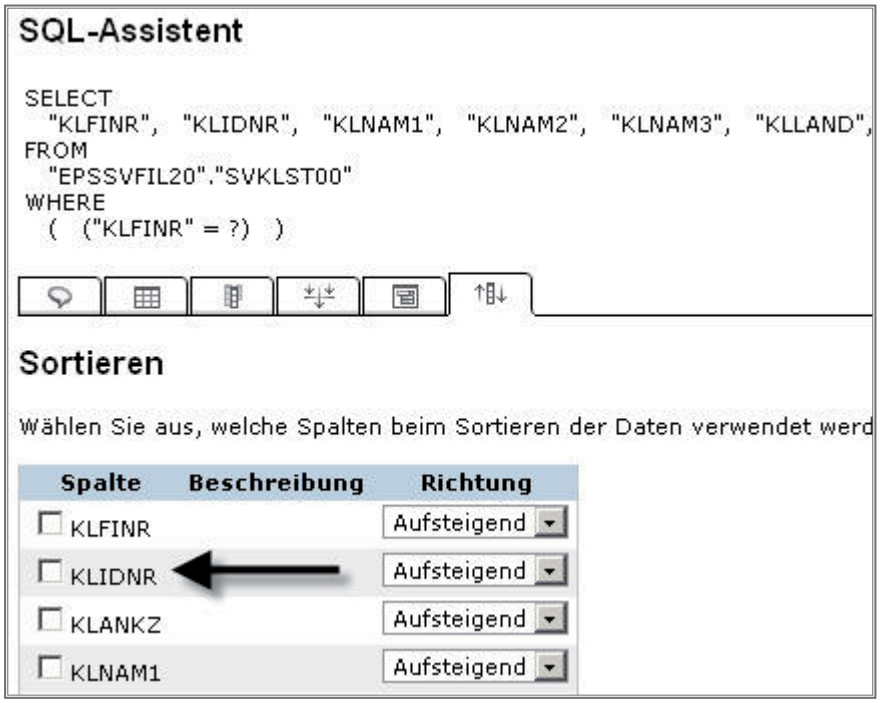
Die meisten Punkte sprechen für sich. Stellvertretend werde ich die Auswahl „Beschriftung“ bearbeiten.

Beschriftung

Für den Report wird der Beschriftungstext „Kunden und Lieferanten nach Firma anzeigen“ angegeben und anschließend auf „OK“ geklickt.

Hervorheben möchte ich an dieser Stelle die Option „Schablonendatei“. Damit haben Sie die Möglichkeit, den Report – bezogen auf das Layout – komplett Ihrer Firmen-CI anzupassen. Wenn in Ihrem Haus eine HTML-Datei als Vorlage für die Firmen-CI existiert, dann kann mit dieser Option die Schablonendatei als Formatierungsvorlage für die Abfrage herangezogen werden.

Im letzten Reiter wird außerdem die Sortierreihenfolge für die Datenausgabe beschrieben.



Reiter „Sortieren“

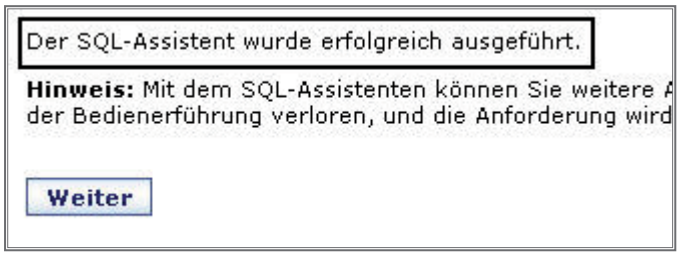
In der angezeigten Liste bestimmen Sie die Sortierfelder für diese Abfrage. Wir aktivieren die Checkbox für das Feld „KLIDNR“ (Identnummer).

Reihenfolge „Sortierfelder“

Mit Hilfe des Knopfs „Sortierreihenfolge“ lässt sich die Hierarchie der Sortierfelder festlegen.

Hierarchie der Sortierfelder

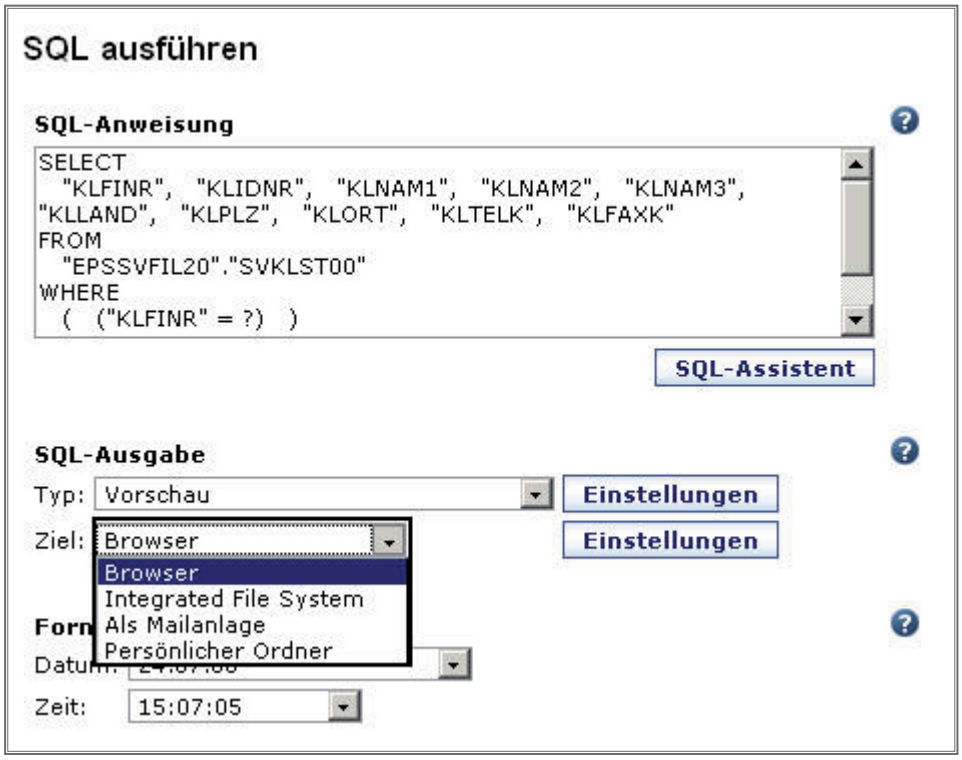
In meinem Beispiel wird lediglich das Feld „KLIDNR“ angezeigt, da ich auch keine weiteren Sortierfelder definiert habe. Durch Änderung der Folgenummer definieren Sie gegebenenfalls die Sortierhierarchie.



Ende-Meldung

Damit ist die Arbeit im SQL-Assistenten beendet; wir gelangen wieder auf die Seite „SQL ausführen“ zurück.

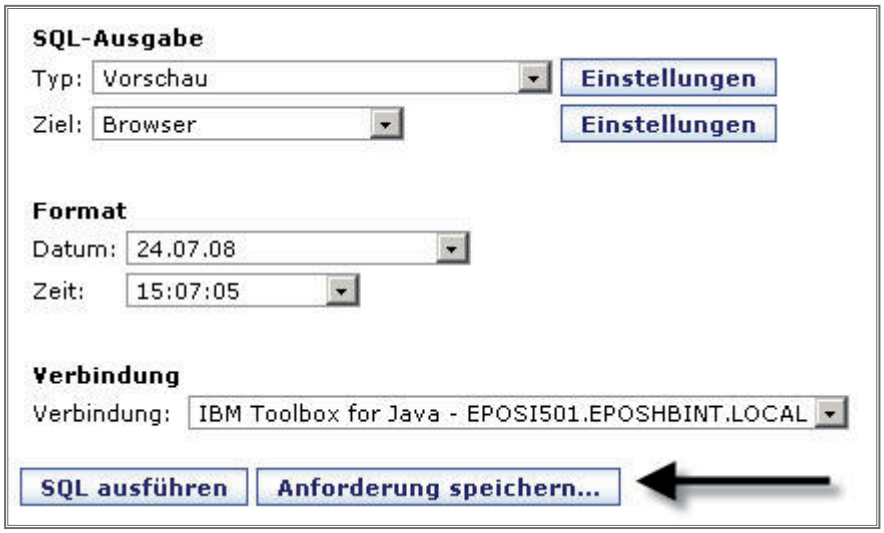
7.4.3.5.2
Seite 16



SQL ausführen

Sie sehen, dass der generierte SQL-Befehl in das Feld „SQL-Anweisung“ übernommen wurde.

Wir könnten jetzt diese Abfrage ausführen.



Abfrage speichern 1

Wir verzichten aber auf diese Option und beantragen stattdessen die Speicherung der Abfrage durch das Anklicken von „Anforderung speichern…“.

Abfrage speichern 2

Im Dialog „Anforderung speichern“ vergeben wir einen Namen für die Abfrage, hinterlegen einen beschreibenden Text und klicken auf „Anforderung speichern“.

Automatisch erscheint folgender Dialog:

Bestätigung der Speicheranforderung

Nachdem die Abfrage erfolgreich gespeichert wurde, erscheint der obige Bestätigungsdialog.

In diesem Dialog gibt es folgende Möglichkeiten:

– eine weitere, neue Abfrage erstellen,

– in den Bereich „Eigene Anforderungen“ wechseln, (zu diesem Thema erfahren Sie mehr im nächsten Abschnitt.)

– die gerade erstellte Abfrage ausführen.

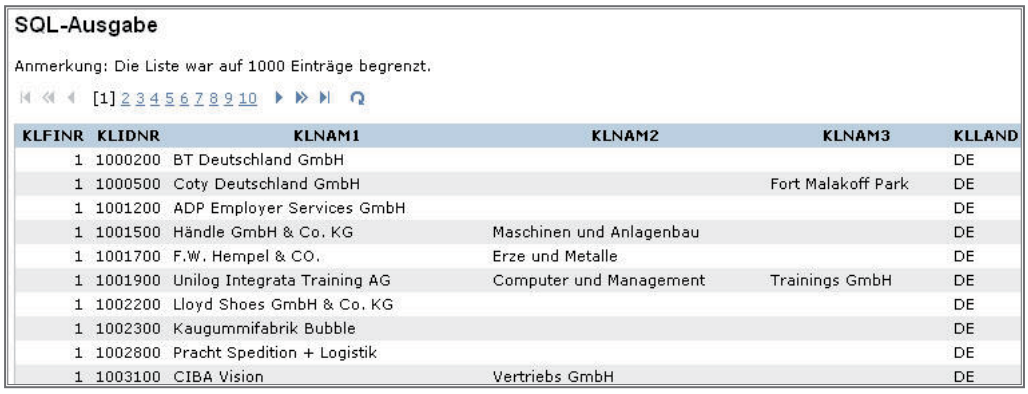
Wir wählen den letzten Link.

7.4.3.5.2
Seite 18

Firmenauswahl

Hier sehen Sie den von uns definierten Firmenauswahldialog. Am Inhalt der Combo-Box erkennen Sie, dass derzeit die Kunden zweier Firmen in der Datei „SVKLST00“ gehalten werden.

Wir wählen „Firma 1“ und klicken auf „OK“.



KLFINR	KLIDNR	KLNAM1	KLNAM2	KLNAM3	KLLAND
1	1000200	BT Deutschland GmbH			DE
1	1000500	Coty Deutschland GmbH		Fort Malakoff Park	DE
1	1001200	ADP Employer Services GmbH			DE
1	1001500	Händle GmbH & Co. KG	Maschinen und Anlagenbau		DE
1	1001700	F.W. Hempel & CO.	Erze und Metalle		DE
1	1001900	Unilog Integrata Training AG	Computer und Management	Trainings GmbH	DE
1	1002200	Lloyd Shoes GmbH & Co. KG			DE
1	1002300	Kaugummifabrik Bubble			DE
1	1002800	Pracht Spedition + Logistik			DE
1	1003100	CIBA Vision	Vertriebs GmbH		DE

Ergebnis der Abfrage

System i Access für Linux

Schon seit einer Reihe von Jahren wird Access für Linux als Download über die Seite www.ibm.com/eserver/iseries/access/linux/ angeboten.

Waren die ersten Versionen noch gekennzeichnet durch viele verbesserungswürdige Eigenschaften, so ist mittlerweile ein Connectivity Produkt entstanden, das robust ist und sicher und zuverlässig einen Zugang von Linux Desktop zum Power System ermöglicht.

Wichtig ist jedoch die Erkenntnis, dass Access für Linux nicht den Funktionsreichtum eines Access für Windows besitzt. Lediglich zwei benutzernahe Funktionen wurden implementiert:

– Eine 5250-Bildschirmemulation,
– in ODBC-Treiber, um auf die DB2 des Power Systems unter IBM i zuzugreifen

Darüberhinaus stehen noch Programmierer-APIs zur Verfügung:

– Extended Dynamic Remote SQL (EDRS)
– RMTCMD

7.5
Seite 2

Voraussetzungen

Auf dem Power System sind folgende Voraussetzungen zu schaffen.

– Das Benutzerprofil QUSER muss aktiviert (*enabled) sein.

– Die Hostserver müssen gestartet sein.

Unter Linux müssen folgende Packages installiert sein:

– GLibc 2.2

– RPM 3.0

– OpenMotif 2.0 oder später für die 5250 Emulation

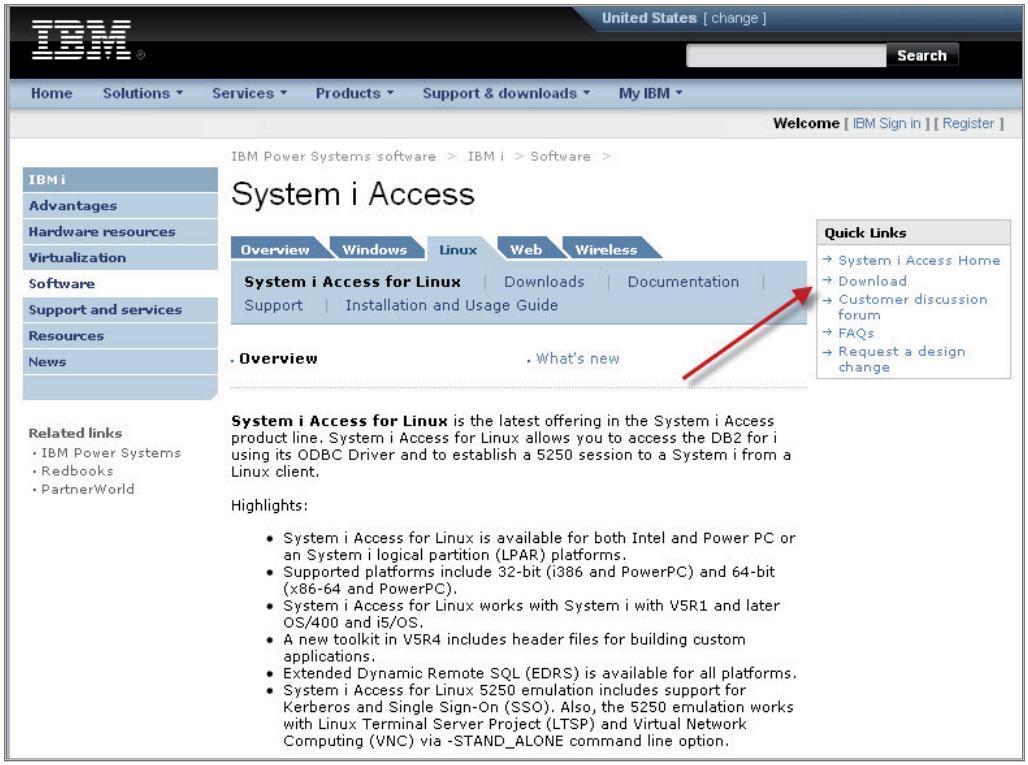
– unixODBC Treiber Manager Version 2.0.11 oder später

7.5.1 Seite 2

Download

Access für Linux wird nicht (!) mit den sonstigen System i Access Produkten ausgeliefert, sondern muss von der System i Access Homepage heruntergeladen werden.

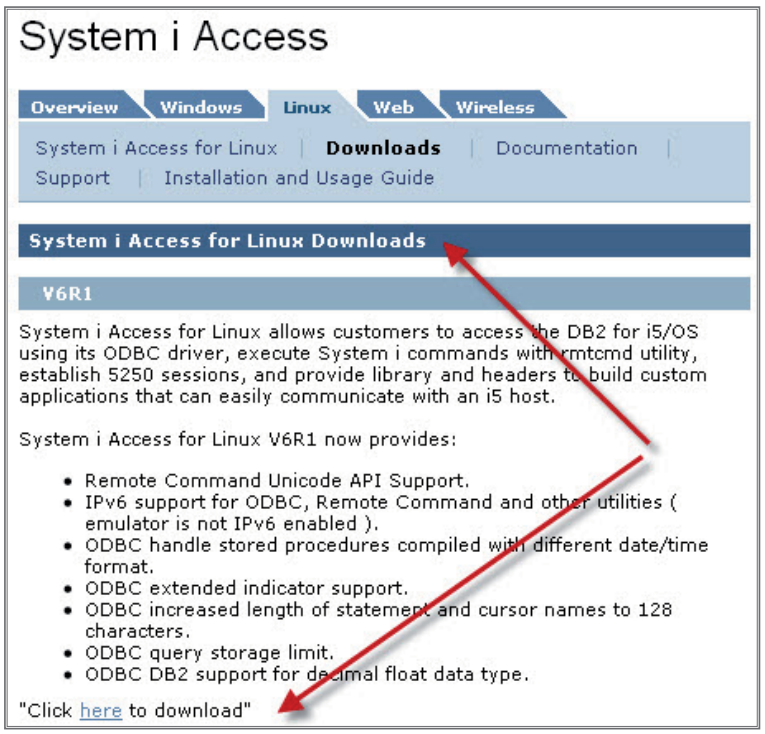
Die System iAccess for Linux Homepage finden Sie unter folgender URL: www.ibm.com/eserver/iseries/access/linux/



System i Access for Linux Homepage

7.5.2
Seite 2

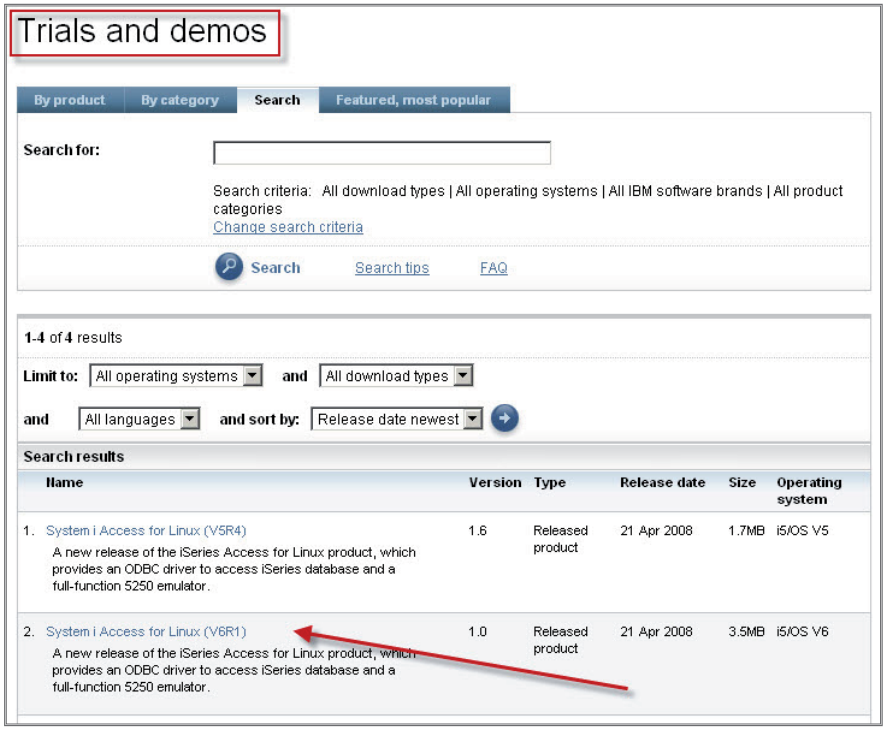
Folgen Sie dem Link Download.



Download Seite

Auf der Download Seite klicken Sie auf den oben bezeichneten Link.

Jetzt werden Sie zu den Download Auswahlmöglichkeiten weitergeleitet.



System i Access for Linux V6R1

Laden Sie die aktuellste Version von System i Access for Linux herunter. Diese Version ist abwärts kompatibel bis zu OS/400 V5R1.

Download bestätigen

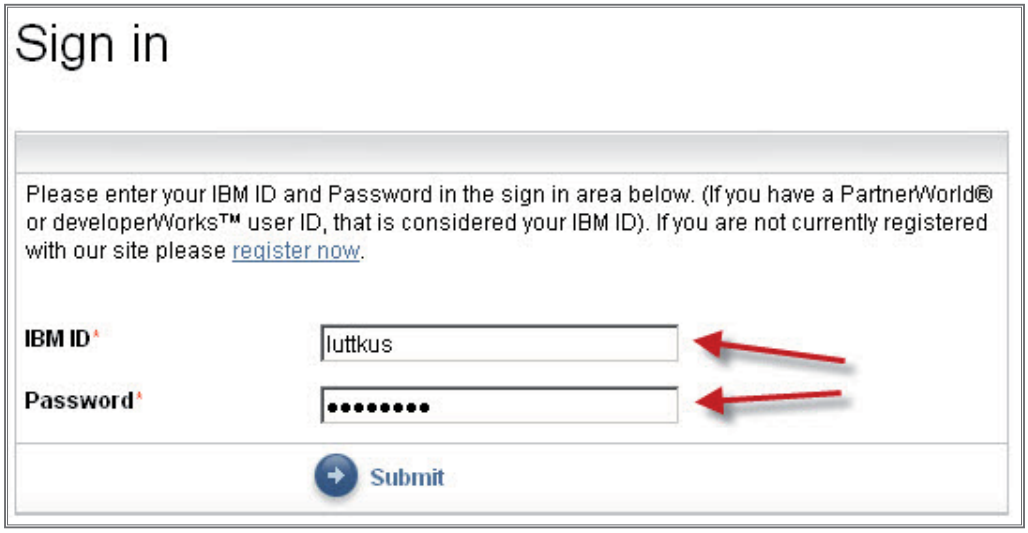
Bestätigen Sie den Download mit einem Click auf Continue.

Sie werden jetzt aufgefordert sich mit Ihrer IBM-ID anzumelden.

Anmeldung 1

7.5.2
Seite 4

Sollten Sie noch nicht über eine IBM-ID verfügen, so sollten Sie diese jetzt beantragen.



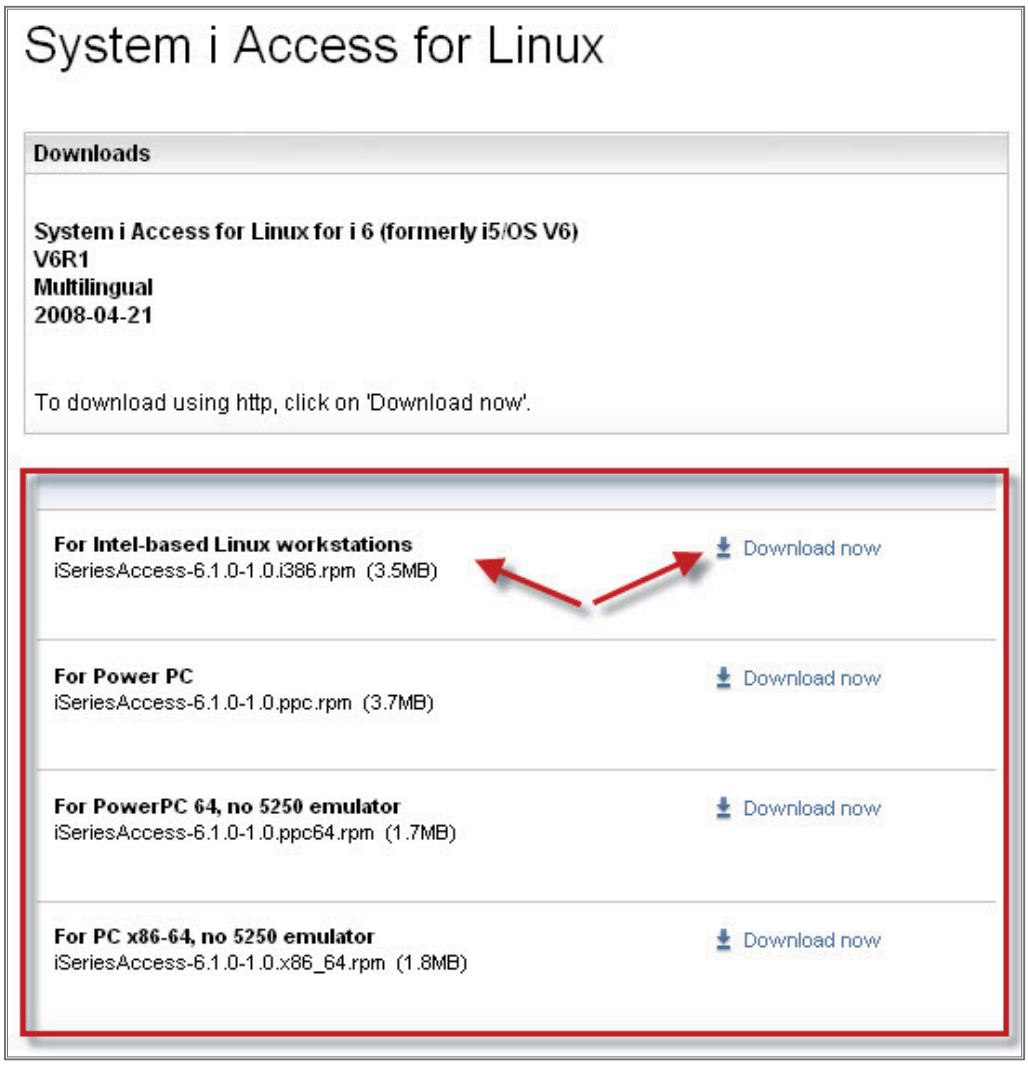
Anmeldung 2

Im oben gezeigten Dialog melden Sie sich dann mit Ihrer IBM-ID an. Sollten Sie noch nicht über eine IBM-ID verfügen, so beantragen Sie sie jetzt mittels des Links register now.

Wichtig

Die IBM-ID ist nicht nur eine der Voraussetzungen, um System i Access for Linux herunterzuladen, sondern versetzt Sie darüber hinaus in die Lage, PTFs und neue Programmversionen von System i Software und Lizenzschlüssel herunterzuladen.

Wählen Sie sodann die für Ihren PC passende Version des System i Acces für Linux herunter.



Intel basierte Workstations

Ich wähle an dieser Stelle For Intel-based Linux workstations und klicke auf Download now.

Ich speichere die Download-Datei iSeriesAccess-6.1.0-10.i386.rpm auf meinem PC in dem Verzeichnis /home/Luttkus/Software.

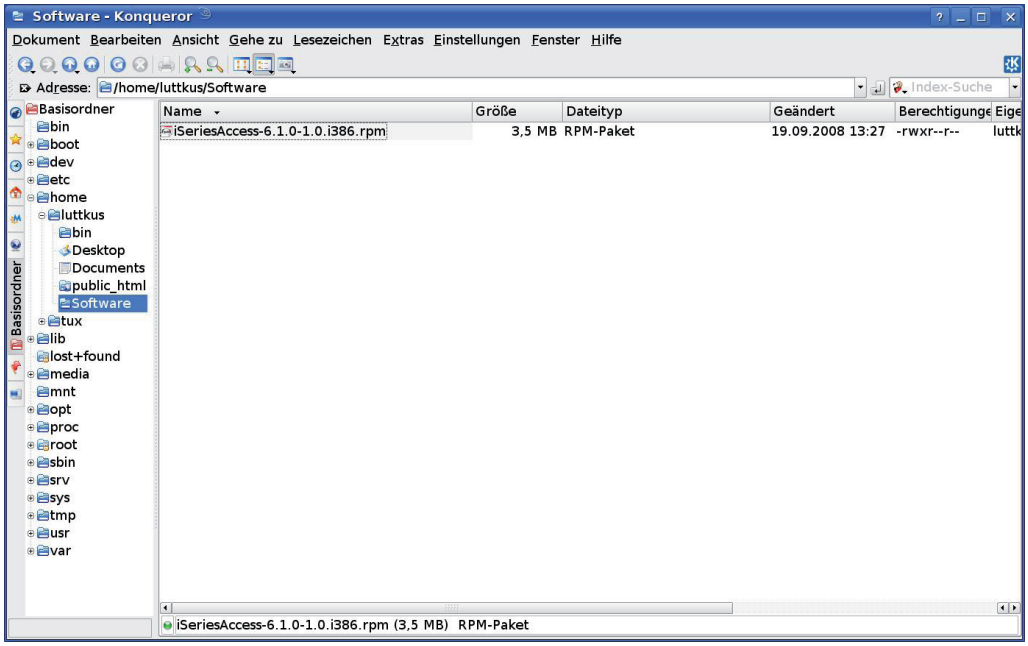
Achten Sie darauf, dass Sie auf dem von Ihnen gewählten Pfad die notwendigen Zugriffsrechte besitzen.

7.5.2
Seite 6

Installation

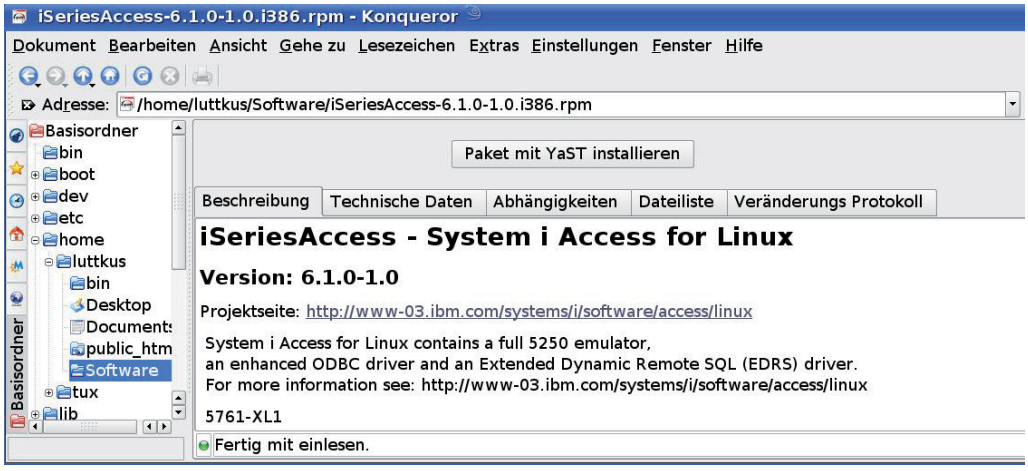
Jetzt beginnt die Installation von System i Access für Linux. Die in der Folge dargestellten Dialoge repräsentieren den Installationsvorgang unter openSuse 10.3 mit dem Desktop KDE 3.5.5.

Im Konqueror sehen Sie die von mir heruntergeladene Datei iSeriesAccess-6.1.0-10.i386.rpm im Pfad /home/Luttkus/Software.



Die Installationsdatei iSeriesAccess-6.1.0-10.i386.rpm

Ein Doppelklick auf die gezeigte Datei öffnet das RPM-Package.



Beginn der Installation

7.5.3
Seite 2

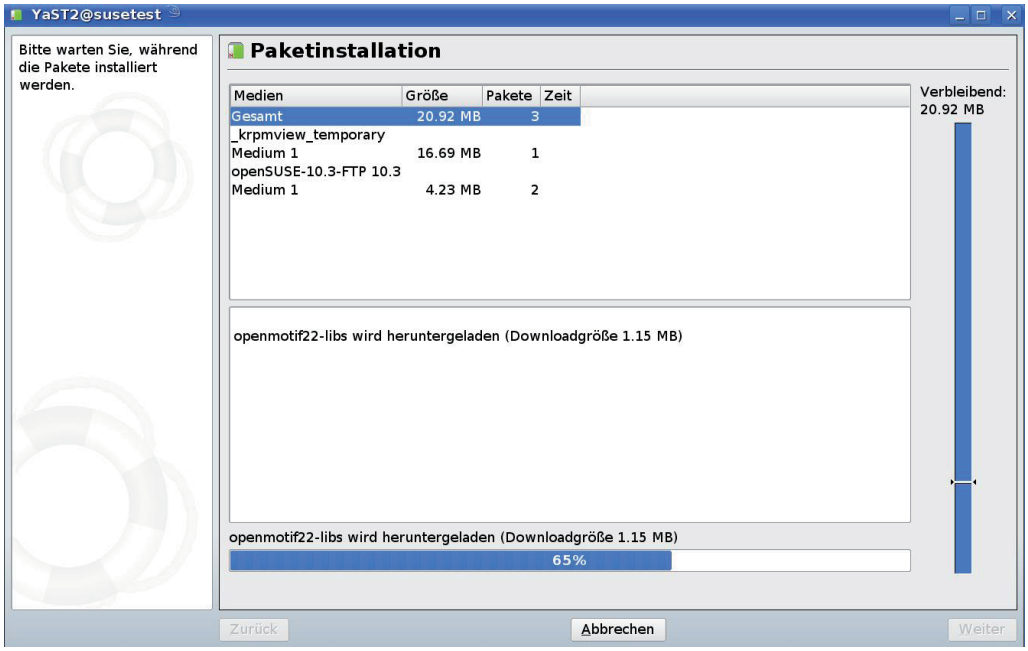
Mit einem Klick auf Paket mit YaST installieren startet die eigentliche Installation.

Besteht eine Verbindung zum Internet, so wird nach Updates für das zu installierende Programm gesucht.

Suche nach Updates

Sollte es bei der Suche nach Updates zu Fehlern kommen, so klicken Sie diese einfach weg.

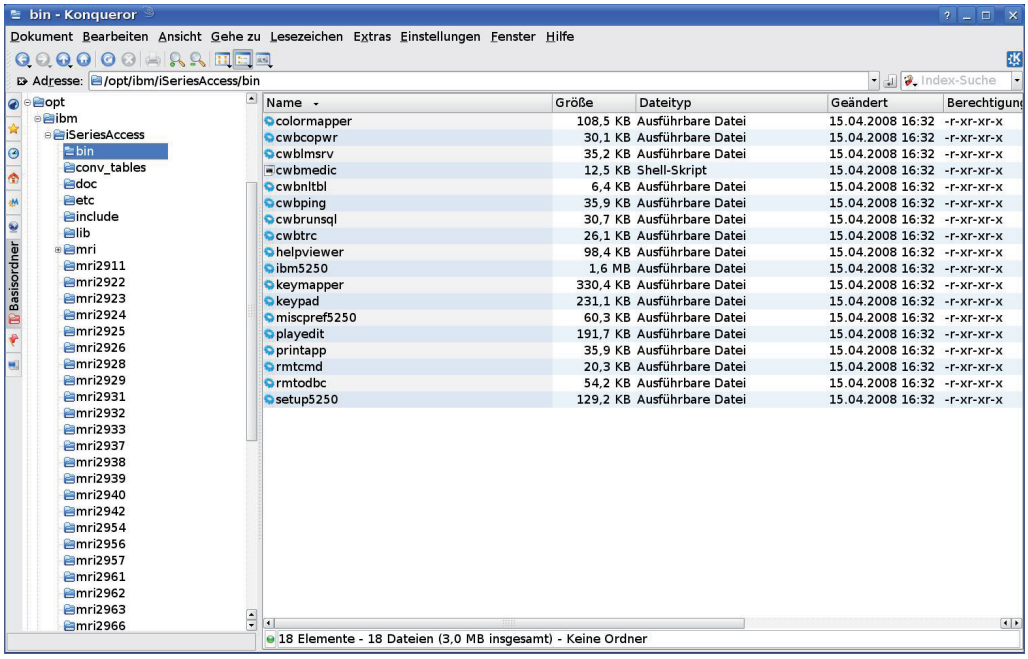
Danach wird der Fortschritt der Installation in folgendem Dialog angezeigt.



Installationsfortschritt

Die eigentliche Installation sollte nur wenige Minuten bis zur Fertigstellung benötigen.

Nach dem Ende der Installation stehen dann alle aufrufbaren Funktionen im Pfad /opt/ibm/iSeriesAccess/bin zur Verfügung.



Installationsverzeichnis

Für uns wichtige Dateien sind:

- Setup5250: Zentrales Konfigurationsprogramm für 5250 – Bildschirmverbindungen zum Power System.
- Ibm5250: Die 5250-Bildschirmemulation
- Rmtcmd: Aufruf von i/OS-Befehlen aus dem Linux – Client heraus
- Cwbping: Prüfprogramm, das den Status der Hostserver zurückmeldet

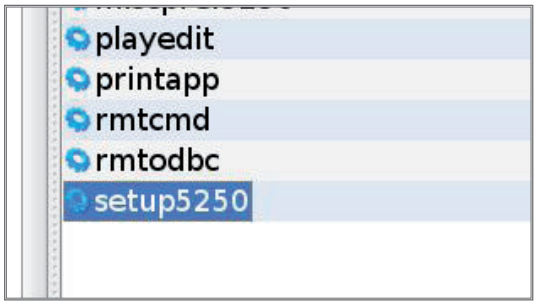
7.5.3
Seite 4

5250 Emulation

Die zentrale Anwendung des Access für Linux ist mit Sicherheit die Bildschirmemulation. Diese kann auf verschiedene Arten aufgerufen werden.

Einmal indem Sie im Pfad /opt/ibm/iSeriesAccess das Programm ibm5250 aufrufen. Diesen Ansatz werde ich hier jedoch nicht verfolgen, da auf diese Weise eine Konfiguration des Bildschirms hauptsächlich mittels Befehlszeilen-Parametern realisiert werden kann und jede Bildschirmsitzung für sich alleine konfiguriert werden muss.

Ich hinterlege lieber meine Konfigurationsdaten zentral mit Hilfe des Konfigurationsprogramms setup5250, dass sich in dem gleichen Ordner befindet wie das Programm ibm5250.



Konfigurationsprogramm setup5250

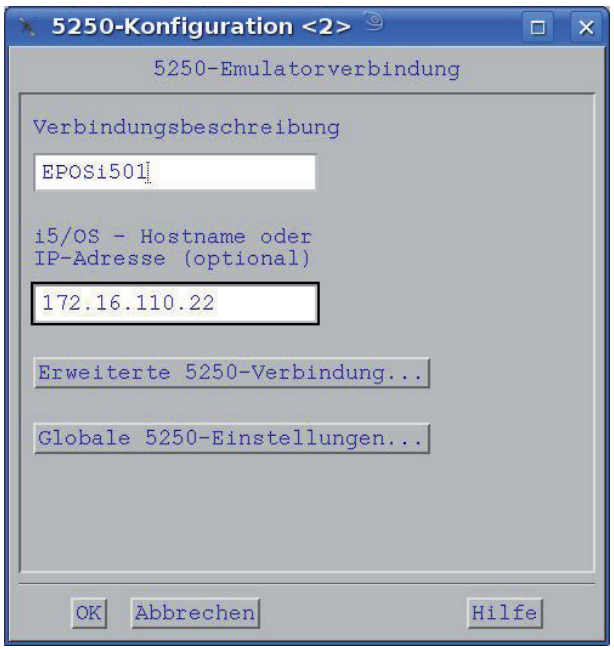
Klicken Sie auf setup5250, so öffnet sich das folgende Fenster.

Beginn der 5250 Konfiguration

Eine neue Bildschirm-Konfiguration beginnen Sie, indem Sie das Menü Verbindung ➜ Neu öffnen.

7.5.4
Seite 2

Jetzt öffnet sich der Dialog 5250-Konfiguration<2>



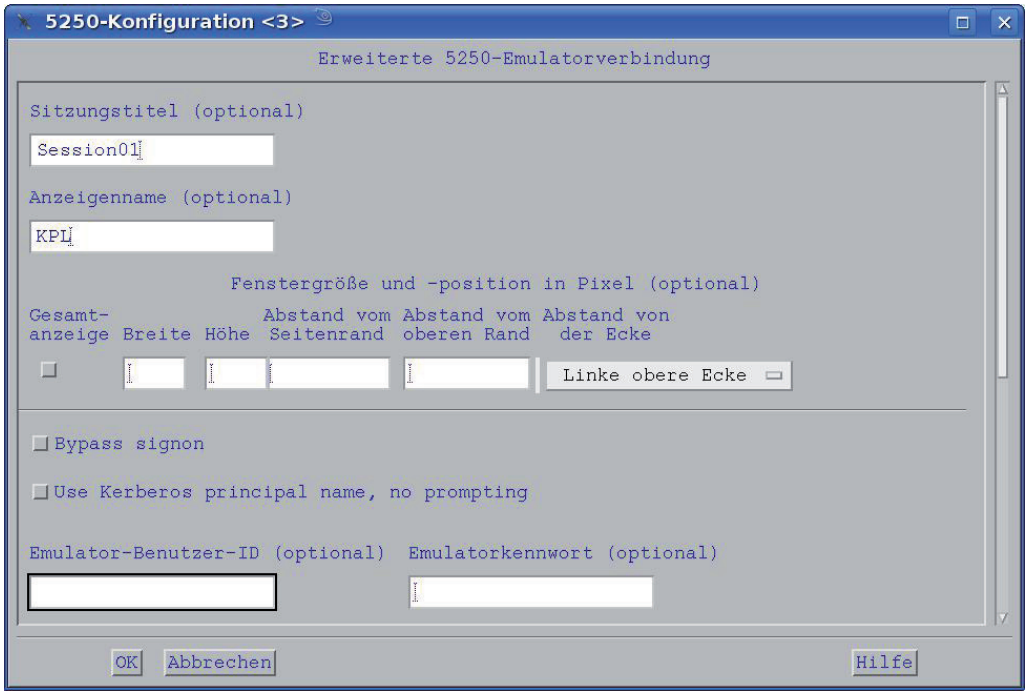
Verbindungskonfiguration

Folgende Parameter müssen versorgt werden:

Verbindungsbeschreibung: Hier tragen Sie einen beliebigen Namen ein. In dem Fall, dass Sie hier den Hostnamen Ihres Power Systems eintragen, kann die Versorgung des folgenden Parameters entfallen. Ich verwende hier den Namen EPOSI501.

IP-Adresse (optional): Falls Sie im vorherigen Parameter nicht den Hostnamen Ihres Systems eingetragen haben, tragen Sie hier die IP-Adresse Ihres Power Systems ein. Ich trage hier die Adresse 172.16.110.22 ein.

Jetzt klicke ich auf den Knopf Erweiterte 5250-Emulatorverbindung. Dieser Arbeitsschritt ist nicht zwingend notwendig.



Erweiterte 5250-Emulatorverbindung

Ich trage hier einen Sitzungstitel ein – ein beliebiger, für mich sinnvoller Name.

Darüberhinaus trage ich hier im Parameter Anzeigenname den Namen der zu generierenden Bildschirmeinheit ein. Sie können hier durchaus mit dem Platzhalter * arbeiten um teilgenerische Namen zu bilden.

Mit Bypass signon wird die gegebenenfalls in diesem Dialog hinterlegte Emulator-Benutzer-ID an die Bildschirmanmeldung übergeben und der Anmeldungsdialog für die 5250-Sitzung entfällt.

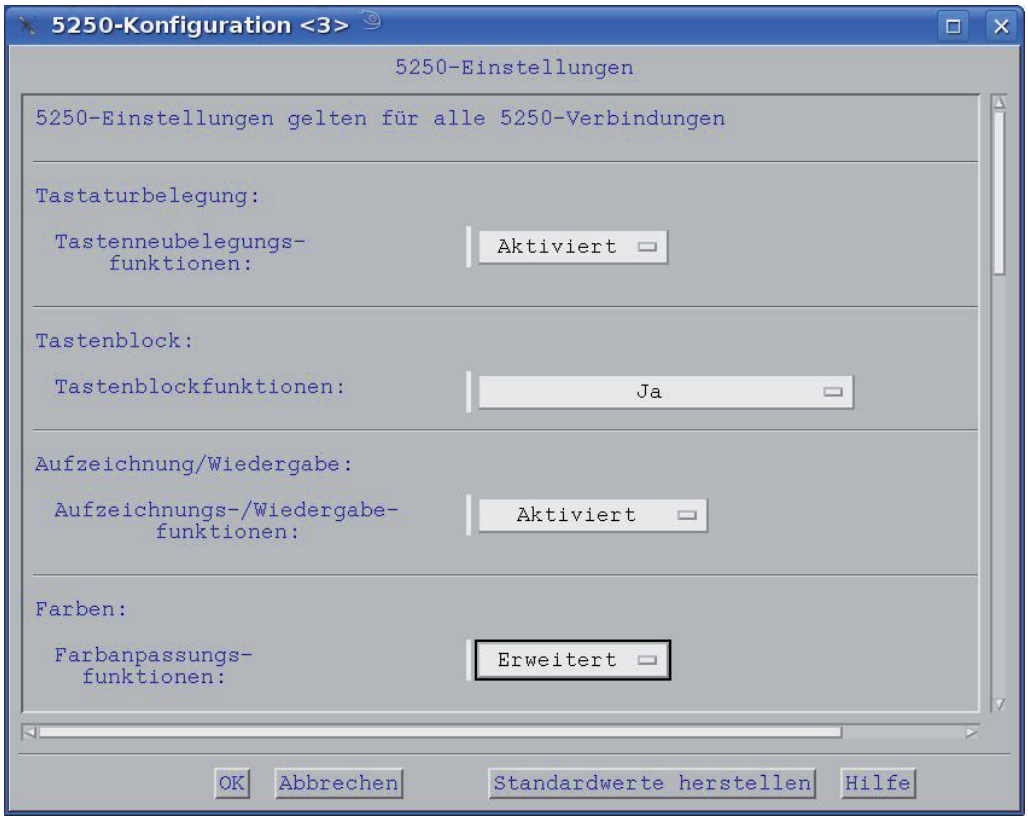
Arbeiten Sie mit Single Signon (SSO)-Verfahren, die auf Kerberos basieren, so kann der Bildschirm eine Fremdauthentifizierung gegebenenfalls übernehmen.

Darüberhinaus kann die Anfangsgröße des Bildschirmanzeige-Fensters in der Maßeinheit Pixel definiert werden.

Ich schließe dieses Fenster.

7.5.4
Seite 4

Ein Klick auf den Knopf Globale 5250-Einstellungen öffnet das nachfolgend gezeigte Fenster.



Globale 5250-Einstellungen

In diesem Dialog werden dem Anwender erweiterte Funktionen der Bildschirmemulation zur Verfügung gestellt.

Tastaturbelegung: Soll der Anwender in der Lage sein seine Tastaturbelegung zu verändern.

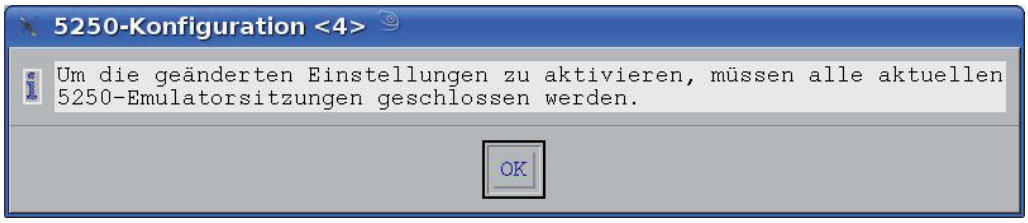
Tastenblock: Soll der Anwender einen Tastenblock verwenden können, mit dem er Makros abspielen kann, die er zuvor mit der Makrofunktion aufgenommen hat.

Aufzeichnung/Wiedergabe: Soll der Anwender Makros aufnehmen und abspielen können.

Farben: Soll der Anwender die Farbgebung individuell gestalten können.

Ich belasse es hier bei allen Standardvorgaben.

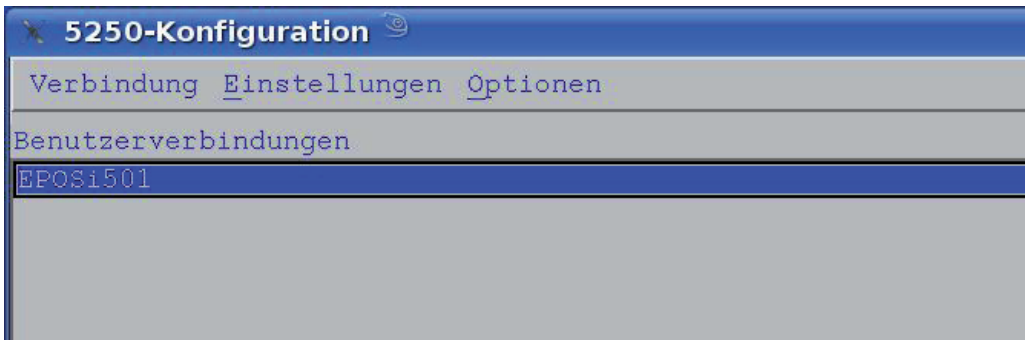
Ich verlasse jetzt das Verbindungskonfigurations-Programm und werde aufgefordert bereits geöffnete 5250-Sitzungen zu beenden.



Beendigungsdialog

Eine bereits geöffnete Bildschirmsitzung würde die gezeigten Konfigurationen nicht übernehmen. Erst ein Neustart der 5250-Emulation würde die Aktivierung der Konfiguration bewirken.

Jetzt sehen Sie den Hauptdialog des Programmes setup5250.

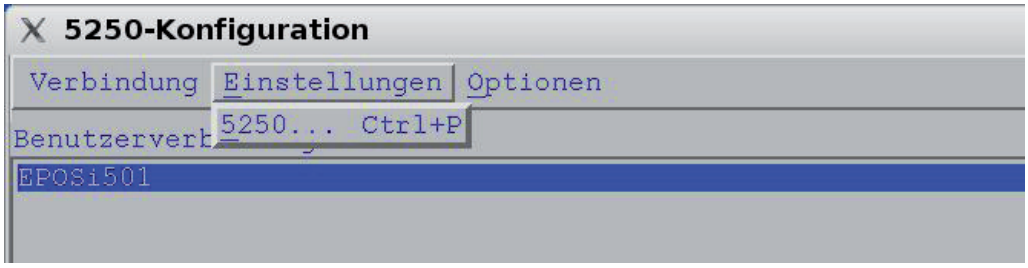


Die erste 5250-Verbindung wurde konfiguriert.

Sie sehen die konfigurierte Verbindung EPOSI501

Die bisher getätigten Konfigurationen beziehen sich auf die Verbindung EPOSI501. Es lassen sich darüberhinaus ebenfalls verbindungsunspezifisch Konfigurationsdaten im set5250-Programm hinterlegen.

Dazu öffnen Sie das Menü Einstellungen ➜ 5250.

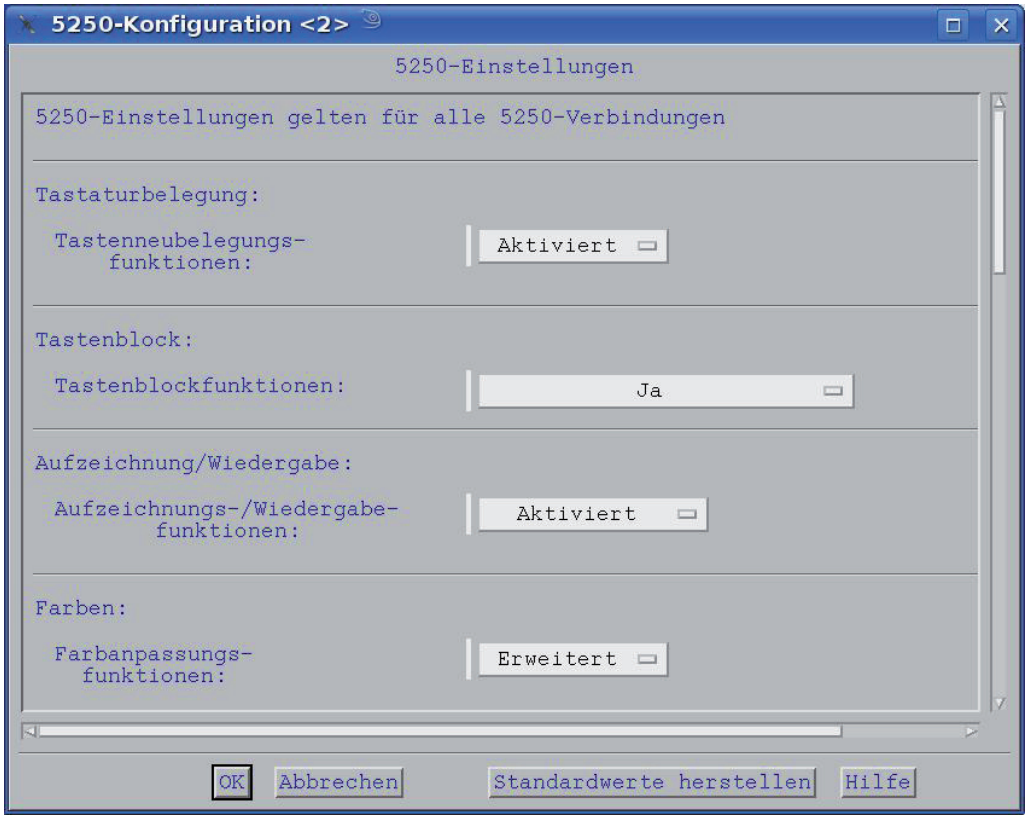


Globale Einstellungen

7.5.4
Seite 6

Nachdem sie das genannte Menü geöffnet haben, erscheint folgender Dialog.

Diesen Dialog kennen Sie bereits.



Globale 5250-Einstellungen

Tastaturbelegung: Soll der Anwender in der Lage sein seine Tastaturbelegung zu verändern.

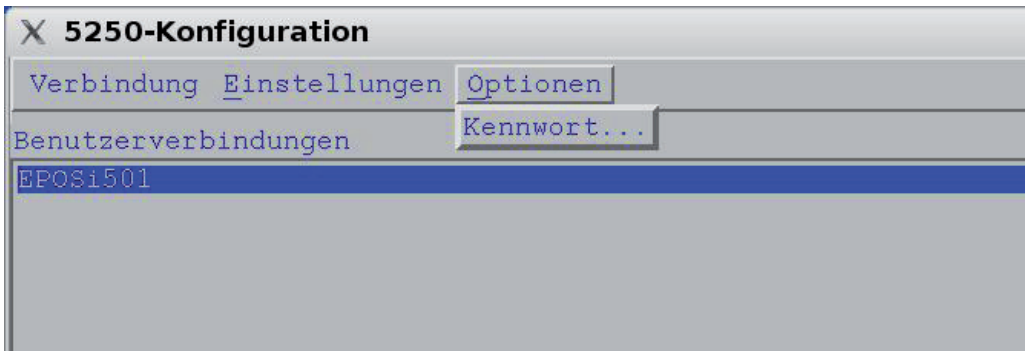
Tastenblock: Soll der Anwender einen Tastenblock verwenden können, mit dem er Makros abspielen kann, die er zuvor mit der Makrofunktion aufgenommen hat.

Aufzeichnung/Wiedergabe: Soll der Anwender Makros aufnehmen und abspielen können.

Farben: Soll der Anwender die Farbgebung individuell gestalten können.

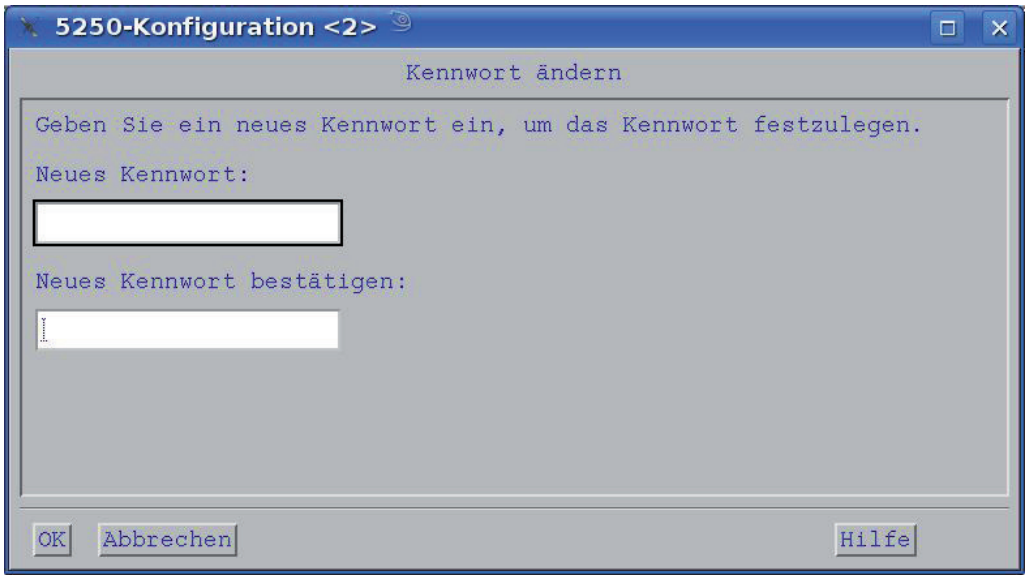
Der Unterschied zu dem bereits weiter oben gezeigtem Dialog besteht darin, dass diese Einstellungen auf alle Verbindungen, die in setup5250 konfiguriert wurden, wirken.

Ich öffne jetzt das Menü Optionen ➜ Kennwort.



Benutzerdaten einpflegen

Im folgenden Fenster konfigurieren Sie ein Standardkennwort,

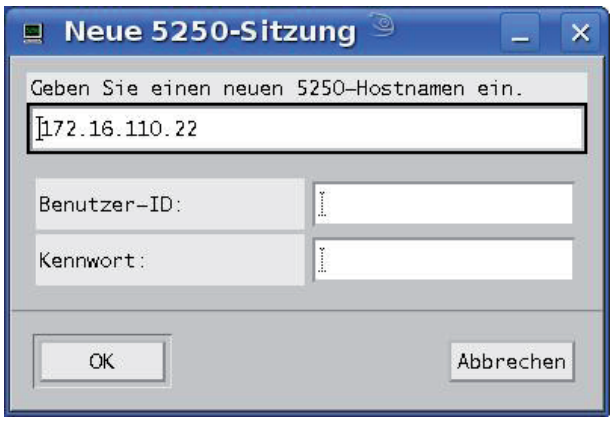


Standardkennwort

welches für alle Verbindungen wirksam wird. Eine derartige Konfiguration geht mit Sicherheit davon aus, dass die Verbindung zum Power System durch einen Standard-benutzer hergestellt wird, dem hier ein Standardkennwort zugeordnet werden kann.

7.5.4
Seite 8

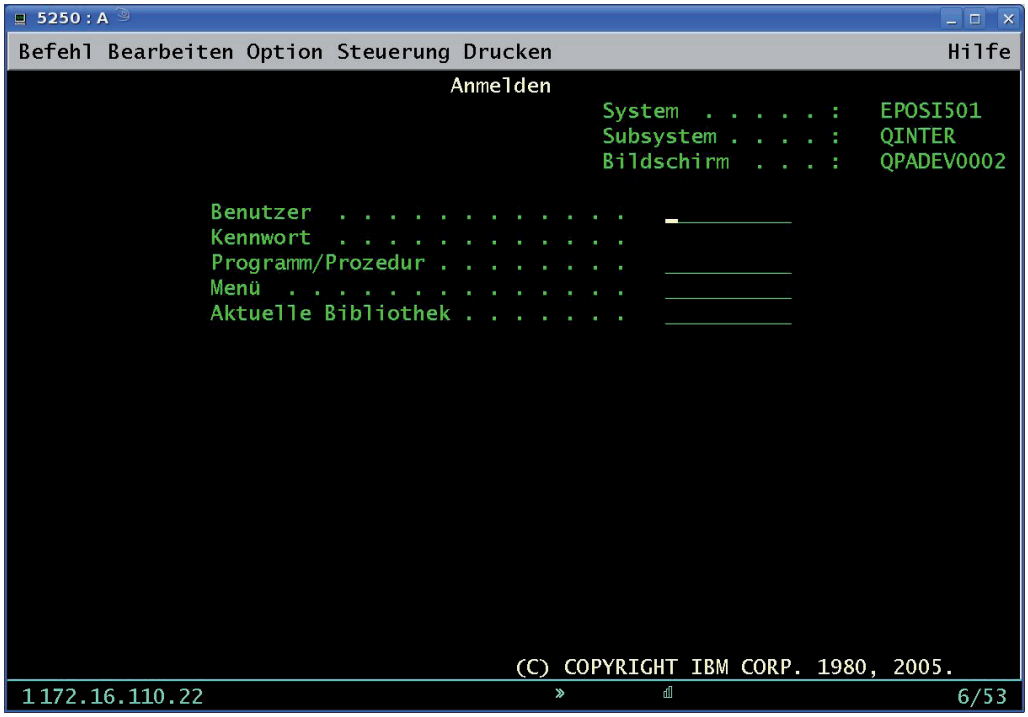
Ein Klick auf die Verbindung öffnet jetzt ein Anmeldungsfenster.



Systemanmeldung

Dieser Dialog erscheint lediglich, wenn Sie weder für die Verbindung noch global einen Benutzer/Kennwort definiert haben.

Jetzt erscheint das typische Bild einer 5250-Emulation.



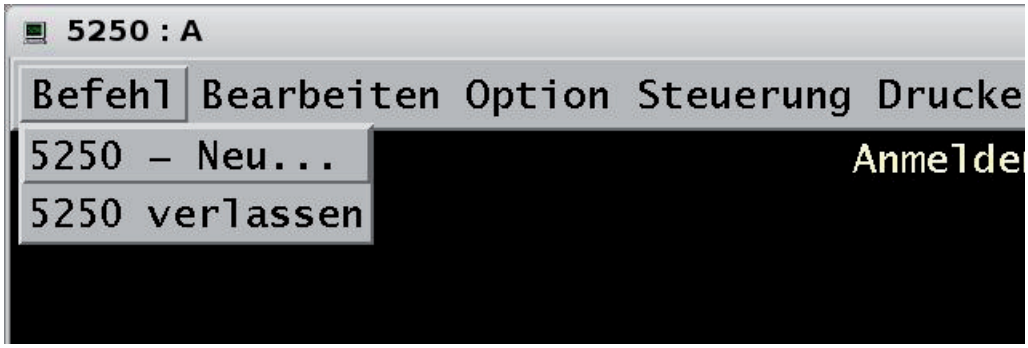
5250-Emulation

Über diese Bildschirmemulation kann man lediglich sagen, dass sie vom Layout her dem gewohnten Schema von System i Access Emulationen folgt.

Ich gehe davon aus, dass Sie bereits mit 5250-Emulationen gearbeitet haben, bzw. noch arbeiten, so dass meines Erachtens eine Bedienungsanleitung für Bildschirme hier entfallen kann.

Die Konfiguration von Emulationsfunktionen und die Steuerung der Emulation als solches werden komplett über die Menüleiste realisiert.

Ich öffne das Menü Befehl.



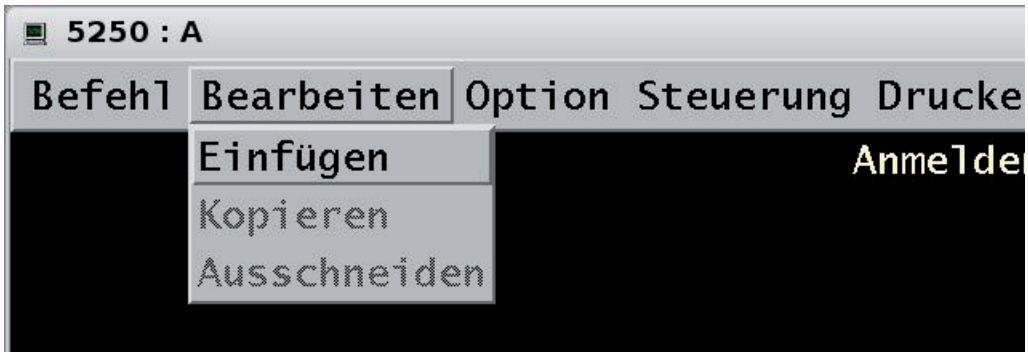
Menü Befehl

Folgende Funktionen lassen sich hier auslösen:

– 5250-Neu: Diese Funktion erstellt eine neue Bildschirmsitzung
– 5250 verlassen: Beenden Sie mit Hilfe dieses Menüpunktes die Bildschirmemulation.

  Ein Klick auf das x-Symbol in der rechten oberen Ecke beendet auch die Bildschirmsitzung, benachrichtigt i/OS jedoch nicht von diesem Vorgang, so dass eine erneute Verbindungsaufnahme via 5250-Sitzung zu Fehlern führen kann.

Ich wechsele in das Menü Bearbeiten.



Menü Bearbeiten

In diesem Menü finden Sie typische Paste-und-Copy-Funktionen von grafischen Benutzeroberflächen.

Kopieren: Markierten Bereich ins Clipboard stellen

Einfügen: Inhalt des Clipboards an der Cursorposition einfügen

Ausschneiden: Inhalt des Clipboards an der Cursorposition einfügen und den Quellbereich löschen

7.5.4
Seite 10

Ich wechsele in das Menü Option.

Menü Option

Schriftarten: In diesem Menüpunkt bestimmen Sie welche Schriftart und welche Schriftgröße die Emulation verwenden soll. Je nach Größe und Auflösung des KDE-Desktops kann die Emulationsanzeige verbessert werden.

Experimentieren Sie mit diesen Einstellungen, um eine optimale Emulationsanzeige einzustellen.

Die gezeigten Hardcopies stammen von einem 24-Zoll Bildschirm mit einer Auflösung von 1920 * 1200 Pixeln. Die Verwendung der Zeichengröße 15 * 28 brachte hier, zumindest was meine Wahrnehmung betrifft, ein gutes Ergebnis.

Die Option Tastenblock zeigt ein Bedienfeld an, mit dessen Hilfe wir Makros, die wir in einer Bildschirmsitzung aufgenommen haben, abrufen können.

Mittels der Option Tastenblock-Apassung kann ein bestehender Tastenblock geändert und den eigenen Bedürfnissen angepasst werden.

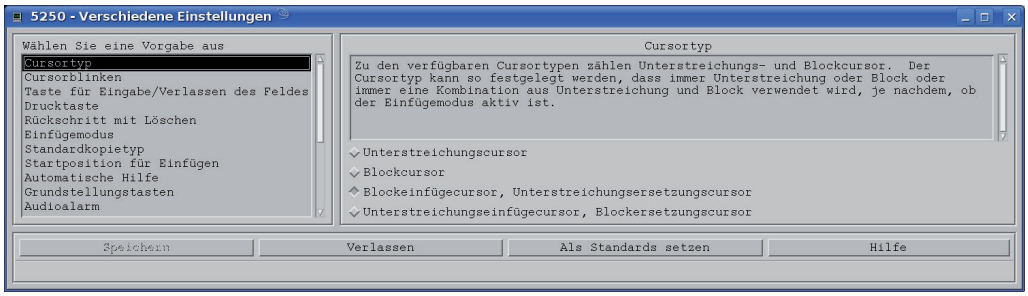
Tastenblock-Anpassung

Ist jedoch bisher kein Tastenblock erstellt worden, so können Sie durch Klick auf den Knopf Neuer Tastenblock einen neuen Tastenblock erzeugen.

Die Menüauswahl Aufzeichnung ermöglicht das Aufnehmen von Makros.

Die Menüauswahl Wiedergabe ermöglicht das direkte Abspielen von Makros. Als Alternative zu diesem Punkt und um eine bessere Bedienbarkeit zu erreichen, erstellen Sie für die Ausführung von Markos einen Tastenblock.

Mit Hilfe der Option verschiedene Einstellungen konfigurieren Sie dann die Eigenschaften der eigentlichen Emulation.



Verschiedene Einstellungen

Der Bedienung dieses Dialogs basiert für alle Einstellungspunkte auf dem gleichen Schema: Wählen Sie links die zu bearbeitende Einstellung aus und wählen Sie sodann auf der rechten Seite den zu modifizierenden Parameter.

Da die meisten Einstellungsparameter ihre Wirkungsweise selbst beschreiben, erwähne ich hier nur einige Punkte.

Cursortyp: Block oder Unterstreichungscursor?

Cursorblinken: ja/nein

Taste für Eingabe…: Löste Enter oder Strg-rechts die Datenfreigabe aus?

Drucktaste: Mit welcher Taste erzeugen Sie eine Hardcopy?

Rückschritt mit Löschen: Soll die Rückschritttaste Feldinhalte löschen?

Einfügemodus: Beginnt Ihre Bildschirmsitzung mit dem Einfügecursor oder nicht?

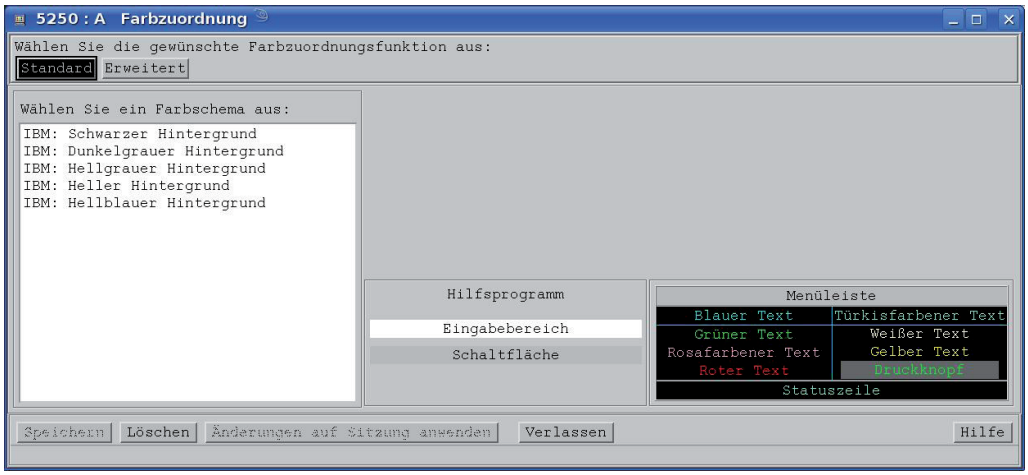
Standardkopietyp: Erstellt eine hardcopy vom Bildschirm eine grafische oder eine textuelle Ausgabedatei?

Grundstellungstasten: Welche Taste bewirkt die Rücksetzung der Tastatur?

Etc.

7.5.4
Seite 12

Mit Hilfe des Menüpunktes Farbzuordnung ändern Sie das Farbschema Ihres Emulations-Bildschirms.

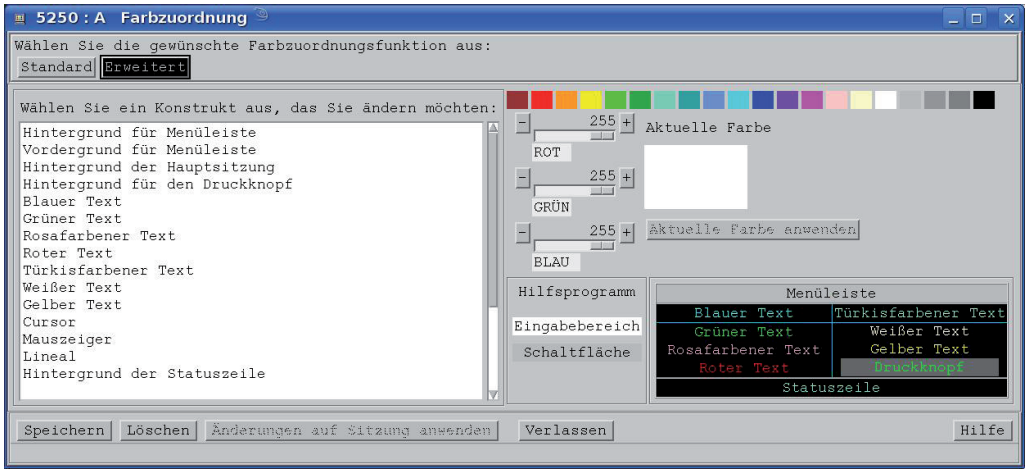


Farbzuordnung – Standard

Mit Hilfe des Knopfes Standard können Sie bereits vordefinierte Farbschemata anwenden.

Klicken Sie einfach auf eines der angebotenen Farbschemata und drücken Sie sodann den Knopf Änderung auf Sitzung anwenden.

Gefallen Ihnen die angebotenen Schemata nicht, so lassen sich die Bildschirmfarben Ihrer Emulation auf differenziert einstellen.



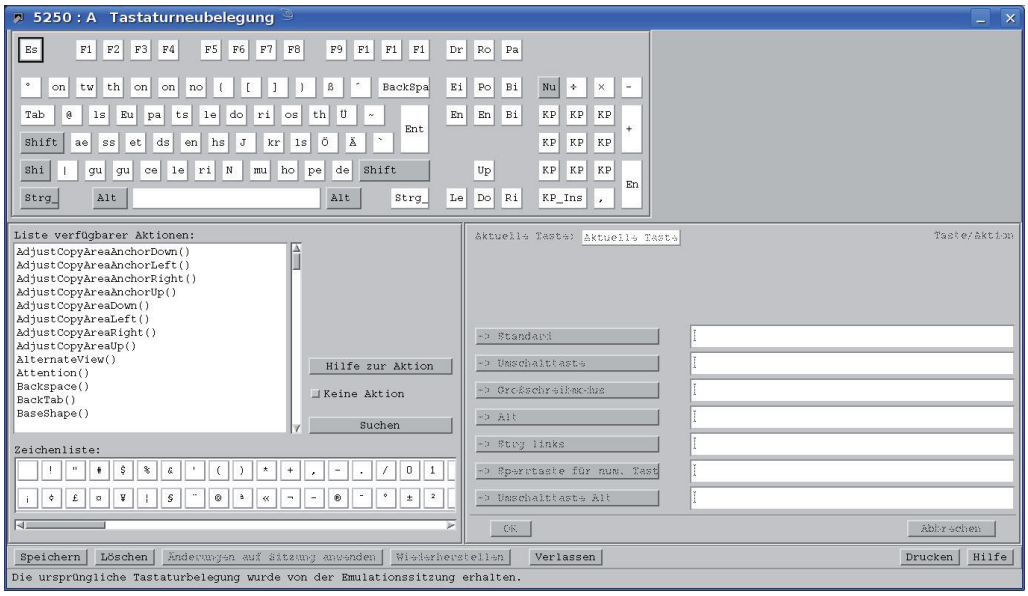
Farbzuordnung – Erweitert

Klicken Sie einfach auf den Knopf Erweitert.

Wählen Sie jetzt links die zu bearbeitende Komponente aus und markieren Sie rechts die gewünschte Farbe.

Drücken Sie dann den Knopf Änderung auf Sitzung anwenden.

Mittels der Option Tastaturbelegung haben Sie die Möglichkeit die Tastatur der Emulation komplett Ihren Wünschen anzupassen.



Tastaturneubelegung

Wählen Sie links in der Liste der verfügbaren Aktionen diejenige aus, die Sie modifizieren möchten.

Markieren Sie dann oben die Taste, die diese Aktion auslösen soll.

Klicken Sie dann rechts auf den mit ➜ bezeichneten Tasten, welche Taste zusätzlich gedrückt sein muss um die gewählte Aktion auszulösen.

- Standard: die markierte Taste allein löst die Aktion aus.
- Umschalttaste: die markierte Taste plus Umschalttaste löst die Aktion aus.

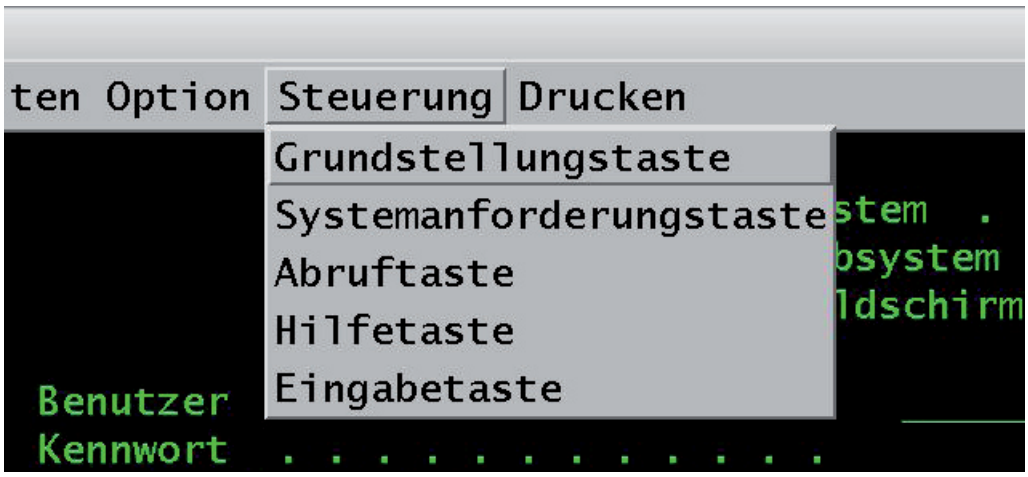
Weitere Belegungsoptionen sind möglich:

- Großschreibmodus
- Alt
- Strg-links
- Umschalt-Alt
- Sperrtaste für numerischen Tastenblock

Daraus ergibt sich insgesamt die Möglichkeit eine Taste siebenfach zu belegen.

7.5.4
Seite 14

Ich öffne das Menü Steuerung.

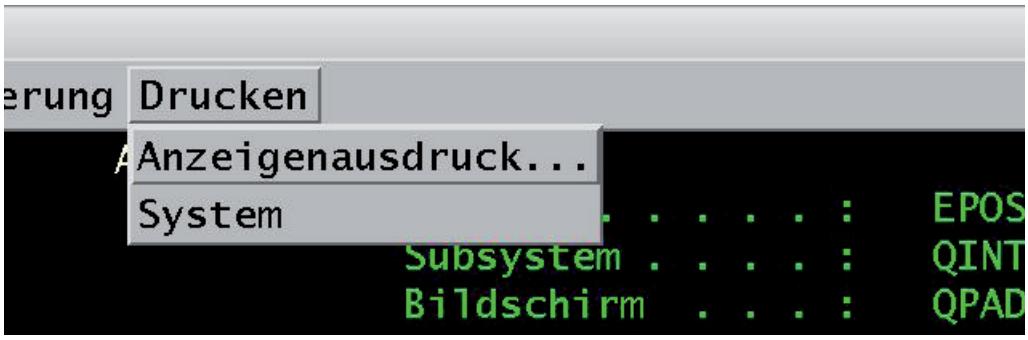


Menü Steuerung

Mit Hilfe dieses Menüs lassen sich folgende Sondertasten auslösen:
– Grundstellungstaste
– Systemanforderungstaste
– Abruftaste
– Hilfetaste
– Eingabetaste

Diese Funktionen liegen alternativ auf Funktionstasten und müssen nicht zwingend notwendig mit Hilfe des gezeigten Menüs ausgelöst werden.

Ich wechsele in das Menü Drucken.



Menü Drucken

Dieses Menü dient zum Ausdrucken vorn Hardcopies.

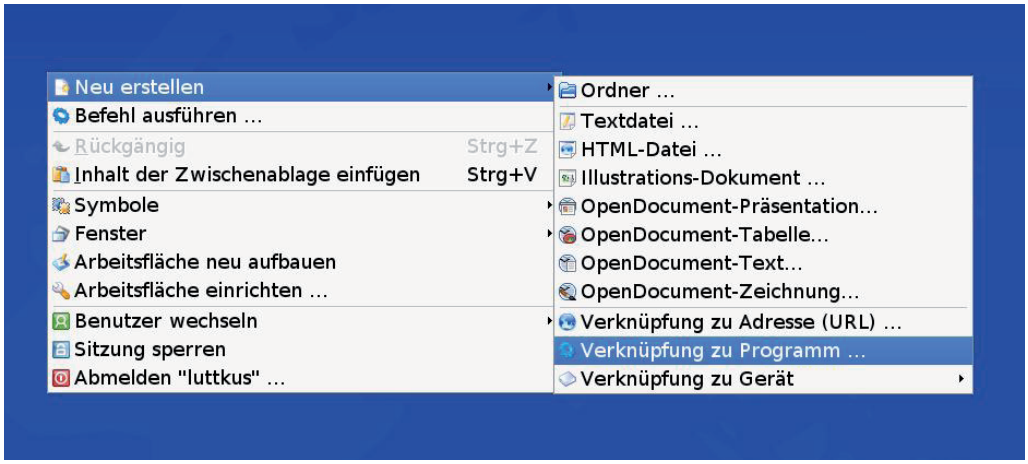
Anzeigendruck: Hardcopy wird über den Standarddrucker des Clients ausgegeben.

Hostdruck: Hardcopy wird mit einem Systemdrucker des i/OS ausgegeben.

Verknüpfung erstellen

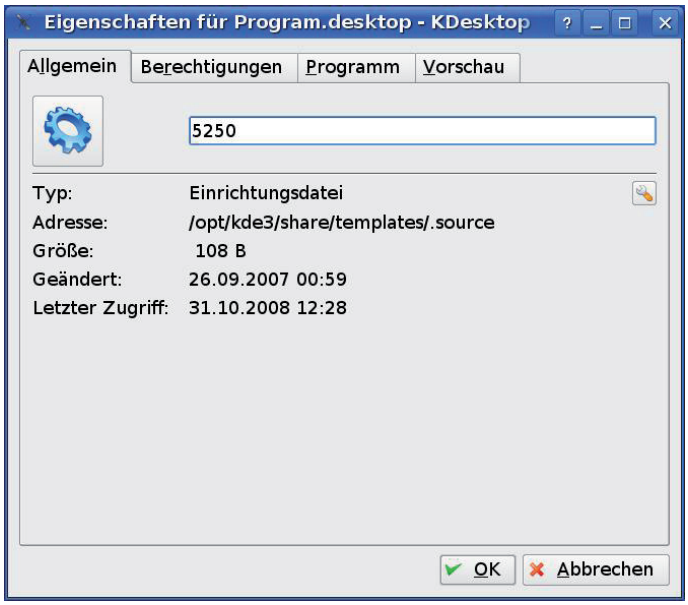
Nachdem ich meine 5250-Emulation konfiguriert habe, werde ich jetzt auf meinem KDE-Desktop eine Verknüpfung erstellen um Bildschirme leicht aufrufen zu können.

Mittels Rechtsklick in den Hintergrund meines Desktops öffne ich das Kontextmenü Neu erstellen ➜ Verknüpfung zu Programm.



Desktop-Verknüpfung erstellen

Jetzt öffnet sich folgendes Fenster.



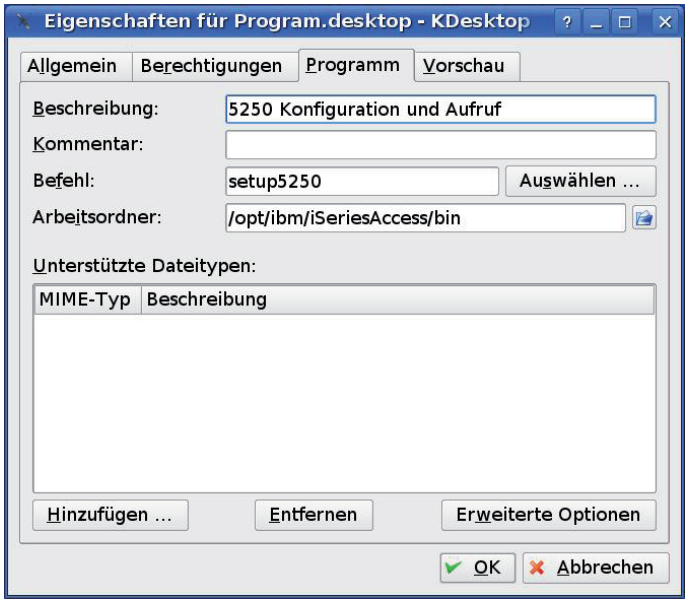
Eigenschaften

In das leere Namensfeld trage ich einen beliebigen und für mich sinnvollen Namen ein: Hier 5250.

Später, nachdem ich die Konfiguration der Verknüpfung fertig gestellt habe, wird auf dem Desktop eine Verknüpfung unter diesem Namen zu finden sein.

7.5.4.1
Seite 2

Sollte Ihnen das angebotene Standardsymbol nicht gefallen, so können Sie durch einen Klick auf die Symbolgrafik in eine Liste weiterer Symbole gelangen und sich aus der Liste ein passendes Symbol aussuchen.



Register Programm

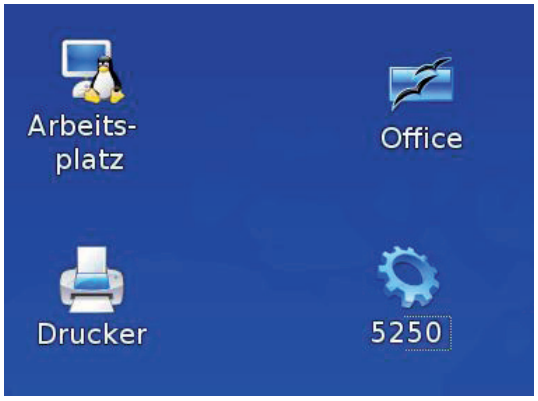
Im Register Programm versorge ich folgende Parameter:

Befehl: Hier trage ich den Namen des aufzurufenden Programms ein. In diesem Fall: setup5250, da dieses Programm bereits meine Verbindungskonfiguration zum Powersystem EPOSI501 enthält.

Arbeitsordner: Hier trage ich den Pfad zum Programm setup5250 ein. Hier: /opt/ibm/iSeriesAccess/bin

Durch einen Klick auf den Knopf Auswählen können Sie die beiden Felder durch interaktive Auswahl füllen.

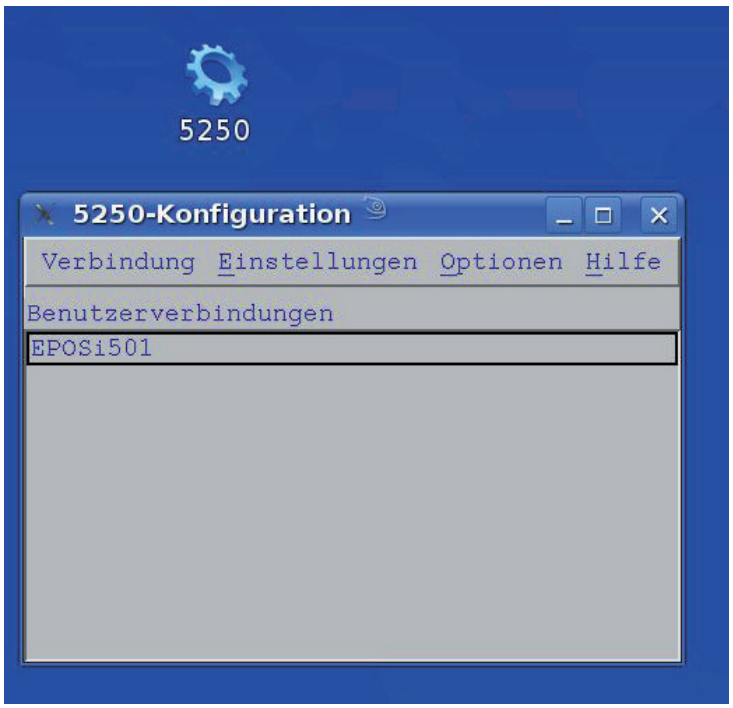
Ich bestätige die gesamte Eingabe mit OK.



Das Symbol 5250

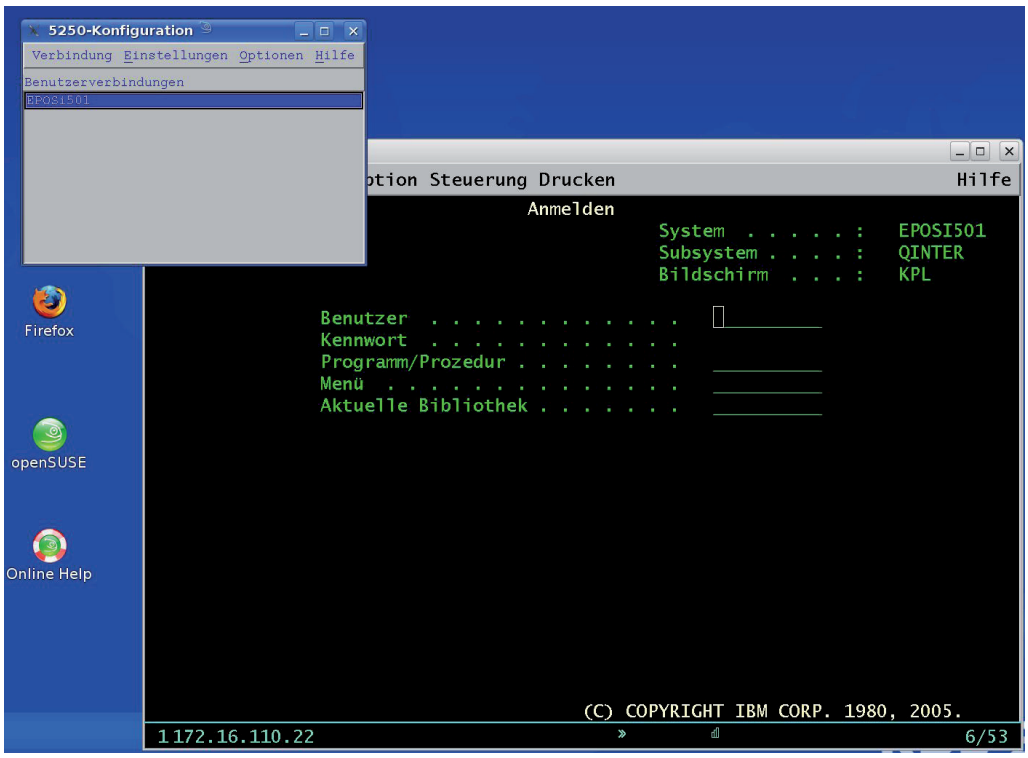
In obigem Bild sehen Sie einen Ausschnitt aus meinem KDE-Desktop, das das neue 5250-Symbol enthält.

Ein Klick auf dieses Symbol öffnet das setup5250-Programm.



Setup5250 aufrufen

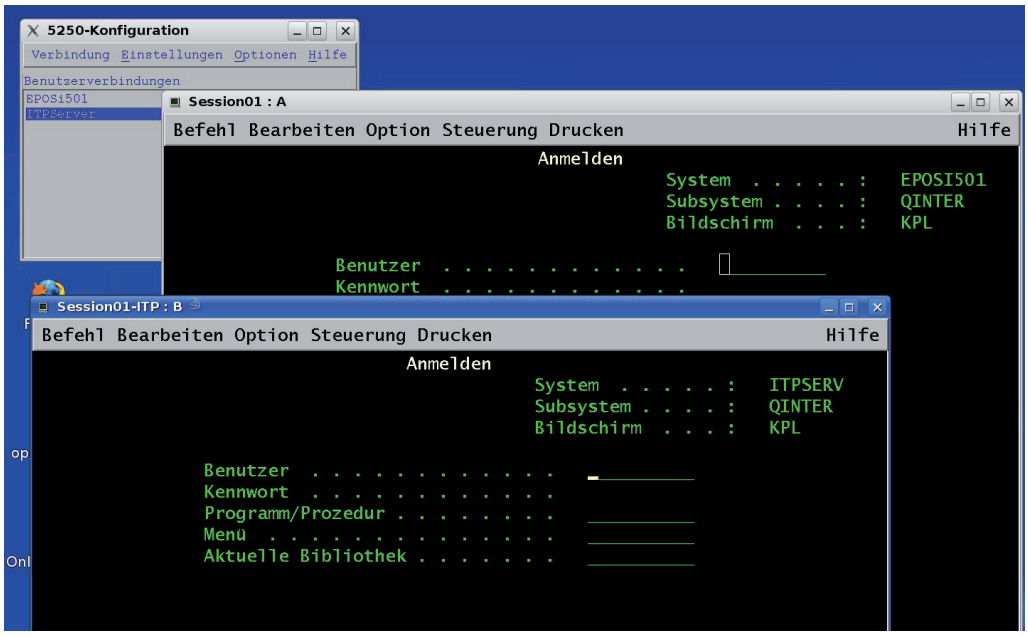
Ein Klick auf den Verbindungsnamen EPOSI501 öffnet den konfigurierten Bildschirm.



Aufruf des 5250-Bildschirms

7.5.4.1
Seite 4

In folgendem Bild sehen Sie, dass ich in setup5250 eine weitere Verbindung konfiguriert habe.



Weitere Verbindung

Auch hier habe ich durch einen Klick auf die neue Verbindung einen Bildschirm geöffnet.

Einführung

Mit der Freigabe von i5/OS V3R1 wurde der TCP/IP-Protokollstapel und damit verschiedene TCP/IP-basierte Dienste als integrierter Bestandteil des iSeries-Betriebssystems zur Verfügung gestellt. i5/OS V3 R1 wurde zum Jahreswechsel 1994/1995 freigegeben. Seitdem sind ein einige Jahre vergangen und wir leben heute mit dem Betriebssystem i5/OS V5R3.

Grundwerk 10/2004

7.6
Seite 2

TCP/IP-Dienste

Neben der Funktion, Verbindungen zwischen TCP/IP-fähigen Systemen zu realisieren, gehören zu diesem Protokollstapel eine ganze Reihe von Funktionen und Diensten, von denen mittlerweile alle im i5/OS implementiert sind. Damit Sie mit diesen Begriffen vertraut werden, stelle ich eine Kurzerläuterung dieser Funktionen an den Anfang dieses Kapitels.

Folgende TCP/IP-Dienste sind bisher implementiert worden:

SNMP: Simple Network Management Protocol

Im TCP/IP-Netzwerk besteht die Möglichkeit der Fernadministration eines Netzwerkes, sofern die beteiligten Hosts das SNMP-Protokoll unterstützen. Soll eine iSeries als TCP/IP-Host an dieser Fernadministration teilnehmen, muss ein entsprechender SNMP-Server gestartet werden, der dann für die Weiterleitung der Administrationsdaten an den fernen Administrator sorgt.

RouteD: Routing Daemon

Arbeiten Sie mit mehreren TCP/IP-Netzwerken, so können Sie mit Hilfe des RIP-Protokolls Leitwege festlegen, die das beschleunigte Auffinden von anderen TCP/IP-Host unterschiedlicher Netzwerke ermöglichen.

BOOTP: Boot Protocol

Das BOOTP-Protokoll ermöglicht den Anschluss von „dummen" NCs und Thin Clients an die iSeries. Dabei sendet der NC seine MAC-Adresse an die verbundene iSeries, die dann über Einträge in einer so genannten BOOTP-Tabelle die Verbindung zum NC herstellt und diesem dann sein Betriebssystem zusendet, sodass er funktionsfähig wird.

DNS: Domain Name Service

Der Domain Name Service ermöglicht es, numerischen TCP/IP-Adressen namensbezogene Adressen zuzuordnen. Diese vergebenen Namen und die ihnen zugeordneten Adressen werden in einer so genannten Zonendatei gehalten. Ein TCP/IP-Host, der auf einen anderen Host (Ziel-Host) zugreifen will, geht also zuerst zu dem System, das den Domain Name Service zur Verfügung stellt, und ermittelt mit Hilfe der Zonendatei die numerische Adresse des Ziel-Host. Bis V4R1 konnte die iSeries auf einen Domain Name Server zugreifen, ab V4R2 kann die iSeries einen DNS-Dienst zur Verfügung stellen. Ab V5R1 ist dieser Dienst in seiner dynamischen Variante verfügbar (DDNS: Dynamic DNS).

7.6.1
Seite 2

DHCP: Dynamic Host Configuration Protocol

DHCP gibt Ihnen die Möglichkeit, TCP/IP-Adressen dynamisch im Netzwerk zu verteilen. Sie brauchen also nicht manuell TCP/IP-Adressen vergeben, sondern lassen den DHCP-Server selbständig gültige Adressen vergeben. Die Lebensdauer dieser Adressen hängt dabei von einer vorgegebenen Lease-Zeit (Gültigkeitsdauer der Adresse) ab. So können Adressen nur für den einmaligen Gebrauch generiert werden oder Ihnen kann über eine längere Lease-Zeit eine längere Lebensdauer zugebilligt werden. Es ist durchaus möglich, sich anfänglich durch den DHCP-Server Adressen generieren zu lassen, die dann eine unendliche Lease-Zeit haben, so dass Sie dann mit statischen Adressen arbeiten können.

FTP: File Transfer Protocol

Das File Transfer Protokoll ermöglicht Ihnen die Übertragung von Objekten zu jedem System, dass FTP-Dienste realisieren kann. FTP ist auf der iSeries als Client und als Server-Dienst implementiert, so dass Sie Daten von der iSeries zu einem UNIX-, Windows NT- oder einem anderen System senden bzw. von diesem empfangen können. Ebenfalls können die genannten Systeme Daten zur iSeries senden und gegebenenfalls empfangen. Zum Aufbau einer FTP-Sitzung ist es notwendig, dass Anwender eine entsprechende Berechtigung für den Zugriff auf den FTP-Server besitzen. Im Falle der iSeries müssen Sie als iSeries-Benutzer ein entsprechendes Benutzerprofil besitzen.

Trivial FTP: Triviales File Transfer Protocol

Das triviale FTP ermöglicht ebenfalls die Versendung von Daten, beinhaltet jedoch nicht die Übergabe von Sicherheitsdaten. TFTP ist kein benutzerbezogener Dienst, sondern wird beim Einsatz von NCs benötigt, damit diese zum Start benötigte Dateien und Objekte – nach Erhalt einer TCP/IP-Adresse – herunterladen können.

TELNET: Bildschirmemulation

TELNET stellt eine einfache systemübergreifende Bildschirmemulation zur Verfügung. Benutzen Sie TELNET von iSeries zu iSeries, so erhalten Sie alle von einer 5250-Bildschirmsitzung übergebenen Attribute. Greifen Sie von einem ASCII-basierten Terminal auf eine iSeries zu, so werden diverse Attribute des 5250-Bildschirms nicht dargestellt. Eine Sonderform der Telnet-Emulation stellen so genannte TN3270- und TN5250-Emulationen dar. Diese nutzen das Telnet-Protokoll und übertragen dabei alle Merkmale und Funktionen eines 3270- bzw. 5250-Bildschirms.

LPD/LPR: Line Printer Daemon/Line Printer Requester

Der Line Printer Daemon (LPD) ist ein Druck-Server, mit dem Sie direkt aus dem Netzwerk heraus auf einen iSeries-Drucker drucken können. Welcher Drucker der iSeries Ihnen zur Verfügung steht, hängt von der Berechtigungssituation der iSeries ab. Der Line Printer Requester (LPR) ist der Client eines LPD. Mit der iSeries als LPR können die auf die LPDs unterschiedlicher Systemumfelder zugreifen. Ebenso können Sie von einem beliebigen LPR auf den LPD der iSeries zugreifen. LPR ist dreifach auf der iSeries implementiert: 1. als Remote Ausgabewarteschlange (wird mit dem Befehl CRTOUTQ erstellt), 2. als virtuelle Druckeinheit (Erstellung mit dem Befehl CRTDEVPRT) und 3. als Befehl LPR.

HTTP: HyperText Transport Protocol

Das HTTP-Protokoll ermöglicht es Ihnen, Ihre iSeries als Internet-Server aufzubauen. Seit V4R1 ist es möglich, mehrere Web-Server auf einer iSeries zu installieren. Mit Hilfe so genannter Access-Listen können dann für diese Web-Server eigene Berechtigungssysteme und Umgebungen aufgebaut werden.

SMTP: Simple Mail Transport Protocol

Das Simple Mail Transport-Protokoll steuert die Versendung von e-Mails, die dann von einem POP3-Server verwaltet werden können.

POP: Post Office Protocol

Der POP3-Server übernimmt die Verwaltung von e-Mails auf der iSeries. Darüber hinaus kann er Adressbuchfunktionen für Clients realisieren und Verbindungen zum Post-Dienst der Office Vision/400 herstellen. Die Verbindung zum POP3-Server der iSeries wird mittels eines Standard-e-Mail-Clients (Microsoft Exchange bzw. Outlook, Netscape Mail etc.) hergestellt.

REXEC: Remote Exec Server

Seit V4R1 ist die iSeries ein so genannter Remote Exec Server. Was heißt das? Von jedem TCP/IP-basierten Remote Exec Client können Sie – entsprechende Berechtigungen vorausgesetzt – Programmprozesse auf der iSeries starten. Es ist also durchaus möglich, dass ein LINUX-basierter Prozess mit Hilfe eines Remote Exec Client einen Dateitransfer auf der iSeries startet. Darüber hinaus kann eine iSeries auch ein Remote Exec Client sein. Dieser Dienst wurde über den CL-Befehl RUNRMTCMD (Fernen Befehl ausführen) auf der iSeries implementiert. Mit Hilfe des Client Access sind Sie in der Lage, jeden Windows 95/NT-Rechner zu einem Remote Exec Server zu machen. Mit RUNRMTCMD sind Sie auch in der Lage, von der iSeries einen PC-Befehl auf dem so konfigurierten Windows-System aufzurufen.

NFS: Netfile Services

Die Net File Services der iSeries sind ein nicht konfigurierbarer Dienst. Das heißt, dass dieser Dienst immer zur Verfügung steht. Er ist als Verzeichnis „QFileSvr.400“ im Integrated File System (IFS) der iSeries implementiert. Indem Sie unterhalb von „QFileSvr.400“ ein Unterverzeichnis mit dem Host-Namen einer zweiten iSeries einrichten, erhalten Sie einen Einstieg ins IFS dieser zweiten iSeries. Erfolgreich kann dieser Vorgang aber nur umgesetzt werden, wenn entsprechende Berechtigungen vergeben wurden.

DDM über TCP/IP

Ab V4R2 ist es nun auch möglich, DDM-Verbindungen mithilfe von TCL/IP zu realisieren. Erste Versuche meinerseits haben gezeigt, dass diese DDM-basierten Dateizugriffe deutlich schneller sind als SNA-basierte.

LDAP: Lightweight Directory Access Protocol

Seit V4R4 verfügen wir über einen Verzeichnis-Server, mit dessen Hilfe wir z.B. zentral Anmeldedaten im Netzwerk hinterlegen können. Seit OS/400 V5R1 ist dieser Dienst integraler Bestandteil des Betriebssystems und wird mit V5R2 noch einmal aufgewertet, indem die iSeries als Single-Signon-Server konfiguriert werden kann. Dieser Dienst wird als EIM (Enterprize Identity Mapping) bezeichnet und ist eigentlich nichts anderes als ein vorkonfigurierter LDAP-Server.

QOS: Quality of Service

Quality of Service ermöglicht die Definition von Bandbreiten-Garantien für bestimmte Dienste im Rahmen von Punkt-zu-Punkt- aber auch von Netzwerk-zu-Netzwerk-Verbindungen.

VPN: Virtual Private Network

Virtuelle Private Netzwerke stellen quasi Standleitungen über das Internet bereit, deren Datenströme verschlüsselt sind.

Kerberos (Network Authentication Service – NAS)

Bei Kerberos handelt es sich um ein Authentifizierungsprotokoll, mit dem zentrale Anmeldevorgänge in heterogenen Netzwerken realisiert werden können. Kerberos ist seit V5R3 sowohl als Client- als auch als Server-Protokoll implementiert.

Ich habe hier nicht alle Dienste aufgezählt, sondern nur die gebräuchlichsten. Wollen Sie sich über alle zur Verfügung stehenden Dienste informieren, so sollten Sie einfach einmal den Befehl STRTCPSVR in einer Befehlszeile eines iSeries-Bildschirms aufrufen und dann die F4-Taste drücken. Im Parameter „Server-Anwendung“ drücken Sie dann erneut die Taste „F4“. Dann erhalten Sie eine Übersicht aller im i5/OS implementierten TCP/IP-Dienste.

Verbindungsaufbau

In diesem Kapitel möchte ich mit Ihnen über den Verbindungsaufbau zwischen TCP/IP-fähigen Systemen sprechen. Der Verbindungsaufbau befolgt dabei grundsätzlich unterschiedliche Wege, verglichen mit Systemen, die unter IBM SNA laufen. Um den Verbindungsaufbau beschreiben zu können, werden Sie in der Folge eine Reihe von Begriffen kennen und verstehen lernen, die grundsätzlich und wichtig für die Funktion sowie den Betrieb von TCP/IP sind.

Lassen Sie uns von folgendem einfachen Szenario ausgehen:

Wir beabsichtigen einen PC direkt mit einer iSeries über ein Ethernet-Netzwerk zu verbinden. Welche Alternativen haben wir?

1.) Im Rahmen eines SNA-Netzwerks bauen wir eine DLC-Verbindung auf. Das DLC-Protokoll stellt eine direkte Verbindung zur iSeries her, d.h., das beide Systeme eine direkte Verbindung unterhalten. Dies wiederum bedeutet, dass beide Systeme eine so genannte SNA-Sitzung aufbauen müssen. Damit der Sitzungsaufbau erfolgen kann, tauschen beide Systeme so genannte Austausch-IDs aus. Wird dieser Prozess korrekt abgeschlossen, so werden Benutzerkennungsdaten in der iSeries überprüft. Und sollte auch dieser Vorgang korrekt abgeschlossen werden, so wird die Verbindung als entsprechend berechtigt gekennzeichnet. Erst danach kann das eigentliche DFV-Geschäft aufgenommen werden, es können Bildschirmemulationen, Dateitransfers, emulierte- und virtuelle Drucker etc. genutzt werden. Diese Vorgänge werden dann als Konversationen bezeichnet. Fazit: Die SNA Sitzung sorgt für die physikalische und logische Verbindung und legt die Sicherheitsdaten der Verbindung fest. Die Konversation steuert dann die auf der Basis der Sitzung geforderten Tätigkeiten und Funktionen.

So weit so gut, was für Sie wichtig ist, ist, dass es sich bei einer SNA-Verbindung um eine Verbindung zwischen **zwei Systemen** handelt, denn die Adressen die wir zu vergeben haben, beziehen sich jeweils auf das gesamte System. Werte die hierbei zu versorgen sind, sind die Netzwerk-IDs beider Systeme, die lokalen Standortnamen beider Systeme usw. Auf unser kleines Beispiel bezogen, führt diese Arbeit dann dazu, dass nach erfolgreicher Konfiguration unser PC unterhalb der Leitungsbeschreibung durch eine APPC-Steuereinheit und durch eine APPC-Device repräsentiert wird. Diese beiden Objekte steuern die Sitzung. Alle Tätigkeiten – also die Konversationen – werden dann als Jobs dargestellt.

2.) Wir bauen eine TCP/IP-Verbindung auf. Der große Unterschied zu der oben beschriebenen Variante besteht darin, dass jetzt keine System-zu-System-Verbindung aufgebaut wird, sondern dass die Verbindungsaufnahme letztendlich auf der Ressourcen-Ebene vollzogen wird. Was sagt uns das? Wenn wir eine TCP/IP-Adresse vergeben, wem wird eigentlich eine Adresse

gegeben? Die IP-Adresse wird auf der iSeries als so genanntes TCP/IP-Interface an die Ethernet-Leitungsbeschreibung gebunden und beim PC wird die Adresse an die Ethernet-Netzwerkkarte gebunden. Fazit: die Verbindung erfolgt auf der Ebene der Netzwerk-Ressourcen, also der Netzwerkkarten. Der funktionale Gegensatz zur SNA-Verbindung besteht darin, dass zum Zeitpunkt der Verbindungsaufnahme keine Berechtigungsdaten zwischen den Systemen übertragen werden. Die IP-Adresse beschreibt lediglich, dass ein System in einem TCP/IP-Netzwerk verfügbar ist, dass seine Adresse sozusagen im Netzwerk bekannt ist und dass dieses System zu einem logischen Netzwerk dazugehört. Mehr nicht! Und wann erfolgt dann die Berechtigungsüberprüfung? Diese erfolgt dann auf der so genannten Funktionsebene. Immer dann, wenn wir eine TCP/IP-Funktion aufrufen, wird zuerst überprüft ob ein Benutzer für diese Funktion die entsprechenden Rechte besitzt und erst dann wird diese Funktion ausgeführt. Das lässt sich auch sehr leicht nachvollziehen. Gehen Sie einmal an einen PC, der mittels TCP/IP auf die iSeries zugreift und starten Sie einfach eine Bildschirmemulation, ohne vorher mit dem Verbindungsmanager die Verbindung aufzubauen. Was passiert? Sie müssen sich zuerst für die Funktion anmelden (nicht für die Verbindung!), erst dann kommt der Bildschirm. Wenn Sie die Emulation beenden und erneut wieder aufrufen – was passiert dann? Sie müssen sich erst wieder für die Funktion anmelden und danach kommt der Bildschirm.

Die Ursache für dieses komplett unterschiedliche Verhalten beider Verbindungsvarianten liegt sicherlich darin, dass die Anforderungen, die an das TCP/IP-Protokoll gestellt wurden, grundsätzlich verschieden sind, von denen die an das SNA-Verfahren gestellt wurden – Stichwort: offenes Netzwerk. Für uns iSeries-Leute ergibt sich diese Unterschiedlichkeit jedoch aus der Implementierung der beiden Schnittstellen. Während das SNA-Verfahren seine Konversationen (siehe oben) auf der Prozessebene des i5/OS abwickelt (also eine Konversation nicht anderes ist, als ein Job des i5/OS, den wir in der WRKACTJOB- [Mit aktiven Jobs arbeiten-] Anzeige als Verbindungsjob im Subsystem QCMN oder QBASE wahrnehmen), so wird das TCP/IP-Geschäft über eine so genannte Socket-Schnittstelle im Betriebssystemkern der iSeries abgewickelt. Oberflächlich betrachtet, finden wir also keinen TCP/IP-Verbindungsjob in den klassischen DFV-Subsystemen. Nutzen Sie bereits TCP/IP, so sollten Sie sich einmal mit WRKACTJOB das Subsystem QCMN oder QBASE ansehen. Sie werden feststellen, dass Sie nichts sehen – jedenfalls keine Jobs, die die TCP/IP-Verbindungen zu den entsprechenden TCP/IP-fähigen Clients darstellen.

Da TCP/IP-Verbindungen auf der Ebene der Funktionen hergestellt werden, muss es für die Menge der TCP/IP-Funktionen eine Schnittstelle geben und diese Schnittstelle wird durch so genannte TCP/IP-Ports dargestellt. Die Ports kann man sich sehr einfach als Eingänge in die TCP/IP-Schnittstelle vorstellen, wobei jeder Eingang (Port) für eine bestimmte Funktion reserviert ist.

Die TCP/IP-Ports werden durchnummeriert von 0 bis 65535, wobei wir jedoch zwei Nummernkreise unterscheiden: Alle Ports zwischen 0 und 1023 sind so genannte Standard-Ports, d.h., dass über sie Standardfunktionen des TCP/IP betrieben werden. Lesen Sie noch einmal die Einführung zum Kapitel 4.1 „TCP/IP“. Dort habe ich beschrieben, welche Funktionen in TCP/IP der iSeries implementiert wurden. Jede dieser Funktionen wird durch einen Port repräsentiert. Soll ein bestimmter TCP/IP-Dienst ausgeführt werden, so sendet er zuerst einmal seine Portnummer. Diese wird dann auf der empfangenden Seite gesehen und die Anforderung wird an den entsprechenden TCP/IP-Server weitergeleitet und verarbeitet.

```
5          Remote Job Entry
7          Echo
11         Active Users
13         Daytime
20         File Transfer (Data)
21         File Transfer (Control)
23         Remote Terminal(TELNET)
25         SMTP-Simple Mail Transfer Protocol
37         Time
42         Host Name Server
43         Who Is
53         Domain Name Server
67         BOOTP/DHCP Relay Agent Server oder Dynamic Host Configu-
           ration Protocol Server
68         BOOTPC/DHCPC Bootstrap Protocol Client oder Dynamic Host
           Configuration Protocol Client


69         Trivial File Transfer Protocol
70         Gopher
79         Finger
80         World Wide Web HTTP
101        Host Name Server
102        ISO TSAP
103        X.400
104        X.400 SND
105        CSNET Mailbox Name Server
109        Post Office Protocol(POP2)
110        Post Office Protocol(POP3)
```

7.6.2
Seite 4

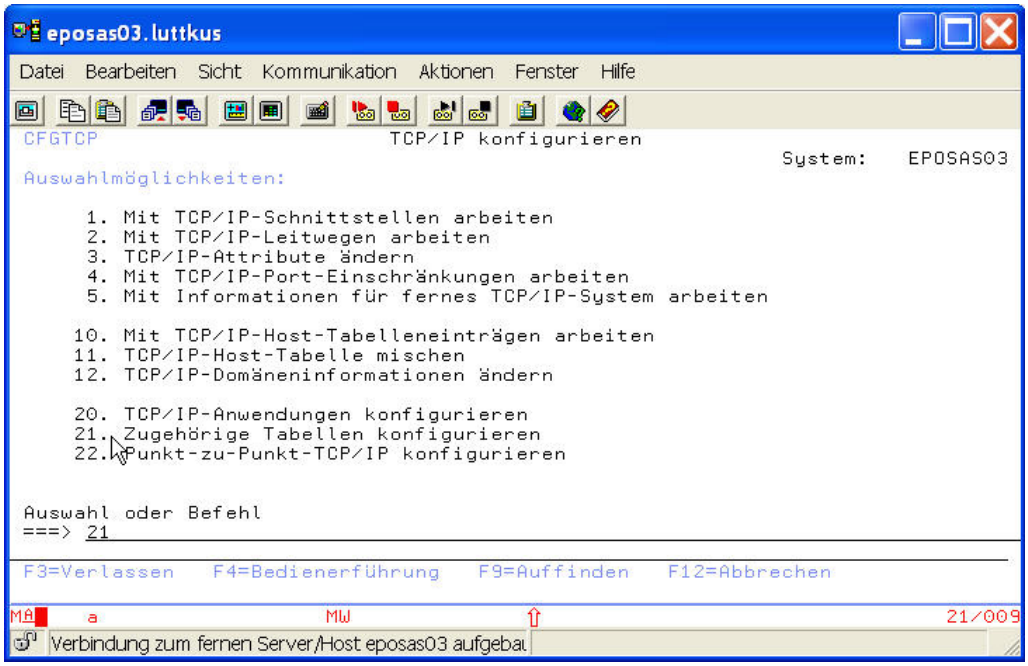
```
111        RPC Portmap
137        NetBIOS Name Service
138        NetBIOS Datagram Service
139        NetBIOS Session Service
161        Simple Network Management Protocol(SNMP)
247        Subnet Broadcast Trivial File Transfer Protocol
512        Remote command execution (REXEC)
515        Remote printing(LPD)
```

Alle Ports im Nummernkreis 1 bis 1023 werden als so genannte Standard-Ports bezeichnet. Das bedeutet, dass alle Standard-Dienste des TCP/IP-Netzwerks Ports in dem oben genannten Nummernkreis nutzen. Innerhalb dieses Kreises wird die Gruppe der TCP/IP-Dienste, die von fast jedem Benutzer genutzt werden, als „Well Known Ports“ bezeichnet. Well Known steht für den Sachverhalt, dass ausschließlich Standard-TCP/IP-Dienste, die in allen Implementierungen vorhanden sind und stets gebraucht werden, diese Ports benutzen. Ports mit einer Portnummer ab 1024 sind dann so genannte Individual-Ports oder auch kurzlebige Ports. Ein gutes Beispiel für einen Individual-Port stellt die Funktion des Webfacing Servers dar. Diese verwendet nämlich den Port 4004. Damit ist eigentlich klar, dass der Webfacing Server keine Standardfunktion des TCP/IP ist, sondern eine Individualfunktion, die wir nicht unbedingt in anderen TCP/IP-Implementierungen erwarten können.

Um einen „hereinkommenden“ Ports zu analysieren, gibt es im TCP/IP die Funktion des so genannten Port-Mappers. Die Aufgabe eines Port-Mappers ist schnell erklärt. Er überwacht hereinkommende Anforderungen für TCP/IP-Dienste, prüft ob auf dem System, auf dem der Port-Mapper läuft, der Dienst bekannt ist und stellt dann die Verbindung zu dem jeweiligen Dienste-Server her. Kommt eine in diesem Sinne ungültige Port- und somit Dienste-Anforderung herein, so wird die Verbindung abgelehnt. Der jeweilige Dienste-Server übernimmt dann auch die Validierung des Benutzers.

Wie kann jetzt der Port-Mapper feststellen, ob ein Port bekannt ist oder nicht? Für die Port-Validierung existiert im System eine so genannte Service-Tabelle. In anderen Architekturen – wie Windows, Unix, Linux etc. – ist diese eine Textdatei mit dem Namen „Service“. In Windows NT, 2000 und XP finden Sie diese Datei im Pfad „/Windows/System32/Drivers/etc.“. Bei Interesse sollten Sie sich diese Datei einfach einmal in den Editor laden. Die Datei kann übrigens nur von einem lokalen Windows-Administrator fortgeschrieben werden. Sind Sie also kein Administrator Ihres Systems, so können Sie hier auch keine Zufallsschäden anrichten. Als Administrator Ihres Systems sollten Sie allerdings Vorsicht walten lassen.

Eine ähnliche Datei existiert auch auf Ihrer iSeries. Rufen Sie einfach durch den Befehl CFGTCP die TCP/IP-Konfiguration Ihres Systems auf.

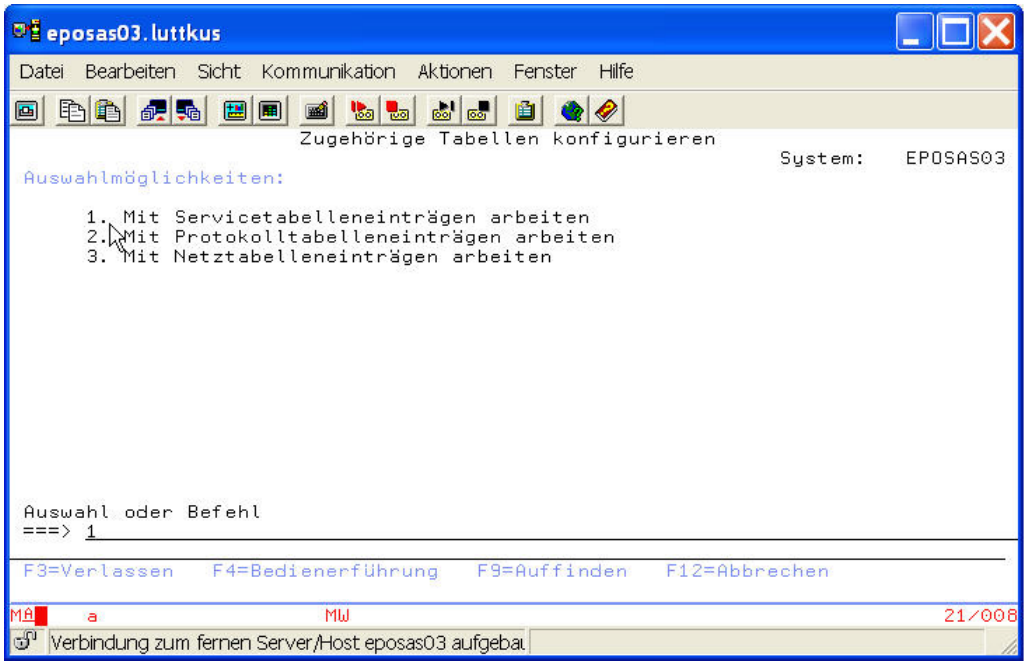


Zugehörige Tabellen konfigurieren

Hier wählen Sie den Punkt 21 – Zugehörige Tabellen konfigurieren – aus.

Daraufhin gelangen Sie in das nachfolgende Menü.

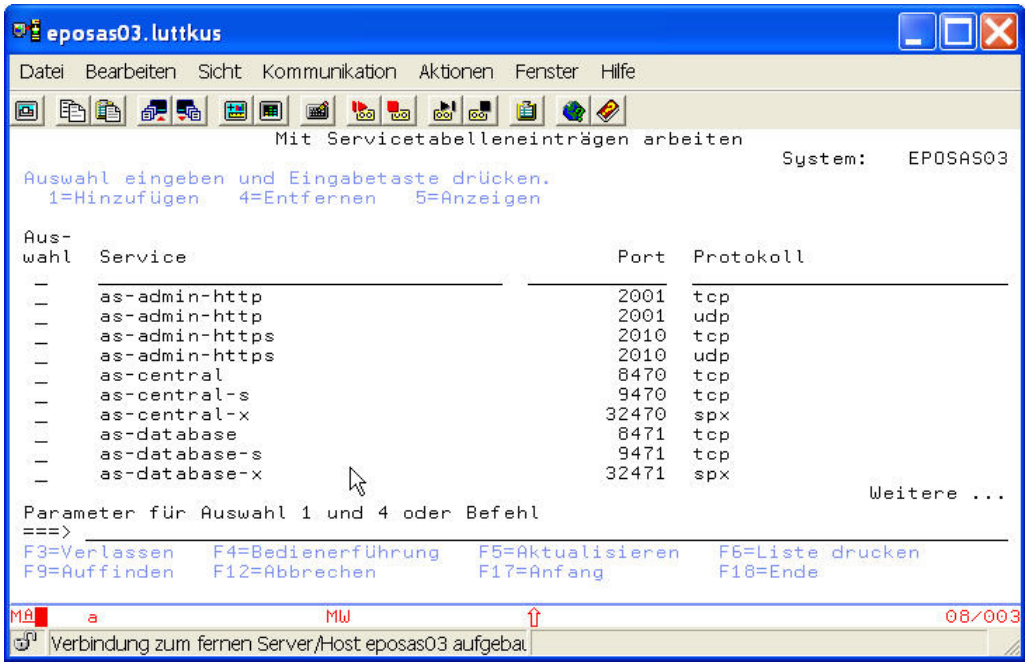
7.6.2
Seite 6



Mit Servicetabelleneinträgen arbeiten

Im oben gezeigten Menü wählen Sie jetzt die Auswahl 1 – Mit Servicetabelleneinträgen arbeiten – aus.

Jetzt sehen Sie folgenden Dialog.



Mit Servicetabelleneinträgen arbeiten

Im oben gezeigten Bildschirm erkennen Sie neben der Spalte „Auswahl“ drei weitere Spalten: „Service“, „Port“ und „Protokoll“.

Mit „Service“ wird eine Zeichenkette bezeichnet, die den Aufruf eines Dienste-Servers startet. Um zum Beispiel den Telnet-Server, der für 5250-Bildschirmemulationen benötigt wird, zu starten, werden Sie in der Spalte „Service“ mindestens zwei Einträge mit dem Namen „Telnet“ finden. Warum zwei Einträge? Dazu später mehr.

Die Spalte „Port“ spricht für sich selbst. Hier ist die für den jeweiligen Dienst genutzte Portnummer eingetragen.

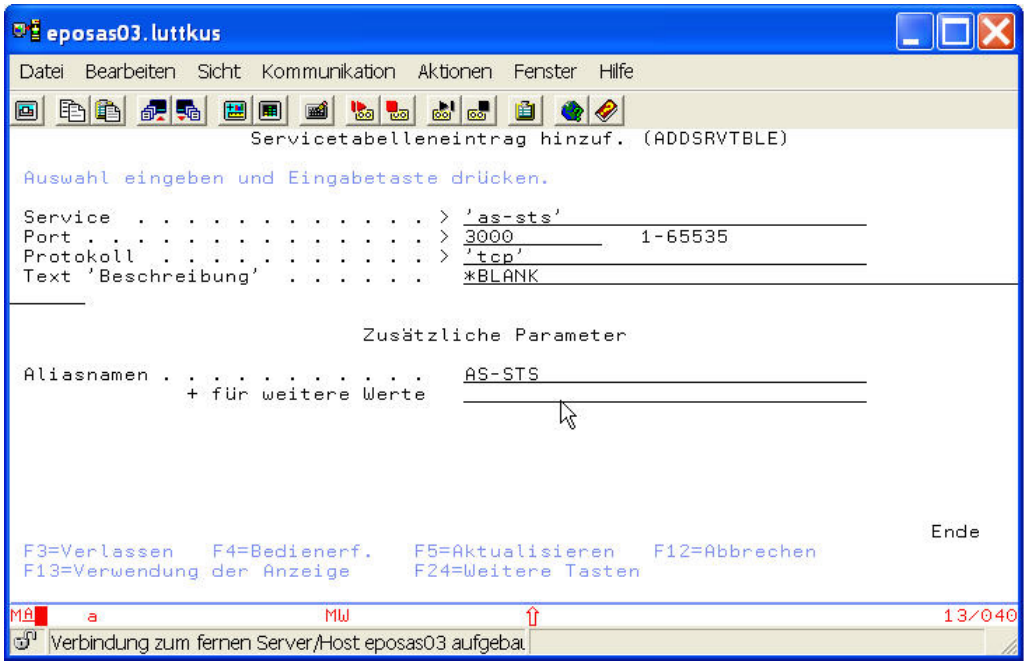
In der Spalte „Protokoll“ sehen Sie dann verschiedene Einträge, wobei für das TCP/IP-Netzwerk nur zwei Begriffe wichtig sind: „tcp“ und „udp“. Diese beiden Namen bezeichnen unterschiedliche verbindungsaufbauende Protokolle. Das tcp-Protokoll betreibt so genannte gesicherte Verbindungen Das bedeutet, dass, wenn zu einem bestimmten Dienst eine Verbindung aufgebaut wurde, neben dem Datenverkehr ein so genanntes Sicherheits-Polling stattfindet, bei dem der Server-Dienst sich versichert, dass der entsprechende Client-Dienst aktiv ist. Sollte also bei der Ausführung eines Dienstes der Client – aus welchem Grund auch immer – seinen Dienst einstellen, so wird der Server das merken und die Kommunikation zum Client einstellen. Das udp-Protokoll dagegen stellt eine Verbindung zu einem Client her. Bei der nach der Verbindungsaufnahme stattfindenden Kommunikation wird jedoch kein Polling zum Client durchgeführt. Sollte also während der Kommunikation der Client seinen Dienst aufgeben, so sendet der Server weiterhin Daten an den nicht existenten Client.

7.6.2
Seite 8

Lassen Sie uns eine sinnvolle Übung machen: Zum Betrieb der hardwarenahen Dienste des Operations Navigator muss der Port 3000 in der Service-Tabelle der iSeries eingetragen werden. Wir fragen uns jetzt nicht, warum dies nicht schon bei der Installation des Systems vom Installationsprozess geregelt wurde. Tatsache ist, dass ab OS/400 V5R1 mit seinen erweiterten iSeries Navigator-Fähigkeiten ein Eintrag in der Service-Tabelle zu hinterlegen ist, sonst kann in Teilbereichen mit dem iSeries Navigator nicht gearbeitet werden.

Um die volle Funktionalität des iSeries Navigators nutzen zu können, tragen Sie in den vorherigen Bildschirm in das Auswahl-Feld eine „1“ ein, um einen neuen Service-Tabelleneintrag zu erzeugen.

Nachdem Sie auf „Datenfreigabe“ gedrückt haben, erscheint folgender Dialog.



Servicetabelleneintrag hinzufügen

In das Feld „Service“ tragen Sie den Wert „as-sts“ ein. Achten Sie darauf, dass dieser Wert in Kleinschreibung eingetragen ist.

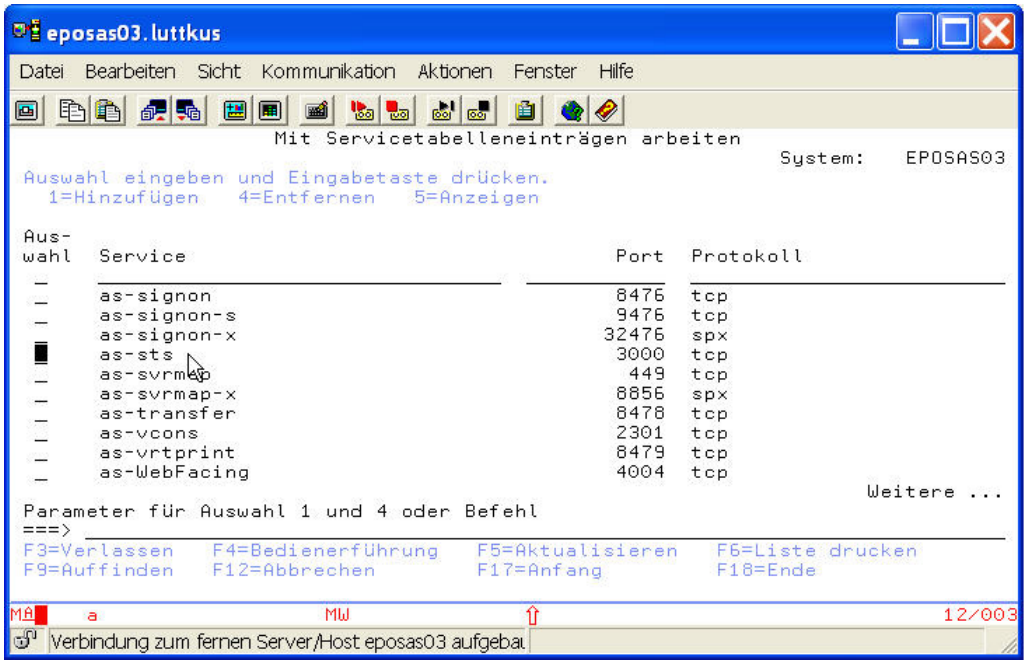
In das Feld „Port“ tragen Sie die Portnummer „3000“ ein.

In das Feld „Protokoll“ tragen Sie „tcp“ ein. Achten Sie auch hier auf die Kleinschreibung.

Kleinschreibung erzwingen Sie dadurch, dass Sie die klein geschriebenen Begriffe in einfache Hochkommata (‘) setzen.

Danach drücken Sie die F10-Taste für zusätzliche Parameter und tragen in das Feld „Aliasnamen“ die Zeichenfolge „AS-STS“ ein. Jetzt bitte ausschließlich in Großschreibung!

Bestätigen Sie den Dialog mit „Datenfreigabe“.



Neuen Eintrag überprüfen

Zur Prüfung des neuen Eintrags blättern Sie solange in oben gezeigtem Bildschirm, bis der neue Eintrag gefunden wurde. Achten Sie auf die Schreibweise! Auch wenn Sie derzeit nicht mit dem iSeries Navigator arbeiten, so sollten Sie zumindest, nachdem Sie diese kleine Aufgabe erledigt haben, in der Gewissheit leben, dass Sie für eine spätere Arbeit mit dem iSeries Navigator für sich und die Kollegen etwas Sinnvolles getan haben.

Die „Servicetabelleneinträge“ sind also das Bindeglied zwischen dem Port-Mapper, der hereinkommende Anforderungen bearbeitet, und den Dienste-Server-Jobs, die wir kurz als TCP/IP-Server bezeichnen. Bei jedem Start der iSeries müssen diese Dienste-Server gestartet werden.

Damit es einen Sinn macht, diese Server zu starten, müssen folgende Voraussetzungen gegeben sein:

- Der Netzwerkschnittstelle muss ein TCP/IP-Interface hinzugefügt worden sein. Auf gut Deutsch: Erfassen Sie für die Netzwerkschnittstelle der iSeries eine IP-Adresse (ADDTCPIFC: TCP/IP-Interface hinzufügen).
- Die Schnittstelle muss gestartet werden (STRTCPIFC: TCP/IP Interface starten).

7.6.2
Seite 10

Jetzt können die TCP/IP-Server mit dem Befehl STRTCPSVR (TCP/IP-Server starten) gestartet werden. Dieser Befehl kennt nur einen Parameter, der den Namen SERVER trägt. Folgende Parameterwerte sind möglich:

Wert	Bezeichnung
*ALL	Alle TCP/IP-Server starten, für die das Attribut AUTOSTART auf *YES gesetzt wurde.
*AUTOSTART	Alle für den automatischen Start definierten Dienste starten
*SNMP	Start des SNMP-Agenten
*ROUTED	Start des Router Daemon
*BOOTP	Start des Bootp-Server
*TFTP	Start von Trivial FTP
*DNS	Start des Domain Name Service
*DHCP	Start des DHCP-Server
*DDM	Start des DDM over TCP/IP
*TELNET	Start des Telnet-Server
*FTP	Start des FTP-Server
*SMTP	Start des SMTP-Server
*LPD	Start des TCP/IP-Druck-Server
*http	Start des Web-Server
*WSG	Start des Workstation Gateway
*POP	Start des POP3-Server
*REXEC	Start des Remote Exec Server
*DCE	Start des DCE-Server
*DIRSRV	Start des LDAP-Server
*USF	Start des Ultimedia System Facilities Server
*NSMI	Start des Network Station Manager Inventory Server
*NSLD	Start des Networkstation Login Daemon
*INETD	Start des Internet Daemon
*MGTC	Start des Management Central
*ONDMD	Start des On-Demand Server
*NETSVR	Start des Netserver (Microsoft Kompatibilität)
*DLFM	Start des Datalink File Managers
*VPN	Start des VPN-Server
*EDRSQL	Start des SQL-Server für Remote, dynamische Zugriffe
*HOD	Start des Host-On-Demand-Server

Wert	Bezeichnung
*ODPA	Start der On-Demand-Plattform Authentifizierung
*NTP	Start des Time-Server Client
*QOS	Start des Quality of Service
*TCM	Start des Triggered Cache Managers
*DOMINO	Start der Domino Server
*WEBFACING	Start des Webfacing Server

Einige dieser Server werden automatisch gestartet: *TELNET, *FTP, *LPD, *HTTP und *POP. Nehmen wir einmal an, Sie beabsichtigen das Workstation Gateway separat zu starten, so geben Sie einfach den Befehl STRTTCPSVR *WSG ein und dann wird dieser Server gestartet.

Sie können sich aber grundsätzlich die Arbeit einfacher machen, indem Sie einfach den Befehl STRTCP (TCP/IP Starten) eingeben, ohne jegliche Parameter aufzurufen. Was passiert jetzt? Dieser Befehl erwartet zwei Parameter:

1.) STRSVR – zu startende Server

2.) STRIFC – zu startende Schnittstellen

Beide Parameter sind mit dem Wert *YES vorbelegt, so dass alle TCP/IP-Server, deren AUTOSTART-Parameter auf *YES gesetzt wurden, automatisch gestartet werden. Der zweite Parameter sorgt dafür, dass alle definierten und funktionsfähigen Schnittstellen gestartet werden.

Es ist sicherlich für Sie eine einfache Routine, den Befehl STRTCP in das Autostart-Programm der iSeries aufzunehmen, so dass alle TCP/IP-Interfaces mit jedem Start des Systems aktiviert werden.

Nachdem wir uns mit Hintergrundwissen beschäftigt haben, was meiner Meinung nach wichtig ist, um TCP/IP vernünftig nutzen zu können, wenden wir uns im nächsten Abschnitt den eher handfesten Dingen zu, die da wären: Erstellung eines TCP/IP-Interface auf der iSeries und TCP/IP für einen Windows PC aktivieren.

7.6.2
Seite 12

Adressierung

Nachdem wir im vorherigen Abschnitt über den Verbindungsaufbau in einem TCP/IP-Netzwerk, Ports und Server-Programme gesprochen haben, kümmern wir uns jetzt darum, wie eine IP-Adresse aufgebaut ist und wie eine iSeries unter TCP/IP adressierungsfähig gemacht wird. Das Umfeld, in dem wir uns dabei gedanklich bewegen, ist das eines lokalen Ethernet-Netzwerks.

Die IP-Adresse besteht aus einer 32 Bit langen Folge von 0 und 1. Jeweils ein Byte – also acht Bit – werden durch eine Dezimalzahl repräsentiert. Damit ergibt sich, dass eine IP-Adresse aus vier 1-Byte-langen Dezimalwerten zusammengesetzt ist. Diese Werte werden dann jeweils durch einen Punkt getrennt.

Formal betrachtet sieht eine IP-Adresse folgendermaßen aus:

n.n.n.n

„n“ steht für einen Wert zwischen 0 und 255. Null repräsentiert die Bit-Folge 00000000 und 255 repräsentiert die Bit-Folge 11111111.

Hieraus ergibt sich, dass die IP-Adresse 172.16.0.1 eine – zumindest formal betrachtet – richtige IP-Adresse ist. Um aber bewerten zu können, ob diese Adresse auch funktional korrekt ist, müssen wir die Struktur einer IP-Adresse genauer kennen.

Jede IP-Adresse teilt sich grundsätzlich in zwei Einheiten auf: in die Net-ID und die Host-ID. Die Net-ID beschreibt dabei die Zugehörigkeit zu einem logischen Netzwerk, während der zweite Teil – die Host-ID – die Adresse der Arbeitsstation darstellt.

Dem Gedanken, Net-IDs zu vergeben, liegt zugrunde, dass ein physisches Netzwerk – als Einheit betrachtet – unter Umständen verwaltungstechnisch zu inflexibel ist. Deshalb erscheint es sinnvoll, die physische Einheit eines Netzwerks in logische Untereinheiten zu zergliedern. Damit können kleinere, organisatorisch einfachere Einheiten entstehen, die besser administriert werden können, und bei denen sich der Netzwerkverkehr besser formal beschreiben lässt. Außerdem können deren Schnittstellen besser kontrolliert werden.

Insofern unterscheidet sich ein TCP/IP-basiertes Netzwerk nicht grundsätzlich von einem SNA-Netzwerk. Auch hier kennen wir die lokale Netzwerk-ID als Beschreibungselement für die Zugehörigkeit zu einem logischen Netzwerk.

Die Host-ID gibt dann einem TCP/IP-Host im logischen Netzwerk eine eindeutige Adresse. Der Begriff des Host sollte aber nicht missverstanden werden. Ein TCP/IP-Host entspricht im SNA-Netzwerk einem Knoten. Host beschreibt also nicht die Steuerungsfunktion eines Systems im Netzwerkbetrieb, sondern beschreibt lediglich ein EDV-System im TCP/IP Netzwerk. So

7.6.3
Seite 2

ist eine TCP/IP-fähige iSeries, ein UNIX-System, ein PC unter Windows NT, ein Windows 95 Notebook – also jedes TCP/IP-fähige System – ein Host im TCP/IP-Netzwerk.

Insgesamt gesehen ist die Struktur einer IP-Adresse zumindest nicht unähnlich einer SNA-Knotenadresse. Der Unterschied besteht jedoch darin, dass die Grenze zwischen der Net-ID und der Host-ID variabel gehalten ist.

Strukturell betrachtet kann eine IP-Adresse folgende Erscheinungsformen haben:

N.H.H.H

N.N.H.H

N.N.N.H

„N“ steht für Net-ID und „H“ für Host-ID.

Alle eben beschriebenen Adressstrukturen sind nicht nur möglich, sondern auch richtig! Die Frage, die sich daraus ergibt, ist die, welcher steuernde Parameter bewirkt die Struktur einer Adresse. Das Geheimnis liegt darin, dass IP-Adressen in so genannte Adressklassen unterschieden werden.

Derzeit werden im Internet drei Adressklassen verwendet: Class A-, Class B- und Class C-Adressen. Über welches Unterscheidungsmerkmal bilden sich aber diese Adressklassen heraus? Zur Unterscheidung wird das erste Byte einer IP-Adresse herangezogen.

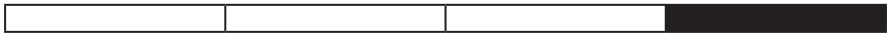
Class A

Adressenbereich: 1.0.0.1 - 126.255.255.254

Class B

Adressenbereich: 128.0.0.1 - 191.255.255.254

Class C



Adressenbereich: 192.0.0.1 - 223.255.255.254

Steht im ersten Byte einer IP-Adresse ein Wert zwischen 0 und 126, so ist dies eine Class A-Adresse. Der Inhalt der anderen Stellen kann dann theoretisch jeweils in den Grenzen 0 bis 255 liegen. Damit hat die vergebene Adresse die Struktur N.H.H.H. Also wäre eine IP-Adresse 57.89.0.1 plausibel. Der Wert 57 stellt die Net-ID dar, während die Werte 89.0.1 die Host-ID beschreiben. Alle Hosts, die die Net-ID 57 haben, sind damit Bestandteil des gleichen logischen Netzwerks.

Steht im ersten Byte einer IP-Adresse ein Wert zwischen 128 und 191, so liegt eine Class B-Adresse vor. Auch hier gilt, dass der Inhalt der restlichen Stellen wiederum zwischen 0 und 255 liegen kann. Eine Class B-Adresse hat die Struktur N.N.H.H. Damit wäre die am Anfang dieses Abschnitts abgebildete Adresse 172.16.0.1 nicht nur formal richtig. Das Wertepaar 172.16 ist in diesem Fall als Net-ID zu betrachten, hingegen ist das Wertepaar 0.1 die Host-ID. Daraus ergibt sich, dass all die Systeme, die die Net-ID 172.16. haben, zum gleichen logischen Netzwerk gehören.

Steht im ersten Byte einer IP-Adresse ein Wert zwischen 192 und 223, so handelt es sich um eine Class C-Adresse. Alle anderen Stellen liegen auch hier in den Grenzen zwischen 0 und 255. Eine Class C-Adresse hat die Struktur N.N.N.H. Die IP-Adresse 195.10.99.10 ist damit eine gültige Class C-Adresse. Die ersten drei Stellen 195.10.99 werden jetzt bei Netzwerkvorgängen als Net-ID verstanden, die letzte Stelle – 10 – ist die Host-ID. Alle die Hosts, die die Net-ID 195.10.99 aufweisen, gehören damit zum selben logischen Netzwerk.

Soweit, so gut – aber wozu ist das gut? In der Klassifizierung von IP-Adressen beschreiben wir letztendlich die Dimensionen unseres Netzwerks. Ein Class A-Netzwerk mit seiner einstelligen Net-ID kann theoretisch in 126 logische Netzwerke unterteilt werden. Jedes einzelne logische Netzwerk kann jedoch eine große Menge an Hosts umfassen. Ein Class B-Netzwerk kann dagegen in eine noch größere Anzahl an logischen Netzwerken unterteilt werden, wobei dann aber jedes dieser Netzwerke weniger Host adressieren kann. Ein Class C-Netzwerk kann noch feiner in logische Netzwerke zerteilt werden, aber jedes dieser Netzwerke kann nur maximal 255 Hosts haben.

Ich habe ganz bewusst darauf verzichtet, Ihnen die numerischen Grenzwerte zu nennen, denn schon ein Class A-Netzwerk dürfte – zumindest für ein lokales Netzwerk – genügend logische Gliederungsmöglichkeiten bieten.

7.6.3
Seite 4

Die privaten Adressen

Sie müssen immer bedenken, dass es sich bei TCP/IP-Adressierungsverfahren, um ein weltweit angewendetes Verfahren handelt. Um jetzt Namensdubletten mit dem Internet zu vermeiden, gibt es in jeder Adressklasse einen so genannten privaten oder freigelassenen Bereich.

Freie Class A-Adressen: 10.0.0.0 - 10.255.255.255

Freie Class B-Adressen: 172.16.0.0 - 172.31.255.255

Freie Class C-Adressen: 192.168.0.0 - 192.168.255.255

Arbeiten wir in einem LAN (Local Area Network), also einem privaten, nichtöffentlichen Netzwerk, so sollten wir uns immer im Bereich privater Adressen aufhalten. Damit sind wir zuerst einmal davor sicher, eine Adresse zu verwenden, die schon einmal im Internet vergeben wurde. Auf der anderen Seite ist kein Internet-Provider in der Lage, eine Information direkt an eine private Adresse durchzurouten, so dass eine private Adresse auch als eine Art Schutzmechanismus verstanden werden kann.

7.6.3.1

Seite 2

Subnetzmasken

Mit Hilfe der Adressklassen dimensionieren wir unsere Netzwerke nach außen hin. Um aber auch intern zu einer differenzierten Struktur zu kommen, muss jeder IP-Adresse eine so genannte Subnetzmaske zugeordnet werden.

Subnetzmasken zu definieren, ist eigentlich recht einfach. Jedoch differenziert deren Funktionalität zu verstehen, ist eine nicht so einfache Aufgabe! Nur wer die Funktion der Subnetzmasken richtig versteht, ist in der Lage ein TCP/IP-Netzwerk richtig zu strukturieren. Deshalb werde ich mit Ihnen in der Folge deren Art und Funktion ergründen.

7.6.3.2

Seite 2

Standard-Subnetzmasken

Die Teilnetzmaske oder auch Subnetzmaske einer IP-Adresse muss mit einem eindeutigen Wert versorgt werden. Die Online-Hilfen der meisten TCP/IP-Implementierungen führen dann die folgenden Werte auf, deren Eintragung von der Adressklasse einer avisierten IP-Adresse abhängt.

Class A-Adresse: 255.0.0.0

Class B-Adresse: 255.255.0.0

Class C-Adresse: 255.255.255.0

Verfahren Sie so, dann wird Ihre TCP/IP-Schnittstelle funktionieren. Das ist gut, aber Sie sollten sich jetzt die Frage stellen, was Sie mit diesen Werten an- oder ausgerichtet haben und welche Funktionalität sich dahinter verbirgt.

Grundwerk 10/2004

7.6.3.2.1

Seite 2

Struktur der Subnetzmaske

Die Subnetzmaske ist eine 32 Bit lange Folge von binären Einsen und Nullen. Aufgrund der besseren Lesbarkeit hat man diese Binärfolge in vier so genannte Oktetts unterteilt, wobei jedes dieser Oktetts Werte in den Grenzen von 00000000 bis 111111111 enthalten kann. Diese Binärwerte werden dann durch Dezimalwerte in den Grenzen von 0 bis 255 (zumindest rein theoretisch) repräsentiert. Kommt Ihnen das bekannt vor? Eine IP-Adresse ist vom Aufbau her identisch, aber nicht von der Funktion! Um es klarzustellen: Eine Subnetzmaske ist keine Adresse. Aber was ist sie dann?

7.6.3.2.2

Seite 2

Funktion der Subnetzmaske

Eine Subnetzmaske ist eine Schablone, die als Binärfolge über die Binärfolge der IP-Adresse gelegt wird und definiert, welcher Teil der IP-Adresse als Net-ID ausgewertet wird und welcher Teil der IP-Adresse als Host-ID betrachtet wird. Aber Halt! Es wird doch auch durch die Adressklasse definiert, wie eine IP-Adresse strukturiert ist. Stimmt, aber durch die Historie von TCP/IP hat es sich so entwickelt, dass wir zwei Verfahren haben, die bestimmen, wie eine IP-Adresse hinsichtlich der Verteilung von Net-ID und Host-ID interpretiert wird. Beide Verfahren sind notwendig und bauen aufeinander auf. Ich kann also nicht sagen: „Ich arbeite nur mit Adressklassen; Subnetzmasken interessieren mich nicht."

Da wir jetzt wissen, was eine Subnetzmaske macht, stellt sich als Nächstes die Frage, wie sie ihre Arbeit ausführt. Das wollen wir uns an einem Beispiel verdeutlichen. Gehen wir einfach davon aus, dass ich für die Ethernet-Karte meiner iSeries die Class B IP-Adresse 172.16.0.1. vergebe. Diese Adresse müssen wir uns jetzt als Binärfolge vorstellen:

IP-Adresse (dezimal)	172	16	0	1
IP-Adresse (binär)	10101100	00010000	00000000	00000001

Da wir jetzt für unsere IP-Adresse eine Subnetzmaske zu vergeben haben, werden wir uns zuerst einmal Adressklassen-konform verhalten. Die in der Tabelle dargestellte IP-Adresse gehört der Adressklasse B an. Um eine für die Adressklasse konforme Subnetzmaske zu vergeben, muss ich wie folgt vorgehen:

IP-Adresse (dezimal)	172	16	0	1
IP-Adresse (binär)	10101100	00010000	00000000	00000001
Subnetzmaske (dezimal)	255	255	0	0
Subnetzmaske (binär)	11111111	11111111	00000000	00000000

Um den Vorgang besser verdeutlichen zu können, sollten Sie jetzt nur die Binärfolgen der IP-Adresse und der Subnetzmaske betrachten:

IP-Adresse (binär)	10101100	00010000	00000000	00000001
Subnetzmaske (binär)	11111111	11111111	00000000	00000000

Ich hatte gesagt, dass die Subnetzmaske eine Schablone ist, die über die IP-Adresse gelegt wird. Folgendes wird jetzt realisiert: Steht ein Bit der Subnetzmaske auf 1, so wird das „darunter liegende" Bit der IP-Adresse als Net-ID interpretiert. Was bedeutet das für unser Beispiel? Jedes Bit des ersten Bytes der IP-Adresse und jedes Bit des zweiten Bytes der IP-Adresse wird als Net-ID ausgewertet. Das ist doch die gleiche Aussage, die mittels der Adressklasse B

7.6.3.2.3
Seite 2

ebenfalls formuliert wird! Richtig! Wir wollten ja auch eine der Adressklasse B konforme Subnetzmaske vergeben.

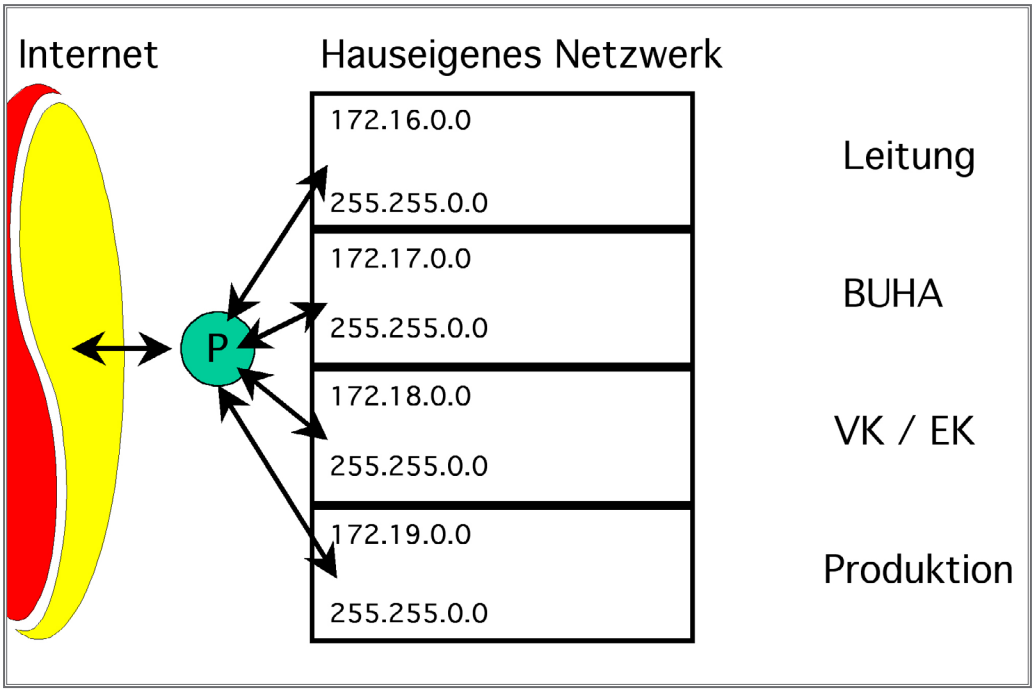
Bitte beachten Sie:

Eine Subnetzmaske muss immer mit einer geschlossenen Folge von binären Einsen beginnen. Steht ein Bit der Subnetzmaske auf Null, so müssen alle folgenden Bits ebenfalls binär Null sein.

Sie werden sich jetzt sicherlich sagen: „Na gut, wenn es denn so ist, das es zwei Verfahren gibt, die ich benutzen muss, dann mache ich mir das Leben einfach und verwende nur Adressklassen-konforme Subnetzmasken." In kleinen, relativ übersichtlichen TCP/IP-Netzwerken dürften Sie auch mit dieser Haltung zum Erfolg kommen. Werden die Netzwerke größer und komplexer, so wird dieser Ansatz nicht mehr funktionieren.

1. Beispiel-Netzwerk

Nehmen wir einmal an, Sie haben folgendes Netzwerk aufzubauen:



Das hauseigene Netzwerk

Aufgabe war es, ein TCP/IP-Netzwerk einzurichten. Jede Abteilung sollte aber einen eigenen Bereich besitzen, so dass nicht alle Clients aller Abteilungen überall adressierungsfähig sein sollten. Wir sehen das hauseigene Netzwerk dargestellt durch den viergeteilten Kasten in der Mitte des Bildes. Jede Ebene in diesem Kasten beschreibt eine Abteilung. Der Netzwerkdesigner hat jetzt unter der Prämisse gearbeitet, dass ausschließlich Adressklassen-konforme Subnetzmasken vergeben werden; er hat deshalb für jede Abteilung ein eigenes logisches Netzwerk aufgebaut. Die Geschäftsleitung hat das Netzwerk 172.16.0.0, die Buchhaltung das Netzwerk 172.17.0.0, der Verkauf/Einkauf das Netzwerk 172.18.0.0 und die Produktion das Netzwerk 172.19.0.0. Soll nun die Geschäftleitung in Bereiche der Buchhaltung sehen, so sind entsprechend den Anforderungen so genannte Router aufzubauen. Router sind im TCP/IP-Netzwerk Hosts in einem logischen TCP/IP-Netzwerk, die die Adressen von Hosts in einem zweiten logischen TCP/IP-Netzwerk kennen. Über Router werden wir uns später noch unterhalten müssen. Damit ist der Übergang zwischen den logischen Netzwerken geregelt. Soweit könnte nun alles gut sein; ich bin davon überzeugt, dass das Netzwerk auch gut funktionieren wird.

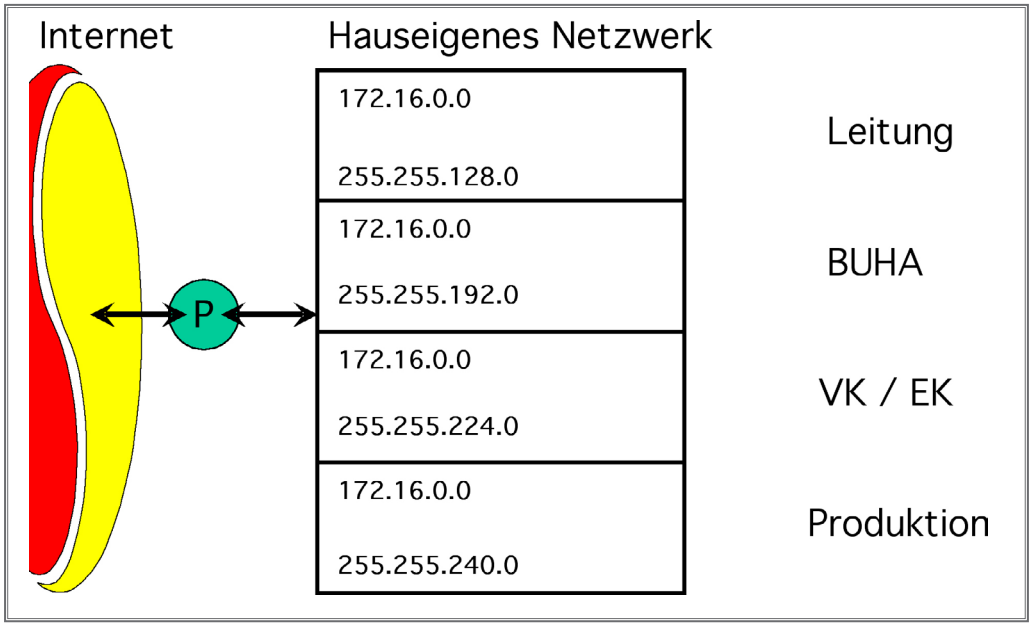
Jetzt wird an den Netzwerkadministrator eine weitere Anforderung herangetragen: „Wir müssen ins Internet“. Da nicht jeder, der ins Internet will, so ohne weiteres dort auch hineinkommt, sondern einen Provider (P) braucht, wird dieser kontaktiert. Der Provider analysiert das Netzwerk und stellt fest,

7.6.3.2.4
Seite 2

dass Sie insgesamt vier Netzwerke haben: 172.16.0.0, 172.17.0.0, 172.18.0.0 und 172.19.0.0. Sollen also Teilnehmer aus allen vier Netzwerken ins Internet, brauchen Sie vier Internet-Zugänge. (TCP/IP-Profis werden jetzt sicherlich sagen, dass man an dieser Stelle diese Problematik abfangen könnte. Sie haben Recht. Ich möchte nur aus Gründen der Einprägsamkeit das Umfeld, so wie es dargestellt ist, sehen.) Sicherlich freut sich der Provider, dass er vier Internetzugänge verkaufen kann, den Netzwerkadministrator und seine Firma freut es mit Sicherheit nicht.

2. Beispiel-Netzwerk

Wie können wir diese Problematik unterlaufen? Dadurch, dass wir uns den Subnetzmasken anvertrauen und unser Netzwerk modifizieren. Die von uns eingesetzte Subnetzmasken-Technik wird erreichen, dass wir nach außen hin – also zum Internet – als ein Netzwerk erscheinen, aber nach innen als vier Netzwerke.



Modifiziertes Netzwerk

Sie sehen im Bild, was der Netzwerkadministrator geändert hat. Statt mit vier logischen Netzwerken zu arbeiten, wird im gesamten hauseigenen Netzwerk mit einem logischen Netzwerk gearbeitet. (Alle Abteilungen sind Bestandteil des Netzwerks 172.16.0.0.) Damit im hauseigenen Netzwerk jedoch vier so genannte Subnetze existieren, hat der Administrator die Werte für die Subnetzmasken geändert.

IP-Adresse (dezimal)	172	16	0	1
IP-Adresse (binär)	10101100	00010000	00000000	00000001
1. Subnetz (dez.)	255	255	128	0
1. Subnetz (bin)	11111111	11111111	10000000	00000000
2. Subnetz (dez.)	255	255	192	0
2. Subnetz (bin)	11111111	11111111	11000000	00000000
3. Subnetz (dez.)	255	255	224	0
3. Subnetz (bin)	11111111	11111111	11100000	00000000
4. Subnetz (dez.)	255	255	224	0
4. Subnetz (bin)	11111111	11111111	11110000	00000000

7.6.3.2.5
Seite 2

Konzentrieren wir uns jetzt noch einmal auf die Binärfolgen der Subnetzmasken:

IP-Adresse (binär)	10101100	00010000	00000000	00000001
1. Subnetz (bin)	11111111	11111111	10000000	00000000
2. Subnetz (bin)	11111111	11111111	11000000	00000000
3. Subnetz (bin)	11111111	11111111	11100000	00000000
4. Subnetz (bin)	11111111	11111111	11110000	00000000

Jetzt sehen wir, was passiert ist. Im ersten Subnetz (Leitung) werden nicht nur die ersten beiden Bytes als Net-ID aufgelöst, sondern zusätzlich wird auch noch das erste Bit des dritten Bytes der IP-Adresse als Net-ID ausgewertet. Im zweiten Subnetz wird das nächste Bit, im dritten Subnetz ein weiteres Bit und im vierten Subnetz das vierte Bit mit zur Net-ID hinzugenommen.

Der Übergang zwischen den einzelnen Abteilungs-Subnetzen wird dann wieder durch entsprechende Router geregelt.

Soweit, so gut. Wo liegt die Einschränkung dieser Lösung? Gehen wir einfach einmal von der Tatsache aus, dass das Netzwerk erweitert werden muss. Weitere Abteilungen oder Teile bestimmter Abteilungen sollen wieder jeweils isolierte Netzwerkbereiche bekommen. So ergibt sich, dass ich maximal nur acht (8) Subnetze mit Hilfe dieses Lösungsansatzes realisieren kann. Wieso? Ganz einfach: Da ja für die Subnetzmasken die Regel besteht, dass eine Subnetzmaske mit einer geschlossenen Folge von binären Einsen beginnen muss und dann an einem definierten Punkt auf eine geschlossenen Folge von binären Nullen „umkippen“ muss, so ergibt sich, dass wir bei diesem Lösungsweg lediglich das dritte Byte der IP-Adresse als Netz-ID mit auswerten dürfen. Das heißt dann folgerichtig, dass im dritten Byte meiner IP-Adresse nur folgende Werte möglich sind:

3. Byte der Subnetzmaske (binär)	3. Byte der Subnetzmaske (dezimal)
10000000	128
11000000	192
11100000	224
11110000	240
11111000	248
11111100	252
11111110	254
11111111	255

Wenn ich mir diesen Sachverhalt vor Augen halte, muss ich mir auch eingestehen, dass ich nicht mehr als acht (8) Subnetze bilden kann. Und das kann sicherlich nicht der Weisheit letzter Schluss sein – oder?

3. Beispiel-Netzwerk

Die Voraussetzungen für dieses Netzwerk sind identisch mit denen der bereits besprochenen. Ich möchte nur jetzt eine Netzwerkstruktur erzeugen, die eine flexiblere Lösung anbietet. Ich vergebe also folgende IP-Adressen und Subnetzmasken:

Abteilung	IP-Adresse	Subnetzmaske
Leitung	172.16.0.0	255.255.224
BUHA	172.16.0.0	255.255.224
VK/EK	172.16.0.0	255.255.224
Produktion	172.16.0.0	255.255.224

Was ist das? Ich arbeite lediglich mit einer Subnetzmaske. Aber sind meine Adressen dann nicht Mitglieder ein und desselben Netzwerks? Das hängt jetzt von der Vergabe der Host-ID in den IP-Adressen ab. Lassen Sie uns das einmal genauer betrachten: Da ich im dritten Byte meiner Subnetzmaske den Wert 224 notiert habe, werden die ersten drei Bits des dritten Bytes der IP-Adresse zusätzlich als Bestandteil der Net-ID mit ausgewertet. Ich muss jetzt für dieses Byte der IP-Adresse Dezimalwerte finden, die in den ersten drei Bits eine identische Struktur aufweisen. Diese stellen dann die Hosts eines Subnetzes dar. Betrachten wir folgende Tabelle:

Bit 1-3 der Subnetzmaske	Byte 3 der IP-Adresse (dezimal)		Byte 3 der IP-Adresse (binär)
000	Erster Host:	0	00000000
	Letzter Host:	31	00011111
001	Erster Host	32	00100000
	Letzter Host:	63	00111111
010	Erster Host:	64	01000000
	Letzter Host	95	01011111
011	Erster Host	96	01100000
	Letzter Host	127	01111111
100	Erster Host	128	10000000
	Letzter Host	159	10011111
101	Erster Host	160	10100000
	Letzter Host	191	10111111
110	Erster Host	192	11000000
	Letzter Host	223	11011111
111	Erster Host	224	11100000
	Letzter Host	255	11111111

7.6.3.2.6
Seite 2

Betrachten Sie die Tabelle genau! Trage ich in das dritte Byte meiner IP-Adresse Dezimalwerte in den Grenzen von 1 bis 31 ein, so sind alle diese Werte, wenn man einmal ihre binäre Repräsentation betrachtet, in den ersten drei Bits identisch. Daraus folgt, dass mein erstes Subnetz folgenden Adressbereich besitzt:

1. Subnetz	172.16. 0.1 bis 172.16.31.254

Für die weiteren Subnetze gilt also analog:

2. Subnetz	172.16. 32.1 bis 172.16.63.254
3. Subnetz	172.16.64.1 bis 172.16.95.254
4. Subnetz	172.16.96.1 bis 172.16.127.254
5. Subnetz	172.16.128.1 bis 172.16.159.254
6. Subnetz	172.16.160.1 bis 172.16.191.254
7.Subnetz	172.16.192.1 bis 172.16.223.254
8. Subnetz	172.16.224.1 bis 172.16.255.254

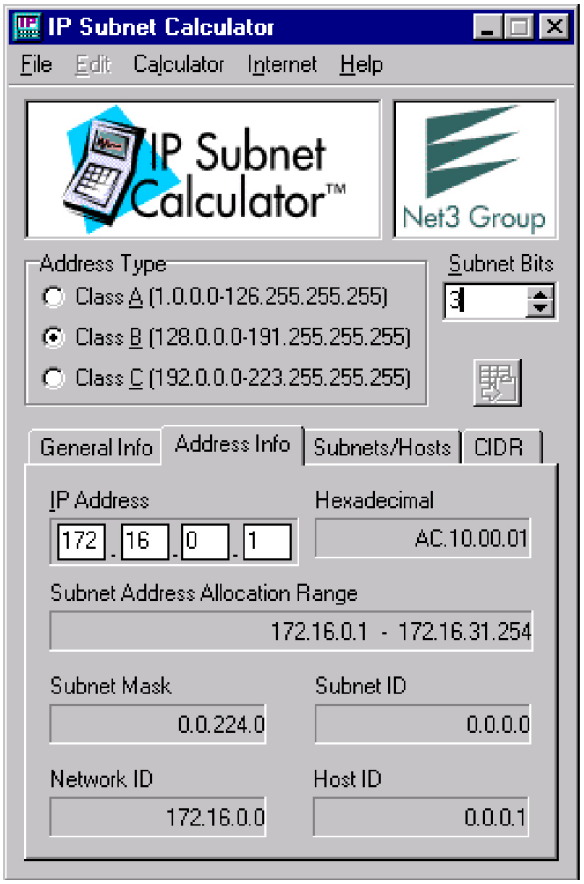
Die Schwierigkeit dieses Ansatzes besteht darin, dass wir mit dezimalen Werten arbeiten, uns aber immer wieder deren binäre Repräsentation vergegenwärtigen müssen – und das ist nicht immer ganz leicht. Belohnt werden wir aber bei diesem Ansatz mit einer flexibleren Netzwerkstruktur.

Sie werden sich jetzt sicherlich fragen: „Muss dass denn jetzt sein?“ In vielen Fällen ist es sicherlich nicht unbedingt notwendig, mit Subnetzmasken zu arbeiten – bei kleinen, einfach strukturierten Netzwerken. In den meisten Fällen wird man jedoch durch die Umstände der Unternehmen dazu gezwungen, in diesen relativ komplexen Bereichen zu arbeiten. Deshalb möchte ich Ihnen noch eine kleine Software empfehlen, die als Freeware im Internet zu beziehen ist.

Bei dieser Software handelt es sich um ein IP-Subnetz-Berechnungsprogramm. Planen Sie ein TCP/IP-Netzwerk, so berechnet Ihnen dieses Programm für die von Ihnen gewählte Netzwerkklasse bei einer gewählten Subnetzmaske die Subnetzgrößen und die jeweiligen IP-Adressbereiche.

Subnetzrechner finden Sie im Internet in großer Anzahl. Suchen Sie einfach mal mittels „Google“ nach dem Begriff „Subnet Calculator“ (Sie finden das Programm „IPSubnetCalc321.exe“ auch auf der beiliegenden CD im Ordner „Tools“). Sie erhalten dann ein Setup-Programm, das Ihnen dabei hilft, die Software zu installieren. Das ist ein ganz einfacher Standard-Windows-Setup-Dialog, der von Ihnen keine schwerwiegenden Angaben erfordert. Danach steht das Programm im Dialog „Start-Programme“ zur Verfügung.

Hier sehen Sie das Register „Address Info“ des „IP Subnet Calculator“:

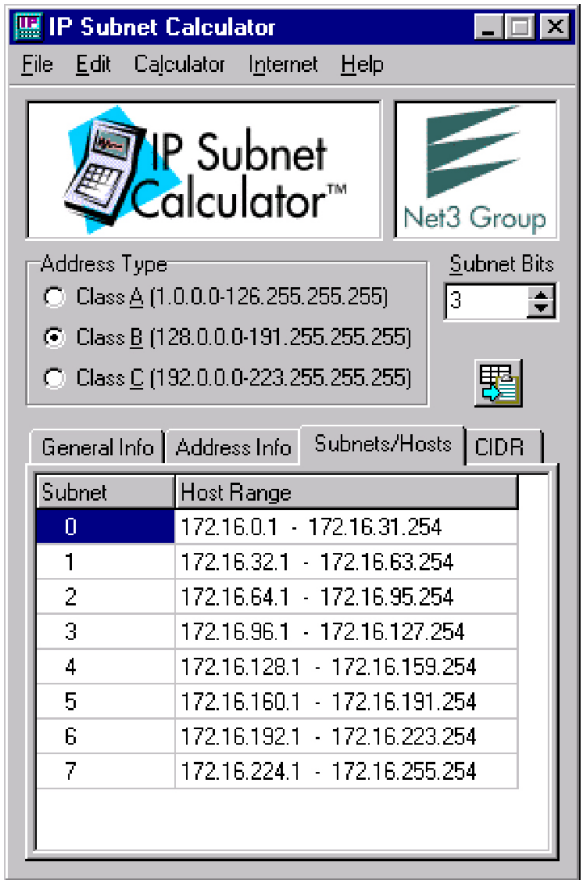


Startadresse des Netzwerks eintragen

Für die Berechnung unseres Beispiel-Netzwerkes habe ich die Adressklasse „Class B“ markiert und in das Feld „IP-Adress“ die Start-Adresse des TCP/IP-Netzwerks eingetragen. In das Feld „Subnet Bits“ habe ich dann noch den Wert „drei“ (3) notiert. Was bedeutet das? Meine projektierte Subnetzmaske heißt 255.255.224.0. Im dritten Byte sehen Sie den Wert 224. Dieser Wert fragt die ersten drei (3) Bits des dritten Bytes der IP-Adresse als Net-ID und somit muss ich also in dieses Feld den Wert „drei“ (3) eintragen.

Da mich die Subnetzstruktur und die Adressierungsbereiche der einzelnen Subnetze interessieren, wechsele ich in das Register „Subnets/Hosts“.

7.6.3.2.6
Seite 4



Register „Subnets/Hosts“

Sie sehen: Was wir uns vorhin mühselig durch Dezimal-zu-Binär-Umrechnungen erarbeitet haben, wird uns hier in Sekundenschnelle automatisch serviert. Wenn das keine Arbeitserleichterung ist…

Mit diesem theoretischen Wissen ausgestattet sollten Sie in der Lage sein, für Ihr Netzwerk IP-Adressen bereitzustellen.

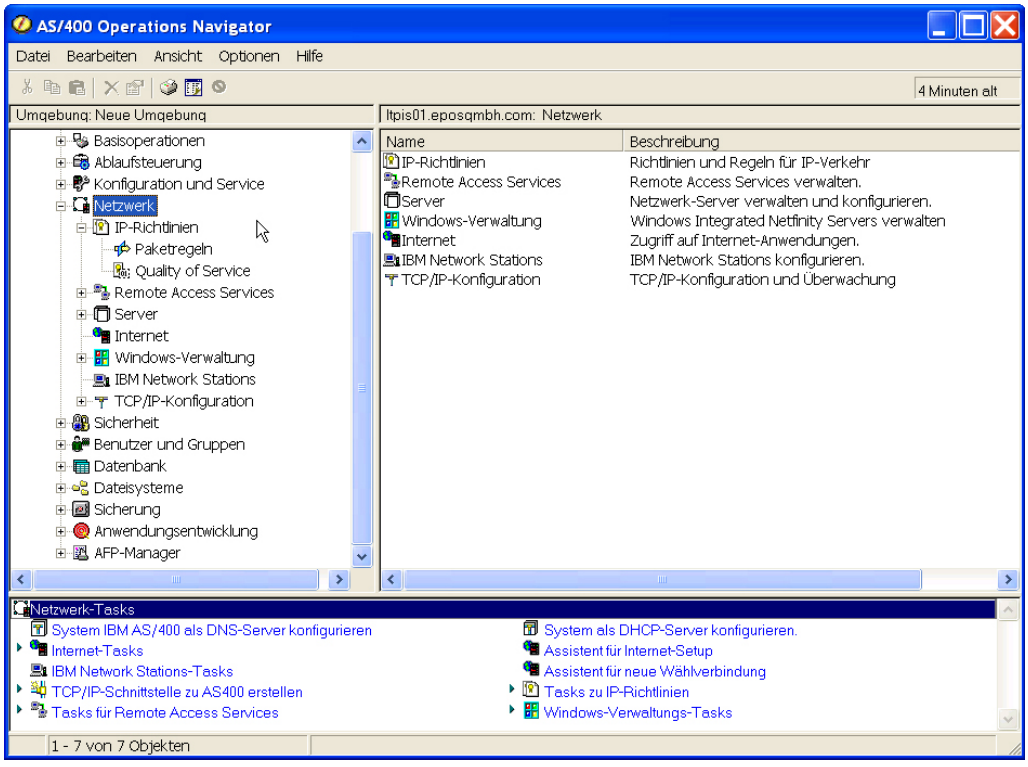
Die iSeries TCP/IP-Konfiguration

Seit OS/400 Version 3 Release 1 werden die TCP/IP-Protokollstapel als integriertes Bestandteil des Betriebssystems ausgeliefert. Dies bedeutet jedoch nicht, dass nicht vorher schon ein TCP/IP für die iSeries verfügbar gewesen wäre. Der Unterschied zur aktuellen TCP/IP-Version besteht darin, dass die Alt-Version ausschließlich als Lizenzprogramm zur Verfügung stand und damit natürlich kostenpflichtig war, und dass das TCP/IP im User Interface des Betriebssystems funktional tätig war. Das führte dazu, dass TCP/IP-Verbindungen weniger performand waren als SNA-Verbindungen. Insgesamt führten diese Gründe dazu, dass TCP/IP-Verbindungen auf der iSeries nicht sonderlich beliebt waren.

Die aktuelle TCP/IP-Implementierung ist von den eben genannten Merkmalen frei. TCP/IP ist in das Betriebssystem integriert. D.h., dass seine Funktionen weitgehend in den Mikrocode des i5/OS eingebracht wurden, was zur Stabilität und Schnelligkeit dieser Verbindungsmöglichkeit erheblich beitrug.

Auch die Bedienbarkeit – sprich die Administration der TCP/IP-Schnittstellen – wurde im Laufe der Jahre erheblich verbessert, indem der iSeries Navigator zur zentralen Administrationsschnittstelle wurde.

Die Komponente „Netzwerk“ im iSeries Navigator bietet Ihnen eine Vielzahl von Funktionen, die der Administration der Netzwerkfunktionalitäten eines iSeries Servers dienen.



Netzwerkfunktionen

7.6.4
Seite 2

Dazu gehört das Arbeiten mit folgenden Funktionen:

- Arbeiten mit IP-Richtlinien
- Arbeiten mit Ein- und Auswahlverbindungen (Remote Access Service)
- Arbeiten mit TCP/IP und Host-Servern
- Arbeiten mit Web-Servern und digitalen Zertifikaten
- Arbeiten mit Integrated X-Servern und -Adaptern
- Arbeiten mit IBM Network Stations
- Arbeiten mit der TCP/IP-Konfiguration

Sie finden die Administrationsfunktionen im iSeries Navigator im Pfad:

Umgebung/Verbindung/Netzwerk/TCP/IP-Konfiguration

Lokale Netzwerke

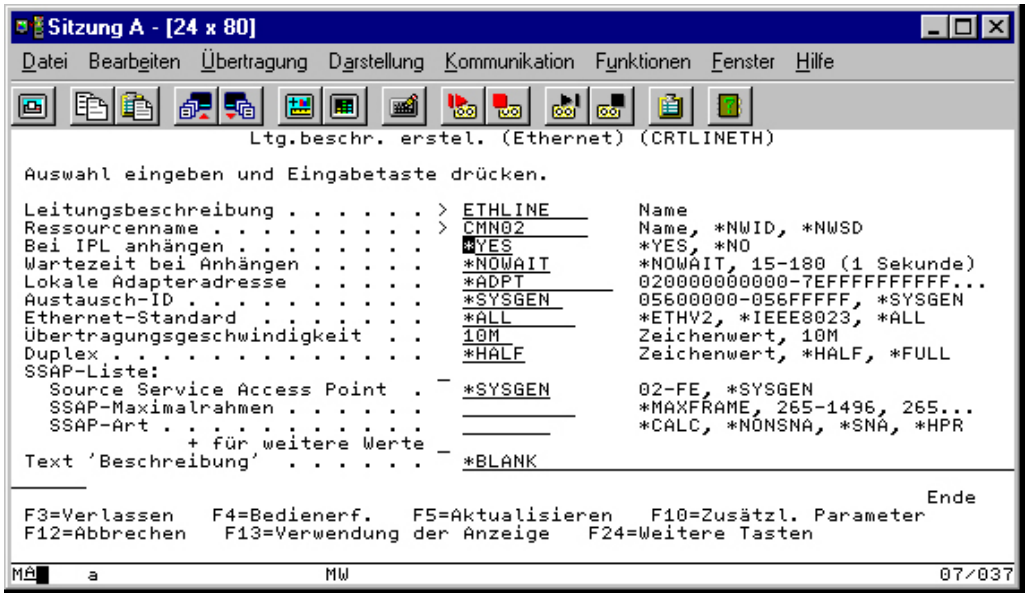
Voraussetzung für eine lokale Verbindung mit Hilfe des TCP/IP-Protokolls ist das Vorhandensein einer physischen Leitung: Ethernet oder Token Ring.

7.6.4.1

Seite 2

Ethernet-Netzwerke

Im Folgenden sehen Sie eine typische Ethernet-Leitungsbeschreibung. Welche Parameter sind wichtig?



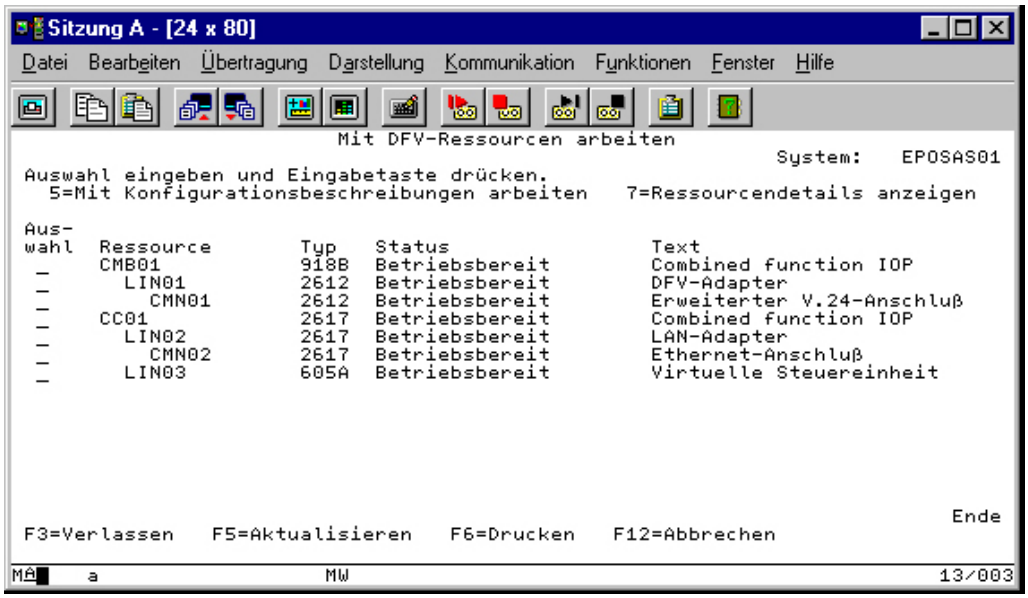
Ethernet-Leitungsbeschreibung

Name der Leitung:

Frei wählbar.

Ressourcenname:

Mit Hilfe der Anzeige des Befehls WRKHDWRSC (Mit Hardwareressourcen arbeiten) ermitteln Sie den Ressourcennamen der LAN-Adapterkarte.



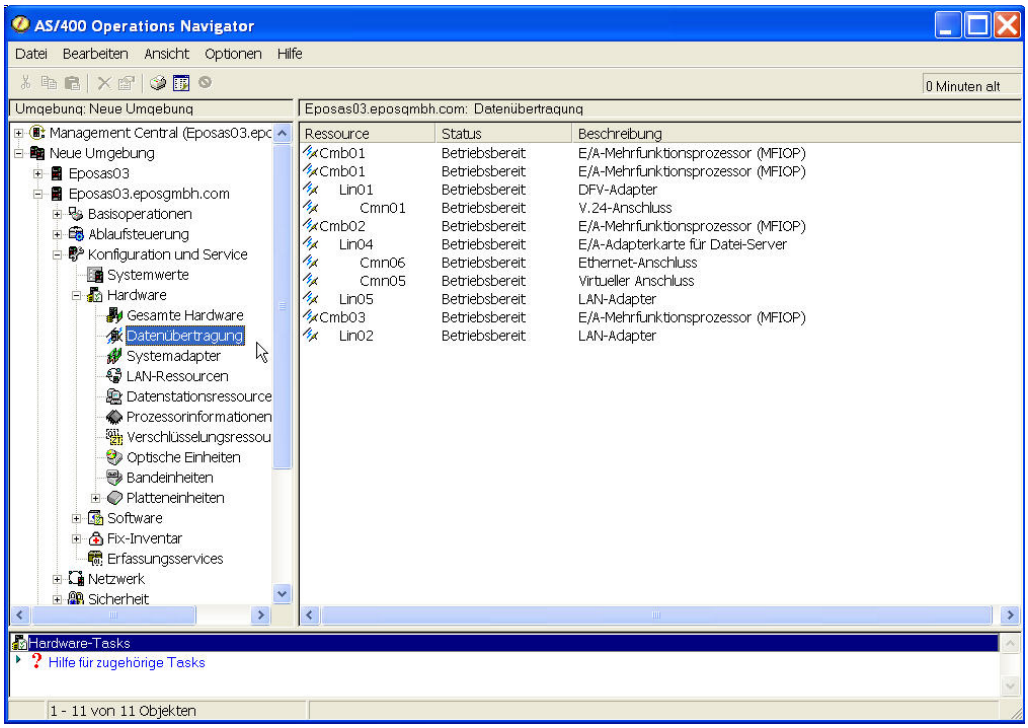
*Anzeige des Befehls WRKHDWRSC *CMN*

7.6.4.1.1
Seite 2

Das Auffinden der Ressourcennamen ist mithilfe des Operations Navigator ebenfalls möglich. Dazu rufen Sie dieses Programm auf und gehen in den Pfad:

Umgebung/Verbindung/Konfiguration und Service/Hardware/Datenübertragung

Auf der rechten Seite des Operations Navigator sehen Sie dann folgenden Dialog:



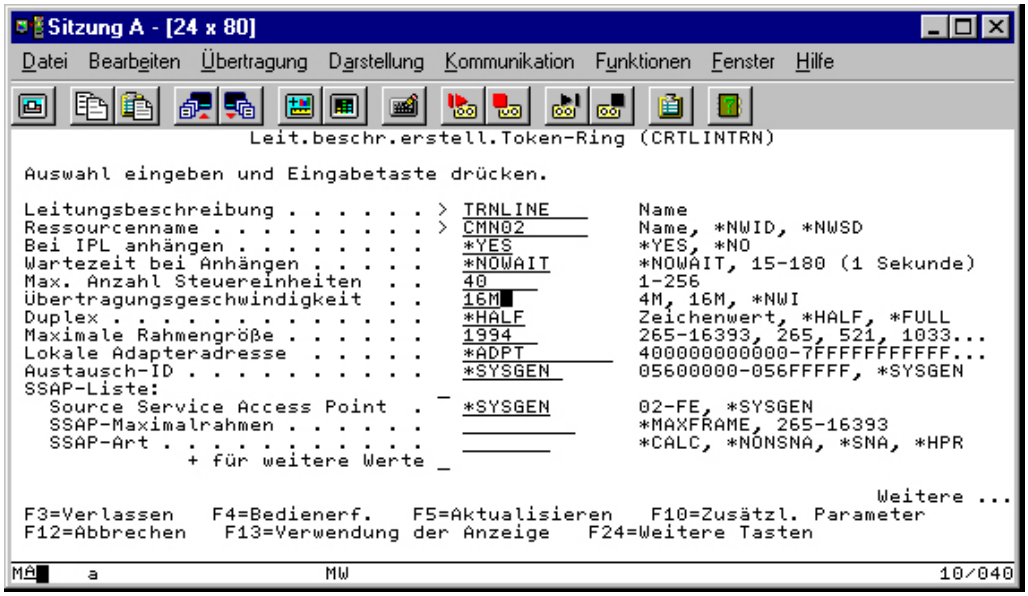
Hardware-Ressourcen mit Operations Navigator finden

Adapteradresse:

Die Adapteradresse wird über den Einsetzwert *SYSGEN vom System ermittelt.

Token Ring Netzwerke

Im Folgenden sehen Sie eine typische Token Ring-Leitungsbeschreibung. Welche Parameter sind wichtig?



Token Ring-Leitungsbeschreibung

Name der Leitung

Frei wählbar

Ressourcenname

Siehe Ethernet-Adapter!

Adapteradresse

Siehe Ethernet-Adapter!

7.6.4.1.2

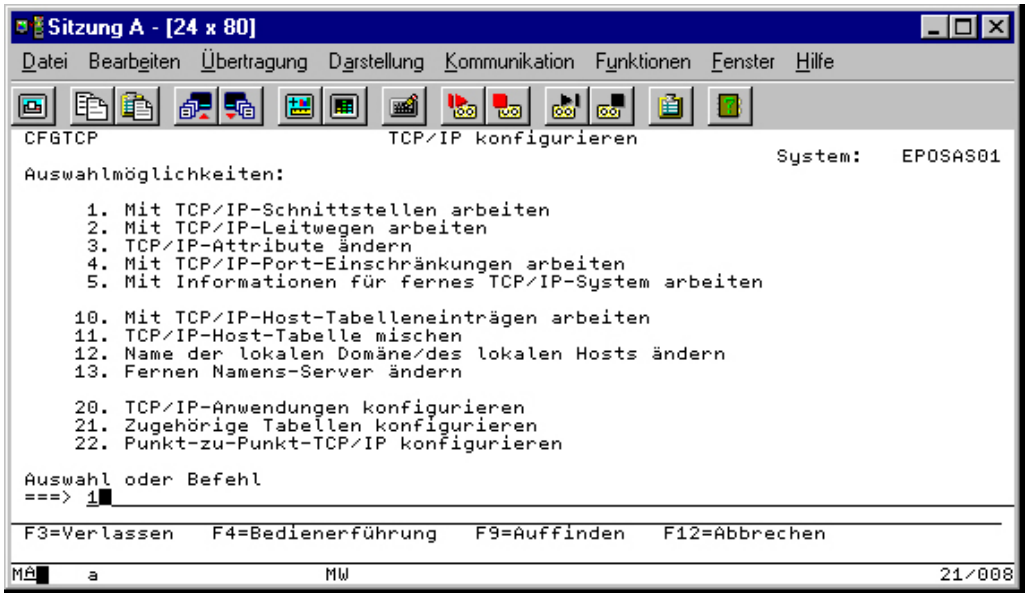
Seite 2

Erstellung einer TCP/IP-Schnittstelle

Die gesamte Konfiguration der TCP/IP-Funktionen kann aus dem iSeries-Menü CFGTCP heraus erfolgen. Dieses Menü erreichen Sie auf zwei unterschiedliche Arten:

1. Geben Sie auf einer beliebigen Befehlszeile den Befehl CFGTCP (TCP/IP konfigurieren) ein.
2. Alternativ können Sie auch den direkten Menü-Aufruf GO CFGTCP eingeben.

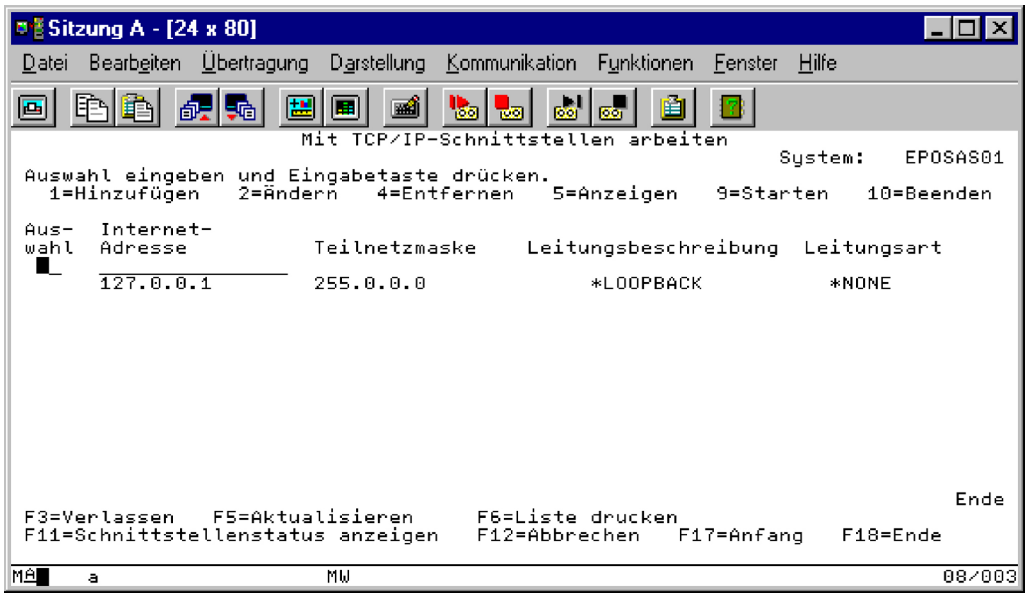
Welchen der beiden genannten Wege Sie gewählt haben, ist letztendlich egal. Sie erhalten auf alle Fälle den folgenden Bildschirm als Anzeige:



Menü CFGTCP

7.6.4.2
Seite 2

Wählen Sie jetzt den Menüpunkt 1 „Mit TCP/IP-Schnittstellen arbeiten“. Sie sehen daraufhin den folgenden Bildschirm:

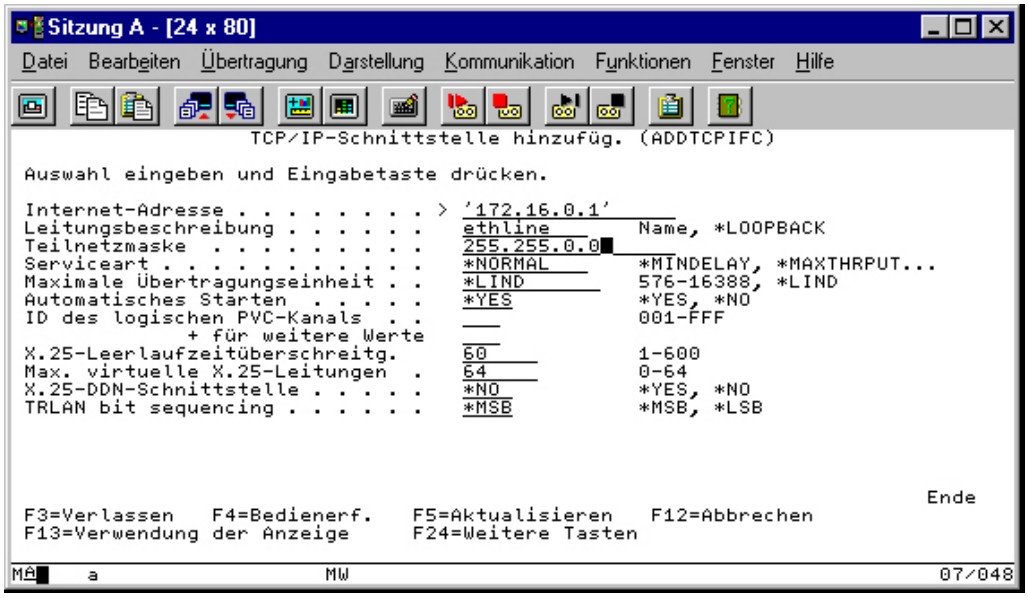


Mit TCP/IP-Schnittstellen arbeiten

Worin liegt der Sinn, eine TCP/IP-Schnittstelle zu definieren? Es geht darum, dass Sie Ihre iSeries in einem TCP/IP-Netzwerk adressierungsfähig machen. Dies erreichen Sie dadurch, dass Sie eine TCP/IP-Adresse an die Leitungsbeschreibung – Ethernet oder Token Ring – Ihrer iSeries binden.

Sie sehen, dass bereits eine Adresse eingetragen ist. Die Adresse 127.0.0.1 ist eine so genannte LOOPBACK-Adresse – eine interne Adresse zur Überprüfung der lokalen TCP/IP-Konfiguration. Wir können dies aber zu diesem Zeitpunkt noch nicht prüfen, denn wir müssen ja erst einmal eine TCP/IP-Schnittstelle erstellen, bevor wir überhaupt etwas zum Prüfen haben.

Geben Sie also „1=Hinzufügen“ im Auswahlfeld gefolgt von Ihrer selbst gewählten IP-Adresse ein; folgender Bildschirm erscheint:



TCP/IP Schnittstelle erstellen

IP-Adresse:

Hier sollte jetzt Ihre eingegebene IP-Adresse als Zeichenkette stehen. Im Bild sehen Sie die Zeichenfolge ‚172.16.0.1'. Aber warum wurde die Net-ID 172.16 gewählt? Aus dem Wertebereich ergibt sich, dass es sich hier um eine Class B-Adresse handelt.

Wie Sie sehen, habe ich mich aus dem Adressen-Pool privater Class B-Adressen bedient. Sollte mein Netzwerk zum Internet hin geöffnet werden, gibt mir das die Sicherheit, dass ich keine bereits im Internet vergebene Adresse verwende. Darüber hinaus haben die freigelassenen Adressen den Vorteil, dass an sie kein Provider seine Dienste direkt durchrouten kann.

Leitungsbeschreibung:

Als Leitungsbeschreibung geben Sie die von Ihnen ausgewählte Leitung ein, über die die TCP/IP-Dienste wahrgenommen werden sollen – in meinem Fall die Ethernet-Leitung ETHLINE.

Subnetzmaske:

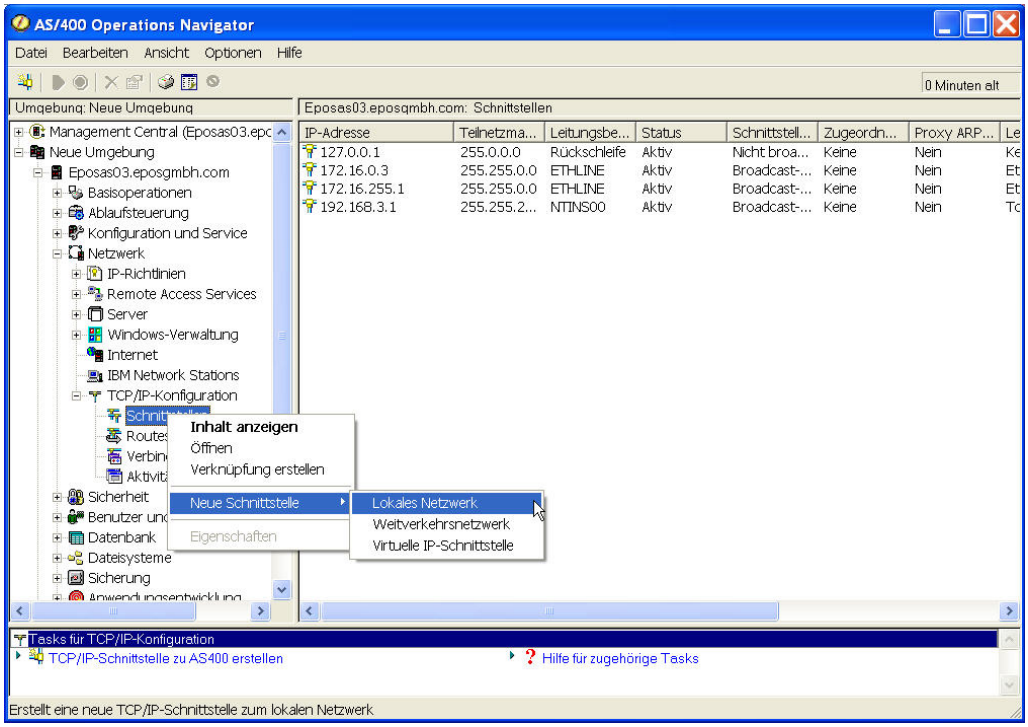
Tragen Sie hier den Wert 255.255.0.0 ein. Dieser Wert besagt, dass alle Bits des ersten und des zweiten Bytes der IP-Adresse komplett als Net-ID interpretiert werden.

Alternativ lassen sich auch mit dem Operations Navigator neue Schnittstellen erzeugen. Dazu gehen Sie in den Pfad:

Umgebung/Verbindung/Netzwerk/TCP/IP-Konfiguration/Schnittstellen

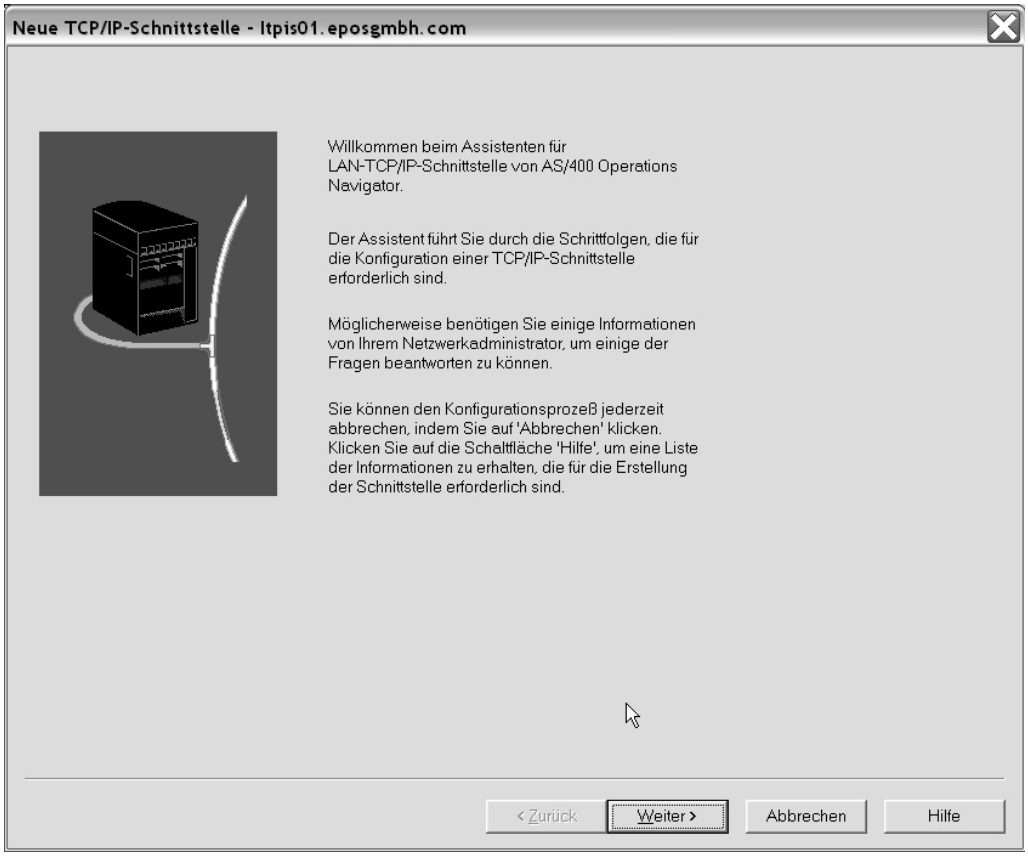
und markieren den Eintrag TCP/IP-Konfiguration mit der rechten Maustaste. Danach öffnet sich das Schnittstellen-Kontextmenü.

7.6.4.2
Seite 4



Eine neue Schnittstelle erzeugen

Im Kontextmenü wählen Sie dann „Neue Schnittstelle – Lokales Netzwerk“ an. Daraufhin wird der Wizzard für Neue Schnittstellen aufgerufen, der Sie dann durch den gesamten Prozess der IP-Schnittstellenerstellung führt.

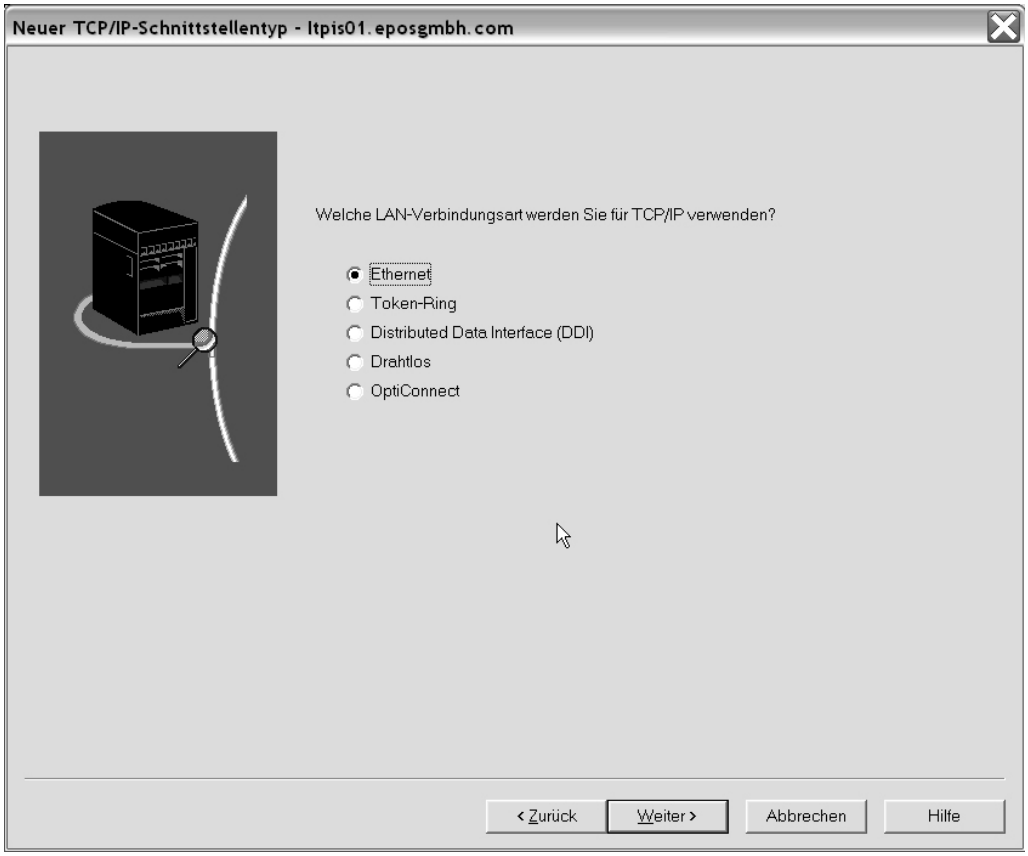


Willkommen!

Nach einem freundlichen Willkommens-Gruß, der Sie global in die Bedienung des Wizzard einführt, bestätige ich den Dialog mit „Weiter“.

Als Nächstes wird die Frage nach dem physikalischen Verbindungsmedium gestellt.

7.6.4.2
Seite 6

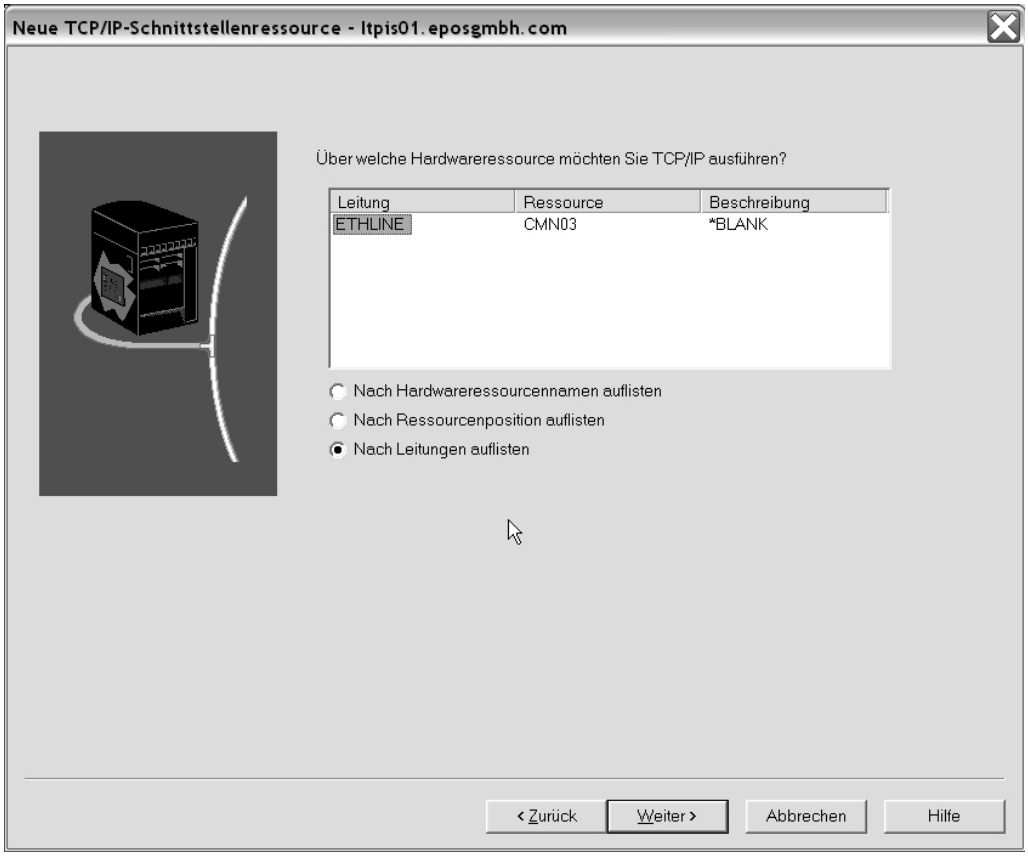


Definition der physikalischen Verbindung

Da in meinem LAN-Netzwerk Ethernet als physikalisches Protokoll verwendet wird, treffe ich in oben gezeigtem Dialog die entsprechende Auswahl.

Die Definition wird mit „Weiter“ bestätigt.

Der nächste Dialog ist dann schon spannender, da hier – sofern gewünscht – eine physische Schnittstelle (Leitungsbeschreibung) erstellt werden kann, die dann mit der logischen IP-Schnittstelle verbunden wird.



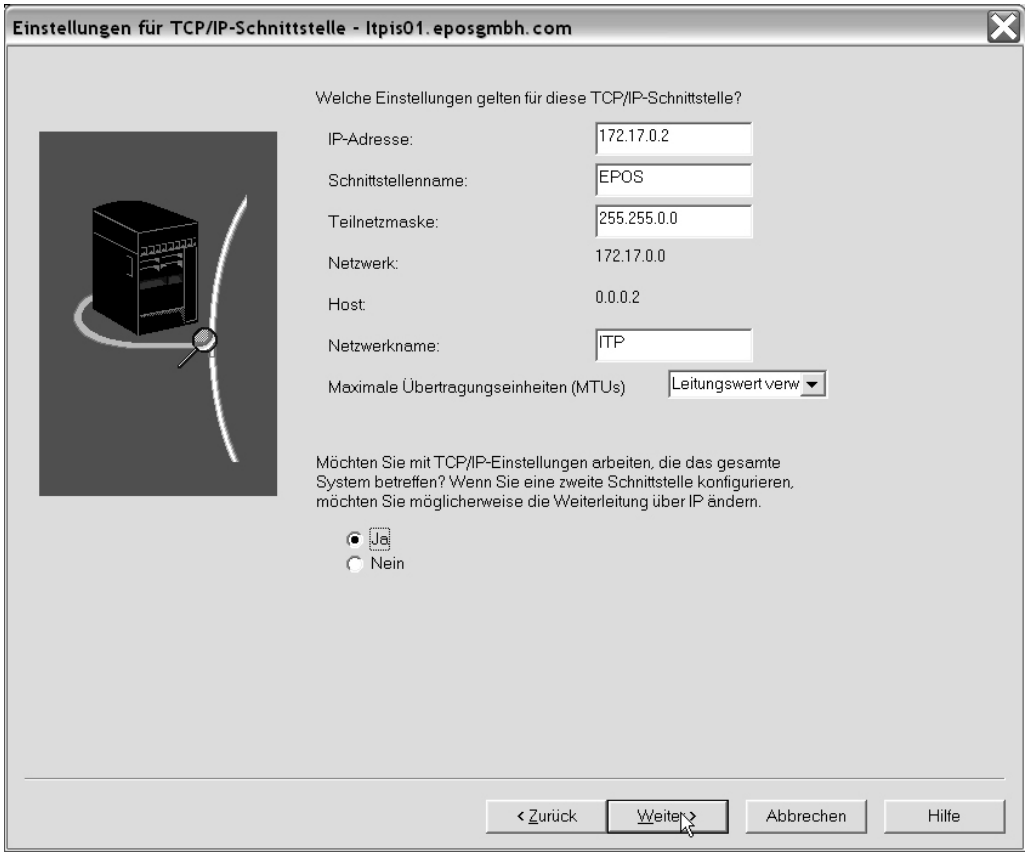
Mit physikalischen Schnittstellen arbeiten

Beabsichtigen Sie eine Leitungsbeschreibung zu erstellen, so markieren Sie den Parameter „Nach Hardwareressourcennamen auflisten“ oder „Nach Ressourcenposition auflisten“. Danach gelangen Sie in den Dialog für die Erstellung einer Leitungsbeschreibung.

Ich habe jedoch bereits eine Leitungsbeschreibung erstellt, so dass nur noch eine IP-Adresse für die bereits vorhandene Leitungsbeschreibung erzeugt werden muss.

Deshalb markiere ich den Parameter „Nach Leitungen auflisten“ und bestätige den Dialog mit „Weiter“.

7.6.4.2
Seite 8



Adressparameter versorgen

Im oben dargestellten Dialog sind jetzt die Adressparameter der IP-Schnittstelle zu bearbeiten. Dazu sind folgende Parameter zu versorgen:

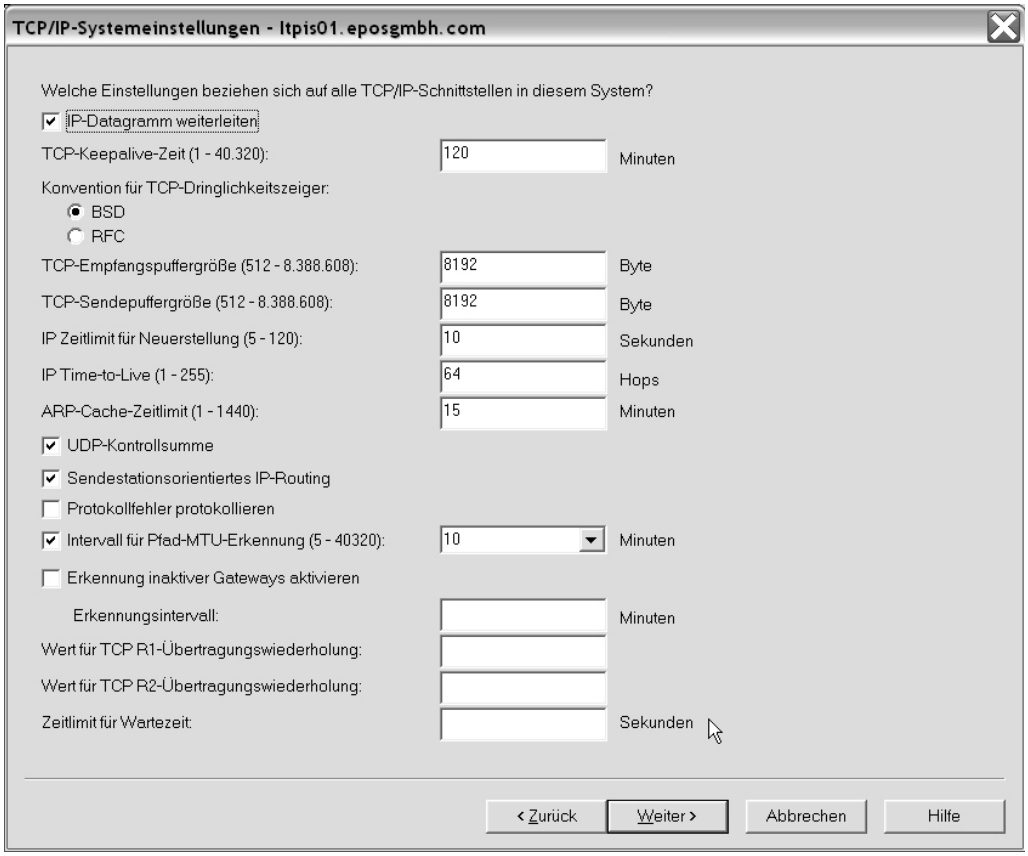
IP-Adresse: Hier trage ich die Adresse 172.16.0.2 ein.

Schnittstellenname: Das ist ein reiner Anzeigewert für die Darstellung im Operations Navigator. Der Wert besitzt keine Funktionalität im TCP/IP-Netzwerk.

Teilnetzmaske: Hier trage ich Adressklassen-konform 255.255.0.0 ein.

Netzwerkname: Das ist ein reiner Anzeigewert für die Darstellung im Operations Navigator. Der Wert besitzt keine Funktionalität im TCP/IP Netzwerk.

Im unteren Teil des Dialogs wird gefragt, ob die globalen Parameter des TCP/IP der iSeries bearbeitet werden sollen. Ich habe hier „Ja“ markiert.

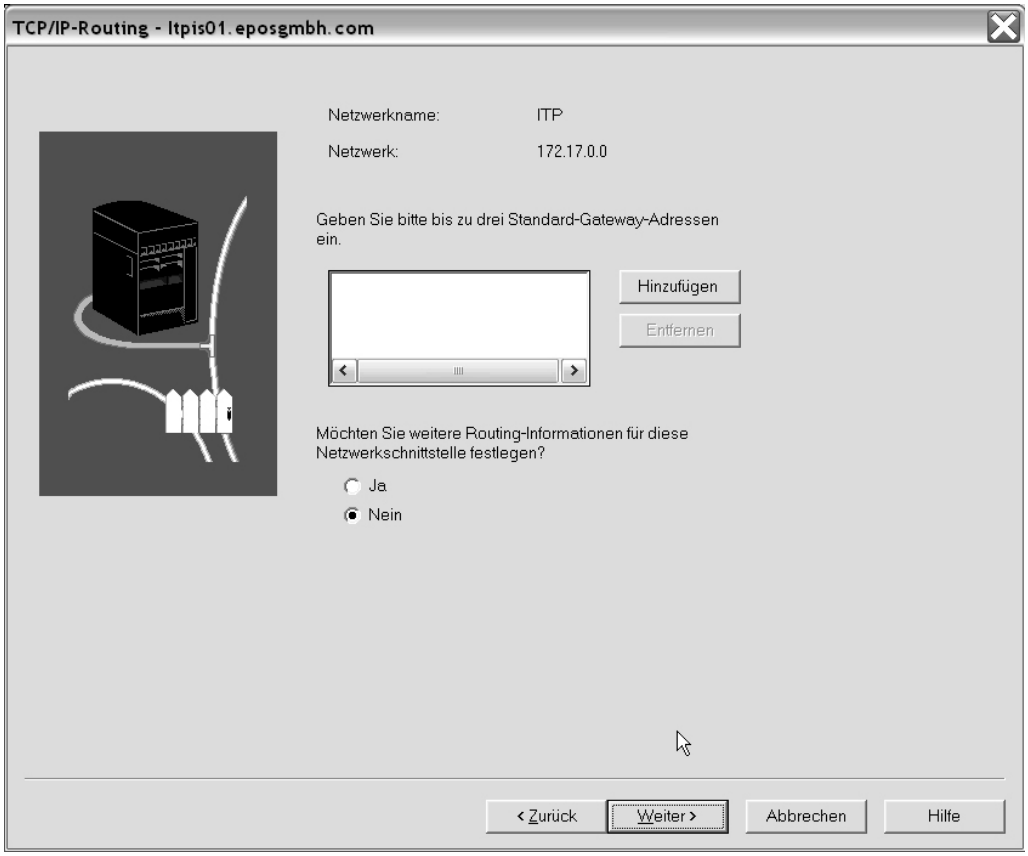


Globale TCP/IP-Parameter

Im Dialog für globale Parameter habe ich nichts verändert und belasse es somit bei den Standardeinstellungen.

Im nächsten Parameter wird nach Routing-Einträgen gefragt, die eventuell erzeugt werden sollen.

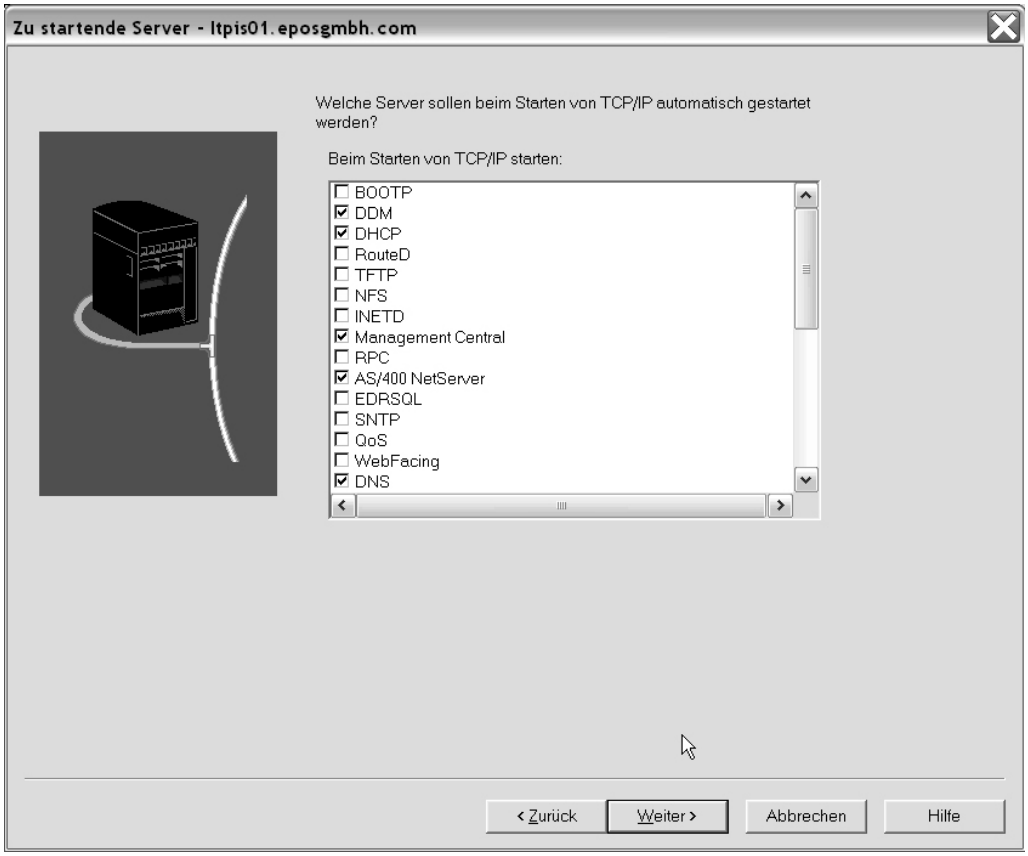
7.6.4.2
Seite 10



Routing-Einträge erzeugen

Auch in diesem Dialog belasse ich es bei den Default-Werten und trage nichts ein.

Danach können Sie einstellen, welche TCP/IP-Server-Dienste automatisch gestartet werden sollen.

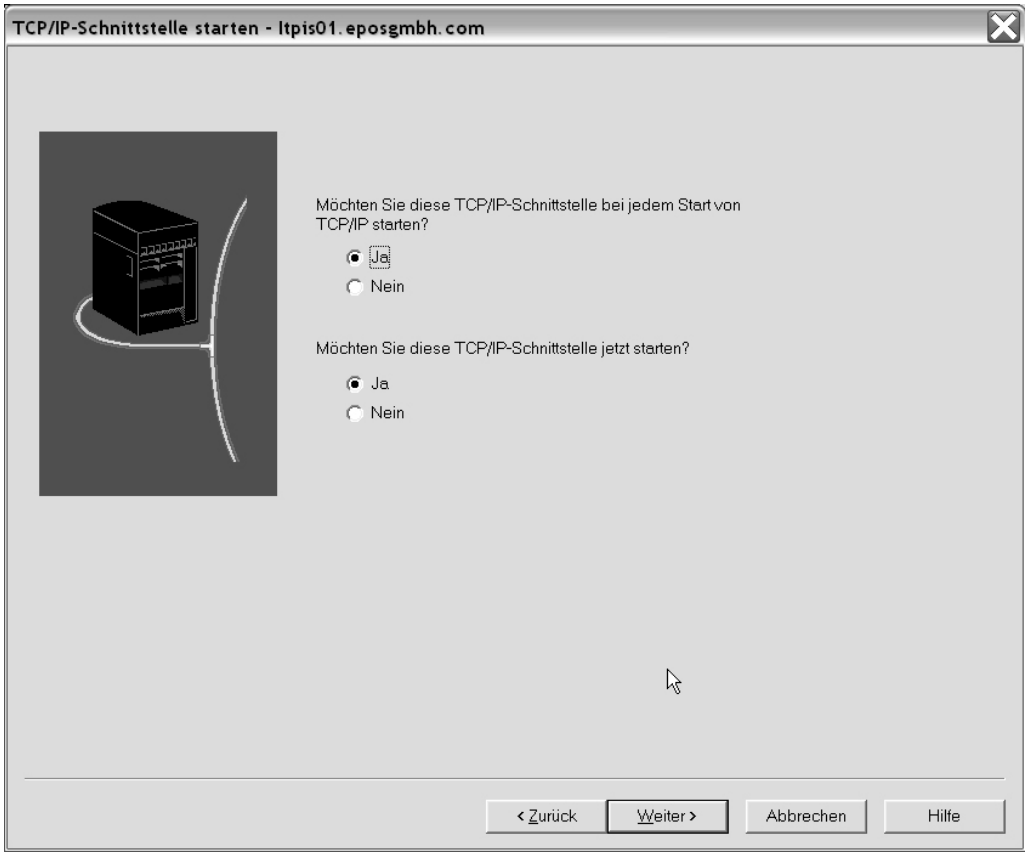


Zu startende Server

Auch hier belasse ich es bei den Standardeinstellungen, die für diesen Server bereits definiert wurden.

Im folgenden Dialog wird das Startverhalten der definierten Schnittstelle festgelegt.

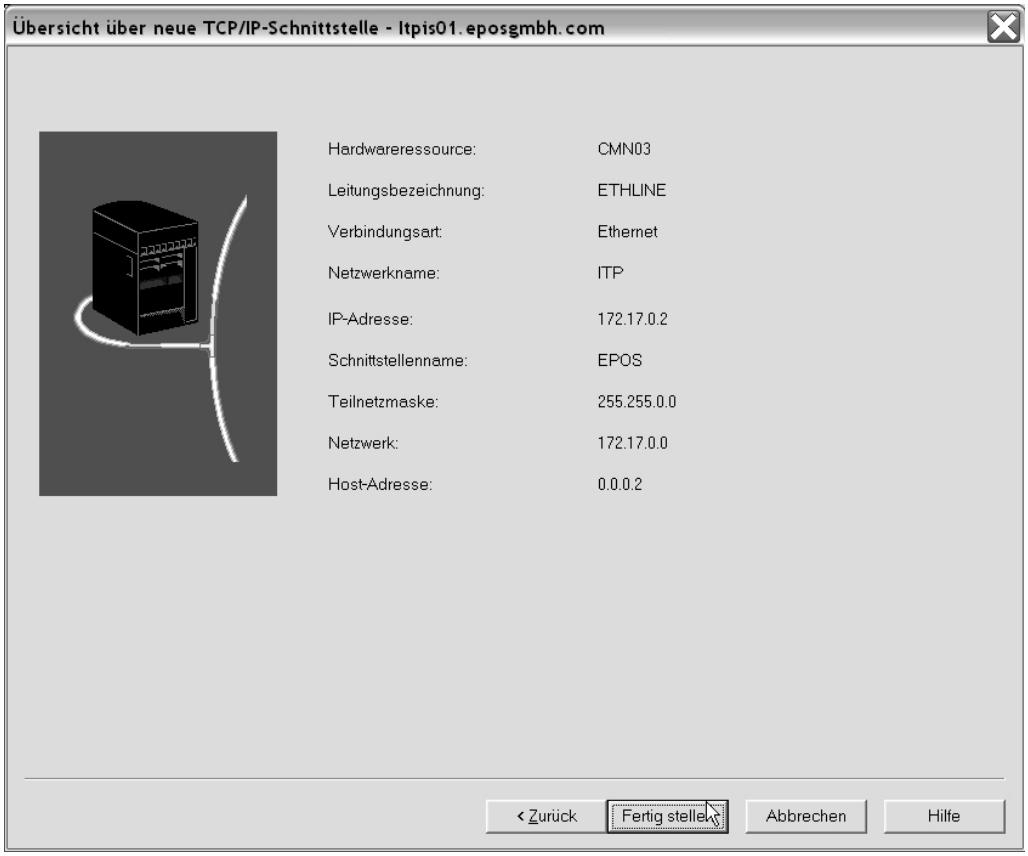
7.6.4.2
Seite 12



Schnittstelle starten

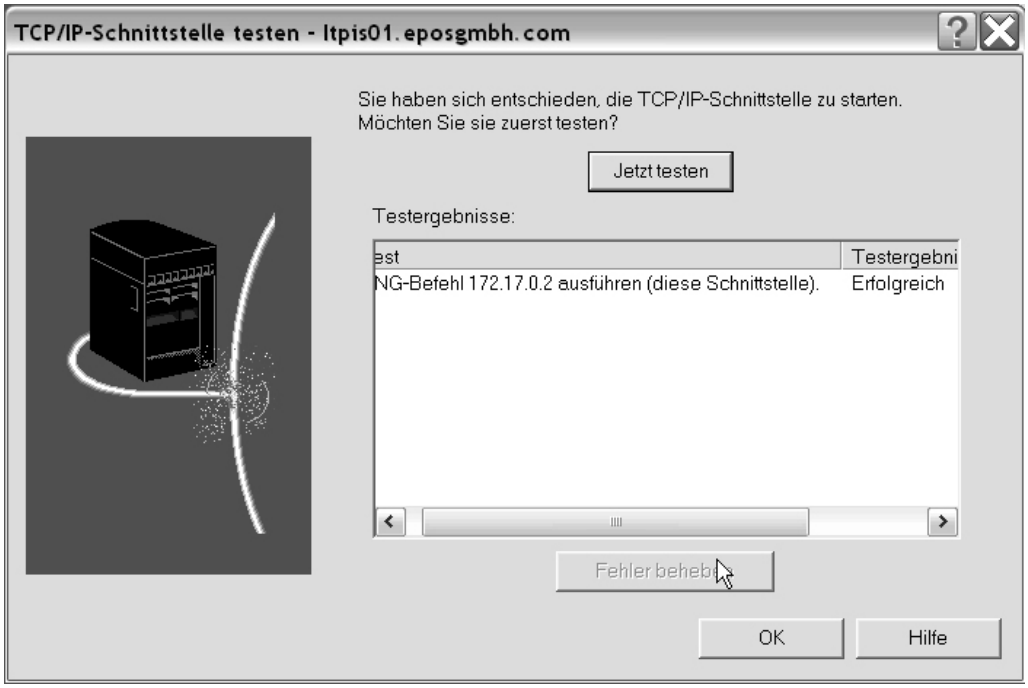
Ich lege hier fest, dass die Schnittstelle bei jedem Systemstart automatisch aktiviert wird und darüber hinaus, das sie jetzt sofort gestartet werden soll.

Danach erscheint ein Zusammenfassungsdialog, der noch einmal die getätigten Einstellungen für die unterschiedlichen Parameter anzeigt.



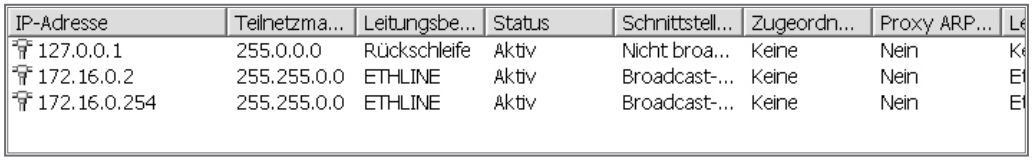
Einstellungen der Schnittstelle

Ich beende den Dialog mit „Fertig stellen“. In der Folge wird mir dann die Möglichkeit gegeben, die neue IP-Schnittstelle zu testen.



Schnittstelle testen

Dazu drücke ich im obigen Dialog den Knopf „Jetzt testen“. Die angezeigte Meldung zeigt mir, dass die von mir erzeugte neue IP-Schnittstelle arbeitsbereit und funktionsfähig ist.



IP-Adresse	Teilnetzma...	Leitungsbe...	Status	Schnittstell...	Zugeordn...	Proxy ARP...	Le
127.0.0.1	255.0.0.0	Rückschleife	Aktiv	Nicht broa...	Keine	Nein	K
172.16.0.2	255.255.0.0	ETHLINE	Aktiv	Broadcast-...	Keine	Nein	Et
172.16.0.254	255.255.0.0	ETHLINE	Aktiv	Broadcast-...	Keine	Nein	Et

Schnittstellenübersicht

Auch die Schnittstellenübersicht im Operations Navigator zeigt mir diesen Zustand an.

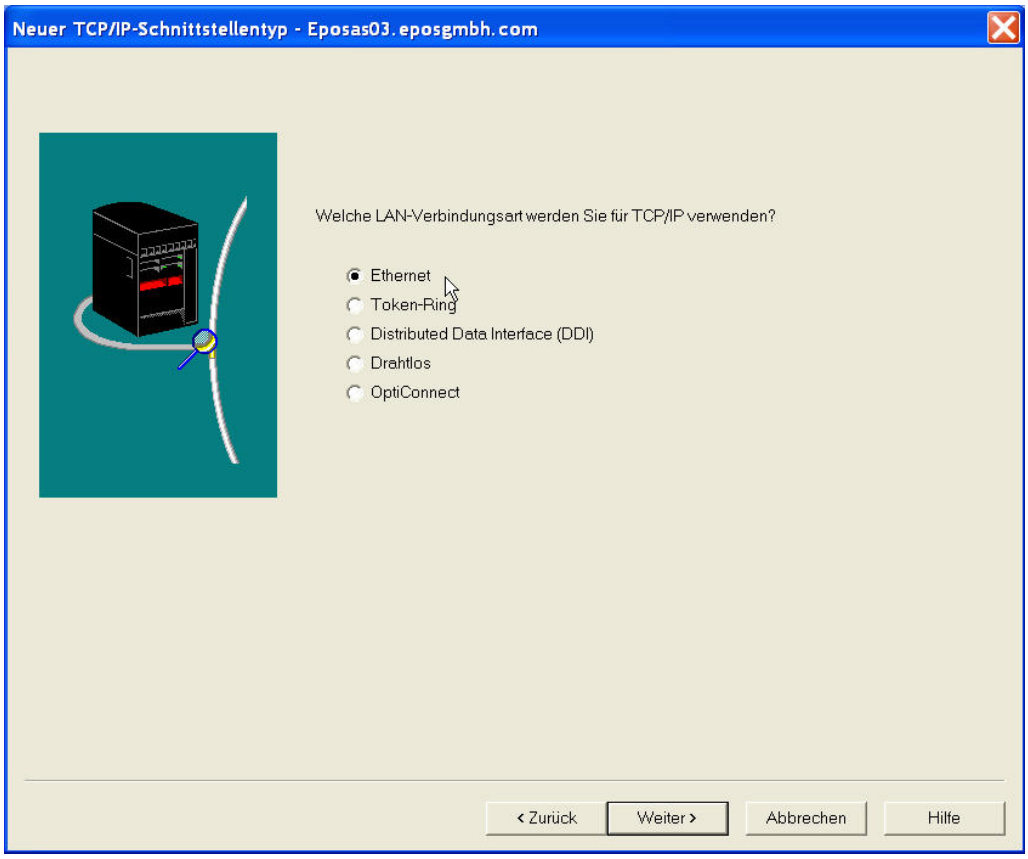
Mit Operations Navigator V5R1 wurde der Schnittstellen-Wizzard noch konsequenter weitergeführt. Er ist jetzt nicht nur in der Lage, eine logische Schnittstelle zu erstellen, sondern kann den gesamten Schnittstellenerstellungsprozess (von der Erstellung der Leitungsbeschreibung bis hin zur arbeitfähigen TCP/IP-Schnittstelle) darstellen.

Ich gehe jetzt davon aus, dass ich für eine neue Ethernet-Netzwerkkarte in meiner iSeries eine Leitungsbeschreibung erstellen muss. Für diese neue Leitung muss dann eine IP-Adresse vergeben werden.

Im Vorfeld ermittle ich den Ressourcennamen für die neue Netzwerkkarte. In diesem Falle gehe ich von „CMN02“ aus.

Ich rufe also wieder den Schnittstellen-Wizzard des Operations Navigator auf.

Nach dem mittlerweile bekannten Willkommens-Dialog öffnet sich der Dialog „Neuer TCP/IP-Schnittstellentyp“.



Ethernet-Schnittstelle

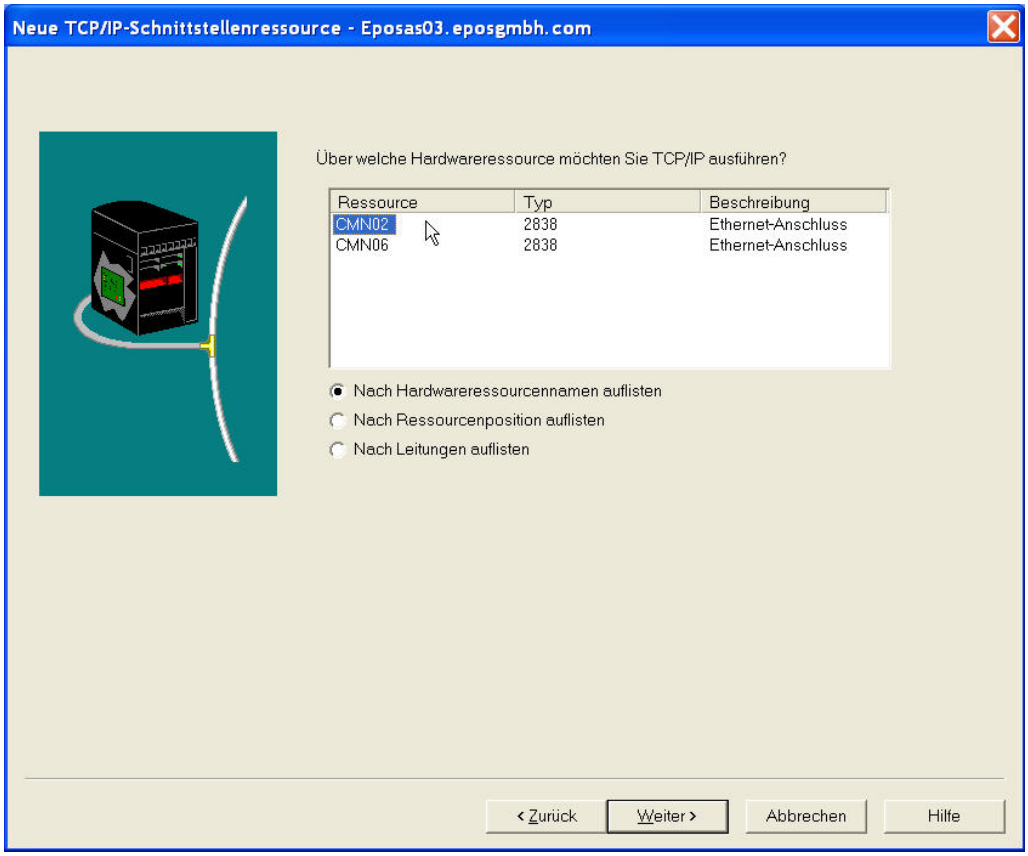
Hier markiere ich die Option „Ethernet“ und bestätige den Dialog mit „Weiter“.

Danach werde ich aufgefordert, die physikalische Ressource zu benennen, mit deren Hilfe die Ethernet-Schnittstelle angesprochen werden kann.

Diese Information hatte ich mir schon im Vorfeld – mit Hilfe des Operations Navigator – ermittelt.

Aus den zur Verfügung stehenden Schnittstellen wähle ich diejenige aus, die für meinen Zweck geeignet ist.

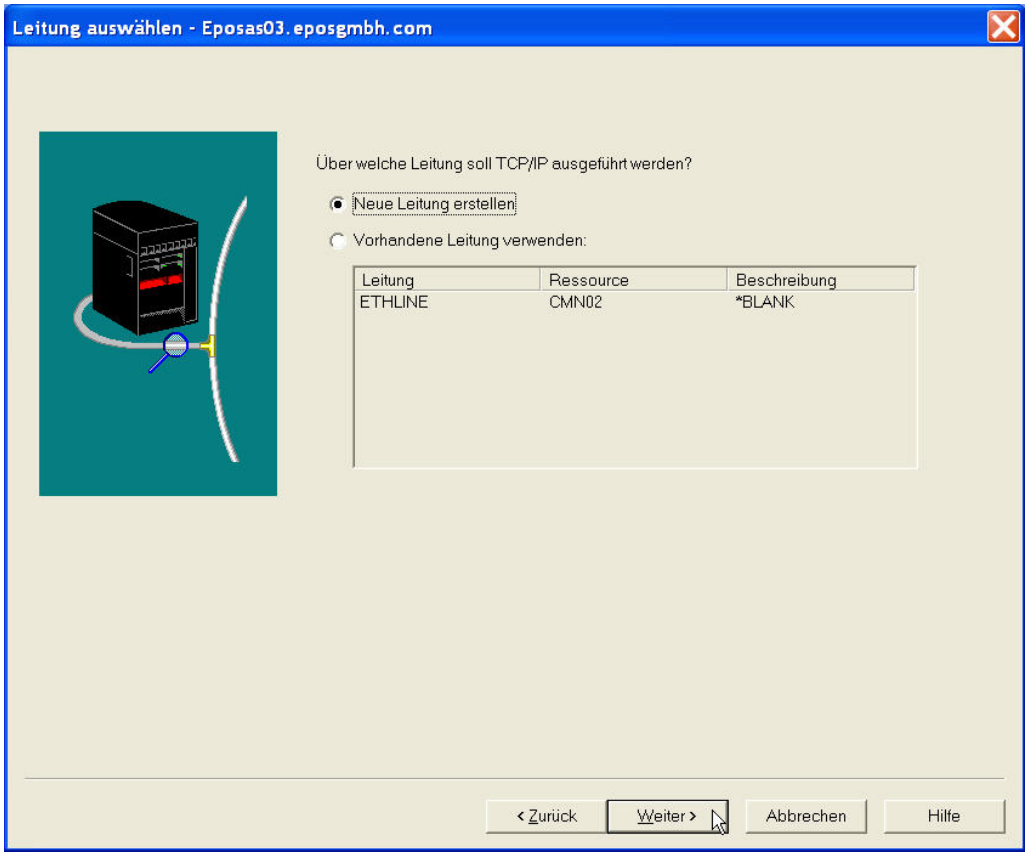
7.6.4.2
Seite 16



Ressource auswählen

Ich markiere also „CMN02“ und bestätige auch diesen Dialog mit „Weiter“.

Der nächste Arbeitsschritt besteht jetzt darin, die Erstellung einer neuen Leitungsbeschreibung zu ermöglichen.



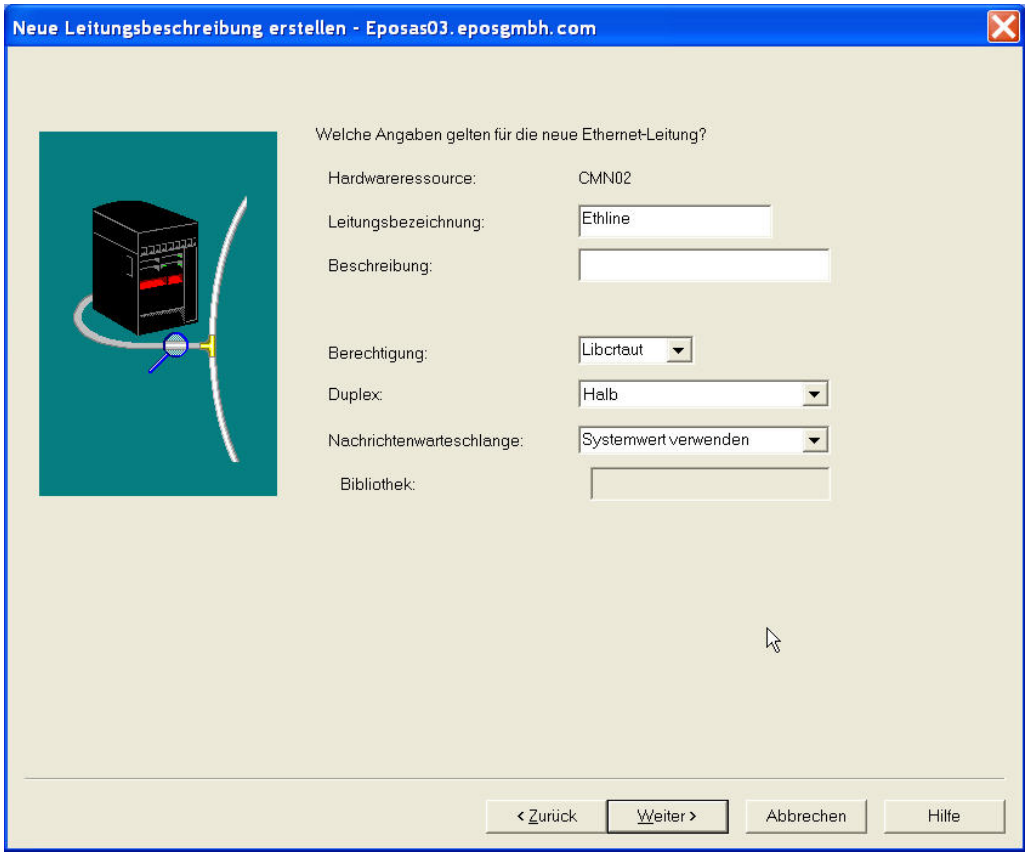
Leitung auswählen

Im Dialog „Leitung auswählen“ markiere ich die Auswahl „Neue Leitung erstellen“. Dadurch wird die Liste unter „Vorhandene Leitung verwenden“ deaktiviert.

Der Dialog wird mit „Weiter“ verlassen.

Daraufhin erscheint der erste Dialog der das Erstellen einer neuen Leitungsbeschreibung steuert.

7.6.4.2
Seite 18

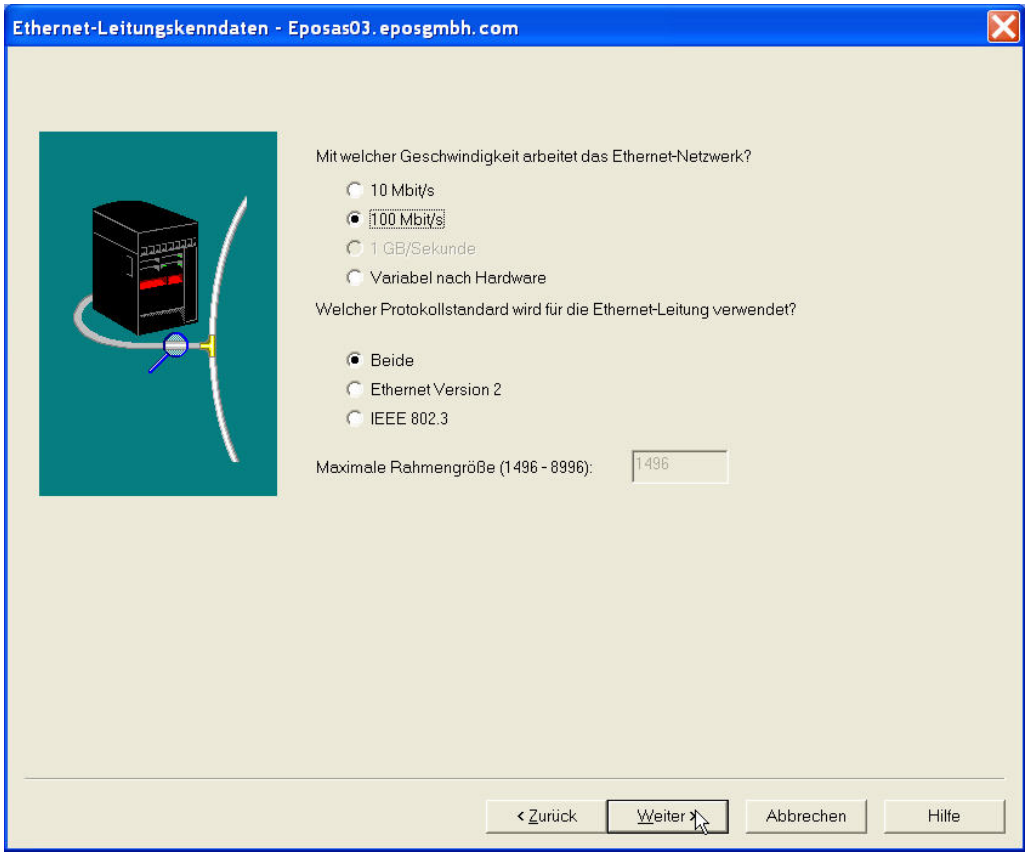


Leitungsbeschreibung erstellen

Ich trage hier lediglich den Namen meiner neuen Leitungsbeschreibung im Parameter „Leitungsbezeichnung“ ein. Gegebenenfalls tragen Sie im Parameter „Beschreibung“ einen sinnvoll beschreibenden Text ein, sofern der Name Ihrer Leitungsbeschreibung in sich selbst nicht sprechend ist.

Mit „Weiter“ wird dieser Dialog bestätigt.

Danach öffnet sich die Dialog „Leitungskenndaten“.



Leitungskenndaten

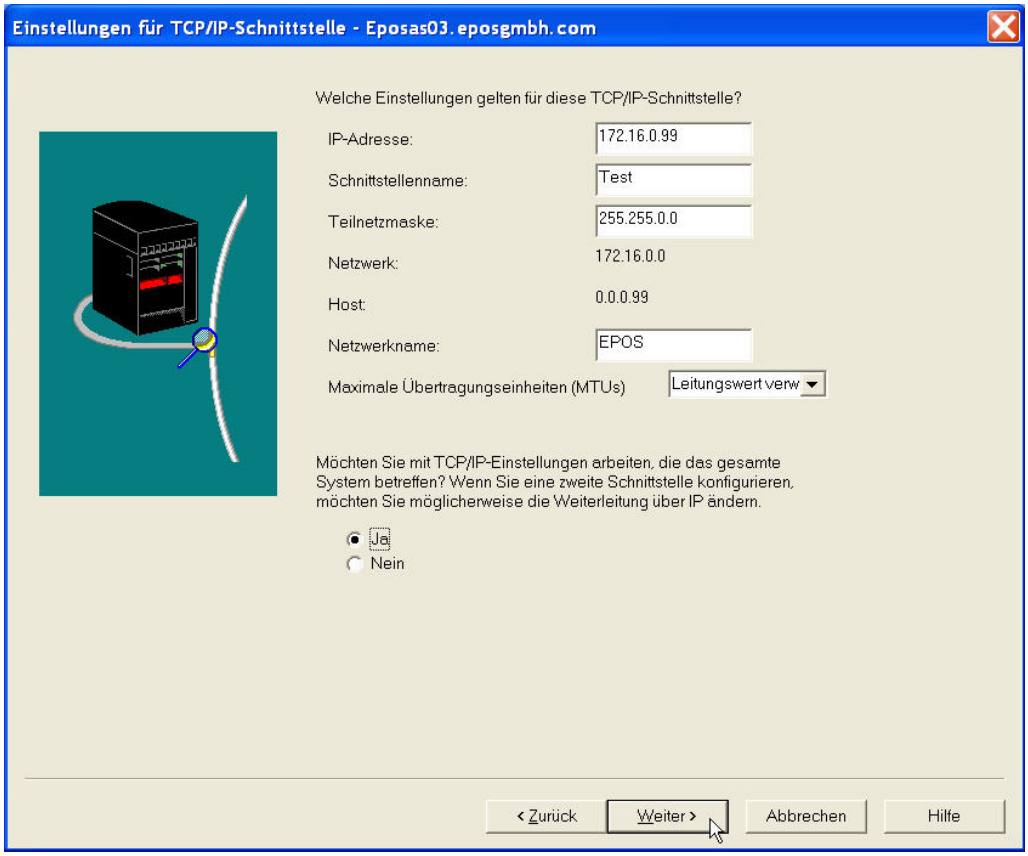
Hier habe ich die maximale Leitungsgeschwindigkeit, die meine Netzwerkkarte ermöglicht, eingetragen: „100Mbit/s“.

Sind Sie sich bezüglich Ihrer Netzwerkkarte nicht sicher, können Sie auch variabel nach Hardware eintragen. Dann ermittelt das System selbst die maximale Leitungsgeschwindigkeit.

Vorsicht: Sollten Sie alte 10 Mbit-Ethernetkarten verwenden, so ist bei diesen eine dynamische Geschwindigkeitsermittlung nicht möglich! In diesem Falle muss 10 Mbit/s fest eingestellt werden.

Neue 100 Mbit- und 1 Gbit-Karten sind in diesem Zusammenhang jedoch unkritisch.

7.6.4.2
Seite 20



IP-Adressendaten erfassen

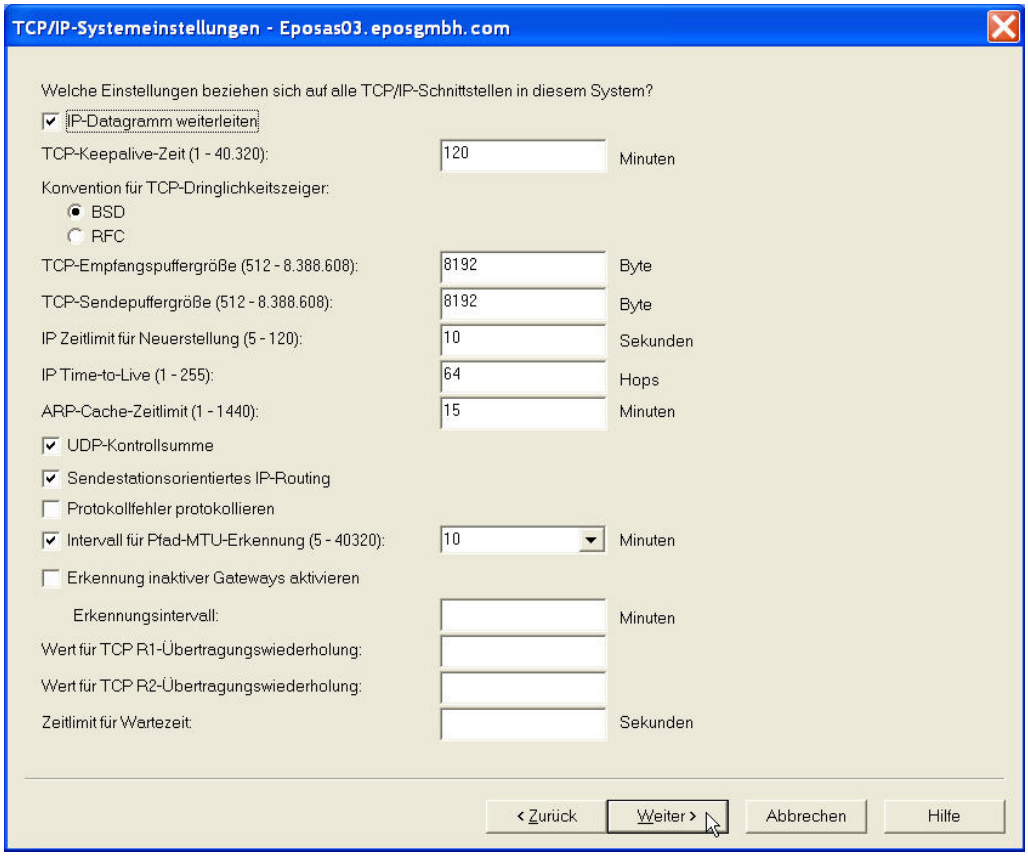
Im Dialog „Einstellungen für TCP/IP-Schnittstelle“ trage ich jetzt die Adressdaten für meine neue Schnittstelle ein. Zwei Parameter sind hier zu finden, die Sie im entsprechenden CL-Befehl des i5/OS nicht finden werden.

Schnittstellenname: Dieser Wert wird nur im Operations Navigator genutzt, um die Anzeige von Schnittstellen transparenter zu gestalten.

Netzwerkname: Auch dieser Parameter dient Dokumentationszwecken. Für operative Zwecke ist er nicht geeignet.

Da ich mit Routings arbeiten möchte, habe ich den entsprechenden Radio-Button auf „Ja“ gesetzt.

Der folgende Dialog ermöglicht den Zugriff auf systemweite Parameter des TCP/IP-Netzwerks.

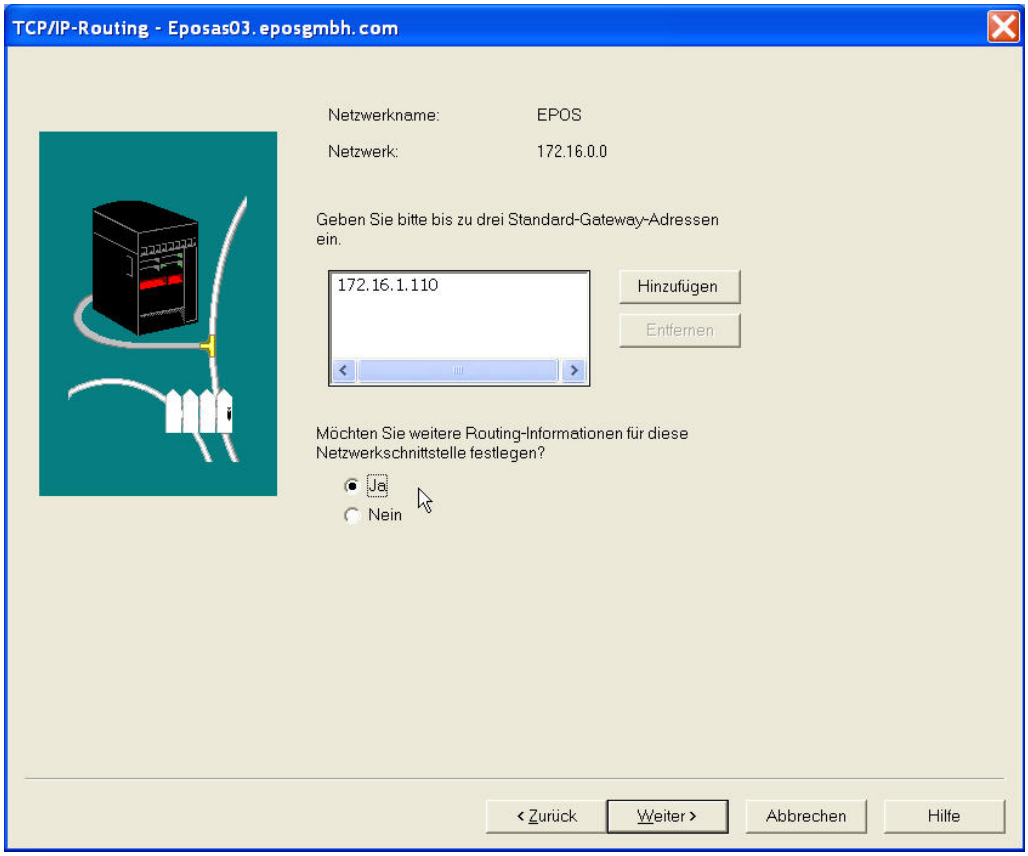


Systemeinstellungen

In diesem Dialog akzeptiere ich die vorgegebenen Werte und beende die Anzeige mit „Weiter“.

Der nächste Dialog ermöglicht mir, ein Standard-Gateway zu benennen.

7.6.4.2
Seite 22



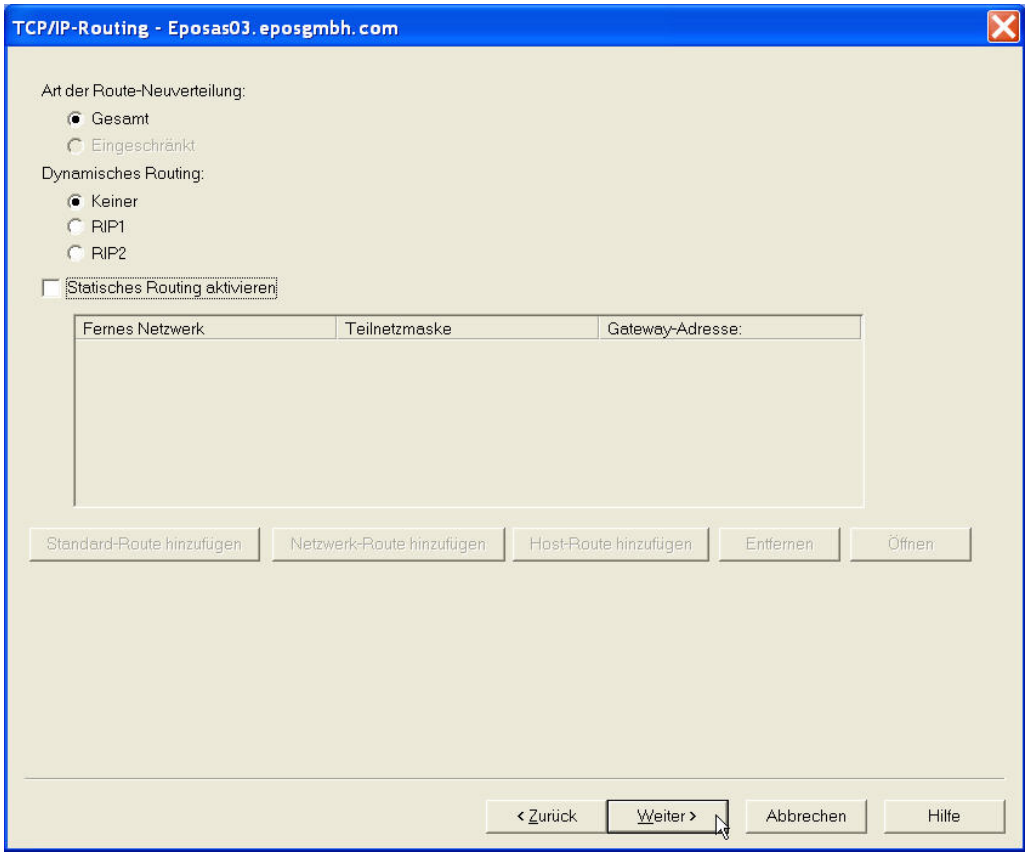
Routing

Falls im System bereits ein Standard-Gateway konfiguriert wurde, so wird es hier angezeigt. Neue Gateways erzeugen Sie durch das Drücken des Knopfes „Hinzufügen“.

Sollten Sie weitere Routings erstellen wollen, so markieren Sie den Radiobutton „Ja“, was ich hier getan habe.

Mit „Weiter“ wird der Dialog beendet.

Unten sehen Sie jetzt die Möglichkeit, Routings einzustellen.



Routings

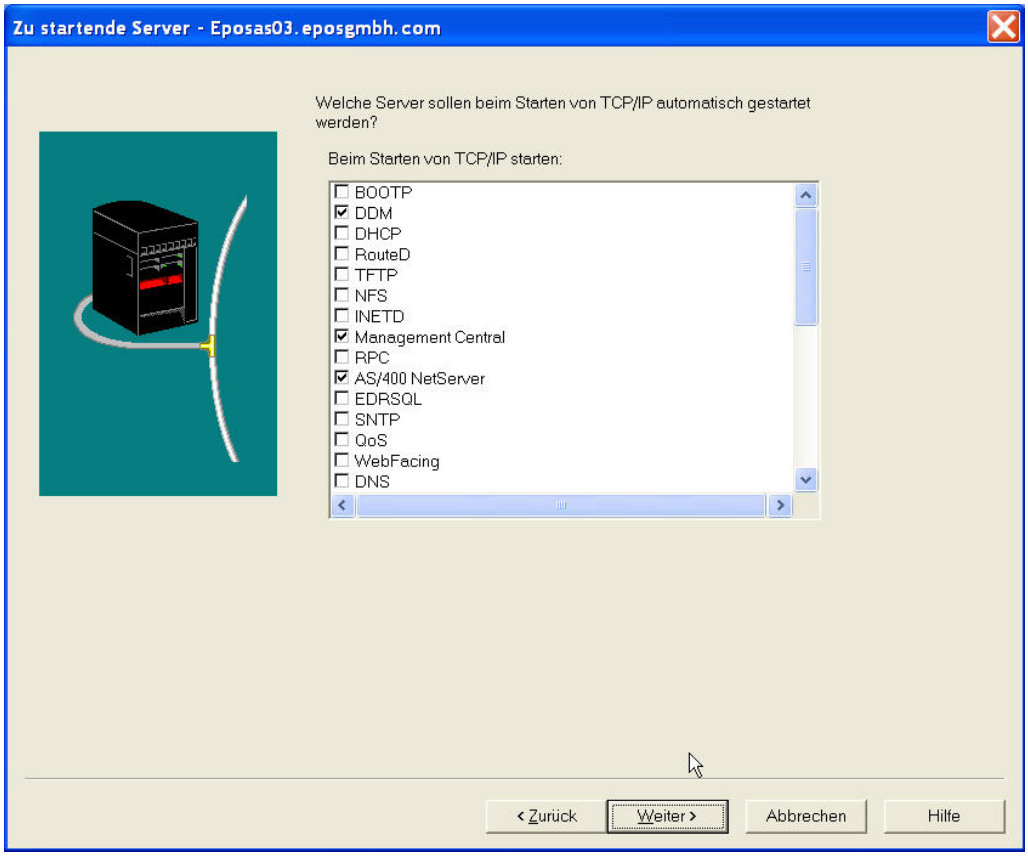
Generell besteht hier die Möglichkeit, sowohl statische als auch dynamische Routen einzutragen.

Ich belasse es hier bei den Default-Einstellungen und bestätige auch diesen Dialog mit „Weiter“.

Damit ist die eigentliche Konfiguration der Schnittstellen abgeschlossen.

Ich werde jetzt nur noch aufgefordert, die TCP/IP-Dienste-Server zu benennen, die automatisch mit dem Start der auf dem System befindlichen TCP/IP-Schnittstellen gestartet werden sollen.

7.6.4.2
Seite 24

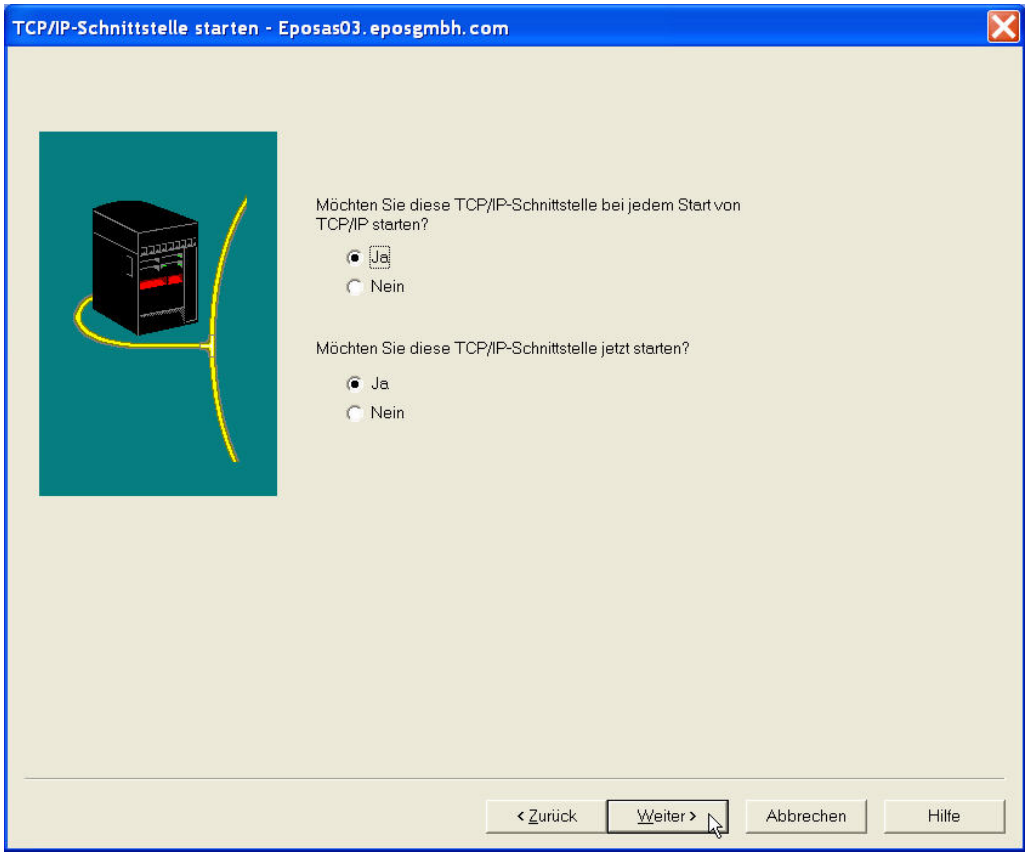


Dienste für Autostart benennen

Auch diesen Dialog bestätige ich mit „Weiter“.

Im nachfolgenden Dialog werden dann zwei abschließende Fragen gestellt:

1. Soll das neue Interface bei jedem Start von TCP/IP aktiviert werden?
2. Soll die Schnittstelle jetzt aktiviert werden?

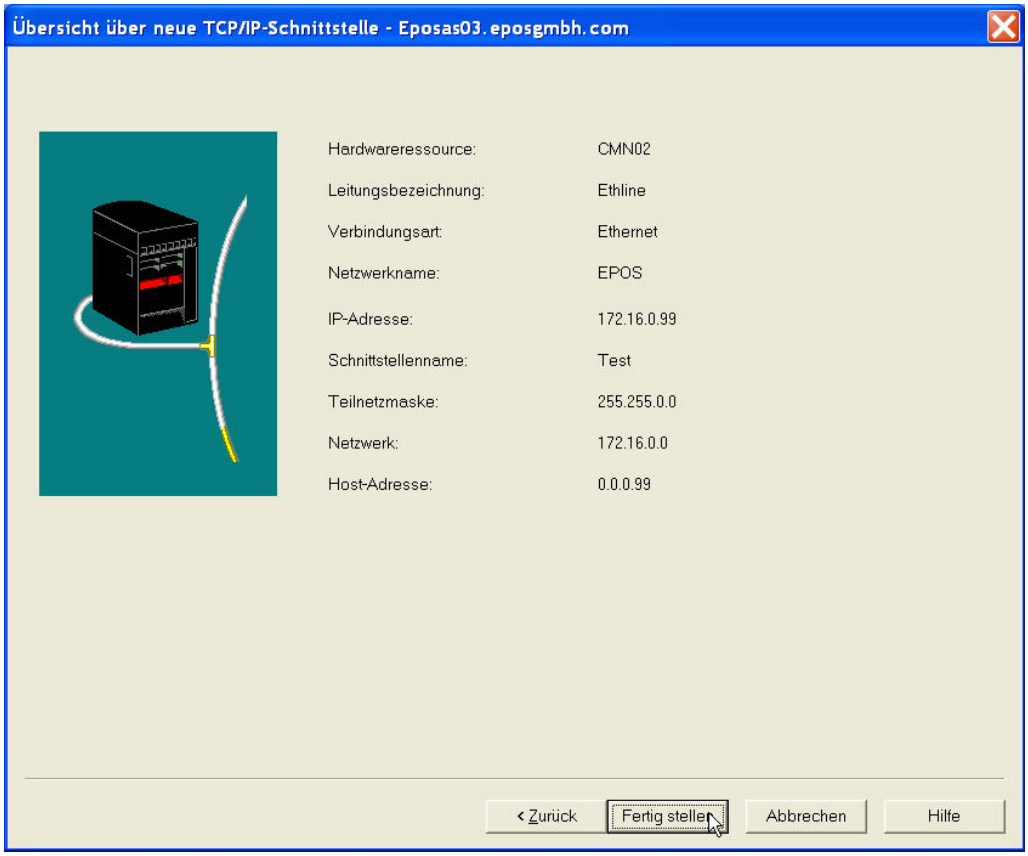


Schnittstelle starten

Sie sehen, dass ich beide Fragen mit „Ja“ beantwortet habe, so dass einem Test dieser neuen Schnittstelle eigentlich nichts mehr im Wege steht.

Nach Beendigung dieses Dialogs erscheint nur noch der folgende:

7.6.4.2
Seite 26



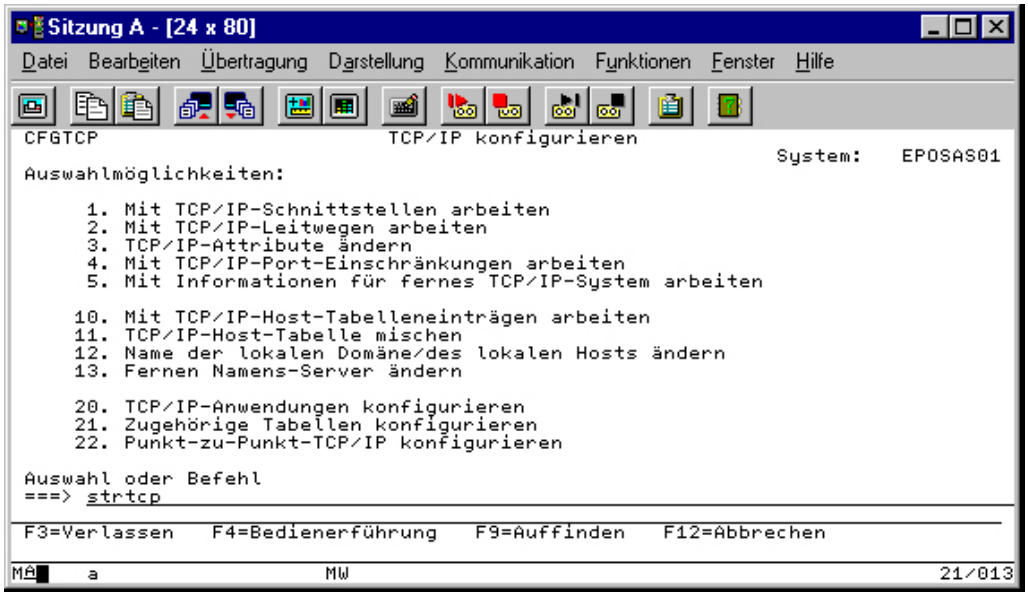
Konfigurationsdaten anzeigen

Er zeigt in der Übersicht einige der vorgenommenen Einstellungen an.

Ich akzeptiere alle getätigten Parameter und beende die Arbeit des Wizzards mit „Fertig stellen“.

Die TCP/IP Schnittstelle starten

Geben Sie zum Start der TCP/IP Schnittstelle den Befehl STRTCP ein.



STRTCP

Dieser Befehl kann eine Reihe von TCP/IP-Diensten starten. Welche der im Folgenden aufgeführten Dienste gestartet werden sollen, wird bei der Einzelkonfiguration auf Dienstebene festgelegt.

FTP: File Transfer Protocol – einfacher Dateitransfer

TELNET: Bildschirm-Emulation

SMTP: e-Mail Protokoll

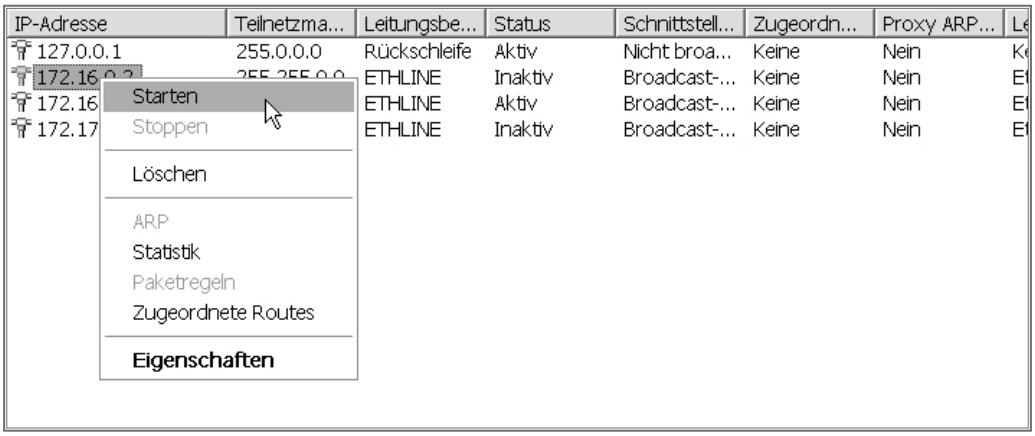
POP3: Post Office Server

HTTP: Internet Server

WSG: Workstation Gateway

Mit dem Operations Navigator wird die TCP/IP-Schnittstelle folgendermaßen gestartet.

7.6.4.3
Seite 2



Schnittstelle starten

Ich markiere in der Schnittstellenübersicht des Operations Navigator die zustartende Schnittstelle mit der rechten Maustaste und wähle die Funktion „Starten“ aus.

Wir gehen jetzt in den Befehl WRKACTJOB und sehen uns im Subsystem QSYSWRK die TCP/IP-Jobaktivitäten an. Folgende Jobs sind u. a bei kompletter TCP/IP-Dienstkonfiguration sichtbar:

TCP/IP-Jobs im Subsystem QSYSWRK und QUSRWRK

QTCPIP	TCP/IP Interface Daemon
QTCPMONITR	TCP Event Monitor
QTGTELNETS	TELNET Server (pre-V4R4)
QTVTELNET	TELNET Server (V4R4)
QTVDEVICE	TELNET Device Manager (V4R4)
QTFTPnnnnn	FTP Server
QTODDHCPR	BootP DHCP Relay Agent
QTODDHCPS	DHCP Server
QTBOOTP	BootP Server
QTRTDnnnnn	RouteD Server
QTTFTnnnnn	Trival FTP Server
QTOBNDS	Domain Name Server
QTOBXFER	DNS (Secondary) Zone Transfer
QTOBXMIT	DNS (Primary) Zone Transfer
QTLPDnnnnn	LPD Server
QTPOPnnnnn	POP Server (Standard)
QTPOCnnnnn	Client Access POP Server
QTPOInnnnn	Client Access IPX POP Server
QTOPSNA	Client Access SNA POP Server
QTRXCnnnnn	REXEC Server

QTSMTPSRVR	SMTP Server
QTMSSRCD	SMTP Server (PJ in V4R4)
QTSMTPCLNT	SMTP Client
QTMSCLCP	SMTP Client (PJ in V4R4)
QTSMTPBRCL	SMTP Bridge Client
QTSMTPBRSR	SMTP Bridge Server
QTWSGnnnnn	Workstation Gateway Server
QTPPPCTL	PPP Control Job
QTPPDIALnn	Point-to-Point Session Dial
QTPPANSnnn	Point-to-Point Session Answer
QTPPPL2TP	L2TP Server (V4R4)
QTPPPL2SSN	L2TP Session (V4R4)
QESISRV	Internet PTF Delivery Server
QTOVMAN	VPN Server (V4R4)
QTOKVPNIKE	IKEYMAN VPN Server (V4R4)
QTOGINTD	INETD Server
QNFSRPCD	RPC Binder Daemon
QNFSBIOD	Block I/O Daemon
QNFSNFSD	NFS Server
QNFSMTND	NFS Mount Server
QNFSNSMD	Network Status Monitor
QNFSNLMD	Network Lock Manager
ADMIN, Server Instance Name	HTTP Server
MSERVERD	Net.Commerce Server
QUSRPYMSVR	IBM Payment Server
QYPSSRV	System Management Server (Management Central)
QDIRSRV	Directory Server
QOAGMAIN	Wireless Server
QYTCNSLD	Network Station Login Server
QTNSMINS	NSM Inventory Server
QRWTLSTN	DRDA Server
QRWTSRVR	DRDA Server
QZSCSRVSD	CA/400 Central Server Daemon
QZSCSRVS	CA/400 Central Server
QZHQSRVD	CA/400 Data Queue Server Daemon
QZHQSSRV	CA/400 Data Queue Server
QNPSERVD	CA/400 Network Print Server Daemon
QNPSERVS	CA/400 Network Print Server

7.6.4.3
Seite 4

QZRCSRVSD	CA/400 Remote Command Server Daemon
QZRCSRVS	CA/400 Remote Command Server
QZSOSGND	CA/400 Sign-on Server Daemon
QZSOSIGN	CA/400 Sign-on Server
QZSOSMAPD	CA/400 Server Mapper
QUMBDMN	Ultimedia SF Server
QIWVPPJT	CA/400 Virtual Print Server
QSQSRVR	SQL Server
QRLGSRVR	OnDemand Server
QTMSNMP	SNMP Server
QTMSNMPRCV	SNMP Server
QSNMPSA	SNMP Server
QAPPCTCP	APPC over TCP/IP (Anynet)
QMSF	Mail Server Framework

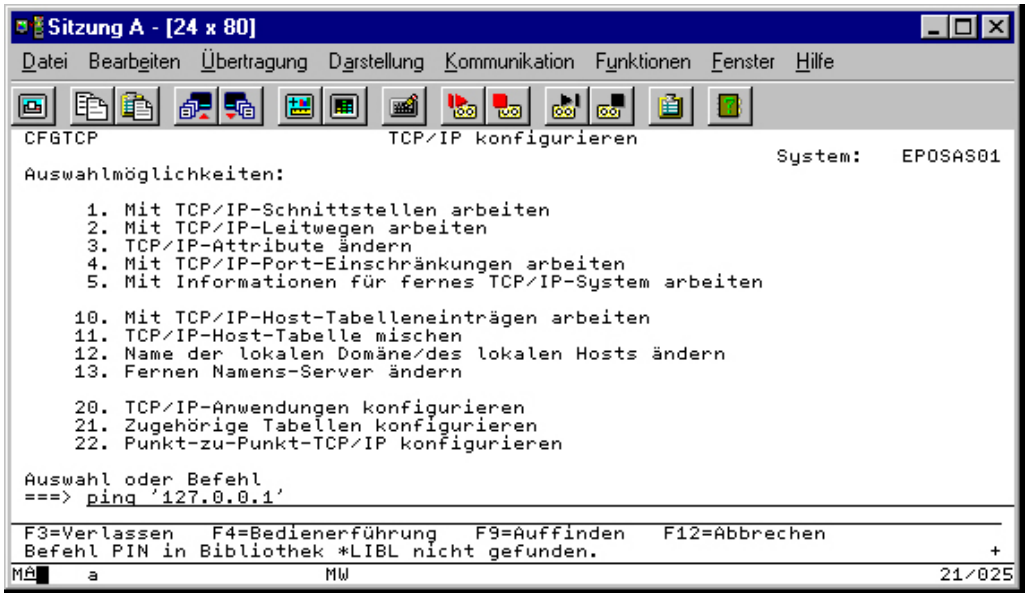
TCP/IP- und iSeries Access für Windows-Jobs im Subsystem QSERVER

QZLSSERVER	NetServer
QZLSFILE	NetServer
QZDASRVSD	CA/400 Database Server Daemon
QZDASOINIT	CA/400 Database Server
QZDASSINIT	CA/400 Database Server for SSL (V4R4)
QPWFSERVSD	CA/400 File Server Daemon
QPWFSERVS0	CA/400 File Server
QPWFSERVS2	CA/400 File Server
QPWFSERVSS	CA/400 File Server for SSL (V4R4)
QTFPJTCP	CA/400 Transfer Funktion

Die Funktion der TCP/IP-Schnittstelle überprüfen

Soweit scheint alles gut gelaufen zu sein. Aber wir brauchen noch ein handfestes Prüfungsergebnis, dass unsere Schnittstelle auch wirklich betriebsbereit ist.

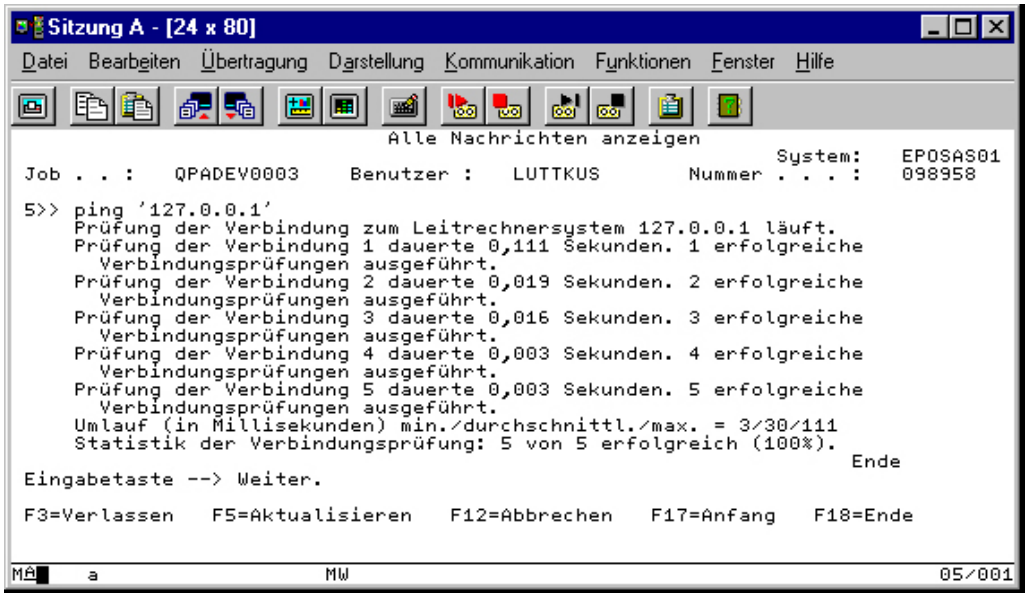
Geben Sie den Befehl PING ein – gefolgt von der LOOPBACK Adresse 127.0.0.1.



Prüfen der Konfiguration: PING ‚127.0.0.1'

Folgende Meldungen sollten in der Statuszeile des Bildschirms erscheinen:

7.6.4.4
Seite 2



PING-Meldungen: Erfolg – die Konfiguration steht!

Das Überprüfen der Funktionsfähigkeit der Schnittstelle kann ebenfalls mit dem Operations Navigator realisiert werden.

Schnittstelle prüfen 1

Dazu markieren Sie mit der rechten Maustaste den Eintrag „TCP/IP-Konfiguration“ und wählen dann „Dienstprogramme > Ping“.

Schnittstelle prüfen 2

Im Parameter PING trage ich die zu prüfende Adresse ein und betätige sodann „PING ausführen“. In der Listbox „Ergebnisse“ sehen Sie das Prüfergebnis.

7.6.4.4

Seite 4

Troubleshooting

Dieses Mal haben wir eine Schnittstelle erfolgreich erstellt. Welche Prüfschritte sind aber für den Fall eines Fehlers zu durchlaufen?

1.) Überprüfung der physikalischen Schnittstelle – Leitungsbeschreibung

Jedes TCP/IP-Paket, das von Ihrer iSeries empfangen werden soll, muss einen Prüfprozess durchlaufen, der das Paketierungsformat dieses Pakets überprüft. Dieser Prüfvorgang wird über einen so genannten Source Service Access Point (SSAP) gesteuert, der mit einer zweistelligen hexanumerischen Adresse benannt wird. Diesen Parameter finden Sie im Rahmen einer SSAP-Liste in der Leitungsbeschreibung:

```
SSAP-Liste:
   Source Service Access Point  .   04
   SSAP-Maximalrahmen . . . . . .   1496
   SSAP-Art . . . . . . . . . . .   *SNA

   Source Service Access Point  .   12
   SSAP-Maximalrahmen . . . . . .   1496
   SSAP-Art . . . . . . . . . . .   *NONSNA

   Source Service Access Point  .   AA
   SSAP-Maximalrahmen . . . . . .   1496
   SSAP-Art . . . . . . . . . . .   *NONSNA
```

Der für die Überprüfung von TCP/IP-Paketen zuständige SSAP trägt die Adresse AA. Sollte dieser SSAP nicht vorhanden sein, so kann keine TCP/IP-Kommunikation stattfinden.

2.) Überprüfung der TCP/IP Serverjobs

Die logische TCP/IP-Schnittstelle kann nur erfolgreich aktiviert werden, wenn das Subsystem QSYSWRK erfolgreich gestartet wurde. In den allermeisten Fällen wird der STRTCP-Befehl durch das Systemstartprogramm der iSeries – QSTRUP – ausgeführt.

Können Sie die Schnittstelle zwar händisch starten, jedoch nicht aus dem Startprogramm heraus, so sind zwei Ursachen typisch:

- Sie haben den Befehl STRTCP im Startprogramm nicht an das Ende gestellt, sondern eher in der Mitte platziert. Da das Startprogramm den Start der Subsysteme einleitet, jedoch das Ende des Vorgangs nicht abwartet, sondern weiterläuft, ist es mehr als wahrscheinlich, dass das Subsystem QSYSRWK noch nicht komplett gestartet wurde, so dass der Job QTCPIP, der die Schnittstellenfunktion darstellt, nicht gestartet werden konnte.

7.6.4.4.1
Seite 2

Plazieren Sie den Aufruf von STRTCP ganz ans Ende des Programms QSTRUP. Sollte das nicht ausreichen, so kodieren Sie vor den STRTCP-Befehl einen DLYJOB (Zeitverzögerung für Job) von 30 bis 60 Sekunden.

- Möglicherweise ist der Befehl STRTCP berechtigungstechnisch so geschützt, dass das Programm QSTRUP nicht das Recht hat, diesen Befehl auszuführen. QSTRUP läuft als Standard unter dem Benutzer QPGMR. Wenn dieser Benutzer nicht das Recht hat, STRTCP auszuführen, sollten Sie QPGMR das *USE-Recht für den Befehl erteilen.

3. Überprüfen der TCP/IP Adressen

Befinden sich die Adressen der zu verbindenden Hosts im gleichen logischen Netzwerk?

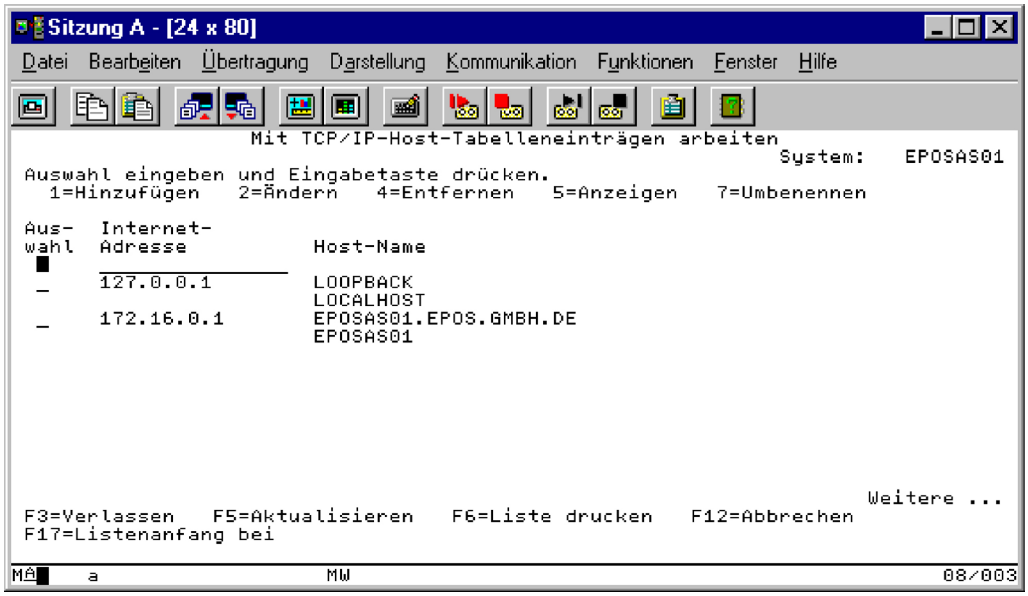
- Überprüfen Sie die Net-ID der Adresse auf Gleichheit.
- Überprüfen Sie die Subnetzmaske auf Gleichheit.

Die Host-Tabelle

Stellen Sie sich einmal vor, dass Sie nicht nur ein, zwei, fünf oder zehn Systeme TCP/IP-fähig machen müssen, sondern summa summarum 200, 300 oder 500. Es wird Ihnen jetzt sicher nicht schwer fallen, diese Anzahl von Adressen zu generieren. Aber wenn es jetzt zu Fehlern in Ihrem Netzwerk kommt und Sie Meldungen zum System 172.16.233.52 oder 172.16.188.47 erhalten, so werden Sie sicher Schwierigkeiten haben, sich zu erinnern, wo und an welchem Arbeitsplatz diese Systeme zu finden sind. Also wäre es doch schön, wenn es die Möglichkeit gebe, „richtige“ Namen für die Systeme zu vergeben, die es Ihnen ermöglichen, ein bestimmtes System genau zu lokalisieren.

Diese Möglichkeit wird Ihnen mit der Host-Tabelle gegeben. Eine Host-Tabelle ist ein lokales Objekt, welches auf jedem TCP/IP-Host implementiert werden kann und einer numerischen TCP/IP-Adresse eine verbale Adresse zuordnet.

Rufen Sie im Menü CFGTCP den Punkt 10 „Mit TCP/IP Host-Tabelleneinträgen arbeiten“ auf. Es erscheint auf der iSeries folgende Anzeige:

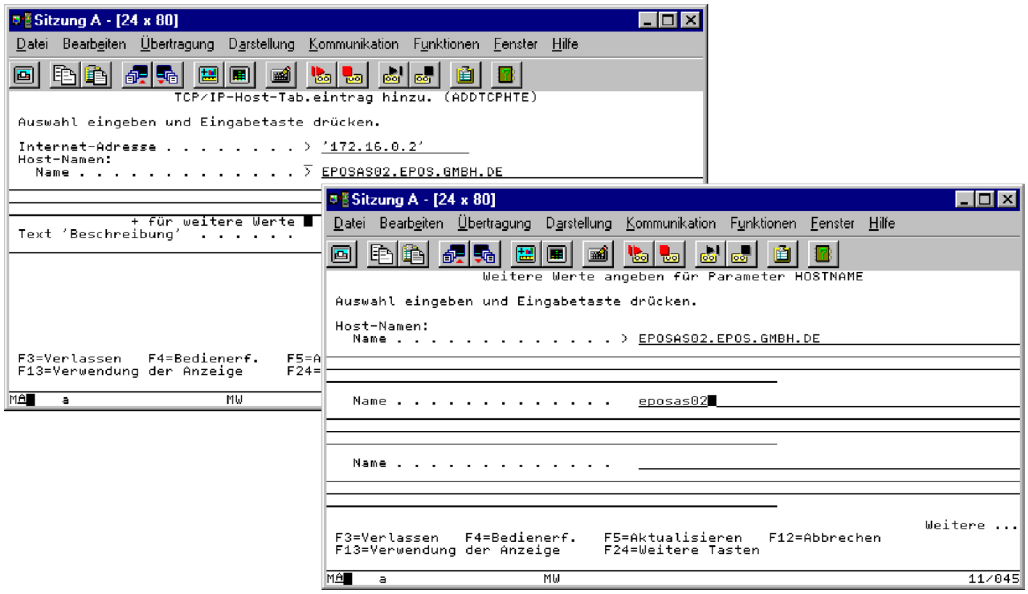


Mit Host-Tabellen arbeiten

Ich werde jetzt eine zweite iSeries – Systemname EPOSAS02 – mit einem sprechenden Namen ansprechen.

Ich wähle also „1=Hinzufügen“ und gebe folgende Werte ein:

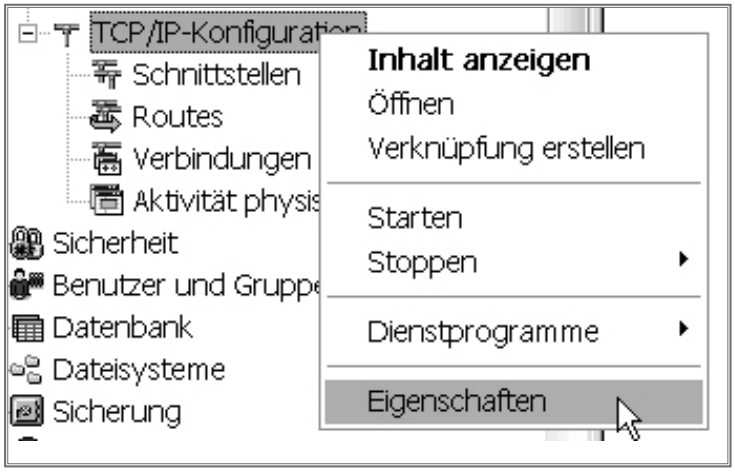
7.6.4.5
Seite 2



Host-Tabelleneintrag erfassen

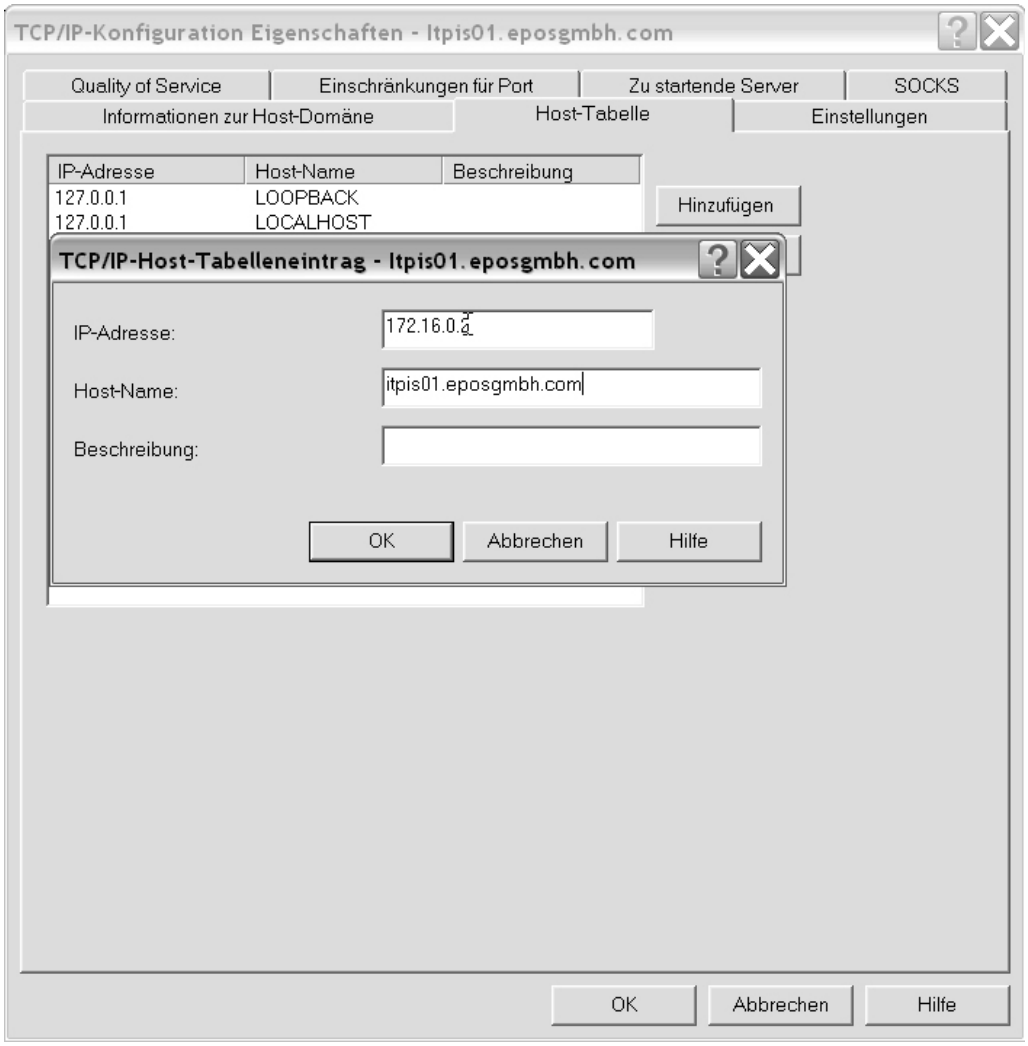
Wie Sie leicht erkennen können, vergebe ich zwei Namen: Der erste Name – EPOSAS02.EPOS.GMBH.DE – wird der IP-Adresse direkt zugeordnet. Der zweite Name – EPOSAS02 – ist als Aliasname hinterlegt und wird generell für „Schreibfaule“ vorgehalten. Ein PING EPOSAS02 gibt sich doch schneller ein als PING EPOSAS02.EPOSGMBH.DE – oder?

Die Verwaltung der iSeries-Host-Tabelle ist ebenfalls mit dem Operations Navigator möglich. Dazu markieren Sie mit der rechten Maustaste „TCP/IP-Konfiguration“ und wählen im Kontextmenü den Punkt „Eigenschaften“ aus.



TCP/IP-Eigenschaften

Im Eigenschaften-Register klicken Sie dann den Reiter „Host-Tabelle“ an.



Host-Tabelleneintrag erzeugen

Jetzt drücken Sie den Knopf „Hinzufügen“ und erzeugen dann den neuen Host-Tabelleneintrag.

Alle geöffneten Dialoge werden anschließend mit „OK“ verlassen.

7.6.4.5
Seite 4

Der Domain Name Service (DNS)

Ein Merkmal des TCP/IP-Netzwerks besteht darin, dass sämtliche für den Transport von Datenpaketen erforderlichen Adressen numerisch gestaltet sind. Aus Gründen, die auf der Hand liegen hat man sich schon frühzeitig darum bemüht, sozusagen als Stellvertreter für numerische Adressen namensbezogene Adressen zu vergeben. In den frühen TCP/IP-Implementationen war für die Umsetzung von numerischen Adressen in Namen die Datei „HOSTS“ zuständig.

Die Verwendung der Datei HOSTS bringt sowohl Vorteile als auch Nachteile mit sich. Ein Vorteil besteht sicherlich in der einfachen Handhabung dieser Datei. Lediglich ein Texteditor ist notwendig, um die Datei zu bearbeiten. Die Einträge, die hinterlegt werden müssen haben einen einfachen und logischen Aufbau. Als gravierender Nachteil kann angesehen werden, dass die Datei HOSTS ein lokales Objekt jedes TCP/IP-Hosts ist. Jeder Host, der eine Adressumsetzung für sich durchführen soll, muss eine eigene Datei HOSTS besitzen, die darüber hinaus auch noch einen bestimmten Standort in der Verzeichnisstruktur eines jeden Hosts besitzen muss. Ein Netzwerkverzeichnis verbietet sich von vornherein für die Datei HOSTS.

Stellen Sie sich einmal vor, dass Sie 300 TCP/IP-Hosts zu administrieren haben, die jeder eine eigene Datei HOSTS besitzen. Wenn jetzt die Forderung besteht, dass alle diese Dateien den gleichen Inhalt haben, so ist die Vorstellung nicht abwegig, dass es irgendwann Asynchronitäten in diesem Konstrukt geben wird. Denken Sie an Murphy: was geschehen kann, wird auch irgendwann geschehen.

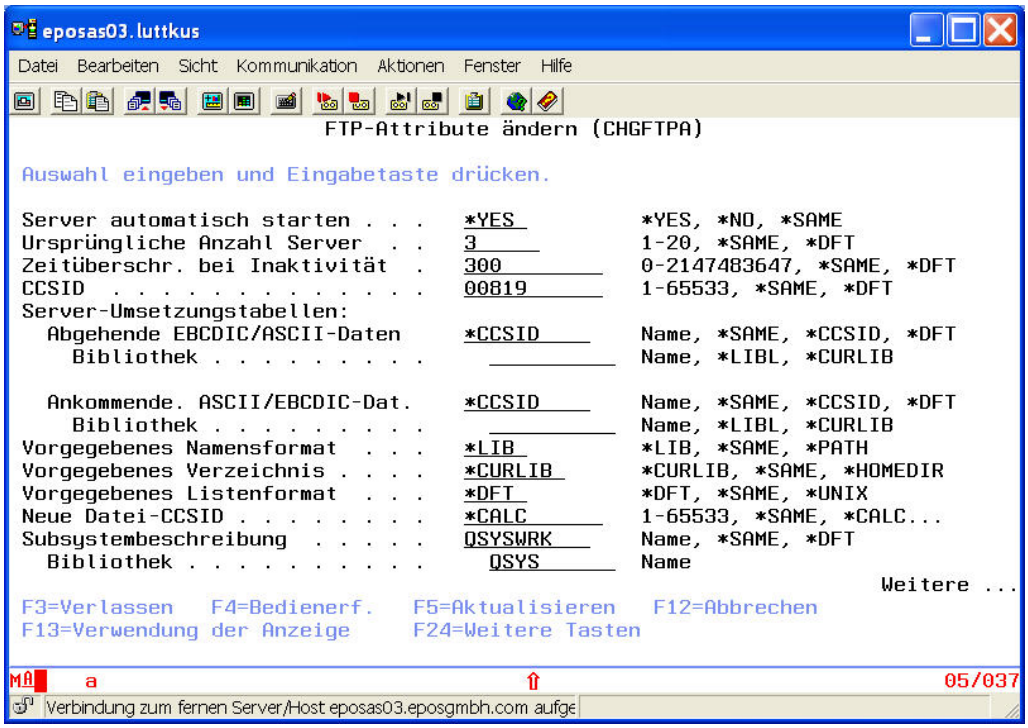
Aus diesem Mangel in der Verwendung der Datei HOSTS hat sich die Entwicklung des Domain Name Service (DNS) ergeben. Dieser Dienst ermöglicht Anwendungen, die miteinander über die Socket-Schnittstelle des TCP/IPs kommunizieren und Host-Namen einfach auswerten, indem sie eine Adressabfrage an einen zentralen Server senden.

7.6.6.1
Seite 2

FTP-Server konfigurieren

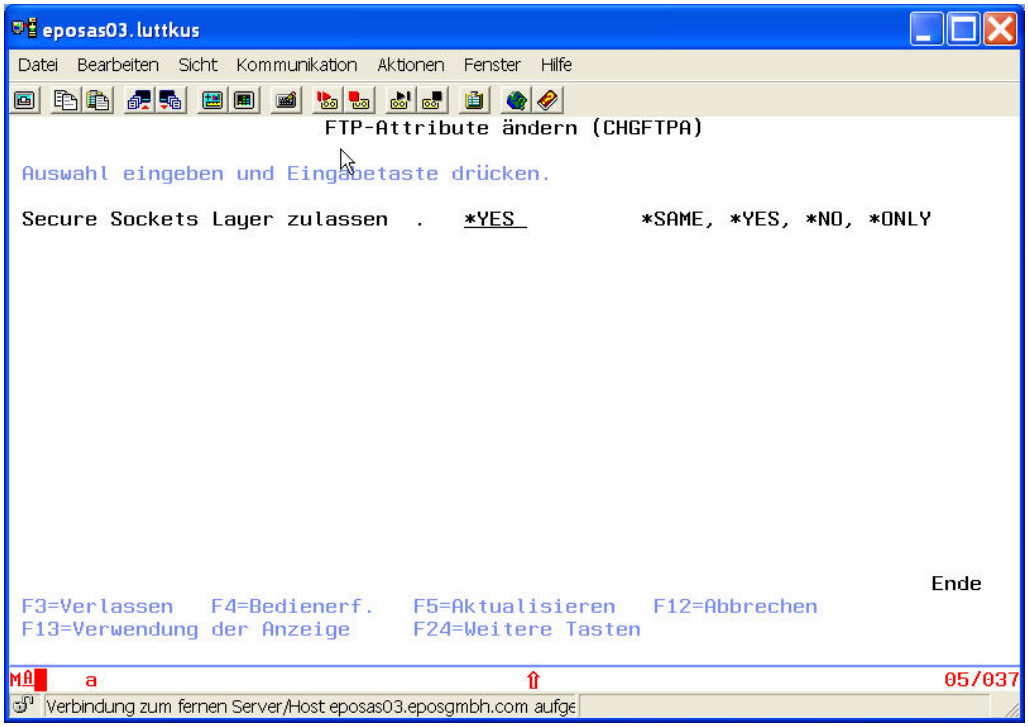
Damit FTP auf einer iSeries aktiv werden kann, muss diese Komponente des TCP/IPs konfiguriert werden. Bevor wir uns jedoch an die eigentliche Konfiguration wagen, müssen zuerst einige Grundlagen geschaffen werden. FTP basiert auf der Funktion zweier TCP/IP-Ports: 20 und 21. Wozu zwei Ports? Der Port 21 ist der so genannte Datenport des FTP. Werden Daten von einem System A nach System B übertragen, so wird dieser Transfer über den Port 21 abgewickelt. Der Port 20 dient als so genannter Steuerungskanal. Was heißt das? Wenn Sie System A beauftragen, Daten von System B herunterzuladen, so wird System A System B einen Befehl – in diesem Fall ein GET – übermitteln, damit der Transfer abgewickelt werden kann. Die Übergabe des Befehls erfolgt dabei über den Port 20. Übrigens wird als Transportmedium für die Befehlsübergabe ein vereinfachtes Telnet-Protokoll verwendet. Bei Funktionsstörungen des FTPs müssen beide Ports beobachtet werden.

Um FTP zu konfigurieren, geben Sie den Befehl CHGFTPA (FTP-Attribute ändern) ein und prompten Ihn mit der F4-Taste.



FTP konfigurieren (1)

7.6.5.1
Seite 2



FTP konfigurieren (2)

Sie sehen hier die Default-Einstellungen des iSeries-FTPs. Folgende Parameter sollten Sie beachten:

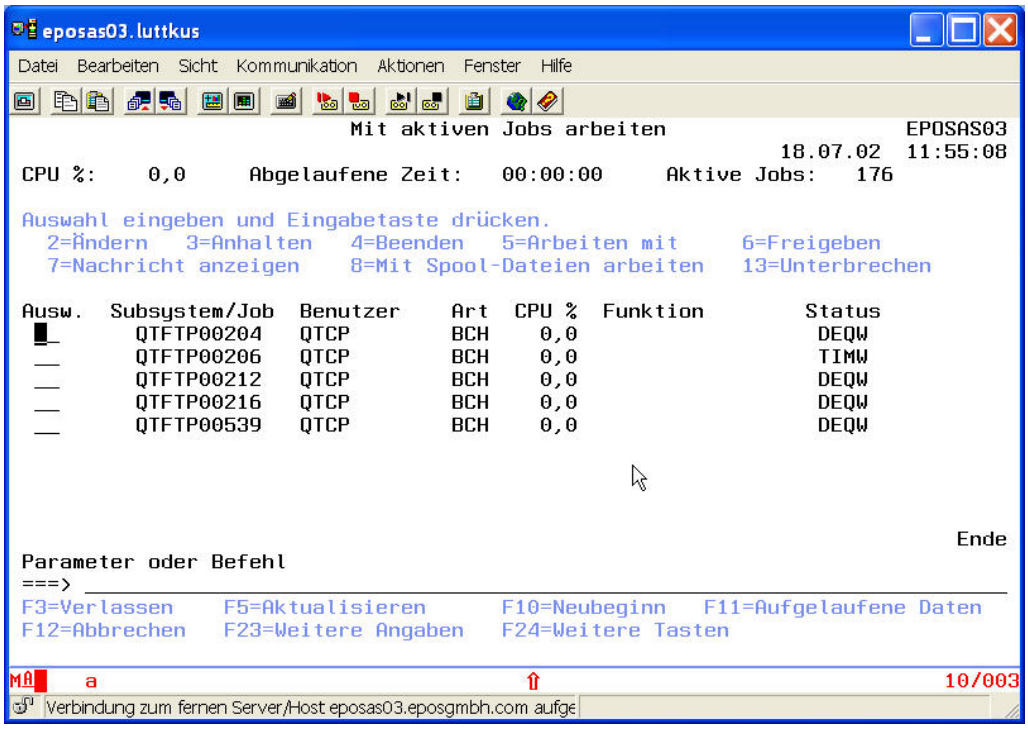
Der Parameter „Server automatisch starten“ ist auf den Wert „*YES“ gesetzt. Dadurch werden die FTP-Serverjobs des OS/400 mit dem Aufruf des Befehls STRTCP (TCP/IP Starten) gestartet.

Der Parameter „Ursprüngliche Anzahl Server“ bestimmt, wie viele der FTP-Serverjobs gestartet werden und steht auf „3“. Warum werden überhaupt mehrere Server-Jobs gestartet? Jedem FTP-Client wird beim Aufbau einer FTP-Sitzung ein eigener FTP-Server-Job zugeordnet. Trifft ein FTP-Client auf einen dieser sozusagen vorgestarteten Jobs, so baut sich zügig eine FTP-Sitzung auf. Ein FTP-Client, für den kein vorgestarteter Server-Job gefunden wurde, muss dann zuerst den Start seines Server-Jobs beantragen – ein etwas länger andauernder Vorgang.

Der Parameter „Zeitüberschreitung bei Inaktivität“ bestimmt dann, wann eine FTP-Sitzung wieder freigegeben wird. Ist eine FTP-Sitzung 300 Sekunden nicht aktiv, so wird der FTP-Server wieder freigegeben und ein anderer Client kann sich dieser Server-Jobs bedienen. 300 Sekunden (5 Minuten) ist der Standardwert.

Soll FTP händisch gestartet werden, so erreichen Sie dies mit dem Befehl STRTCPSVR *FTP. Die Beendigung der FTP-Server wird mit dem Befehl ENDTCPSVR *FTP realisiert.

Ist FTP gestartet, so sollten Sie im Subsystem QSYSWRK mehrere Jobs mit dem Namen QTFTPx sehen (x steht für eine vom OS/400 generierte Zahl).



Die vorgestarteten FTP-Server-Jobs

Überprüfen Sie auch die Funktion der Ports 20 und 21, indem Sie den Befehl NETSTAT *CNN eingeben. In der Spalte „Lokale Ports“ finden Sie die angegebenen Portnummern, allerdings erst wenn Sie die F14-Taste betätigt haben. Beide Ports sollten den Status „Empfangsbereit“ aufweisen.

7.6.5.1

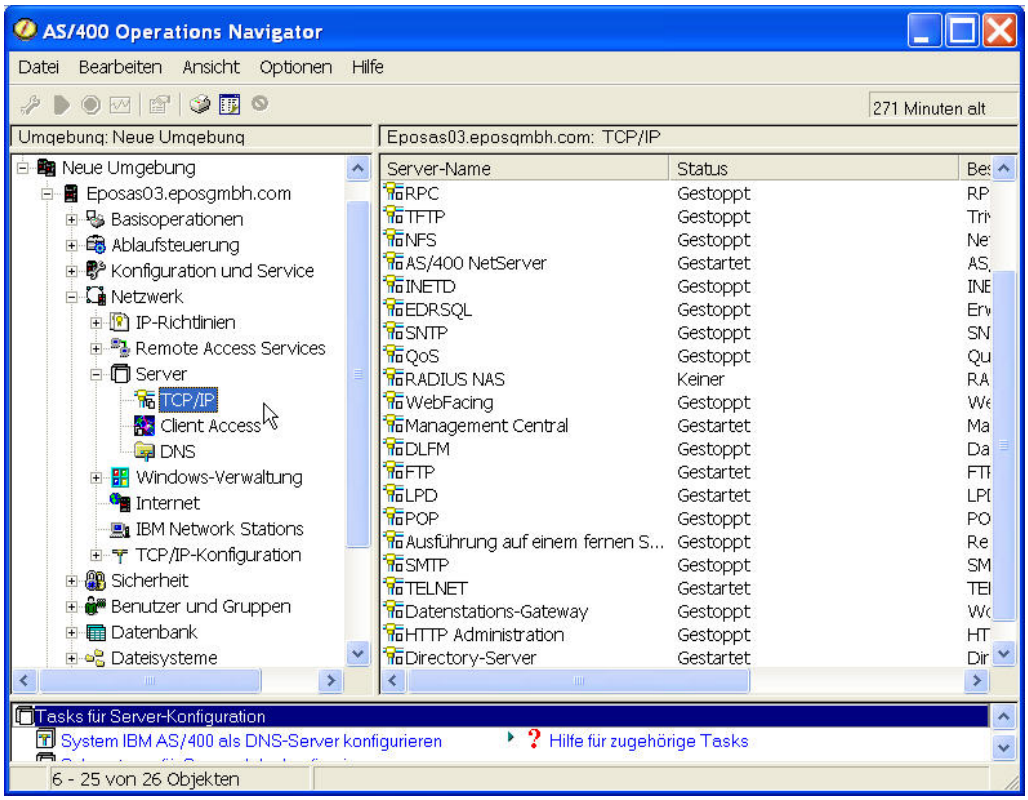
Seite 4

FTP-Server mit Operations Navigator konfigurieren

Ähnlich wie bei der Konfiguration des Telnet-Servers lässt sich auch der FTP-Server mit Hilfe des Operations Navigator einfach konfigurieren und in der Folge administrieren.

Um die Konfigurationsarbeit zu beginnen, öffne ich den Operations Navigator und verwende folgende Ordner:

<Umgebung>→<Verbindung>→Netzwerk→Server→TCP/IP

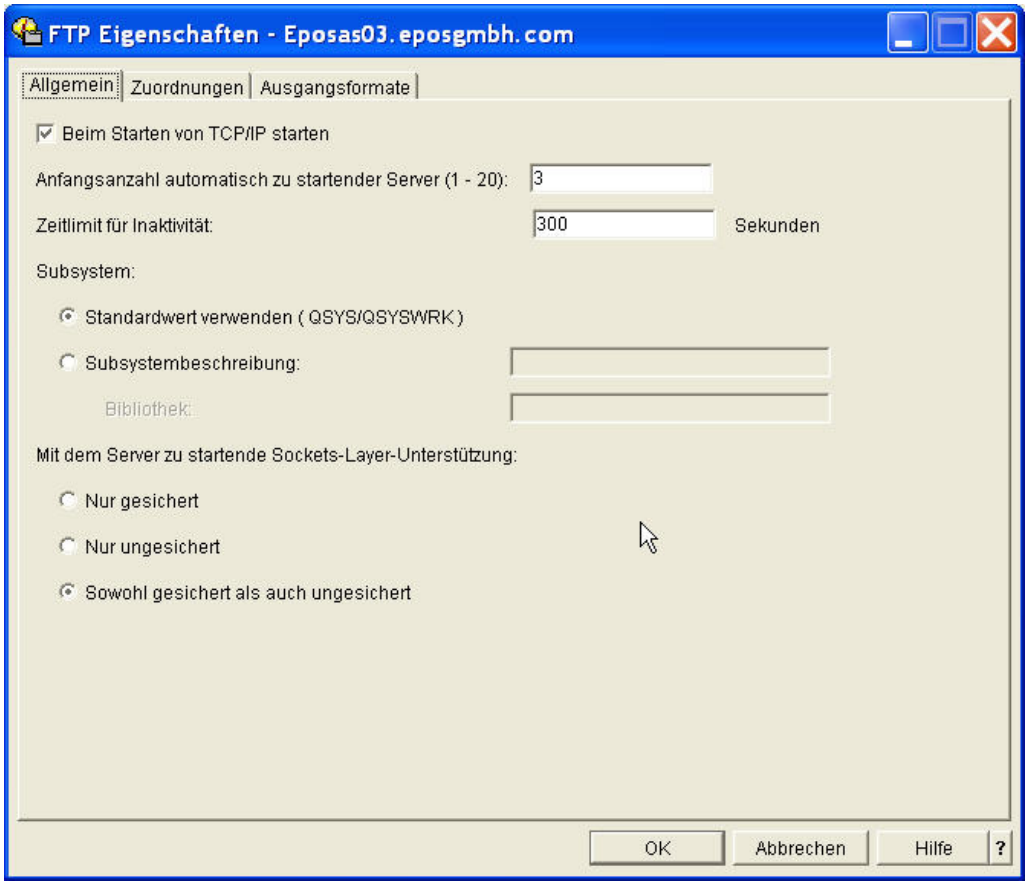


TCP/IP Server-Dienste

Ich markiere jetzt im linken List-Dialog den Eintrag „FTP“ und führe einen Doppelklick aus. Es öffnen sich die Konfigurationsparameter des iSeries-FTP-Servers.

Die im Folgenden gezeigten Parameter entsprechen denen des 5250-Dialogs.

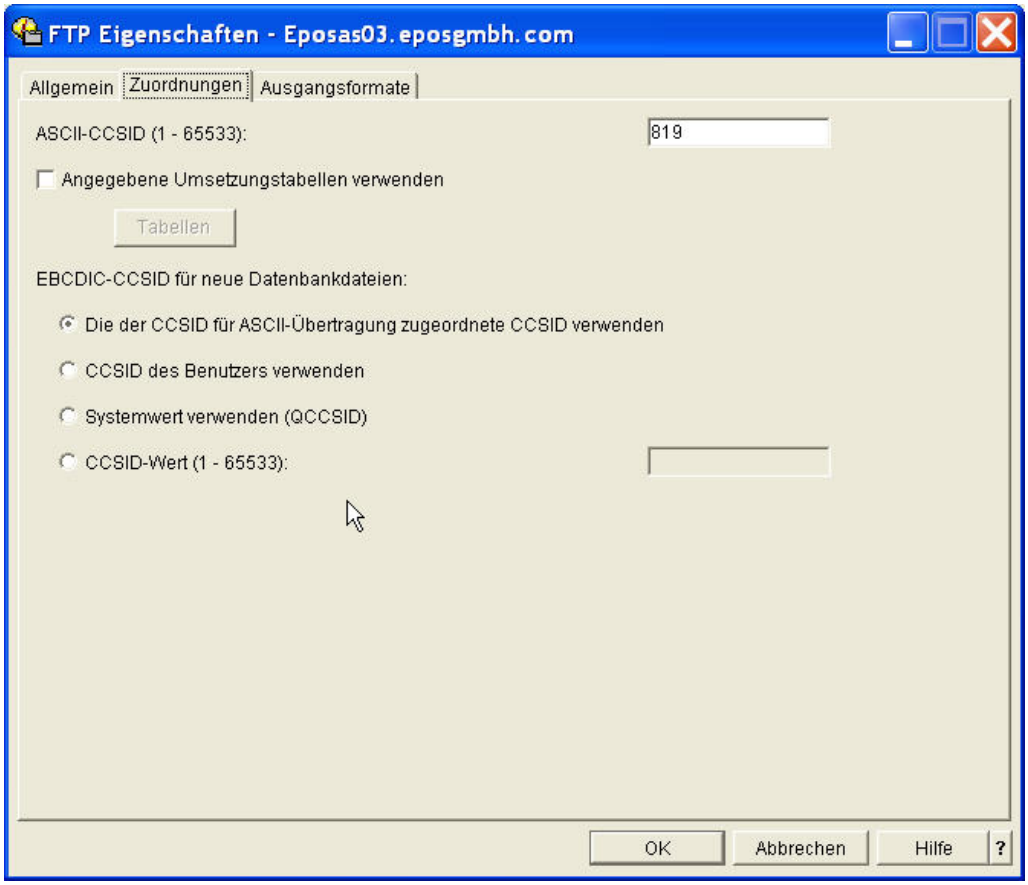
7.6.5.2
Seite 2



FTP-Attribute (1)

Im Register „Allgemein“ finden wir Standardparameter, die ich nicht verändert habe:

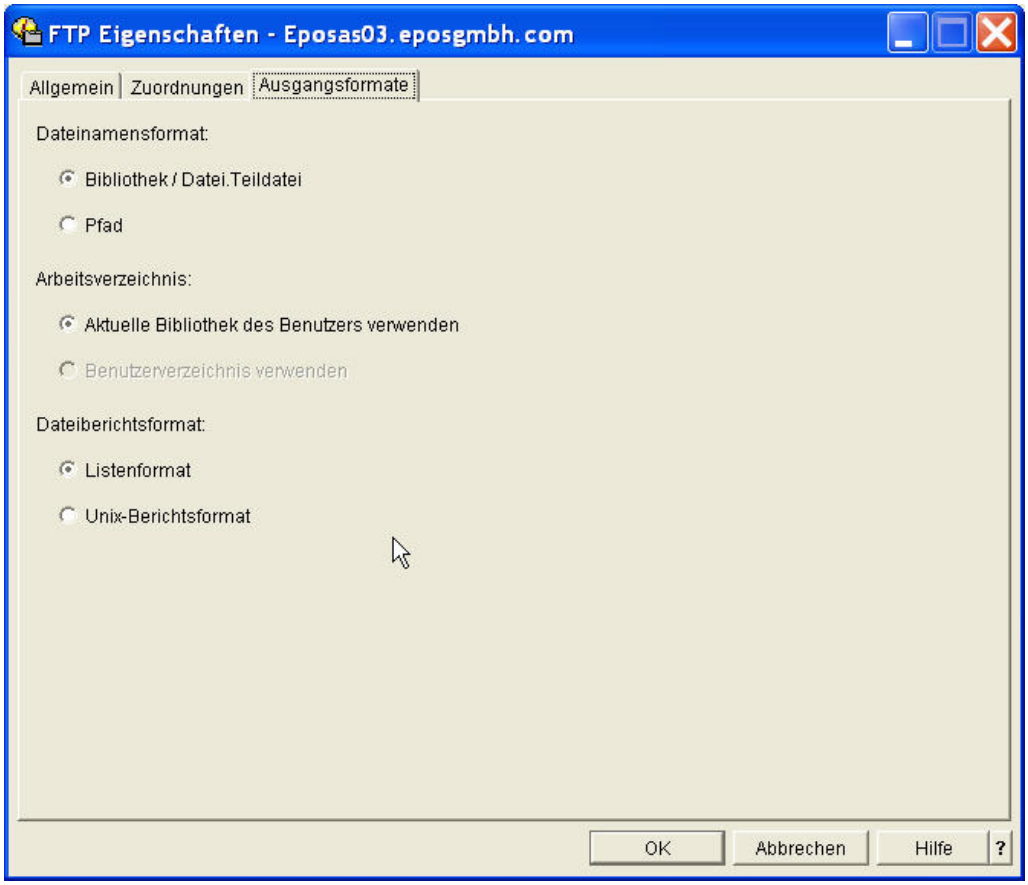
- Anzahl der zu startenden Verbindungs-Server
- Zeitlimit, nachdem eine inaktive FTP-Verbindung beendet wurde
- Auswahl des Subsystems, in welchem die FTP-Verbindungsjobs laufen sollen
- Mögliche Auswahl, ob nur gesicherte, ungesicherte oder beide Arten von Verbindungsaufnahmen unterstützt werden



FTP-Attribute (2)

Das Register „Zuordnungen“ stellt Parameter dar, die die Zeichenumsetzung von Server zu Client und umgekehrt beeinflussen.

7.6.5.2
Seite 4



FTP-Attribute (3)

Die Parameter des Registers „Ausgangsformate“ sprechen grundsätzlich für sich selbst, deshalb erläutere ich hier nur den Parameter „Dateinamensformat“.

Dateinamensformat

- Markieren Sie den ersten Parameter „Bibliothek/Datei.Teildatei“, so sind sämtliche Pfadangaben gemäß dem Bibliothekssystem der iSeries anzugeben.

  Bibliothek/Datei.Teildatei

Arbeiten Sie mit dieser Einstellung, so können nur Dateien übertragen werden.

- Markieren Sie den 2. Parameter „Pfad“, so haben alle Zugriffe mit der Zugriffssyntax des Integrated File System zu erfolgen.

  /Verzeichnis/Objekt.Objektart

Hierzu finden Sie weitere Erläuterungen in Kapitel 7.6.5.7 „Das Integrated File System“.

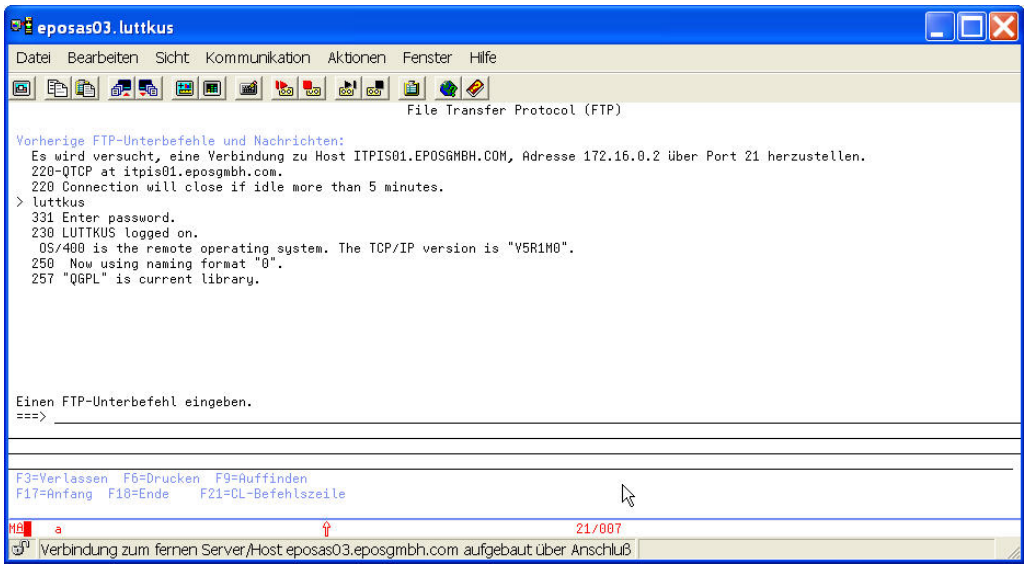
FTP-Client starten

In diesem Kapitel werde ich davon ausgehen, dass Daten zwischen zwei iSeries-Systemen transferiert werden sollen. Die erste iSeries ist über die IP-Adresse 172.16.0.2 zu erreichen. Dieser Adresse ist der Host-Name ITPIS01.EPOSGMBH.COM zugeordnet worden. Das zweite System ist mit Hilfe der IP-Adresse 172.16.0.3 zu erreichen und trägt den Host-Namen EPOSAS03.EPOSGMBH.COM. Ich habe mich am Host EPOSAS03.EPOSGMBH.COM angemeldet.

Auf einer beliebigen Befehlszeile gebe ich folgenden Befehl ein: FTP ITPIS01.EPOSGMBH.COM.

Sie erfahren aus diesem Bildschirm, dass die Datenübertragung über den Port 21 realisiert wurde und dass diese Sitzung nach fünf Minuten wieder aufgeben wird, wenn sie in dieser Zeitspanne keine Befehle eingeben. Sie werden aufgefordert, eine korrekte Anmeldung zu vollziehen.

Nach kurzer Zeit baut sich der gezeigte Bildschirm auf.



Eine FTP-Verbindung ist aufgebaut

Haben Sie sich auf der entfernten Seite korrekt angemeldet, so besteht jetzt eine FTP-Verbindung zwischen EPOSAS03 und ITPIS0101.

Mit Hilfe der Befehlszeile übermitteln wir dann unserem FTP-Server-Job auf dem Host ITPIS01.EPOSGMBH.COM, welche Objekte wir zu transferieren beabsichtigen. Im Folgenden habe ich Ihnen einige der wichtigsten Befehle aus der FTP-Dokumentation zusammengetragen:

7.6.5.3
Seite 2

Befehl	Bedeutung
\	Verzeichniswechsel im entfernten System
CD	Dito
CDUp	Verzeichniswechsel aufwärts
?	Hilfe anfordern
Help	Dito
Ascii	Datenformat anwählen
Binary	Dito
DELete	Objekt löschen
Mdelete	Mehrere Objekte löschen
REName	Objekt neu benamen
Dir	Verzeichnisinhalt anzeigen
Mkdir	Neues Verzeichnis erstellen
RMDir	Verzeichnis entfernen
Get	Objekt holen
Mget	Mehrere Objekte holen
Open	Sitzung anfordern
Put	Objekt bringen
Mput	Mehrere Objekte bringen
PWd	Aktuelles Verzeichnis anzeigen
User	Benutzerprofileingabe anfordern
Password	Kennworteingabe anfordern
QUIt	FTP-Sitzungen beenden

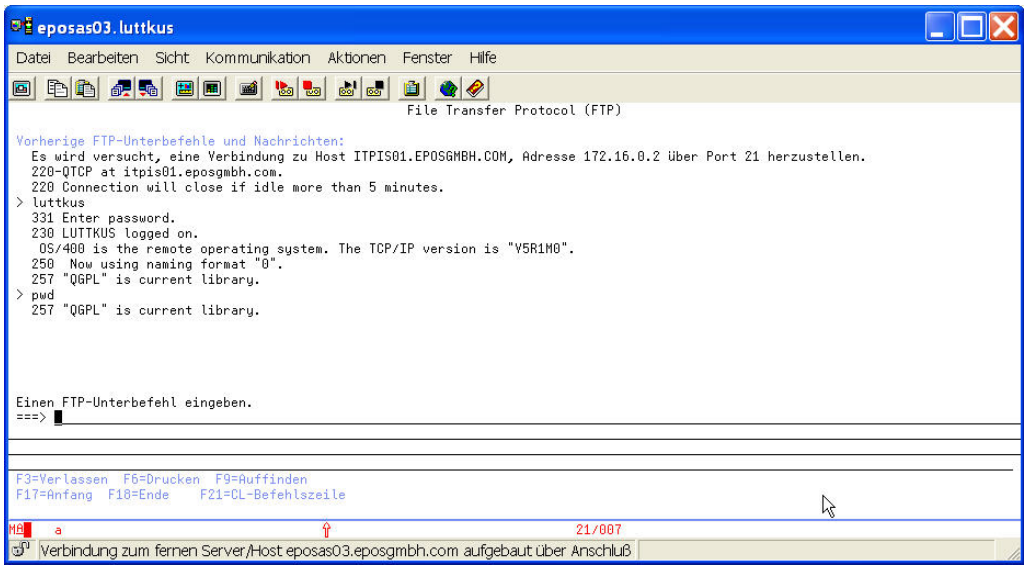
Wichtige FTP-Befehle

Soweit so gut. Was wissen wir bis jetzt? Wir wissen, dass uns der FTP-Client eine interaktive Oberfläche zur Verfügung stellt, mit der wir Daten von einem FTP-Server abholen und zu einem FTP-Server senden können.

Einen einfachen Transfer durchführen

Um Daten zu einem Server zu senden, wird der FTP-Befehl PUT verwendet. Mehrere Dateien übertragen Sie mit dem Befehl MPUT (Multiples Senden). Um Daten von einem Server zu holen, wird der FTP-Befehl GET verwendet. Mehrere Dateien holen Sie mit dem Befehl MGET (Multiples Holen). Wir werden jetzt versuchen, Daten aus der Bibliothek EPSSVFIL des Hosts ITPIS01.EPOSGMBH.COM zu holen (herunterzuladen).

Aber Halt! Wir haben zwar eine Verbindung zum FTP-Server des Systems ITPIS01.EPOSGMBH.COM aufgebaut, aber wo – in welcher Bibliothek dieser iSeries – befinden wir uns jetzt. Um diese Frage beantworten zu können, geben Sie einfach den Befehl PWD (Print Work Directory) ein.

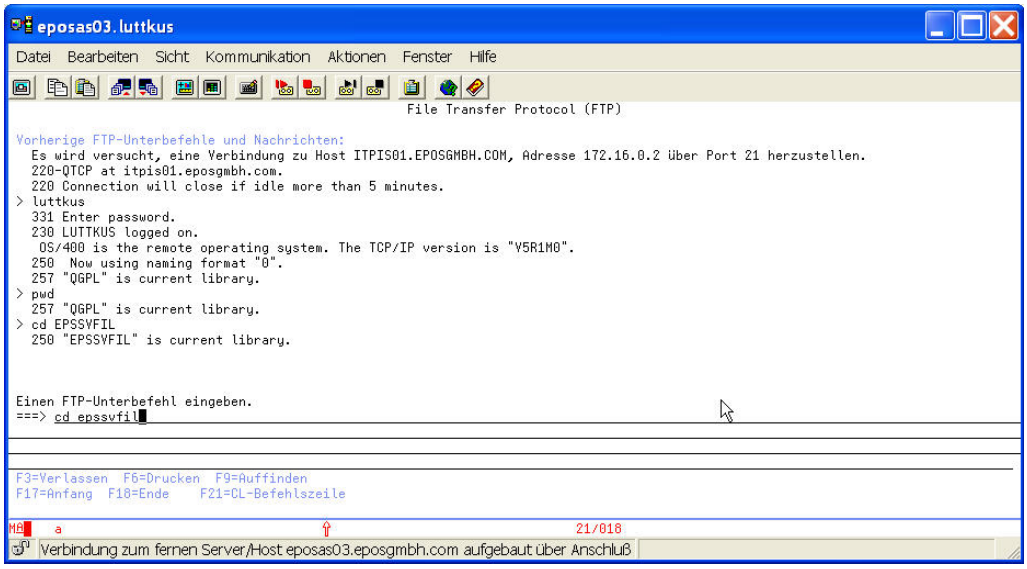


FTP-Befehl: PWD – Print Work Directory

Ist in dem Benutzerprofil, das die Verbindung zum FTP-Server steuert, eine aktuelle Bibliothek definiert worden, so erscheint jetzt der Name der aktuellen Bibliothek im Anzeigebildschirm des FTPs. Ist im Benutzerprofil jedoch keine aktuelle Bibliothek bezeichnet worden, so wird QGPL als aktuelle Bibliothek ausgegeben.

Da in meinem Benutzerprofil keine aktuelle Bibliothek bestimmt wurde, besteht jetzt eine Verbindung zur Bibliothek QGPL des entfernten Systems. Ich verwende den Befehl CD (Change Directory – Verzeichnis wechseln), um jetzt in die Bibliothek EPSSVFIL zu wechseln, und gebe folgenden Befehl ein: CD EPSSVFIL.

7.6.5.4
Seite 2



Mit CD in die Bibliothek EPSSVFIL wechseln

Existiert diese Bibliothek auf dem entfernten System und besitze ich Zugriffsrechte für diese Bibliothek, so wird die Meldung im FTP-Bildschirm ausgegeben, dass ab jetzt eine Verbindung zu dieser Bibliothek besteht.

Ich möchte jetzt Daten aus der Datei SVKDST00 (Kundenstammdaten) herunterladen. Diesen Vorgang realisiere ich mit dem FTP-Befehl GET. Das allgemeine Format dieses Befehls sieht wie folgt aus:

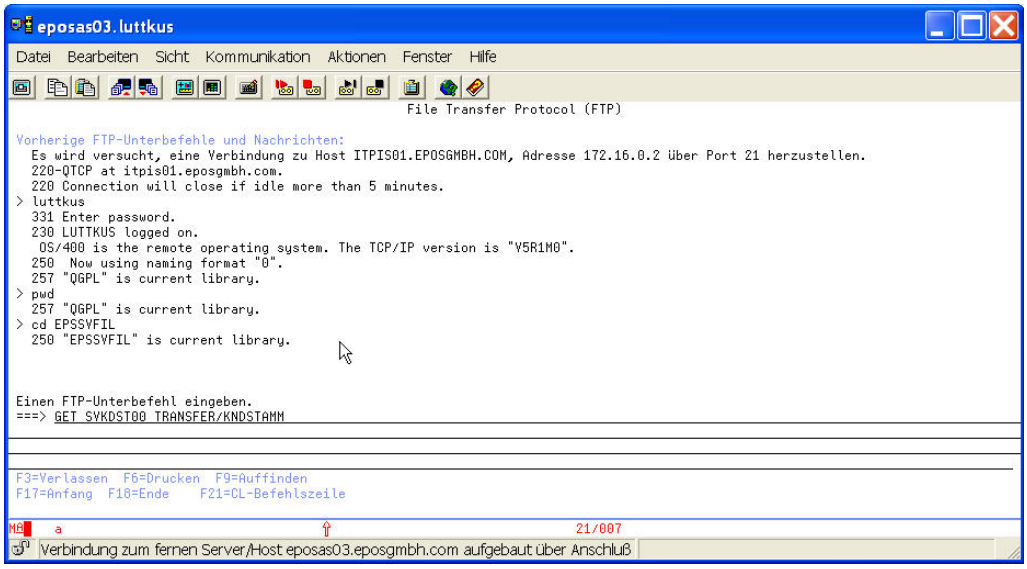
```
Get entfernte Datei [lokale Datei] [(Replace]
```

Entfernte Datei: Ist die Datei, die heruntergeladen werden soll. Die Angabe des gesamten Zugriffspfads ist möglich.

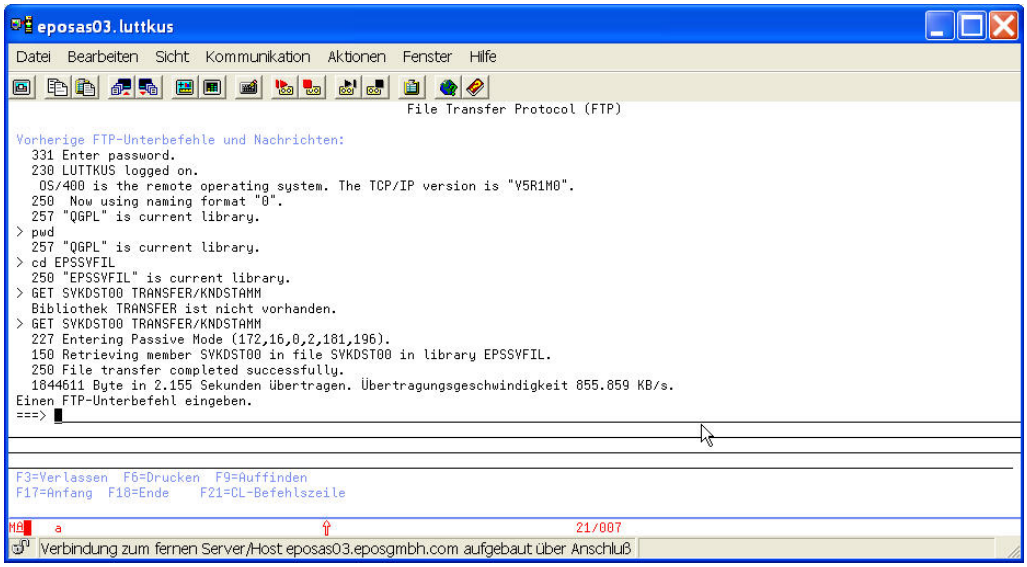
Lokale Datei: Wahlfreier Parameter. Falls die Datei auf dem lokalen System noch nicht vorhanden ist, kann Sie unter einem anderen Namen erstellt werden. Ist die Datei lokal nicht vorhanden und wird dieser zweite Parameter nicht genutzt, dann wird die Datei lokal unter dem gleichen Namen angelegt – wie auf dem „entfernten System".

Replace: Wahlfreier Parameter. Ist die Datei bereits lokal vorhanden und befinden sich in ihr Daten, so werden diese ersetzt, sofern dieser Parameter gesetzt wurde.

Ich gebe also folgende Befehlsfolge auf der Befehlszeile des FTP-Clients ein:



GET SVKDST00 TRANSFER/KNDSTAMM



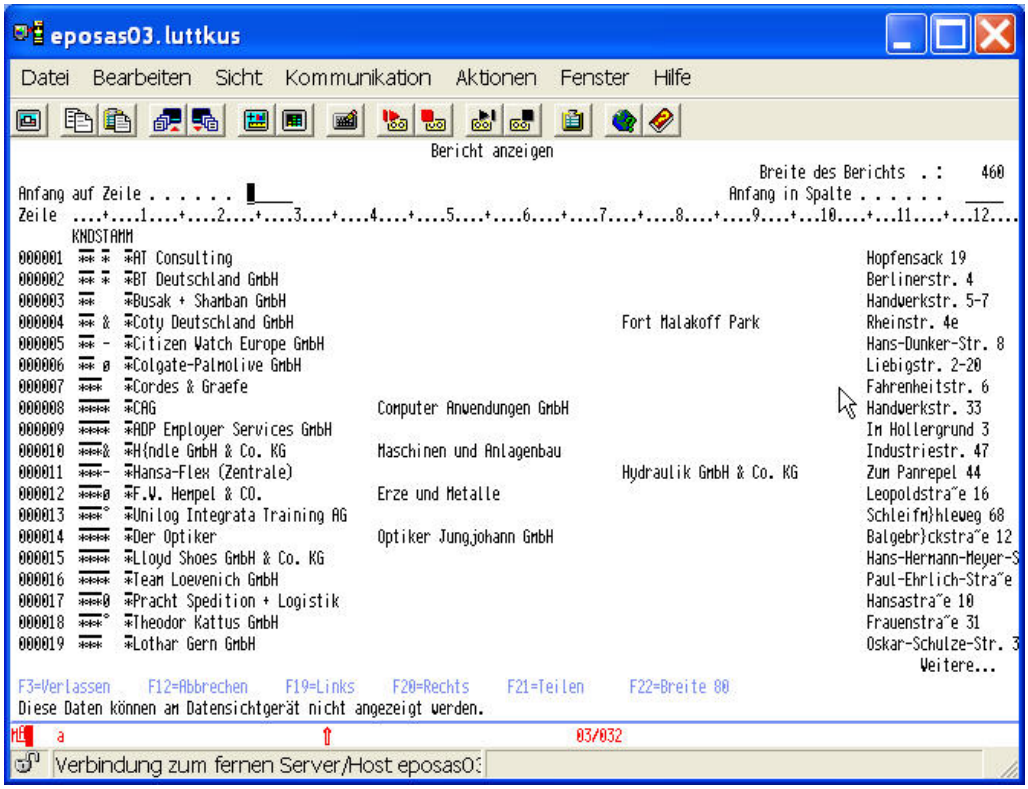
Der Transfer ist beendet

Nach Bestätigung der Eingabe mit Datenfreigabe wird der Befehl ausgeführt. Ist alles korrekt verlaufen, stehen jetzt die Daten der Remote-Datei SVKDST00 auf dem lokalen Host EPOSAS03.EPOSGMBH.COM in der Bibliothek TRANSFER in der Datei KNDSTAMM zur Verfügung.

Sehen wir uns jetzt einmal an, wie unsere Daten auf dem lokalen System EPOSAS03.EPOSGMBH.COM aussehen. Die Datei auf dem Host ITPIS01 war eine ganz normale physische Datei, deren Felder extern beschrieben wurden. Ich möchte damit sagen, dass die Feldbeschreibung für die Datei SVKDST00 Bestandteil der Objektbeschreibung der Datei ist.

7.6.5.4
Seite 4

Ich werde jetzt einfach den Befehl RUNQRY ausführen, um den Inhalt der Datei KNDSTAMM auf dem System EPOSAS03 anzuzeigen.



Anzeige der Daten in KNDSTAMM

Zwei Dinge fallen sofort auf:

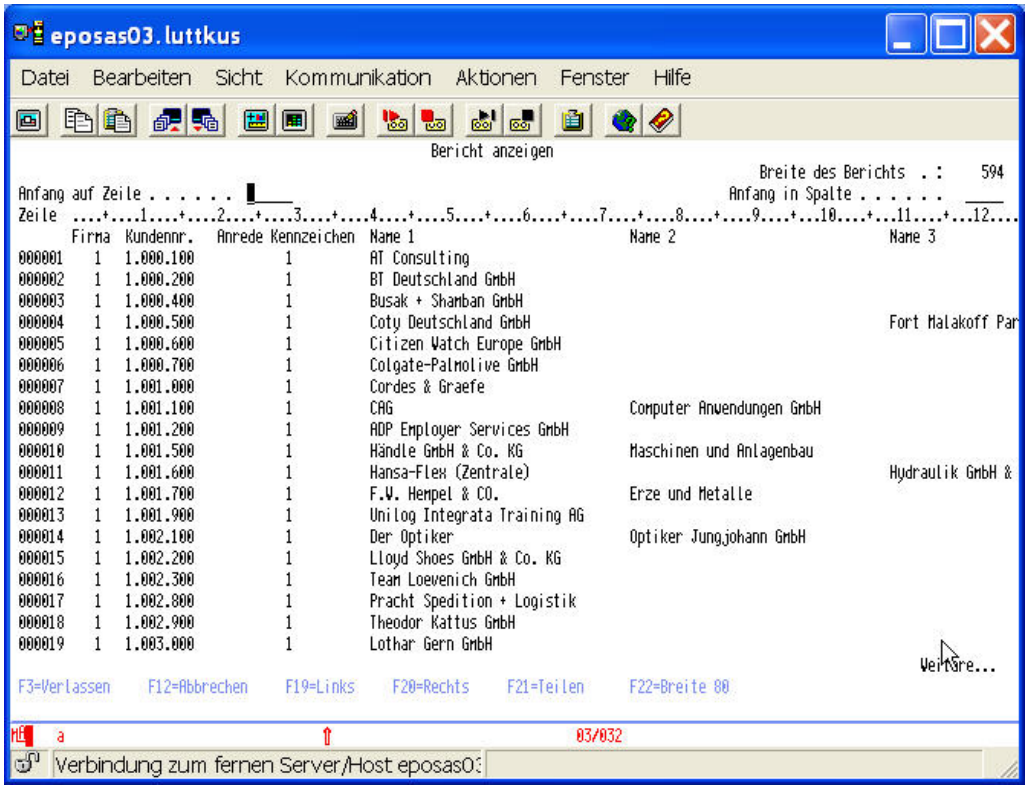
1.) Ein Datensatz besteht nur noch aus einem einzigen Feld, d.h.: Die Feldbeschreibungen wurden beim Transfer nicht mit übertragen.

2.) Am Anfang jedes Datensatzes befinden sich eigentlich drei numerische, gepackte Felder. Diese sind ebenfalls beim Transfer nicht korrekt umgesetzt worden.

Was können wir dagegen unternehmen? Der Trick liegt darin, dass wir die Zieldatei als Objekt erstellen.

Aber Vorsicht!

Erstellen Sie die Datei mit dem Befehl CRTPF (Create Physical File – Physische Datei erstellen) oder mit CRTDUPOBJ (Create Duplicate Object – Objekt duplizieren), so wird automatisch eine leere Teildatei angelegt. Diese verhindert den Transfer. Entfernen Sie also die leere Teildatei mit dem Befehl RMVM (Remove Member – Teildatei entfernen) und führen Sie dann den Transfer erneut durch.



Inhalt der Datei KNDSTAMM in erstellter Zieldatei

Sie sehen, dass beide oben angeführten Probleme in einem Arbeitsgang behoben wurden. Wir sehen anhand der Überschriften, dass externe Feldbeschreibungen angelegt sind und dass die gepackten numerischen Felder feldgerecht übertragen wurden.

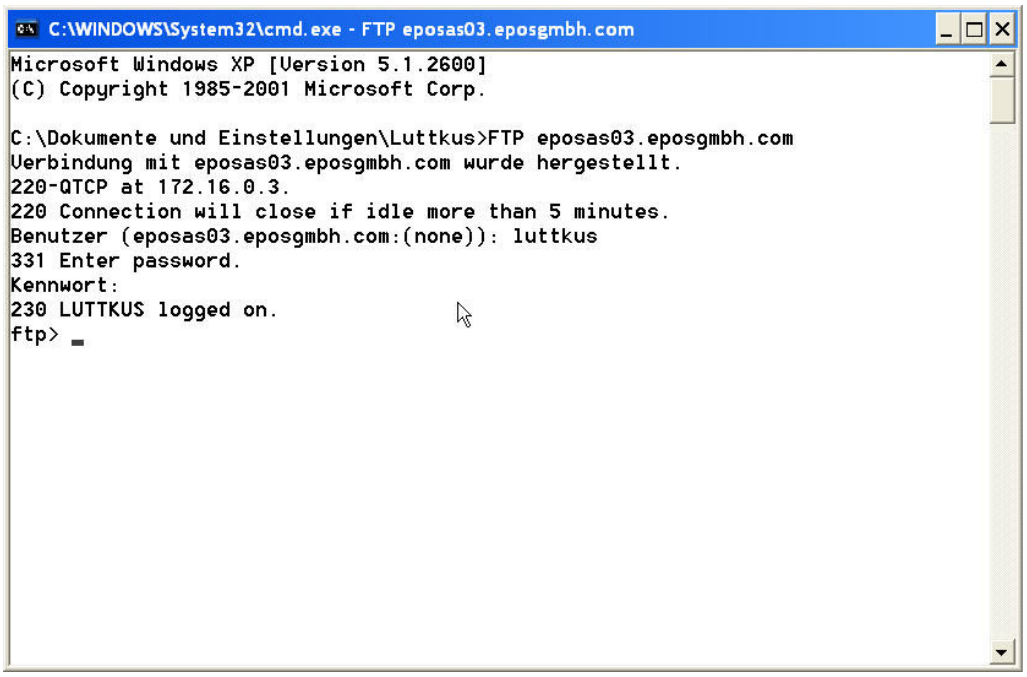
7.6.5.4

Seite 6

FTP auf dem Windows-PC

Haben Sie auf Ihrem Windows-PC – und damit meine ich einen PC unter Windows 95/98/ME sowie Windows NT/2000/XP – TCP/IP als Transportprotokoll installiert und konfiguriert, so stehen Ihnen auch hier die gängigen TCP/IP-Dienste zur Verfügung. Um den Windows-FTP-Client zu starten, gehen Sie einfach in die DOS-Box Ihres Windows-PCs und rufen dort den FTP-Client auf. Der Aufruf hat folgendes allgemeines Format:

FTP <Adresse bzw. Host-Name des FTP-Servers>



Der Verbindungsdialog

Da ich auf der iSeries den FTP-Server konfiguriert und gestartet habe und diese iSeries via TCP/IP erreichbar ist, stelle ich die Verbindung mit folgender Syntax her:

FTP 172.16.0.3 bzw. FTP eposas03.eposgmbh.com

In der ersten Variante dieser Adresse benutze ich die IP-Adresse, in der zweiten den Host-Namen, der entweder in der Host-Tabelle des Clients oder in einem DNS (Domain Name Service) hinterlegt wurde. Sollte die Verbindung zustande gekommen sein, so werden Sie aufgefordert, sich an der iSeries anzumelden. Sie sehen den folgenden Dialog:

7.6.5.5
Seite 2

Zeilen-Nr.	
10	ftp eposas03.eposgmbh.com
20	Verbunden zu EPOSAS03.EPOSGMBH.COM.
30	220-QTCP at EPOSAS03.EPOSGMBH.COM.
40	220 Connection will close if idle more than 5 minutes.
50	Benutzer (EPOSAS03.EPOSGMBH.COM:(none)): luttkus
60	331 Enter password.
70	Kennwort:
80	230 LUTTKUS logged on.
90	Ftp>

Die Zeilennummern habe ich eingefügt; sie dienen der besseren Übersicht.

- In Zeile 10 sehen Sie den DOS-Prompt mit dem Aufruf des Befehls „FTP“.
- Zeile 20 ist eine Statusmeldung des FTP-Clients und spricht für sich selbst.
- Zeile 30 ist eine Statusmeldung des FTP-Servers, die besagt, dass mit Hilfe des Benutzers QTCP die Verbindung hergestellt wurde.
- Zeile 40 besagt, dass die Verbindung wieder aufgegeben wird, sofern der Benutzer innerhalb von 5 Minuten keinen FTP-Befehl eingeben wird. Sollte das einmal der Fall sein, so lässt sich die FTP-Sitzung mit dem Befehl OPEN wieder aufnehmen.
- Zeile 50 fordert dann die Eingabe eines aktiven iSeries-Benutzerprofils an.
- Zeile 60 fordert die Eingabe des entsprechenden Kennworts an.
- Zeile 70 stellt das Eingabefeld zur Verfügung. Eingaben sind nicht sichtbar.
- Zeile 80 ist eine Statusmeldung des FTP-Server-Jobs der iSeries und besagt, dass die Anmeldung korrekt war.
- Zeile 90 ist der Eingabe-Prompt für FTP-Befehle.

An dieser Stelle können wir jetzt FTP-Befehle eingeben. Bevor wir jedoch damit beginnen, einen Transfer auszuführen, sollten wir uns einige Gedanken bezüglich der Fähigkeiten von FTP machen. Welche Daten lassen sich mit Hilfe von FTP übertragen? Die Antwort ist eigentlich simpel: Alles was auf der iSeries in irgendeiner Form Objektstatus hat, kann mittels FTP übertragen werden, d.h.: also nicht nur Datenbankinhalte, sondern rein theoretisch auch Programme, Bildschirmbeschreibungen usw. Soweit, so gut.

Was für uns jedoch wichtig ist, ist der Standort der Daten. Bedenken Sie, dass wir es auf der iSeries mit zwei unterschiedlichen Datenhaltungssystemen zu tun haben: einmal dem Integrated File System, in dem Daten in hierarchischer Ordnung abgelegt werden, auf der anderen Seite das Bibliothekssystem, das keine hierarchische Ordnung kennt.

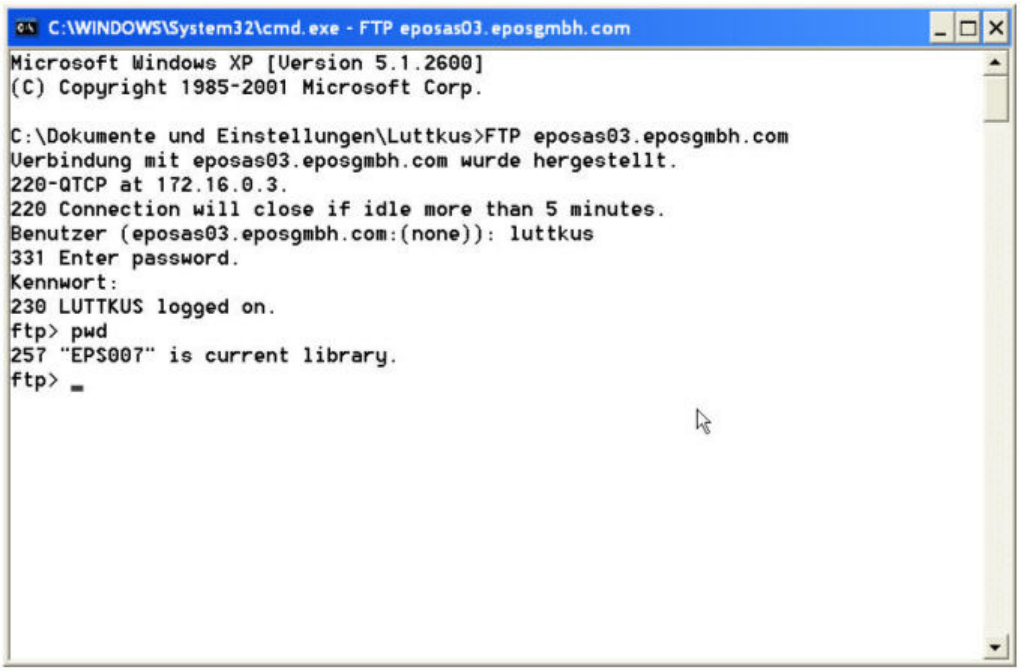
7.6.5.5

Seite 4

Das iSeries-Bibliothekssystem

Haben wir eine FTP-Sitzung aufgebaut, so greifen wir über das iSeries-Bibliothekssystem auf unsere Daten zu. Das bedeutet für uns, dass wir mittels FTP nur auf die Datenbankobjekte der iSeries zugreifen können. Dies ist für uns ebenfalls Stand der Dinge. Auf welche Datenbankobjekte können wir jetzt zugreifen. Auch hier wieder eine einfache Antwort: Auf alle Datenbankdateien, für die wir die entsprechenden Zugriffrechte besitzen, da wir auf solche Objekte mit Hilfe der Pfadangabe „Bibliothek/Datenbank-Objekt“ zugreifen. Dieses Format wird von FTP als das Namensformat „0“ (NAMEFMT 0) bezeichnet.

Die für uns interessante Frage ist jetzt sicherlich, in welcher Bibliothek wir jetzt stehen. Diese Frage lässt sich mit Hilfe des Benutzerprofils beantworten, mit dem wir die FTP-Sitzung aufgebaut haben. Ist über das verwendete Benutzerprofil keine aktuelle Bibliothek bezeichnet worden, so befinden wir uns nach Aufbau der FTP-Sitzung in der Bibliothek QGPL. Wird mittels des Benutzerprofils eine Bibliothek bezeichnet, so befinden wir uns nach FTP-Sitzung in der bezeichneten Bibliothek. Im Benutzerprofil „Luttkus“ ist EPS007 als aktuelle Bibliothek bezeichnet – also befinden wir ins in der Bibliothek QGPL.



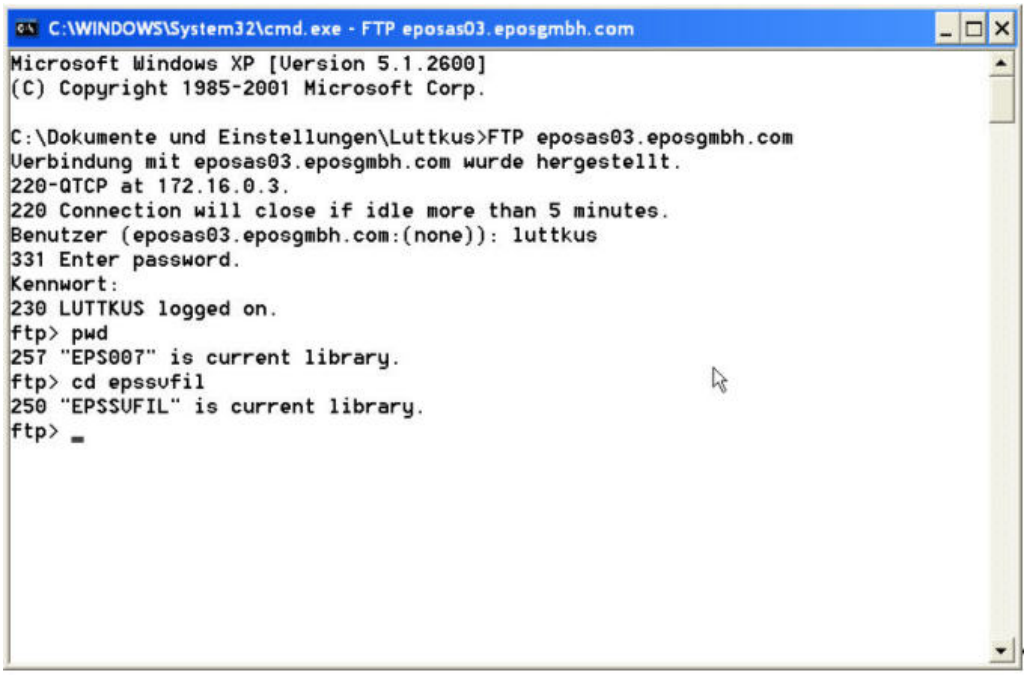
PWD – Print Work Directory

Mit dem Befehl PWD (Print Work Directory) können wir uns unseren aktuellen Standort anzeigen lassen.

7.6.5.6
Seite 2

```
Ftp> pwd
257 "EPS007“ is current library.
Ftp>
```

Um die Bibliothek zu wechseln, nutzen wir den Befehl CD (Change Directory). Dieser Befehl wechselt auf dem Server-System die aktuelle Bibliothek.



In die Bibliothek EPSSVFIL wechseln

Ich werde jetzt in die Bibliothek EPSSVFIL wechseln und gebe folgenden Befehl am FTP-Prompt ein:

```
Ftp> cd epssvfil
250 "EPS007“ is current library.
Ftp>
```

Mit Hilfe von GET-Befehlen können jetzt Daten an den PC-Client heruntergeladen werden. Mit PUT-Befehlen übertragen wir Daten vom Client auf die iSeries.

Das Integrated File System

Beabsichtigen wir jedoch nicht nur Daten aus dem Datenbankbereich der iSeries zu übertragen, so kommen wir mit dem bisher gesehenen nicht weiter. Lassen Sie uns ein einfaches Beispiel entwickeln: Wir beabsichtigen ein Dokument aus dem Verzeichnis „Home“, das auf dem Host EPOSAS03.EPOSGMBH.COM eingerichtet wurde, an einen PC zu übertragen. Hierbei handelt es sich um ein Dokument mit dem Namen ADRESSEN.TXT.

Wenn wir jetzt einfach mit dem Befehl CD in den Ordner PCQTEMP wechseln, so erhalten wir eine Fehlermeldung.

```
Ftp> cd pcqtemp
550-Specified library does not exist or cannot be accessed.
550 Current library not changed.
Ftp>
```

Aus welchem Grund erhalten wir diesen Fehler? Die iSeries-Verzeichnisse sind nicht Bestandteil des Bibliothekssystems, sondern sie sind als hierarchischer Lagerort direkter Bestandteil des Integrated File Systems (IFS) und können damit auch nur über das IFS angesprochen werden. Beabsichtigen wir unsere Daten über das IFS anzusprechen, so muss der FTP-Server der iSeries auf das Namensformat „1“ umgestellt werden. Jetzt greifen wir auf unsere Daten mittels der folgenden Pfadangabe durch:

/Verzeichnis/.../.../Objekt.Extension

Auf unser Beispiel bezogen wäre die korrekte Pfadangabe:

/HOME/ADRESSEN.TXT

Also modifizieren wir die vorhin verwendete CD-Anweisung und geben Folgendes ein:

```
Ftp> cd /home
250-NAMEFMT set to 1.
250 Current directory changed to /home.
Ftp>
```

Sie sehen, dass die modifizierte Eingabe des Befehls CD das Namensformat auf den Wert „1“ gesetzt hat. Alle objektbezogenen Zugriffe werden jetzt IFS-bezogen dargestellt. Ich gebe jetzt einfach den Befehl PWD ein.

```
Ftp> pwd
257  Current directory is "/home“.
Ftp>
```

7.6.5.7
Seite 2

Auch hier sehen Sie jetzt die entsprechende pfadbezogene Syntax.

Jetzt schreiten wir zur Tat und übertragen das Dokument ADRESSEN.TXT aus dem IFS-Pfad „/home“ in das Verzeichnis D:\LUTTKUS meines PCs. Das erreichen wir mit Hilfe des Befehls GET:

```
Ftp> get adressen.txt d:\luttkus
```

Befinden wir uns nicht in dem Verzeichnis, aus dem Daten transferiert werden, so kann das Quellenobjekt auch mit seiner gesamten Pfadangabe angesprochen werden:

```
Ftp> get /home/adressen.txt d:\luttkus
```

Beabsichtigen wir jetzt, Daten (SVKDST00) aus einer Datenbankdatei (EPSSVFIL) zu übertragen, so muss die Syntax folgendermaßen aussehen:

```
Ftp>
get /qsys.lib/epssvfil.lib/svkdst00.file
200 PORT subcommand request successful.
150 Retrieving member SVKDST00 in file SVKDST00 in library EPSSVFIL.
250 File transfer completed successfully.
1658652 Bytes empfangen in 2,39 Sekunden (693,13 KB/s)
Ftp>
```

Beachten Sie, dass die Objektarten der iSeries als Namenserweiterungen den Objektnamen hinzugefügt werden müssen!

Worin besteht – abgesehen von den unterschiedlichen Syntaxen – der Unterschied zwischen Transfers mit bibliotheksbezogenen und pfadbezogenen Namen? Existiert beim bibliotheksbezogenen Transfer das Zielobjekt nicht, so wird eine intern beschriebene 1-Feld-Datei erstellt. In diese Datei werden dann die Daten transferiert. Den Effekt haben wir vorher gesehen.

Benutzen wir jedoch die pfadbezogene Syntax, so wird ein Objekt-Transfer durchgeführt, der sowohl die Objektbeschreibung als auch die Daten überträgt. D.h.: Die pfadbezogene Syntax sorgt dafür, dass auf dem Zielsystem des Transfers eine Datei erstellt wird, die ein Abbild der Quelldatei ist. In diese Quelldatei werden die Daten gestellt.

Erfolg! Die Daten wurden übertragen. Jedoch: Sie werden jetzt sagen, diese Pfadangabe, die relativ lang und tippfehleranfällig ist, könnte auch vermieden werden, wenn man das bibliotheksbezogene Namensformat wieder zurückstellen könnte. Nichts einfacher als das. Hierzu verwenden wir einen Befehl Namens QUOTE, der dazu dient, so genannte Subcommands an den FTP-Server der iSeries zu übertragen.

```
Ftp> quote site namefmt 0
250  Now using naming format "0".
Ftp>
```

Der gleiche Transfer wie eben, sieht im Namensformat „0“ dann so aus:

```
Ftp> get svkdst00
200 PORT subcommand request successful.
150 Retrieving member SVKDST00 in file SVKDST00 in library EPSSVFIL.
250 File transfer completed successfully.
1658652 Bytes empfangen in 2,36 Sekunden (701,63 KB/s)
Ftp>
```

Wollen Sie das Namensformat des Zugriffs auf bestimmte Daten ändern, so geben Sie folgende FTP-Befehle ein:

```
Ftp> quote site namefmt 0
```

Bibliotheksbezogener Zugriffspfad ist eingestellt. Achtung: Nur Objekte, die mit Hilfe der Bibliothek QSYS verwaltet werden, können übertragen werden.

```
Ftp> quote site namefmt 1
```

IFS-verzeichnisbezogener Zugriffspfad ist eingestellt. Jedes Objekt des gesamten Integrated File Systems kann übertragen werden.

7.6.5.7
Seite 4

FTP mit grafischen Clients

Der Austausch von Daten von und zu Windows-Betriebsystemen kann, wie sie bereits gesehen haben, mit dem On-Board-FTP-Client realisiert werden. Wenn wir jedoch bedenken, dass wir uns heute im Zeitalter grafischer Benutzerschnittstellen befinden, so muss doch mehr möglich sein, als in der DOS-Box von Windows Befehle einzugeben.

Auf dem Markt befindet sich eine Reihe von Windows FTP-Clients, mit denen wir sozusagen per „Drag and Drop" Daten von einer iSeries zum Windows-PC und umgekehrt übertragen können.

Ein Produkt, mit dem ich persönlich sehr gerne arbeite, ist WS FTP Pro. Hierbei handelt es sich nicht um eine Freeware, sondern um eine kostenpflichtige Software der Firma Ipswitch. Warum ich nicht eines der viele Free- bzw. Shareware-Produkte nutze? Ganz einfach: WS FTP Pro hat eine sehr gute Anpassung an iSeries-FTP-Server, die mir an anderen Produkten in dieser Form nicht begegnet ist.

WS FTP Pro beinhaltet zwei Programme:

- den WS FTP Pro Windows-Client, mit dem wir Daten vom PC zur iSeries und von der iSeries zum PC übertragen können,
- den WS FTP Pro Explorer, mit dem wir mehrere FTP-Server in einen Verzeichnisbaum einbinden, so dass wir Daten von iSeries zu iSeries oder zu jedem anderen Server transferieren können.

Informationen zu WS FTP Pro finden Sie im Internet unter www.ipswitch.com. Eine Download-Version, die voll funktional ist und die sich für 30 Tage betreiben lässt, finden Sie ebenfalls unter diesem Link.

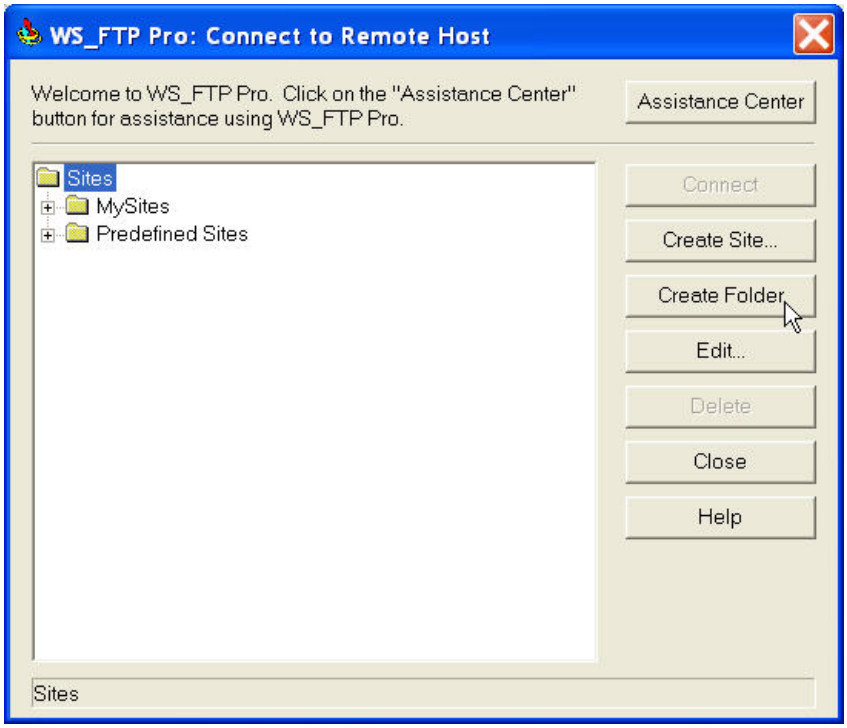
7.6.5.8

Seite 2

Objekte mit WS FTP Pro transferieren

Nach der Windows-typischen Installation von WS FTP Pro finden Sie auf dem Desktop Ihres Windows-PCs ein Symbol für den Aufruf des Programms.

Nachdem das Programm das erste Mal aufgerufen wurde, wird sofort der folgende Dialog angezeigt.



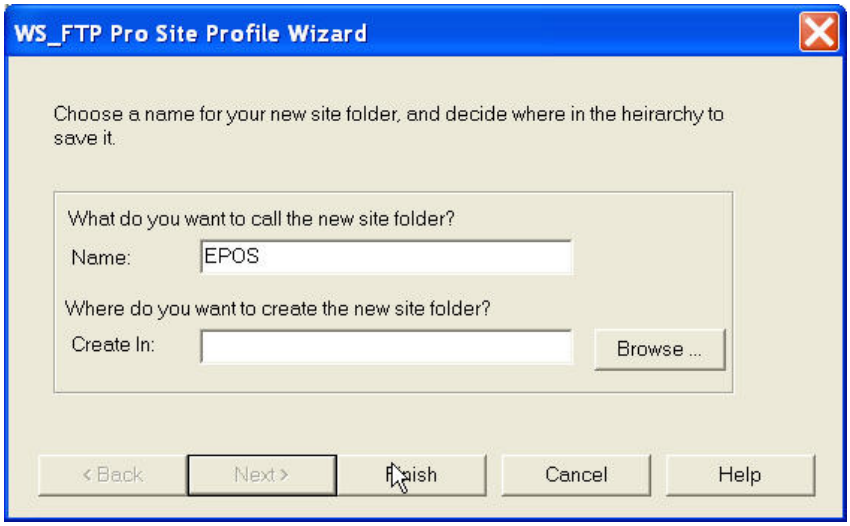
Verbindungen konfigurieren

In diesem Dialog geht es darum, Verbindungen zu einem oder mehreren FTP-Servern zu konfigurieren. Dabei ist es unerheblich, ob der angesprochene FTP-Server eine iSeries ist oder ein anderes System. Wichtig ist jedoch, dass man Verbindungen in eine Art Verzeichnissystem hierarchisch gliedern kann, so dass die verschiedensten Verbindungen übersichtlich angeordnet werden können.

Ich markiere in obigem Dialog das Root-Verzeichnis „Sites“.

Ich werde zuerst ein Verzeichnis EPOS erstellen, deshalb drücke ich „Create Folder“. Übrigens: Die hier gezeigte Version von WS FTP Pro ist die englischsprachige. Es sei hier angemerkt, dass auch deutsche Versionen zur Verfügung stehen.

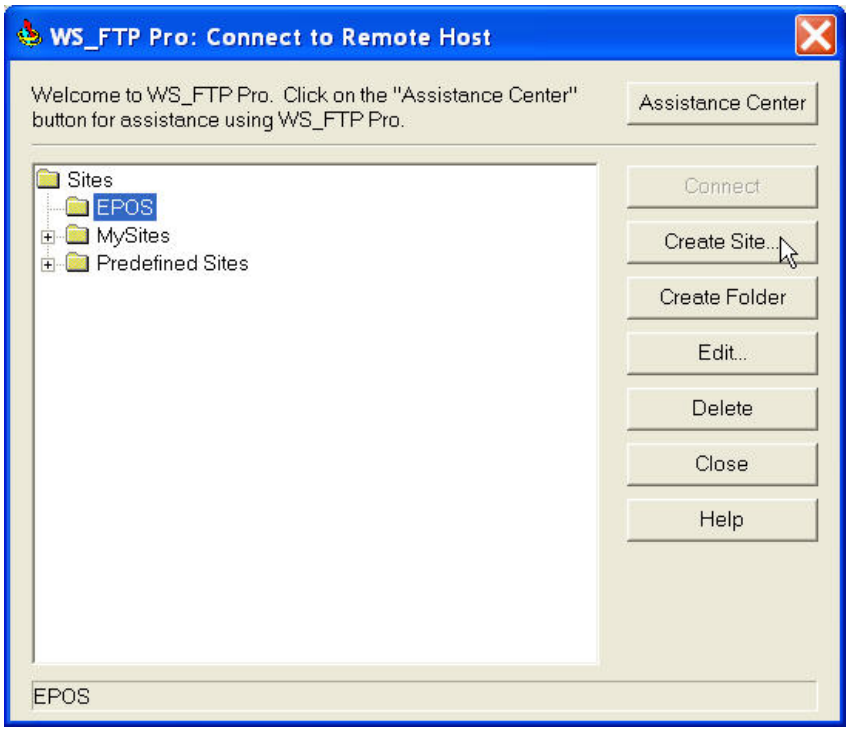
7.6.5.8.1
Seite 2



Verzeichnis erstellen

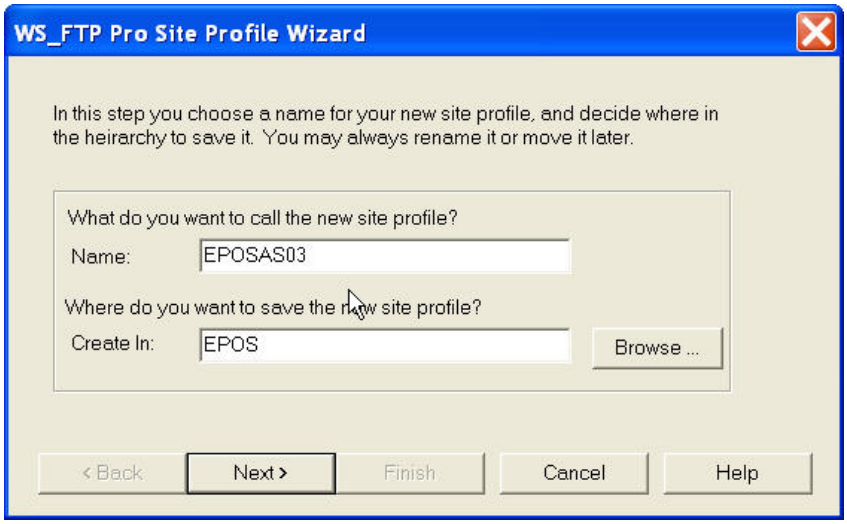
Im Parameter „Name“ trage ich EPOS ein und drücke dann „Finish“.

Sie sehen im nachfolgenden Dialog, dass das Verzeichnis EPOS erstellt wurde. Jetzt muss eine erste Verbindung zu einem FTP-Server erstellt werden.



Verbindung konfigurieren

Ich markiere das Verzeichnis EPOS und drücke „Create Site…“. Folgender Dialog öffnet sich:



Create Site

Unter „Name“ trage ich einen Verbindungsnamen ein. Dieser Name ist im WS FTP Pro ein reiner Anzeigename für die Verbindungsaufnahme. Unter TCP/IP hat dieser Name keine Funktion.

Dann beende ich den Dialog mit „Next“.

Im Folgenden muss jetzt die IP-Adresse oder der Host-Name des FTP-Servers eingegeben werden.

Adresse eingeben

Ich beabsichtige auf das in den vorherigen Beispielen bereits bekannte System mit dem Host-Namen EPOSAS03.EPOSGMBH.COM zuzugreifen. Deshalb trage ich diesen Namen in das Feld „Name“ ein.

Der Dialog wird wieder mit „Next“ beendet.

7.6.5.8.1
Seite 4

Im nächsten Dialog geht es um die Anmeldung am FTP-Server und um Zugriffsrechte auf dem Client.

Anmeldungen

Zu diesem Zweck werden drei Parameter zur Verfügung gestellt:

User Id: Hier trage ich den Namen eines aktiven iSeries-Benutzerprofils ein – in diesem Falle LUTTKUS.

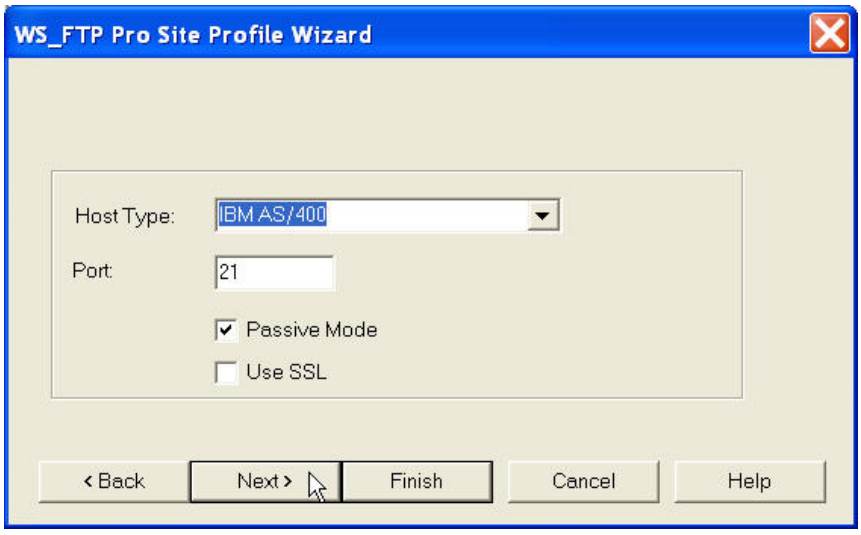
Passwort: Hier wird das Kennwort des Benutzers LUTTKUS eingetragen. Ich aktiviere die Option „Save Password“, so dass bei wiederholten Anmeldungen am iSeries-FTP-Server keine Anmeldedialoge initiiert werden.

Account: Dieser Parameter ist nur für Windows NT/2000/XP aktiv. Hier gebe ich an, ob von der iSeries überhaupt Daten auf den Client-PC übertragen werden können! Ich trage hier den Namen eines Windows-Benutzerkontos ein – in diesem Fall LUTTKUS. Wenn jetzt Daten von der iSeries zum PC übertragen werden, so überprüft das Windows-Berechtigungssystem, 1.) ob ein Benutzer „Luttkus“ lokal auf dem PC angelegt ist. Ist das nicht der Fall, so können die Daten nicht transferiert werden. 2.) Falls das Benutzerkonto „Luttkus“ lokal existiert, wird geprüft, ob dieses Konto die Rechte hat, Daten in das gewählte Verzeichnis zu übertragen. Ist das nicht der Fall, so wird der Transfer nicht durchgeführt.

Auch diesen Dialog beende ich mit „Next“.

In diesem Dialog geht es darum, den Server-Typ einzustellen. Eigentlich ist FTP ein standardisiertes Protokoll. Es lässt sich jedoch nicht leugnen, dass die FTP-Server unterschiedlicher Hersteller gewisse Eigenarten haben. Denken Sie nur einmal an die Gegensätzlichkeiten von dem iSeries Integrated File System und dem Bibliothekssystem.

Einer der Vorteile von WS FTP Pro ist es, dass ich den Server-Typ direkt angeben kann, mit dem ich FTP-Verbindungen aufbauen will.



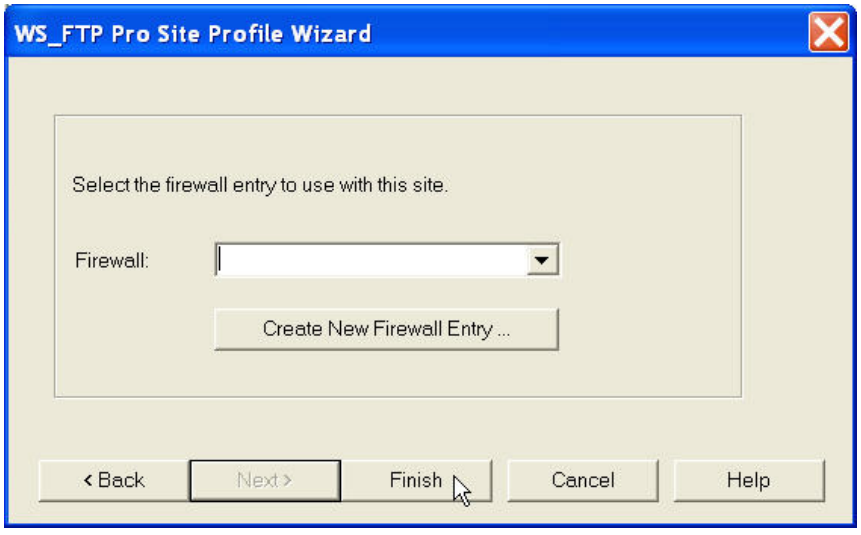
Server-Typ einstellen

Im Parameter „Host Type“ expandiere ich das Pulldown und markiere „IBM iSeries“.

Die anderen Parameter belasse ich auf ihren Standardwerten.

Auch diesen Dialog verlasse ich mit „Next“.

Im nächsten Dialog werden dann Firewall-spezifische Parameter abgefragt:

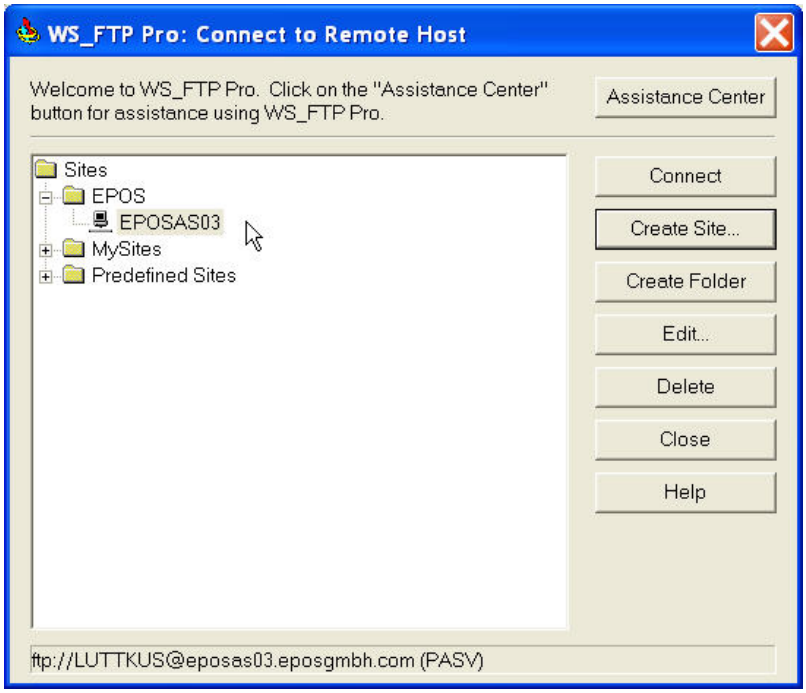


Firewall-Einträge

Im obigen Dialog habe ich keine Werte eingetragen und beende den Dialog mit „Finish“.

7.6.5.8.1
Seite 6

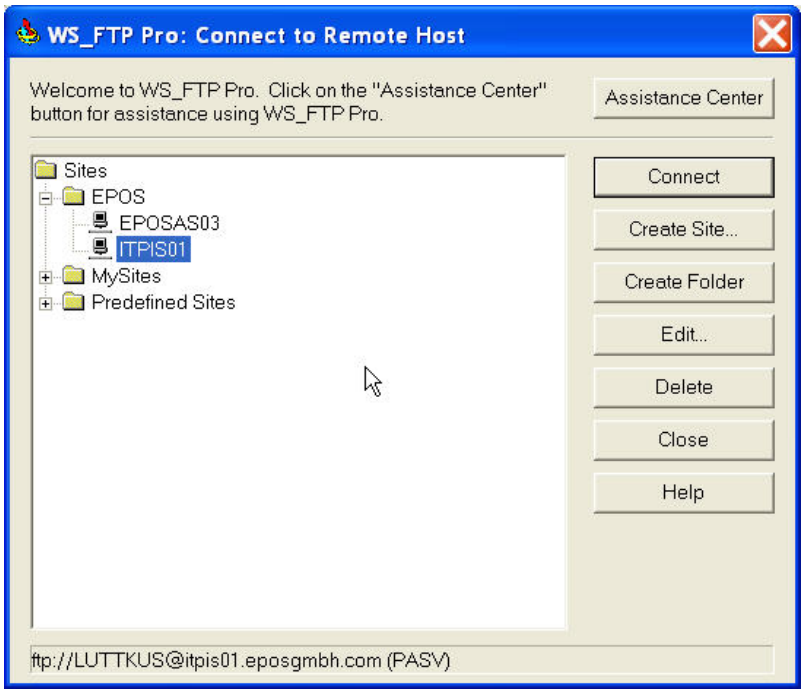
Im Folgenden sehen Sie, dass die neue Verbindung unter dem Namen EPOSAS03 unterhalb des Verzeichnisses EPOS eingetragen wurde.



Verbindung EPOSAS03

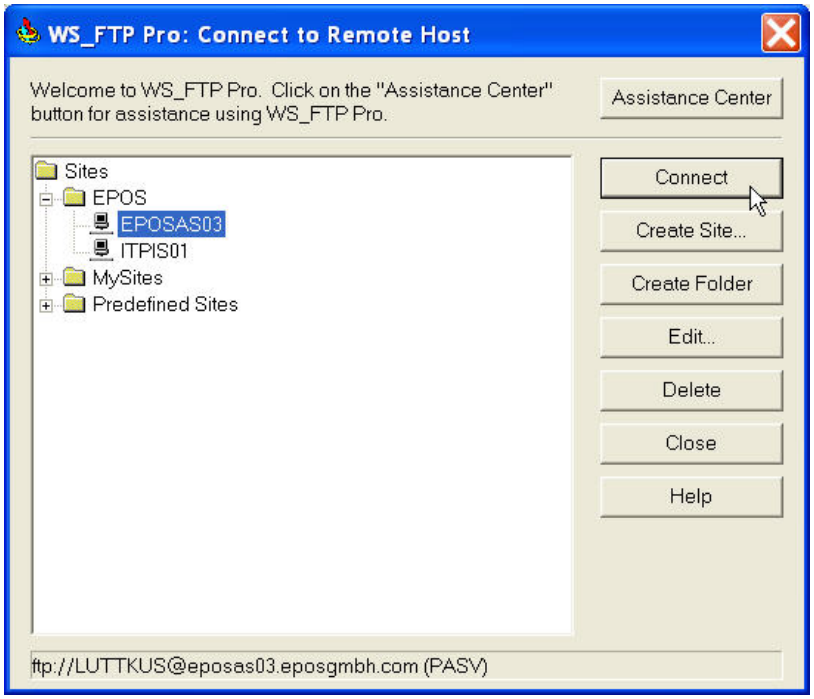
Danach konfiguriere ich eine weitere Verbindung zum System ITPIS01.

Im nächsten Dialog sehen Sie den entsprechenden Eintrag.



Verbindung ITPIS01

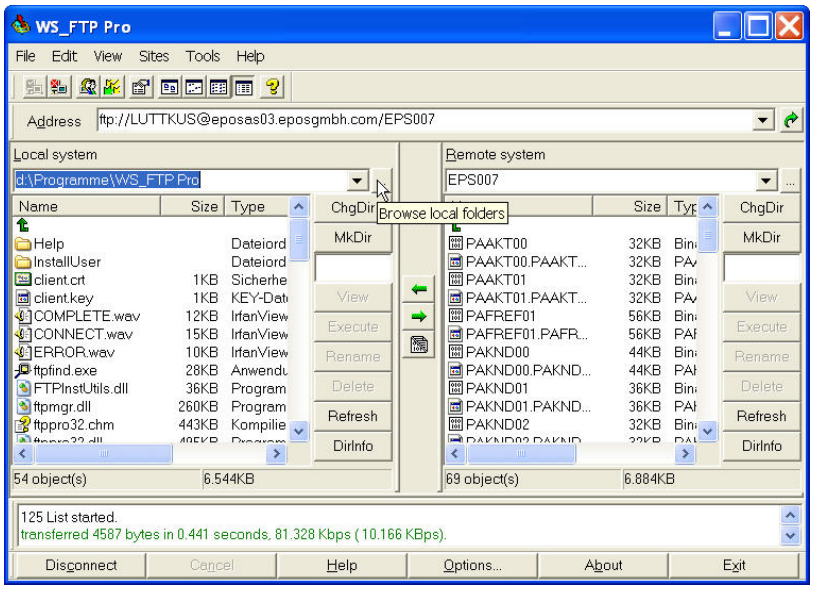
Jetzt markiere ich mit der Maus den Eintrag EPOSAS03 und drücke auf „Connect“.



Verbindung aufbauen

Sind alle Parameter bei der Verbindungseinrichtung richtig gesetzt worden, so öffnet sich der Transfer-Dialog.

Dieser Dialog ist in zwei Seiten unterteilt: links die Darstellung der lokalen Plattenspeicher, rechts die Darstellung des Plattenspeichers der iSeries.



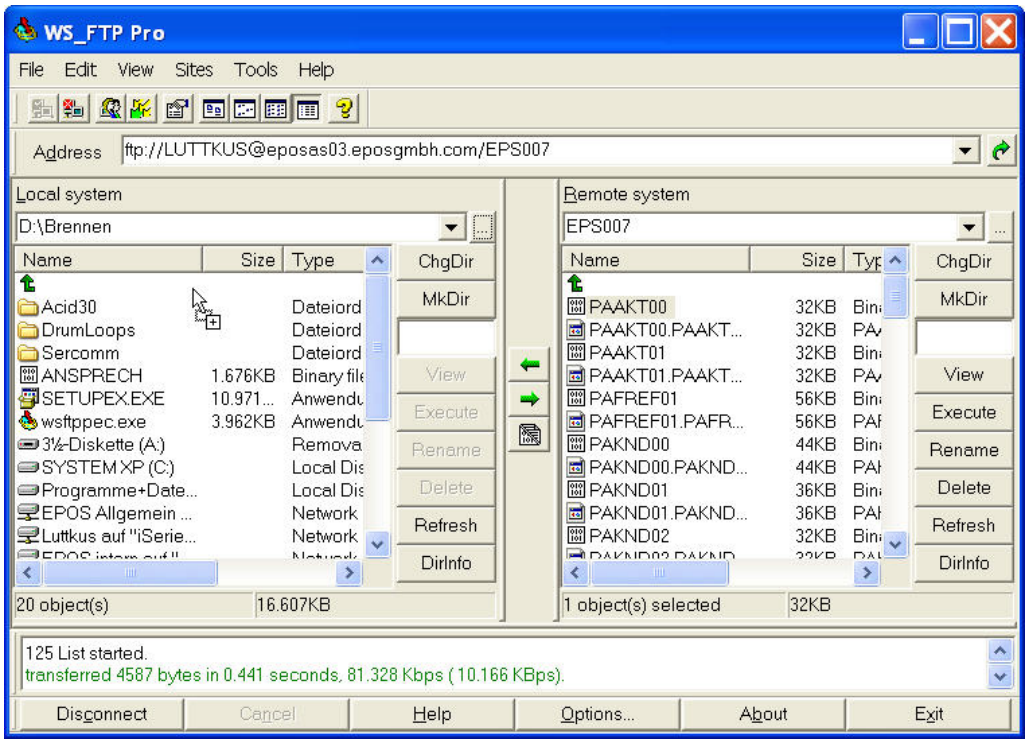
Transfer-Dialog

7.6.5.8.1
Seite 8

Im lokalen Bereich wird der WS FTP Pro-Pfad als Default angezeigt. Auf der rechten entfernten Seite wird die aktuelle Bibliothek des Benutzers, der die Verbindung zur iSeries hergestellt hat, angezeigt.

Beide Seiten lassen sich durch Drücken des „...“-Knopfes einstellen.

Sollen Daten von der iSeries zum Windows-PC übertragen werden, so wird der Transfer einfach durch eine „Drag-and-Drop“-Operation ausgeführt.



Daten transferieren

Dazu fasse ich die gewünschte Datei – hier PAAKT00 – mit der Maus an und ziehe bei gedrückter Maustaste die Datei in die lokale Seite.

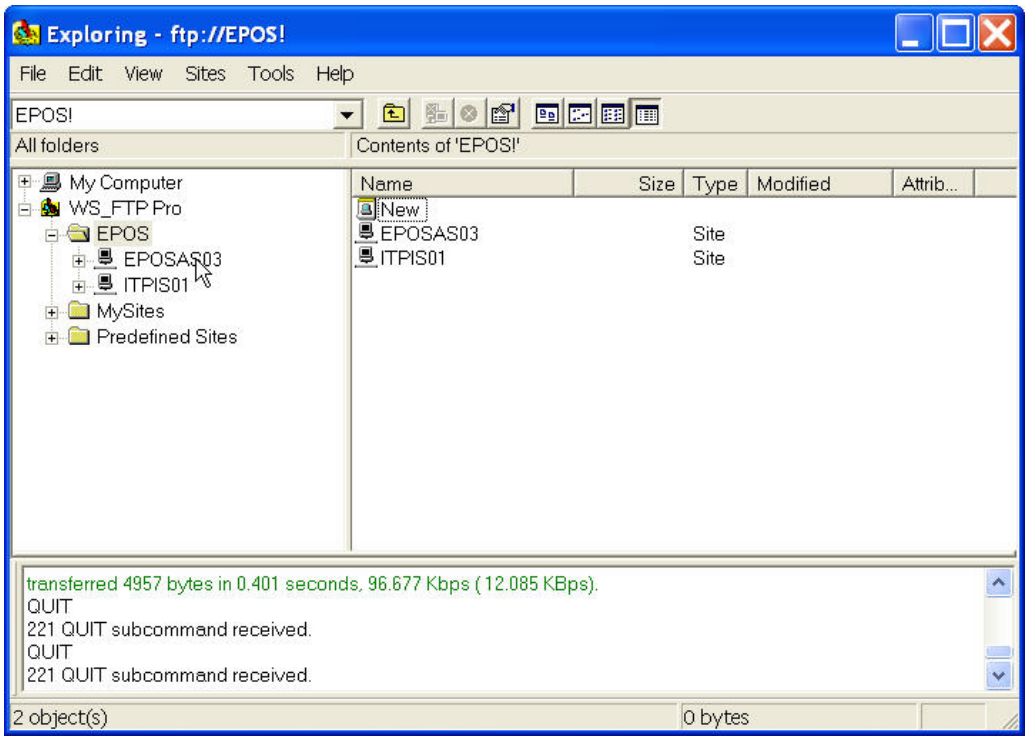
Danach lasse ich die Maustaste los. Das war’s!

Der WS FTP Pro Explorer

Ein zweites Programm, das Sie mit WS FTP Pro erhalten, ist der WS FTP Pro Explorer. Mit diesem Programm können Sie auf einfache Weise Daten zwischen verschiedenen FTP-Servern transferieren.

Über ein entsprechendes Icon auf dem Desktop rufe ich den FTP Pro Explorer auf.

Folgender Dialog öffnet sich:



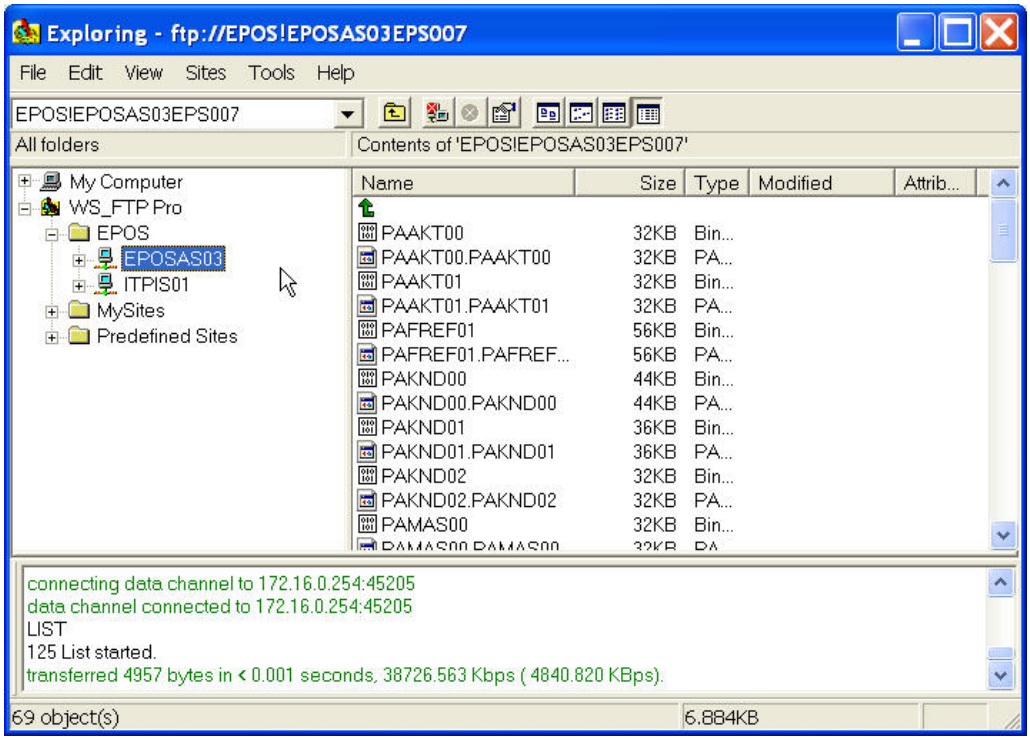
WS FTP Pro Explorer

Hier sehen wir die im vorherigen Abschnitt definierten Verbindungen wieder. Sie erkennen unschwer das Verzeichnis EPOS mit den beiden Verbindungen EPOSAS03 und ITPIS01.

Sind die Bildschirm-Symbole vor den Verbindungsnamen geschwärzt, so besteht keine Verbindung zu diesen Systemen.

Ein Klick auf den jeweiligen Verbindungsnamen genügt, um eine Verbindung zu aktivieren.

7.6.5.8.2
Seite 2



Aktivierte Verbindungen

Ein Datentransfer von EPOSAS03 zu ITPIS01 ist wiederum eine einfache „Drag-and-Drop"-Operation. Ich fasse die Datei PAAKT00 einfach mit der Maus an und ziehe Sie über den Verbindungsnamen ITPIS01.

Dann beginnt der Transfer. Sollte auf dem Zielsystem eine Datei gleichen Namens existieren, so erscheint eine Sicherheitsabfrage, ob die bestehende Datei überschrieben werden oder der Transfer abgebrochen werden soll.

FTP im Batch

Sowohl der FTP-Client der iSeries, als auch der FTP-Client von Windows können im Batch-Betrieb ausgeführt werden. In den folgenden Abschnitten werde ich das Vorgehen für beide Clients beschreiben.

7.6.5.9

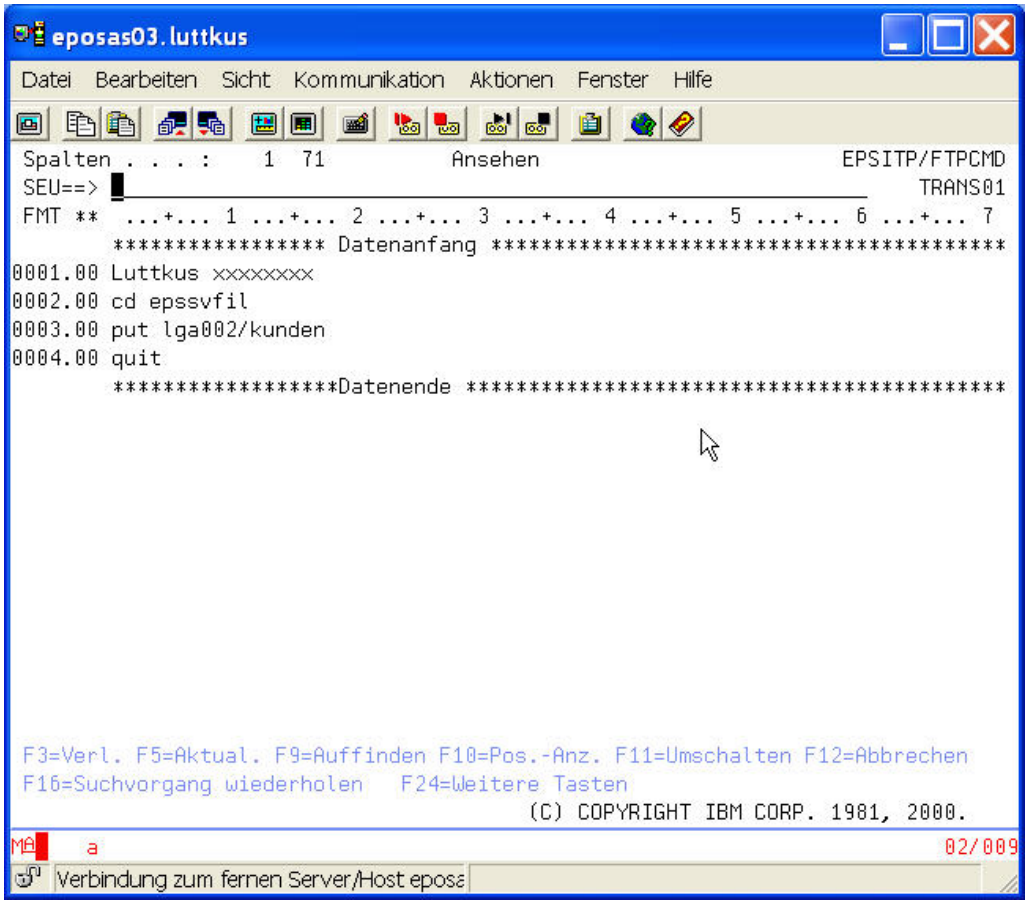
Seite 2

iSeries-FTP im Batch

Der FTP-Client der iSeries ist eigentlich eine rein interaktive Komponente, jedoch lässt sich die Interaktivität einfach beseitigen. Der Bildschirm wird vom FTP-Client als so genannte Standard-Ein- und -Ausgabeeinheit angesehen. Die Standardeingabeeinheit wird systemtechnisch mit dem Namen INPUT angesprochen, die Standardausgabeeinheit mit dem Namen OUTPUT. Wir könnten einfach formulieren: Die Befehlseingabe wird eigentlich aus einer Datei namens INPUT erwartet. Ist diese Datei physisch nicht vorhanden, so schaltet der FTP-Client einfach auf Bildschirmeingabe um. Das Gleiche gilt für protokollarische Ausgaben. Es wird davon ausgegangen, dass eine Datei mit dem Namen OUTPUT auf dem System ist. Ist dieses nicht der Fall, so wird die Ausgabe auf den Bildschirm gelenkt.

In diesen Aussagen steckt auch schon der Ansatz für die Vorgehensweise beim Batch-gesteuerten Dateitransfer.

In einem ersten Arbeitsschritt erzeuge ich eine Quellendatei mit dem Namen FTPCMD. In diese Datei nehme ich dann eine Teildatei mit dem Namen TRANS01 auf. Diese Teildatei enthält dann die Befehle, die für die Ausführung des Transfers notwendig sind.

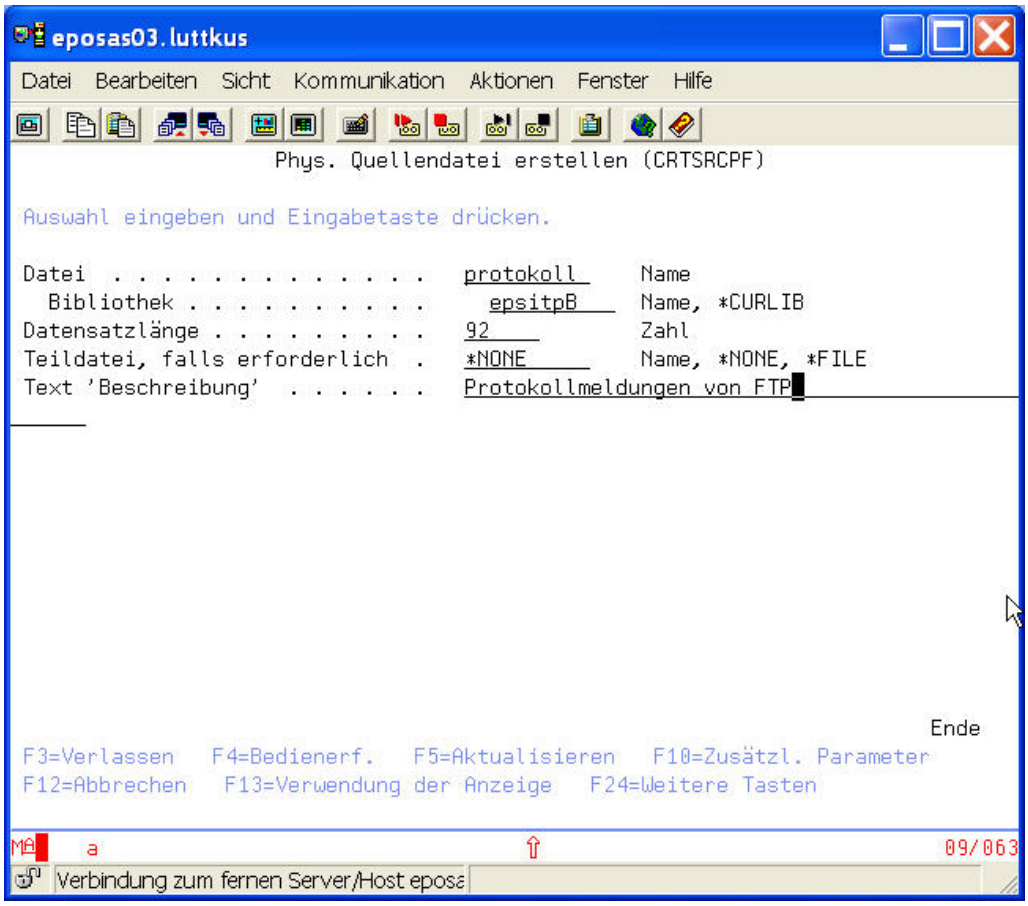


FTP-Befehle in der Datei FTPCMD MBR(TRANS01)

7.6.5.9.1 Seite 2

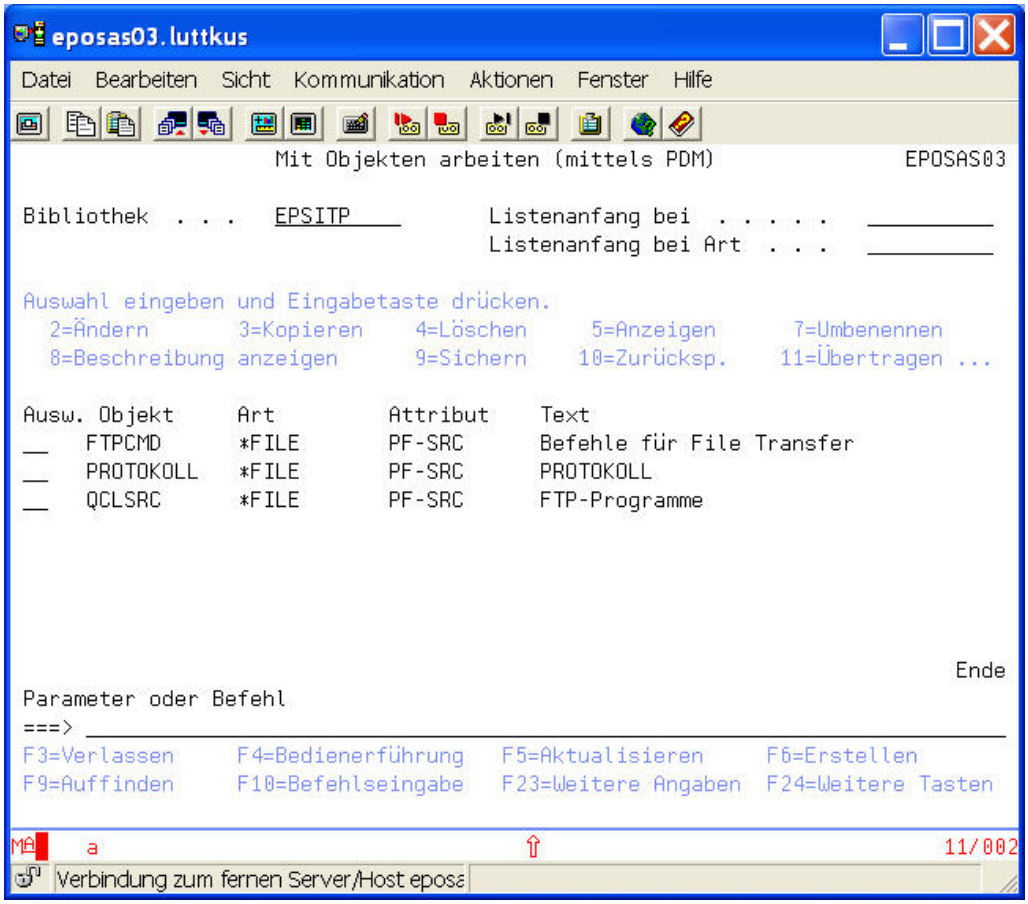
Zeilennummer	Beschreibung
1	Übergabe der Berechtigungsdaten. Achtung: Bei einem Transfer von bzw. zu einem anderen System iSeries müssen Sie hier ein valides Benutzerprofil inklusive Kennwort angeben. Seien Sie sich dieses Sicherheitsaspektes bewusst und schützen Sie diese Datei auf der Objektebene!
2	In die Bibliothek wechseln, in der die zu transferierende Datei steht.
3	Datei transferieren. Hier wird eine Datei von einem entfernten System heruntergeladen.
4	Die FTP-Verbindung beenden.

In einem zweiten Arbeitsschritt wird jetzt die Ausgabedatei PROTOKOLL mit einer Teildatei PROTO01 erstellt.



CRTSRCPF – Create Source Physical File

In einem dritten Schritt erstelle ich auch eine Quellendatei für das CL-Programm, das den Transfer letztendlich ausführen soll.

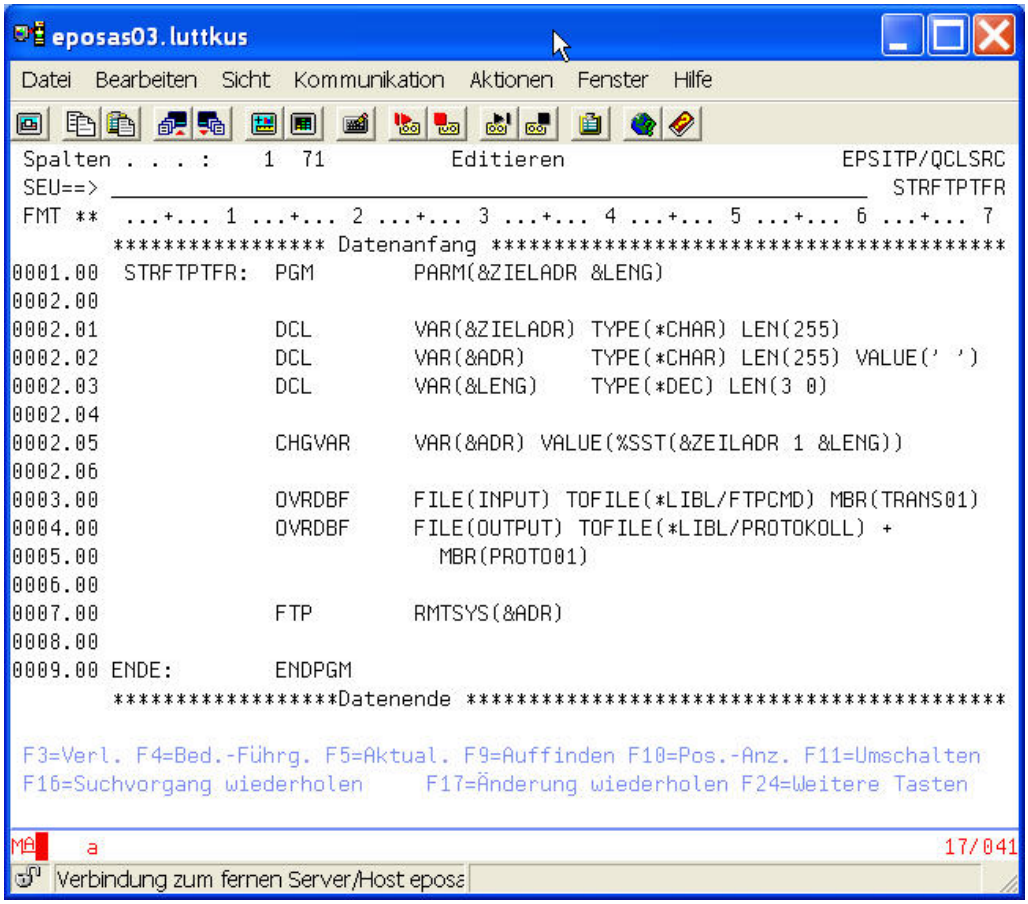


Die Quellendatei QCLSRC ist erstellt

Im vierten Arbeitsschritt wird jetzt das CL-Programm erstellt.

Dieses Programm heißt STRFTPTFR (Start FTP-Transfer). Die Quelle sehen Sie im folgenden Bild.

7.6.5.9.1
Seite 4



Das CL-Programm STRFTPTFR

Zeilennummer	Beschreibung
1	Übergabeparameter: Zieladresse und Länge der übertragenen Zieladresse für den FTP-Befehl
2.01 2.03	Deklaration der Übergabeparamter. Deklaration einer lokalen Variable &ADR. Diese enthält später die Zieladresse für den FTP-Befehl.
2.05	Ermitteln des signifikanten Inhalts der Zieladresse. Übertrag des signifikanten Inhalts an das Feld &ADR.
6	Den Namen der Datei INPUT mit dem Namen der Befehlseingabedatei FTPCMD/TRANS01 überschreiben. Dadurch wird die Befehlseingabedatei als Datei INPUT angesprochen.
7	Den Namen der Datei OUTPUT mit dem Namen der Befehlseingabedatei PROTOKOLL/PROTO01 überschreiben. Dadurch wird die Befehlseingabedatei als Datei OUTPUT angesprochen.
10	Den Transfer durchführen.

Jetzt wird das Programm STRFTPTFR mit dem Befehl CRTCLPGM umgewandelt und kann dann ausgeführt werden.

```
CALL PGM(STRFTPTFR) PARM('172.16.0.1' 10)
```

Als Parameter wird die Adresse des TCP/IP-Host angegeben, von dem die Datei zu transferieren ist und die Länge der übergebenen Adresse.

Folgende Meldungen sind dann der Datei PROTOKOLL zu entnehmen:

Aufnahme der Verbindung mit dem Remote-System.

```
Ausgabe wurde in eine Datei umgeleitet.
Eingabe aus angegebener Überschreibungsdatei wurde gelesen.
Versuch, eine Verbindung zum fernen Host 172.16.0.1 über Port 21
herzustellen.
220-QTCP at EPOSAS02.EPOSGMBH.COM.
220 Connection will close if idle more than 5 minutes.
```

7.6.5.9.1
Seite 6

Anmeldung des Benutzers, der für den Transfer verantwortlich ist.

```
Anmelde-ID (luttkus) eingeben:
331 Enter password.
230 LUTTKUS logged on.
OS/400 is the remote operating system. The TCP/IP version is "V5R1M0".
250 Now using naming format "0".
257 "QGPL" is current library.
Einen FTP-Unterbefehl eingeben.
```

Wechseln in die Bibliothek, aus der die Daten entnommen werden sollen.

```
>> cd epssvfil
250 Current library changed to EPSSVFIL.
Einen FTP-Unterbefehl eingeben.
```

Die Daten herunterladen.

```
>> get svkdst00
250 Now using naming format „0".
257 "EPSSVFIL" is current library.
227 Entering Passive Mode (172,16,0,1,4,1).
150 Retrieving member SVKDST00 in file SVKDST00 in library EPSSVFIL.
250 File transfer completed successfully.
969449 Byte in 31.415 Sekunden übertragen. Übertragungsgeschwindigkeit
30.859 KB/s.
```

Die FTP-Verbindung beenden.

```
Einen FTP-Unterbefehl eingeben.
>> quit
221 QUIT subcommand received.
```

Windows-FTP im Batch

Auch der Windows FTP-Client kennt den Stapel-Modus. Im Prinzip wird hier analog zum iSeries FTP verfahren. Die Eingabe der FTP-Befehle wird aus einer Eingabedatei entnommen, die der Anwender zu erstellen hat.

Um den Vorgang zu verstehen sehen wir uns zuerst die Parameter des Windows-Befehls-FTP an:

FTP [-v] [-d] [-i] [-n] [-g] [-s:Dateiname] [-a] [-w:Puffergröße] [Host]

-v Unterdrückt die Anzeige von Rückmeldungen des Remote-Servers.

-n Unterdrückt die automatische Anmeldung beim Verbindungsaufbau.

-i Schaltet interaktive Eingabeaufforderungen beim Transfer von mehreren Dateien ab.

-d Aktiviert Debug-Modus.

-g Deaktiviert Globbing (siehe Befehl Ftp: glob).

-s: Dateiname
Gibt eine Textdatei mit FTP-Befehlen an; die Befehle werden automatisch ausgeführt, wenn FTP gestartet wird.

-a Verwendet beim Binden von Datenverbindungen eine beliebige lokale Schnittstelle.

-w: Puffergröße ändert die Standardgröße des Übertragungspuffers in 4096.

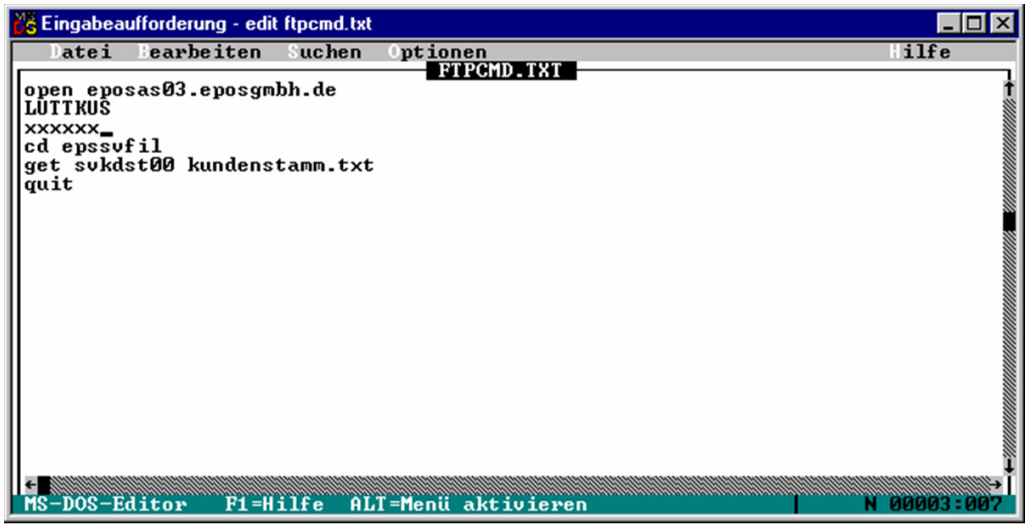
Host Gibt den Host-Namen oder die IP-Adresse der Remote-Hosts an, zu dem eine Verbindung hergestellt werden soll.

Ich habe den Parameter, der für unser Vorhaben maßgeblich ist, grau hinterlegt.

Verwenden wir den Parameter –s, so werden die FTP-Befehle mit Hilfe einer Input-Datei an den FTP-Client übergeben. Daraus folgt folgendes Vorgehen.

1.) Ich erstelle auf meiner Windows NT Workstation eine Textdatei mit dem Namen FTPCMD.TXT. Dazu nutze ich den Befehl EDIT der DOS-Box meiner Workstation.

7.6.5.9.2
Seite 2



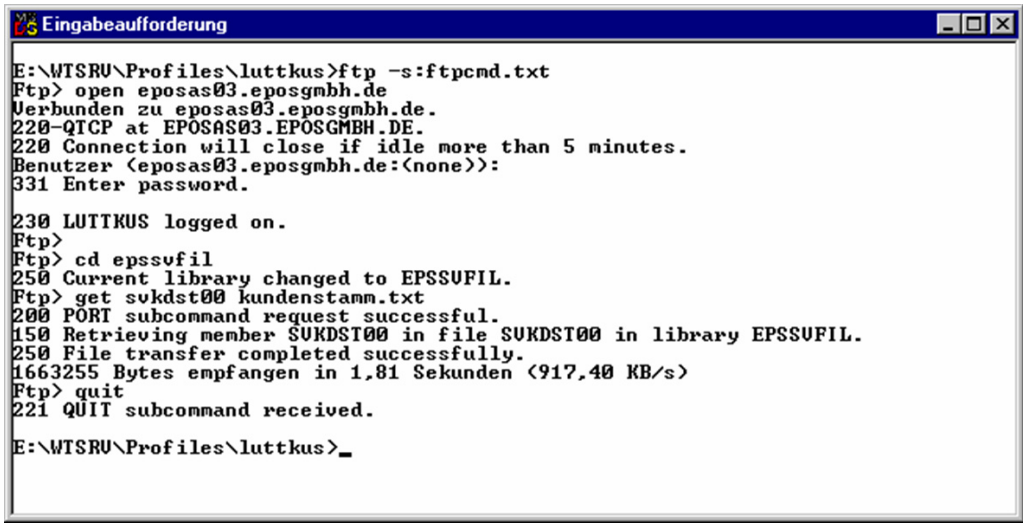
Die Datei FTPCMD.TXT

Achtung: Beachten Sie die Eingabe des Benutzerprofils und des Kennworts. Die hier gezeigte Syntax ist für den iSeries FTP-Server peinlich genau einzuhalten!

2.) Jetzt rufe ich den Windows FTP-Client mit folgender Syntax auf:

```
FTP -S:FTPCMD.TXT
```

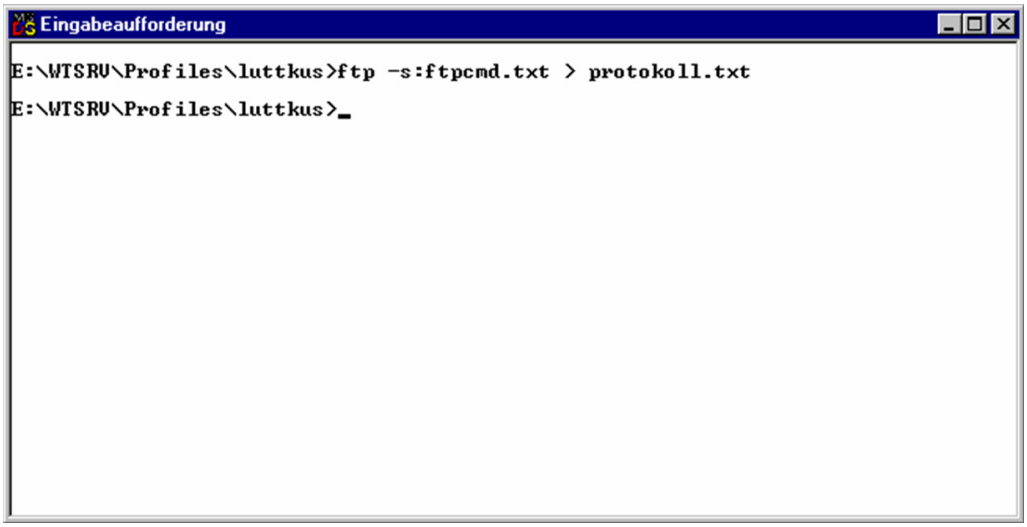
Bei der Befehlsausführung werden alle Ausführungsschritte auf dem Bildschirm dargestellt.



Textausgabe des Batch-gesteuerten Windows-FTPs

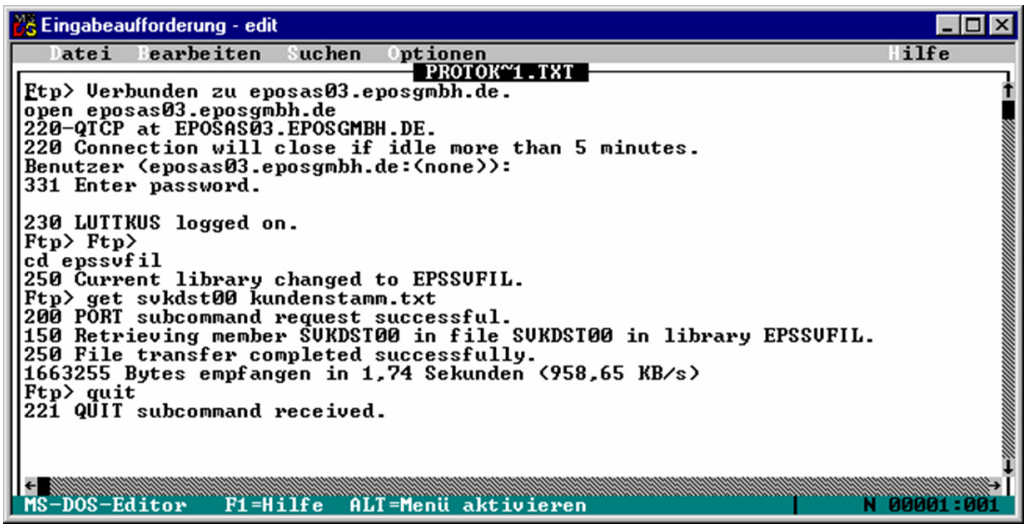
Soll diese Textausgabe unterdrückt werden, so kann die Bildschirmausgabe mit Hilfe einer DOS-Pipe in eine Datei umgelenkt werden. Dazu nutze ich folgende Syntax:

```
FTP -S:FTPCMD.TXT >> PROTOKOLL.TXT
```



Unterdrückte Bildschirmausgabe

Um eine Aussage darüber treffen zu können, wie der Transfer ausgegangen ist, öffne ich jetzt mit Hilfe des Befehls EDIT die Datei PROTOKOLL.TXT.



Inhalt der Datei PROTOKOLL.TXT

7.6.5.9.2

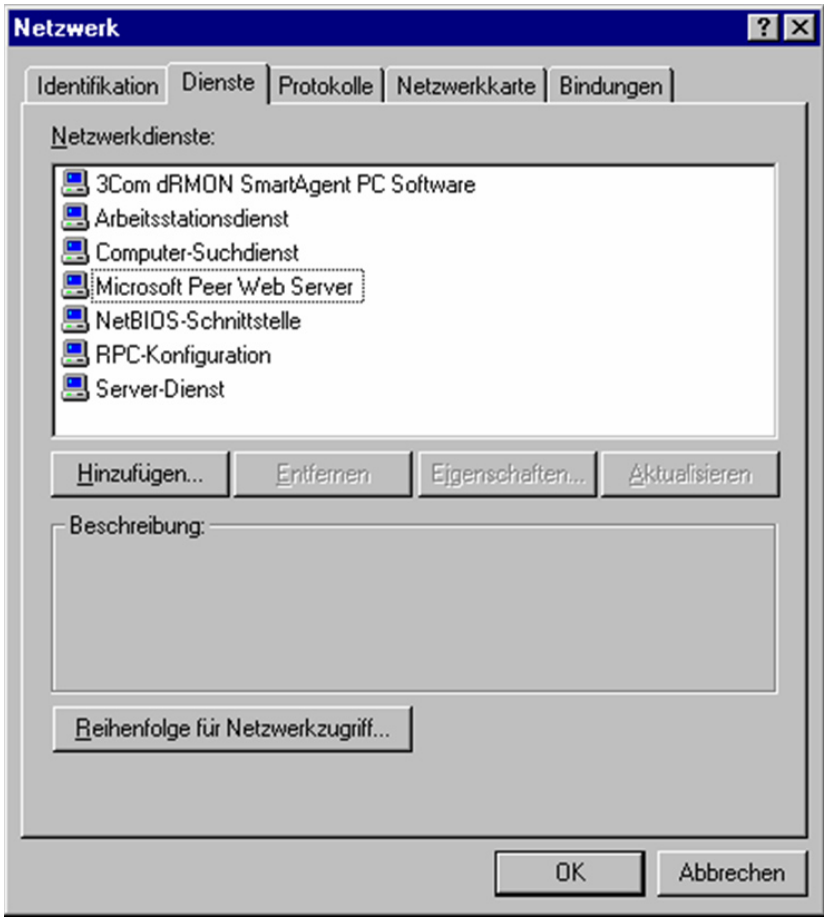
Seite 4

FTP zum Windows PC

Nachdem wir jetzt Daten quasi automatisch von einer iSeries und einem Windows PC herunter- bzw. heraufladen können, sollte es eigentlich möglich sein, diesen Prozess auch von der iSeries aus gesehen gegen einen PC zu richten. Wir wollen erreichen, dass die iSeries einem PC-Client direkt Daten zusenden kann. Also nicht, dass der PC sich Daten von der iSeries abholt, sondern dass die iSeries aktiv Daten an einen PC sendet.

Dazu müssen wir auf einem Windows PC einen FTP-Server installieren. Besitzen Sie eine Windows NT Workstation, so sind Sie bereits im Besitz eines FTP-Servers. Dieser wird zwar nicht automatisch installiert, aber kann leicht nachinstalliert werden. Bestandteil der Windows NT Workstation ist ein so genannter Peer Web-Server und Bestandteil dieses Web-Servers ist wiederum ein FTP-Server.

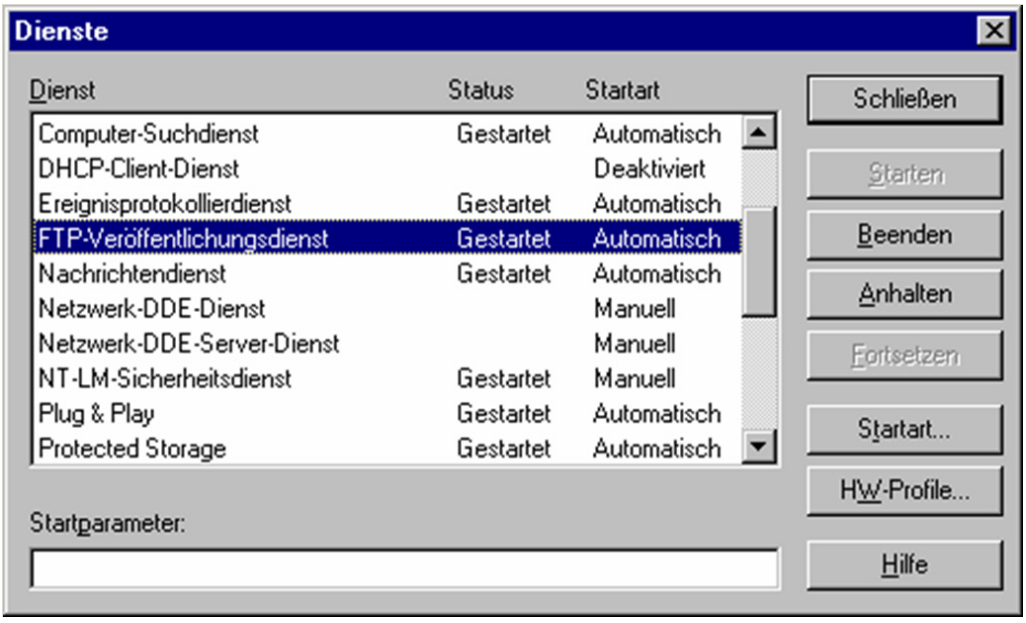
Mittels des Windows NT-Dialogs Netzwerkumgebung-Eigenschaften-Dienste-hinzufügen installieren Sie einfach den Microsoft Peer Web-Server nach.



Der Microsoft Peer Web-Server wurde installiert

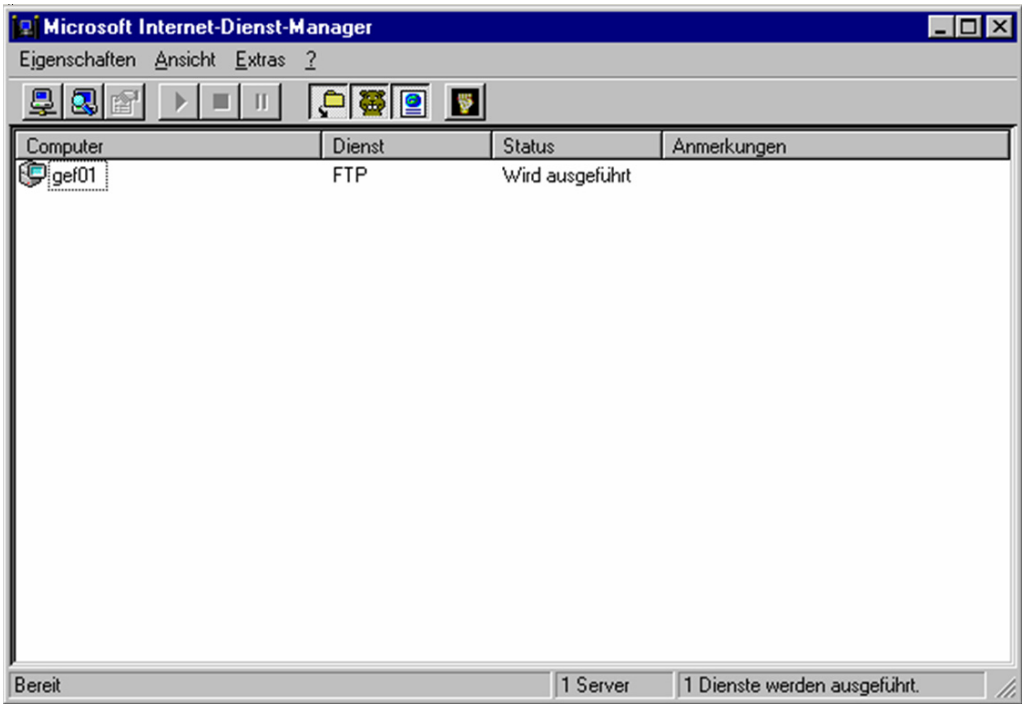
7.6.5.10
Seite 2

Um diesen Server in Betrieb nehmen zu können muss der soeben installierte Dienst manuell gestartet werden. Soll jedoch der FTP-Server ständig in Betrieb sein, so sollten Sie den Start dieses Dienstes von manuell auf automatisch ändern. Dieses erreichen Sie im Dialog Systemsteuerung-Dienste.



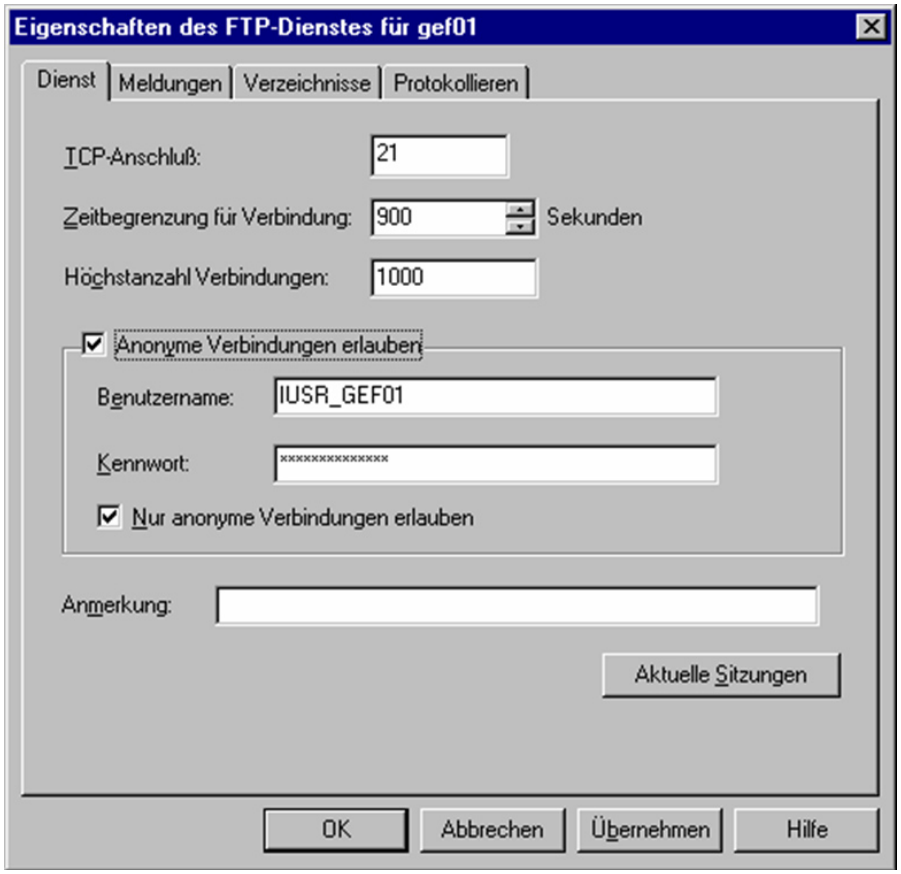
FTP automatisch starten

Bevor Sie jedoch den FTP-Server in Betrieb nehmen können, muss er zuerst einmal konfiguriert werden.



Der Internet Dienst-Manager

Rufen Sie daher aus dem Dialog Start-Programm-Microsoft Peer Web Services-Internet Dienst Manager das Verwaltungsprogramm auf und konfigurieren Sie folgende Merkmale über den Eigenschaften-Dialog des angezeigten FTP-Dienstes:

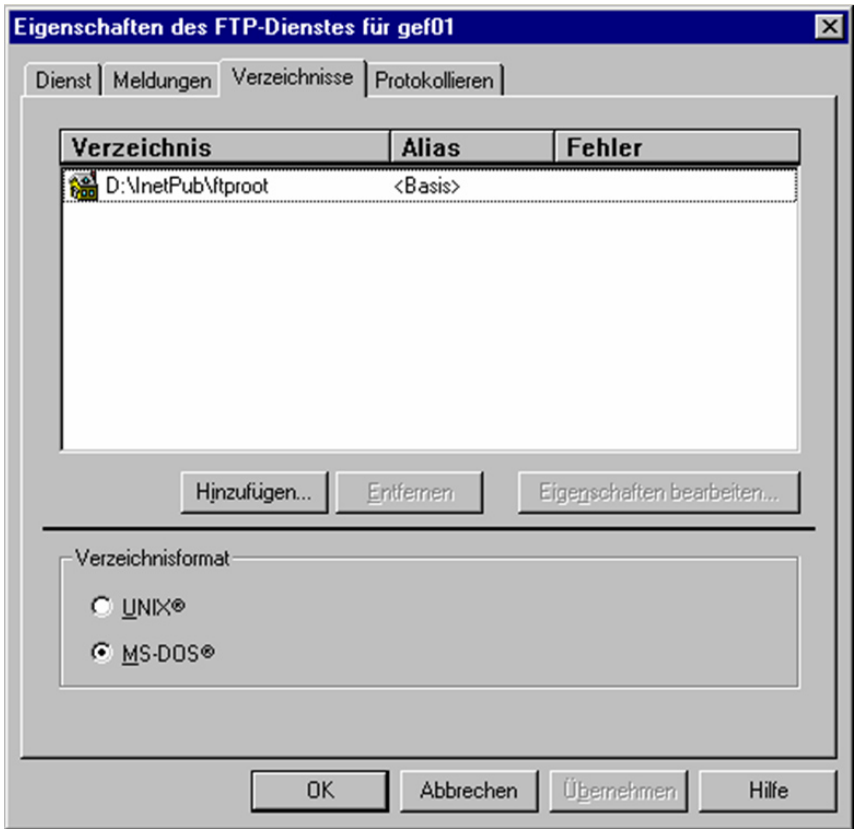


Register Dienst

Register Dienst: Anonyme Verbindungen erlauben oder entziehen.

Gestatten Sie keine anonyme Verbindung, so richtet sich der Anmeldungsprozess der iSeries als FTP-Client gegen einen registrierten lokalen Benutzer der Workstation. Es wird bei der Installation des Peer Web-Servers ein lokales NT-Benutzerkonto erstellt. Der Name wurde vom System generiert. Für diesen Fall habe ich das Kennwort FTP vergeben. Diese Angaben sind wichtig! Der FTP-Client der iSeries wird sich später mit diesen Berechtigungsdaten bei Windows NT FTP berechtigen lassen müssen. Beachten Sie, das Windows NT bei der Schreibweise des Kennworts Groß- und Kleinschreibung unterscheidet!

7.6.5.10
Seite 4



Register Verzeichnisse

Register Verzeichnisse: Richten Sie hier das Verzeichnis ein, zu dem die Daten von der iSeries heruntergeladen werden sollen. Ich habe es hier bei den Default-Einstellungen belassen.

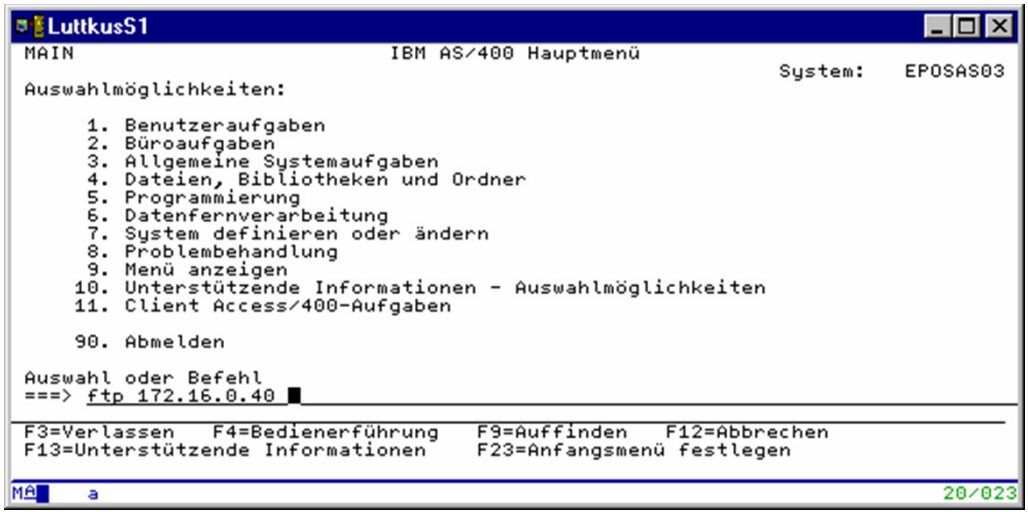
Wichtig:

Ändern Sie das Verzeichnisformat auf MS-DOS. Vergeben Sie sodann mit Hilfe des Eigenschaften-bearbeiten-Dialogs Schreibrechte für das angezeigte Verzeichnis. Die iSeries soll ja Daten auf den PC herunterladen, also benötigen Sie auch die entsprechenden Rechte.

Haben Sie alles erledigt, so sollten Sie kontrollieren, ob der Dienst korrekt arbeitet. Dieses erreichen Sie indem Sie in das Windows NT Ereignisprotokoll gehen. Läuft alles korrekt, so kann das Herunterladen von iSeries-Daten beginnen.

Wir werden zuerst einmal den Vorgang interaktiv testen:

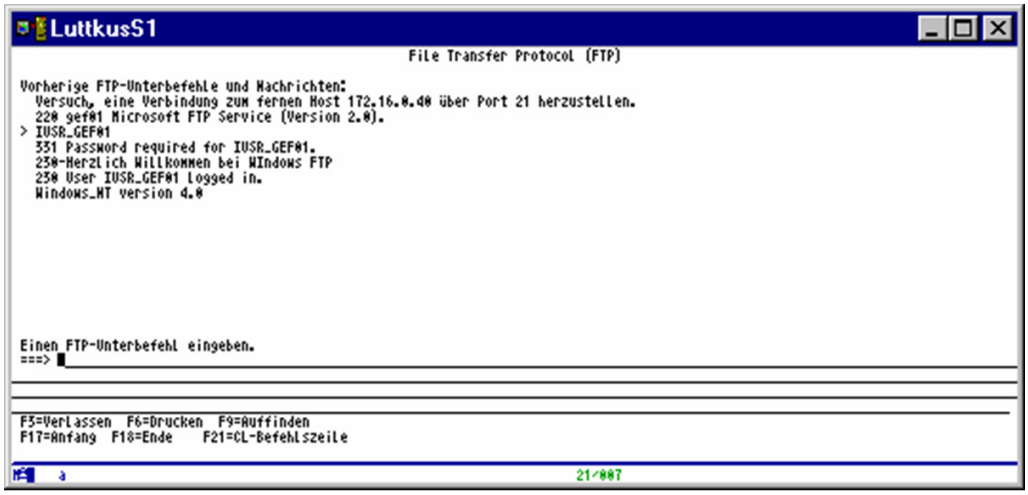
1.) FTP auf der iSeries aufrufen.



FTP aufrufen

Ich übergebe hier als Adresse die IP-Adresse des PCs, auf dem der FTP-Server-Dienst ausgeführt wird.

3.) FTP-Berechtigungsdaten eingeben.

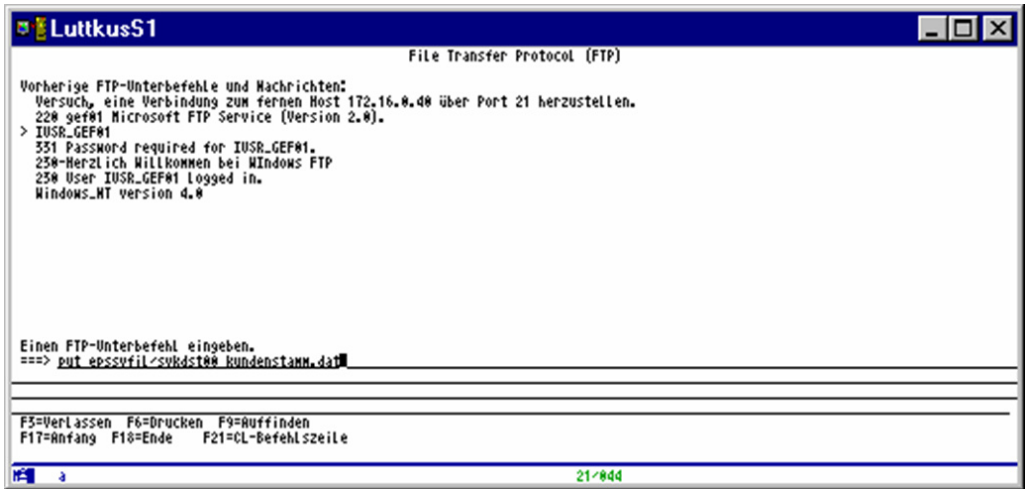


Die Anmeldung

Bei der Anmeldung habe ich das lokale Benutzerkonto IUSF_GEF01, welches bei der Installation des FTP-Servers erzeugt wurde verwendet.

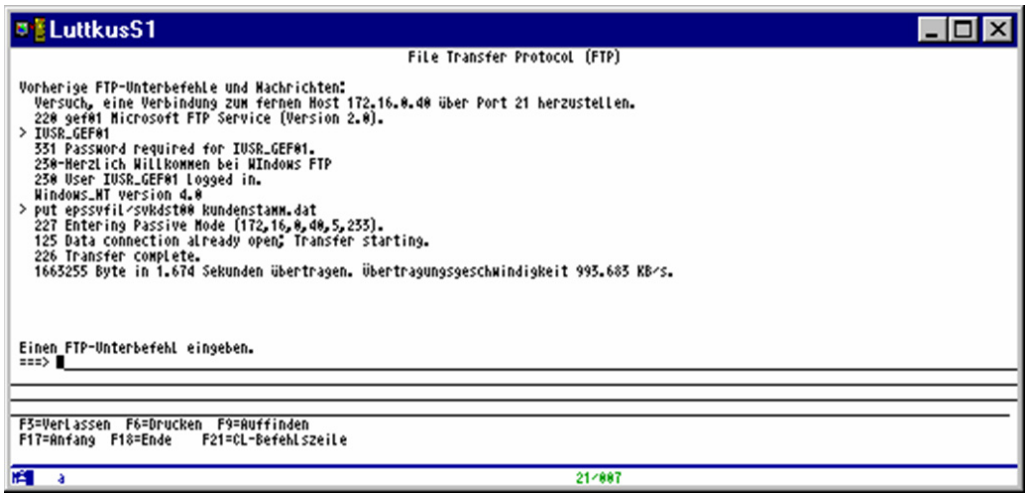
4.) Daten transferieren.

7.6.5.10
Seite 6



Daten senden

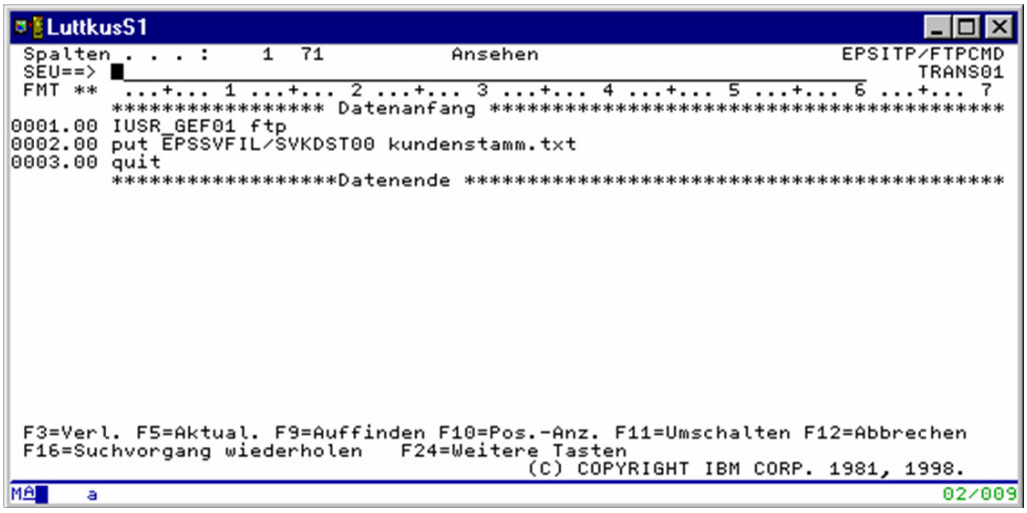
Mit Hilfe des FTP-Befehls PUT werden die Daten auf den PC transferiert.



Die Daten wurden gesendet

Danach wird mit QUIT die FTP-Verbindung beendet. Die Daten stehen zur weiteren Verarbeitung auf der Workstation zur Verfügung.

Soll der Transfer automatisiert werden, so greife ich auf meine vorhin erstellte Datei FTPCMD zurück, die ein wenig verändert werden muss:



Die Datei FTPCMD MBR(TRANS01)

Wichtig ist die Zeile 1! Hier tragen Sie den Namen eines auf der NT Workstation registrierten Benutzers ein. Achtung: Windows NT unterscheidet keine Groß- und Kleinschreibung im Benutzernamen. Die Schreibweise des Kennwortes wird jedoch peinlich genau unterschieden.

Wichtig ist ebenfalls Zeile 2! Der Zugriffspfad zur iSeries-Datei muss in Großschreibung eingegeben werden, sonst wird die Datei nicht gefunden.

Ich rufe also unser Transferprogramm auf und gebe als Zieladresse den Windows NT PC an.

```
CALL PGM(STRFTPTFR) PARM('172.16.0.40')
```

Jetzt wird der Transfer ausgeführt. Um das Ergebnis des Transfers zu überprüfen sehe ich einfach in die Protokoll-Datei.

Die Datei PROTOKOLL MBR(PROTO01)

Die erlösende Meldung steht in Zeile 14: Transfer complete.

7.6.5.10

Seite 8

FTP als Befehlsschnittstelle zur iSeries

Neben seiner Funktion als Objekttransfer-Tool bietet FTP jedoch auch noch eine weitere interessante Möglichkeit. Es können nämlich Befehle an den Server gesendet werden, um bestimmte Aufgaben zu erledigen.

Zwei Arten von Befehle sind möglich:

- CL-Befehle
- FTP-Subcommands

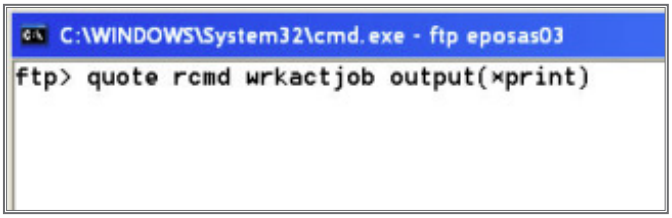
7.6.5.11

Seite 2

CL-Befehle mit FTP ausführen

Um CL-Befehle auszuführen, starte ich den Windows eigenen FTP-Client aus der Microsoft-DOS Eingabeaufforderung heraus.

Ich gebe folgenden Befehl ein:



*ftp>> quote rcmd wrkactjob output(*print)*

Lassen Sie uns einmal diesen Befehl auseinandernehmen.

Quote:

Dieser FTP-Befehl sendet einen sogenannten FTP-Subcommand an den iSeries FTP-Server.

Rcmd

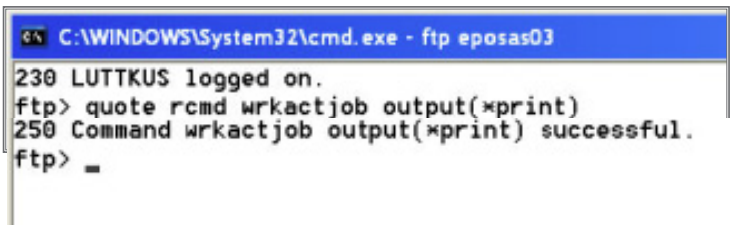
Dieses ist der durch Quote übertragene FTP-Subcommand „Remote Command" - entfernten Befehl ausführen. Dieser Subcommand trägt Sorge dafür, dass ein CL-Befehl an die iSeries übergeben wird.

wrkactjob output(*print)

Der übergebene CL-Befehl.

Sie sehen, dass diese Funktion relativ einfach auszuführen ist. Der schwierige Teil der Angelegenheit besteht darin, den CL-Befehl syntaktisch korrekt einzugeben. Eine F4-Eingabehilfe-Funktion existiert hier natürlich nicht.

So sieht es aus, sollte der Befehl richtig eingegeben worden sein:



Erfolgreiche Ausführung

Und so sieht es aus, wenn Sie einen Fehler gemacht haben:

7.6.5.11.1

Seite 2

Fehler!

FTP-Subcommands ausführen

Neben der Funktion mittels RCMD CL-Befehle auszuführen, gibt noch eine Reihe von Subcommands, die direkt auf der iSeries ausgeführt werden.

Ich zeige Ihnen im Folgenden eine Liste aller FTP-Subcommands für OS/400 V5R1.

Subcommand	What It Does
ABOR	Vorherigen Subcommand abbrechen.
ADDM	Einer Datei eine Teildatei hinzufügen – entspricht dem CL-Befehl ADDPFM.
ADDV	Einer physischen Datei eine Teildatei variabler Satzlänge hinzufügen – entspricht dem CL-Befehl ADDPFVLM.
APPE	Daten an eine Datei hängen.
AUTH	Authentifizierung für aktuelle Sitzung einstellen.
CDUP	In ein übergeordnetes Verzeichnis wechseln.
CRTL	Bibliothek erstellen – CRTLIB.
CRTP	Physische Datei erstellen – CRTPF.
CRTS	Quellendatei erstellen – CRTSRCPF.
CWD	Aktuelles Verzeichnis setzen – CD bzw. CHGCURLIB.
DBUG	Startet / beendet einen Server Trace.
DELE	Löschen einer Datei, Teildatei oder eines Dokuments.
DLTF	Datei Löschen – DLTF.
DLTL	Bibliothek löschen – DLTLIB.
HELP	Hilfe für Subcommands anfordern.
LIST	Inhalt eines Verzeichnisses anlisten.
MKD	Verzeichnis erstellen – MD/CRTDIR.
MODE	Datenformat für Transfer einstellen.
NLST	Liste von Dateien oder Verzeichnissen erstellen.
NOOP	Prüfen ob Server antwortet.
PASS	Passwort an Server senden.
PASV	Nächste Verbindung passiv öffnen.
PBSZ	Größe von Datenpuffern einstellen.
PORT	Port abfragen.
PROT	Sicherheitsprotokoll einstellen.
PWD	Aktuelle Bibliothek/Verzeichnis anzeigen.
QUIT	Abmelden.
RCMD	CL-Befehl an Server senden.
REIN	Sitzung wieder aufnehmen.
RETR	Daten vom Server holen.

7.6.5.11.2
Seite 2

Subcommand	What It Does
RMD	Verzeichnis löschen.
RNFR	Datei umbenennen.
RNTO	Datei umbenennen.
SITE	Informationen an Server senden.
STAT	Statusinformationen vom Server abrufen.
STOR	Daten auf Server in Savefile sichern.
STOU	Daten auf Server in Savefile sichern.
STRU	Specifies the Structure of a File.
SYST	Betriebssystemversion anzeigen.
TIME	Timeout für Server setzen.
TYPE	Transfermodus setzen.
USER	Benutzeranmeldung anfordern.
XCUP	Ins höhere Verzeichnis wechseln.
XCWD	Aktuelles Verzeichnis setzen.
XMKD	Verzeichnis erstellen.
XPWD	Aktuelle Bibliothek bzw. Verzeichnis anzeigen.
XRMD	Verzeichnis entfernen.

Diese Liste finden Sie auch im iSeries Information Center unter folgendem Link:

http://publib.boulder.ibm.com/pubs/html/as400/v5r1/ic2929/index.htm

Mit Hilfe der Suchfunktion und dem eingegebenen Suchbegriff „FTP Subcommands“ finden Sie dann diese Liste.

Sie sind sicherlich bezüglich dieser Möglichkeiten beeindruckt, jedoch drängt sich uns sicherlich ein Gefühl der Unsicherheit auf: Was passiert eigentlich, wenn ein Anwender diese Befehlsübergabe-Möglichkeiten in Erfahrung bringt? Kann derjenige kurzerhand Dateien löschen?

Im Prinzip ja, oder wenn wir den Operations-Navigator nutzen, können wir diese Schnittstelle absichern. Wie das geht, erkläre ich Ihnen im nächsten Abschnitt.

FTP-Server mit Operations-Navigator überwachen

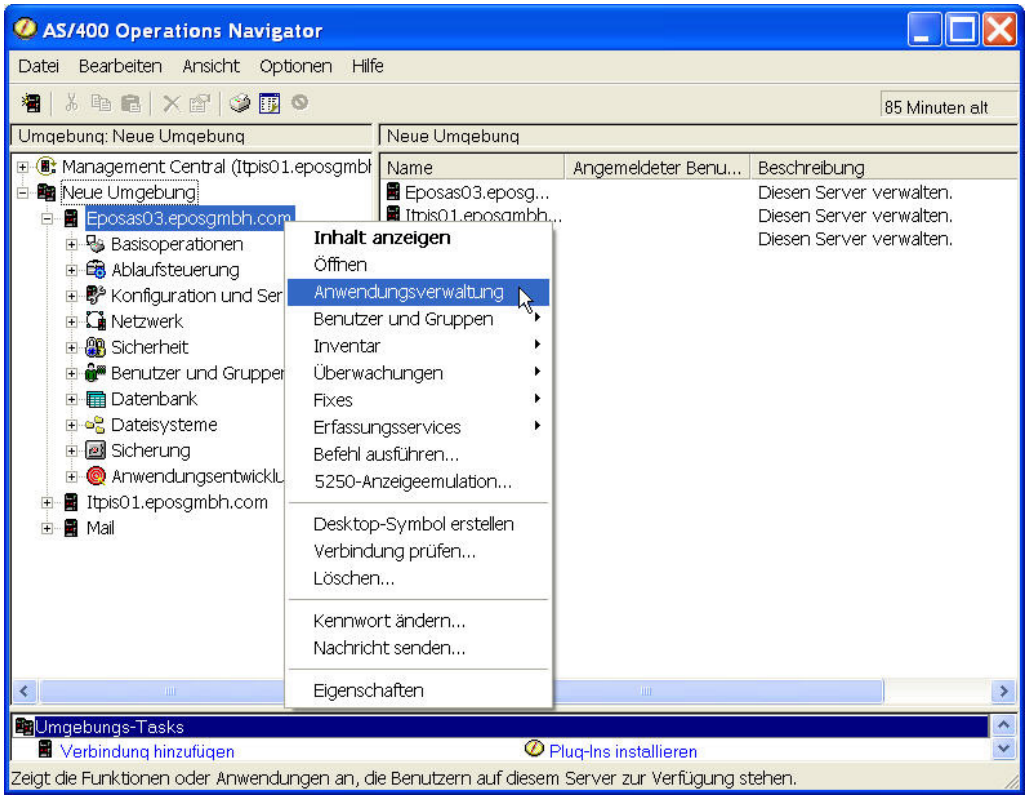
Der Operations-Navigator stellt uns eine Sicherheitsfunktion zur Verfügung, die mittels 5250-Dialogen nicht erreichbar ist.

Diese Funktion nennt sich – wenig dramatisch – die Anwendungsverwaltung. Die Anwendungsverwaltung ermöglicht Ihnen die restriktive Verwaltung von Netzwerkzugriffen auf iSeries Ressourcen und Funktionen.

Was heißt das? Sollte ein Benutzer bzw. eine Benutzergruppe nur eingeschränkt Operations-Navigator-Funktionen ausführen können, soll ein Anwender bzw. eine Anwendergruppe eine Client Access Funktion wie ODBC nicht ausführen dürfen, soll ein Benutzer bzw. eine Gruppe keine CL-Befehle an den FTP-Server der iSeries übermitteln dürfen, dann können Sie mit Hilfe der Operations-Navigator-Anwendungsverwaltung Richtlinien erzeugen, die das Ausführen der oben genannten Funktionen verhindern.

Die oben beschriebenen Fälle stellen nur eine kleine Auswahl der bestehenden Möglichkeiten dar. In diesem Kapitel beschränke ich mich auf die Erklärung für die Einschränkung der Nutzung des iSeries FTP-Servers.

Um die Anwendungsverwaltung zu starten, rufe ich den Operations-Navigator auf und markiere die Verbindung zu System EPOSAS03.EPOSGMBH.COM mit der rechten Maustaste.

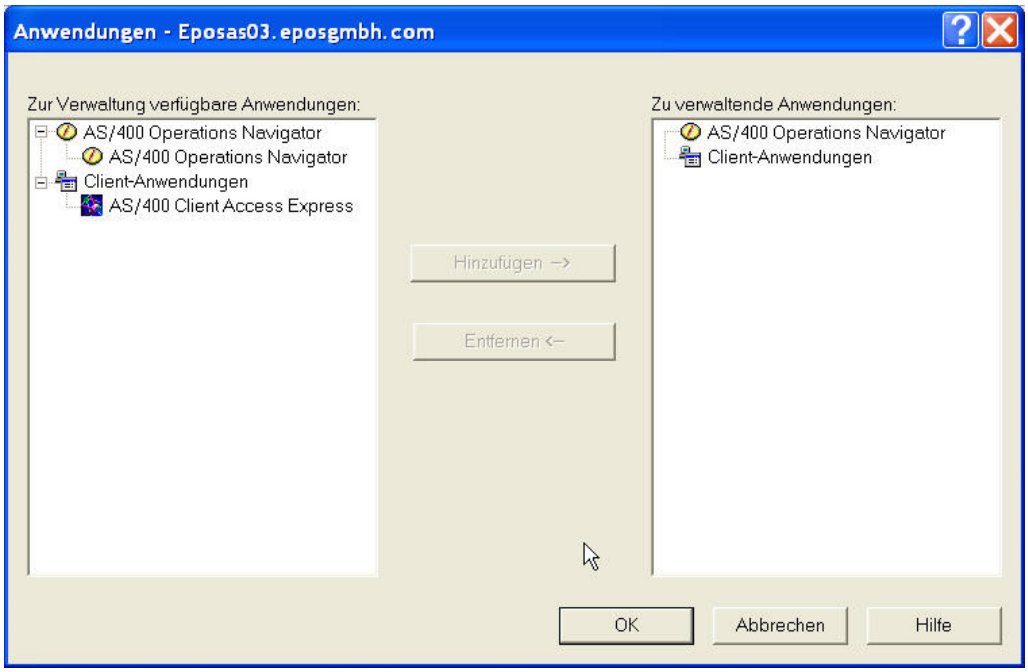


Anwendungsverwaltung aufrufen

7.6.5.12
Seite 2

Sie sehen, dass sich ein Kontextmenü geöffnet hat. In diesem wähle ich die Option Anwendungsverwaltung.

Sollten Sie die Anwendungsverwaltung bisher nicht ausgeführt haben, erscheint folgender Dialog, in dem Sie angeben, welche Client/Server-Anwendungen netzwerktechnisch überwacht werden sollen.



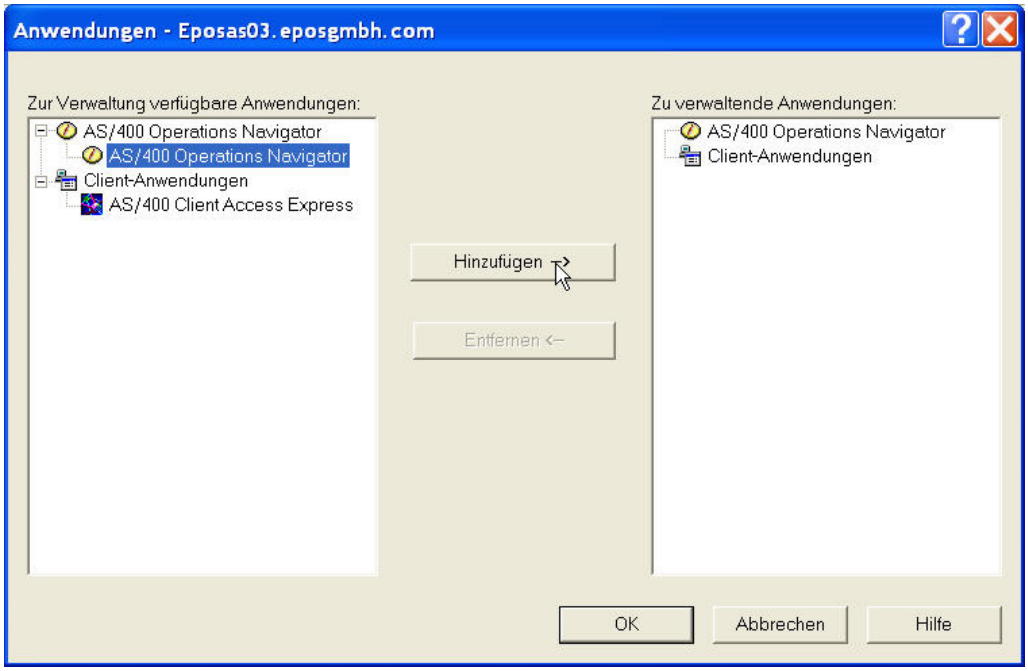
Anwendungen auswählen

Sie sehen, dass auf meinem System zwei Anwendungen (links) zur Überwachung vorgeschlagen werden:

- Operations-Navigator
- iSeries Access Express

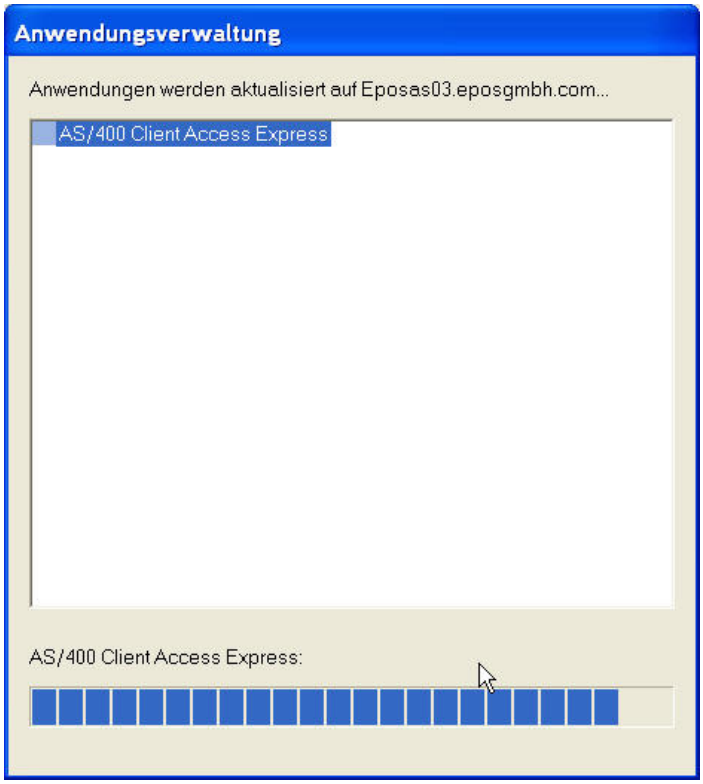
Implizit gibt es noch eine dritte Anwendung, die überwacht werden kann und das ist das Betriebssysem Ihrer iSeries. Es besteht die Möglichkeit, dass Sie in diesem Dialog noch weitere Anwendungen finden, je nachdem welche Lizenzprogramme Sie von IBM erworben haben.

Ich wähle jetzt iSeries Access aus.



iSeries Access für Überwachung aktivieren

Ich klicke auf „Hinzufügen", woraufhin folgender Dialog erscheint.



Client Access Express wird aktiviert

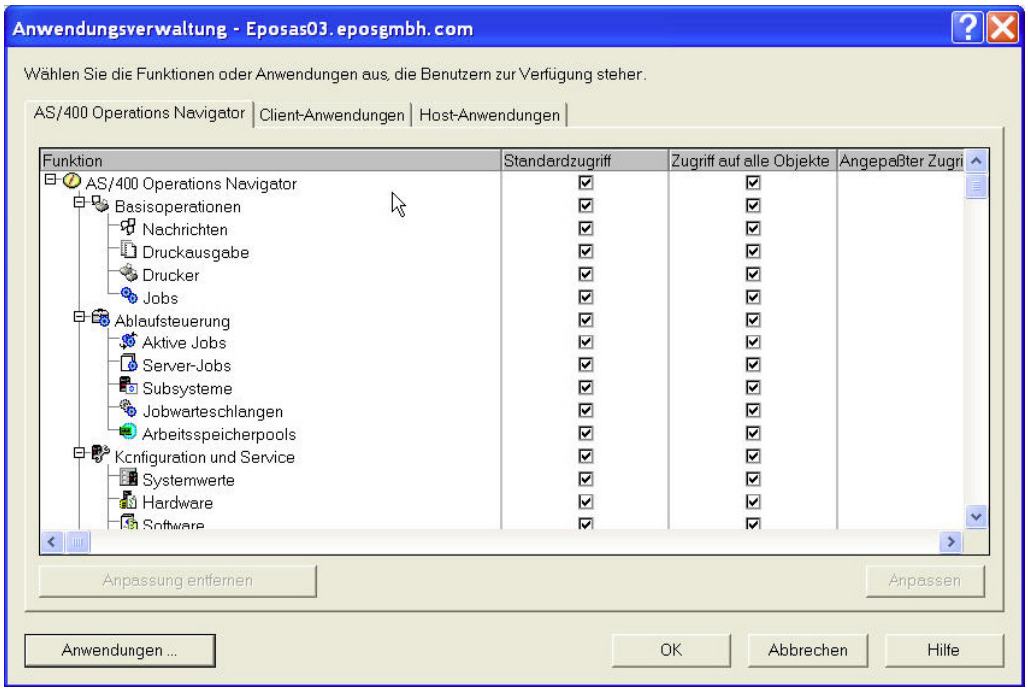
Gleichermaßen verfahre ich mit dem Operations-Navigator.

7.6.5.12
Seite 4

Diese Arbeitsschritte sind nicht notwendig, wenn Sie nur das Betriebssystem überwachen wollen. Ich möchte Ihnen nur empfehlen, mindestens diese beiden Anwendungen zu aktivieren, denn von ihnen gehen mit Sicherheit die größten Gefahren aus.

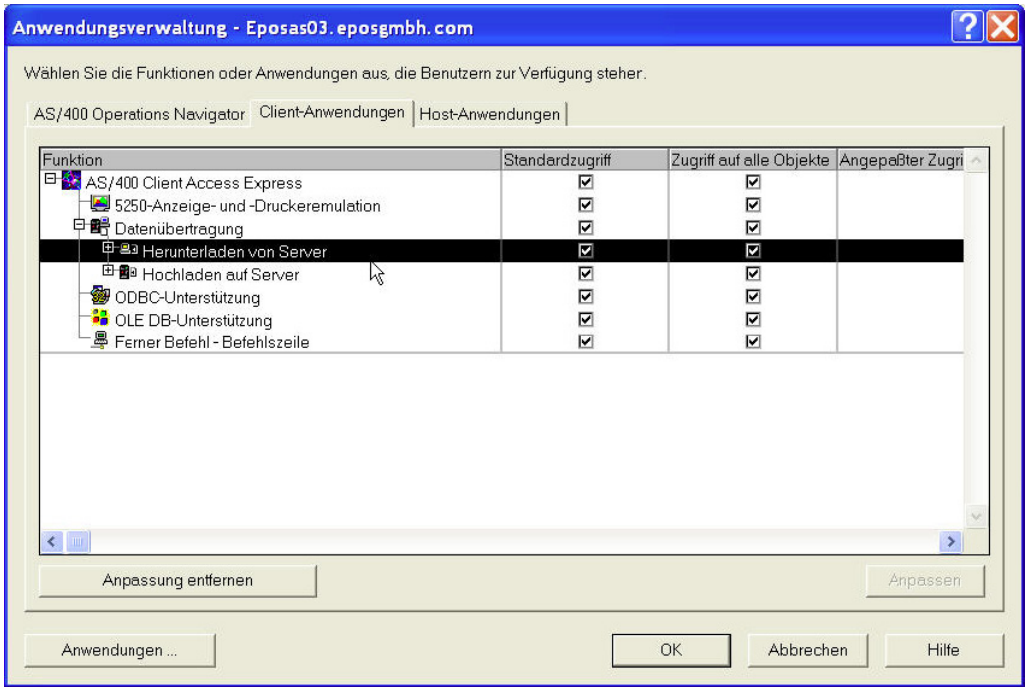
Danach öffnet sich die eigentliche Anwendungsverwaltung.

Sie erkennen, dass es drei Register in diesem Dialog gibt, die die für die Überwachung aktivierten Anwendungen darstellen.



Register Operations-Navigator

In Register Operations-Navigator sichern Sie Funktionsbereiche dieses Programmes gegen unberechtigte Nutzung ab.



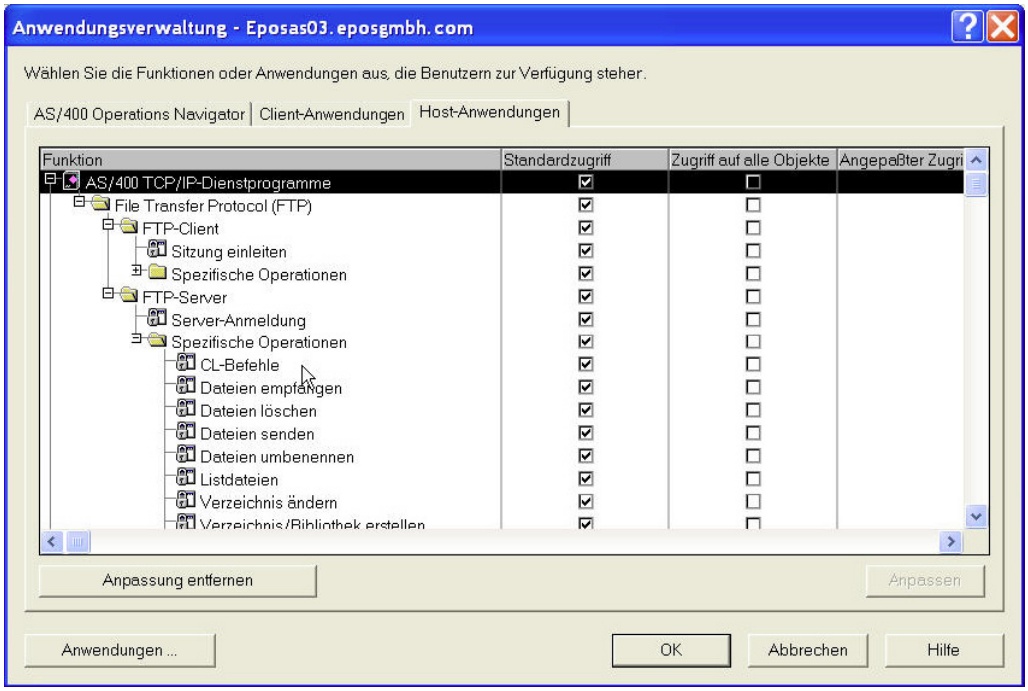
Register Client-Anwendungen

Durch die Aktivierung von iSeries Access Express in der Anwendungsverwaltung können Sie auch die Funktionen dieses Programmes einschränken.

Achtung: Denken Sie daran, dass es sich hier nicht um eine lokale Hinterlegung von Sicherheitsrichtlinien auf Ihrem PC handelt, sondern dass wir mit der Anwendungsverwaltung zentral im OS/400 Sicherheitsrichtlinien aufbauen, die für das gesamte System gelten.

Im letzen Register Host-Anwendungen können dann TCP/IP-Funktionen berechtigungstechnisch gesteuert werden.

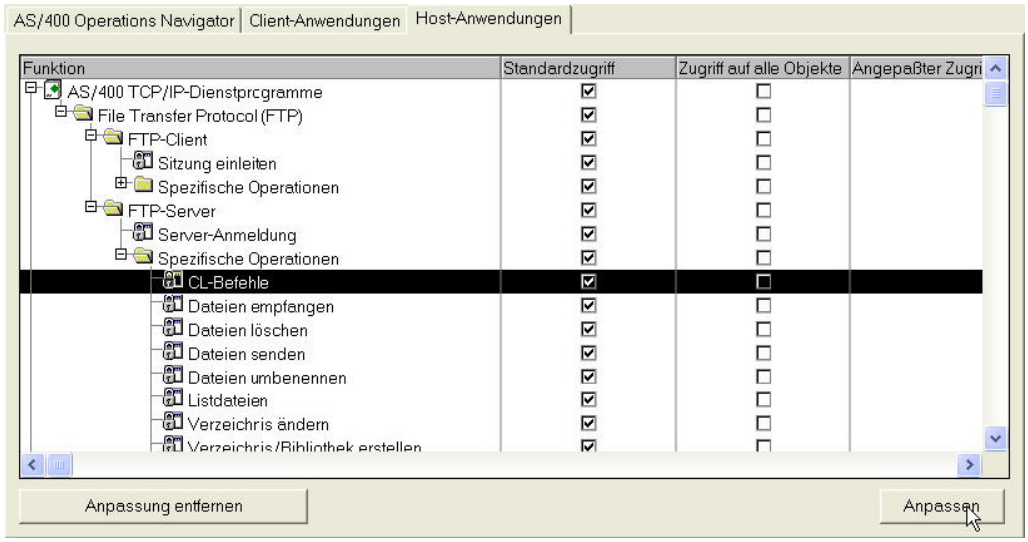
7.6.5.12
Seite 6



Register Host-Anwendungen

Ich expandiere folgenden Pfad:

TCP/IP Dienstprogramme ➜ FTP-Server ➜ Spezifische Operationen ➜ CL-Befehle



Richtlinie für CL-Befehlsausführung erstellen

Sie sehen, dass für CL-Befehle der Standardzugriff aktiviert ist, der Zugriff auf alle Objekte nicht. Diese Einstellung bedeutet, dass ein Anwender im Rahmen seiner Benutzerprofilrechte CL-Befehle über den FTP-Dienst ausführen kann.

Hat ein Anwender z.B. aufgrund seines Benutzerprofils nicht das Recht den Befehl DLTLIB – Bibliothek löschen auszuführen, so kann er es auch nicht über die FTP-Schnittstelle.

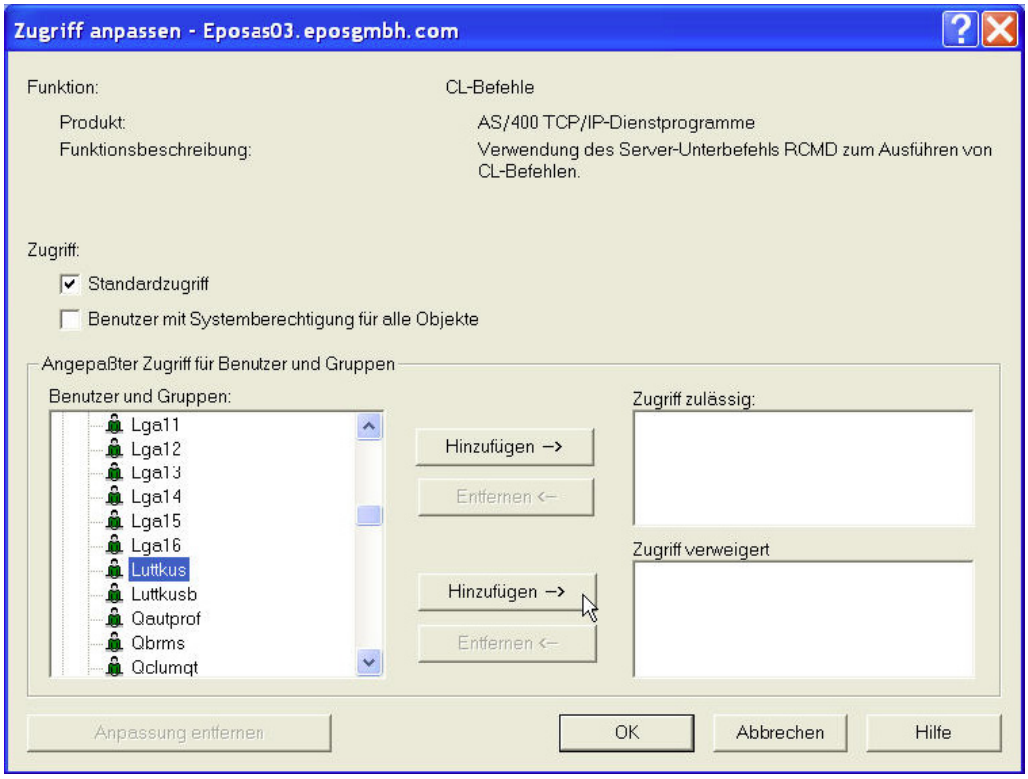
Wäre in der Spalte Zugriff auf alle Objekte das Häckchen ebenfalls gesetzt, so hätte er unabhängig von seinem Benutzerprofil das Recht auf alle Objekte des Systems zuzugreifen.

Zum Glück ist zumindest hier diese Grundeinstellung nicht gewählt worden.

Hier ein kleines Fallbeispiel: Der Benutzer LUTTKUS besitzt auf dem Host EPOSAS03.EPOSGMBH.COM die *SECOFR Rechte. Trotzdem soll er via FTP keine CL-Befehle an die iSeries übermitteln können.

Um dieses Problem zu lösen, markiere ich den Eintrag CL-Befehle und klicke auf „Anpassen".

Jetzt öffnet sich folgender Dialog.



Zugriffe anpassen

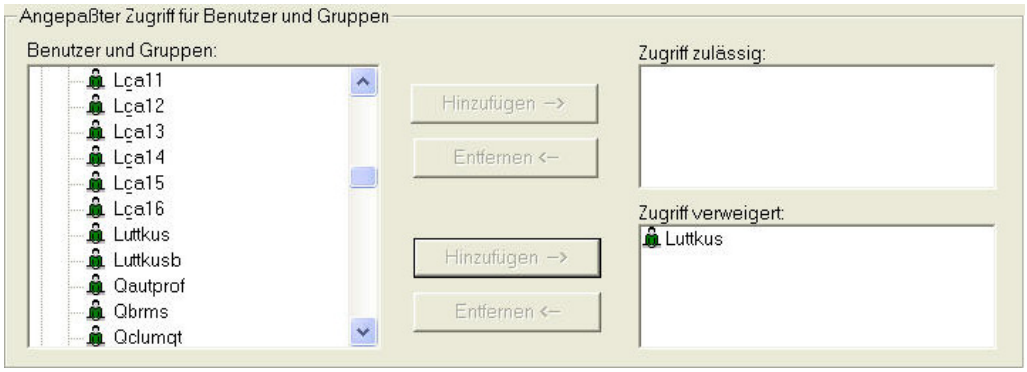
Parameter-Zugriff:

Hier sehen Sie den derzeitigen Stand der Dinge. Benutzer dürfen anhand des Standardzugriffs, der ihnen anhand des eigenen Benutzerpofils gewährt wird, auf das Systemobjekt zugreifen.

7.6.5.12
Seite 8

Ihnen ist kein Zugriff auf alle Objekte des Systems gewährt.

Im Dialog „Benutzer" und „Gruppen" suche ich nach dem Benutzer LUTTKUS, markiere ihn und drücke dann auf „Hinzufügen".



Zugriffe verweigern

Sie sehen, dass der Benutzer LUTTKUS in die Liste „Zugriff verweigert" aufgenommen wurde.

Benutzer, für die der Zugriff explizit erlaubt werden soll, müssen in die Liste „Zugriff zulässig" aufgenommen werden.

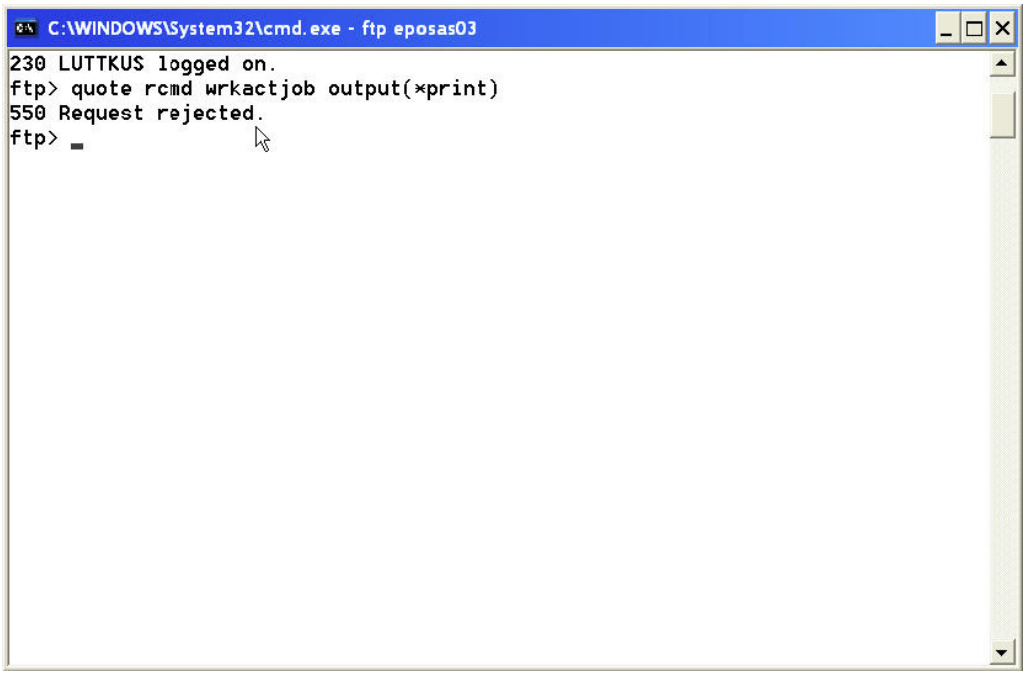
Es ist nur logisch, dass ein Benutzer nicht in „Zugriff zulässig" und gleichzeitig in „Zugriff verweigert" aufgenommen werden kann. Eine entsprechende Fehlermeldung würde erscheinen.

Nachdem wir alle Dialoge mit „OK" beendet haben, müssen wir noch die getätigten Einstellungen testen. Zu diesem Zweck führe ich wieder den Windows FTP-Client aus und melde mich als Benutzer LUTTKUS an.

Ich führe den FTP-Befehl

```
QUOTE RCMD WRKACTJOB OUTPUT(*PRINT)
```

aus.



Befehlsausführung verweigert!

Das Ergebnis spricht für sich. Obwohl der Benutzer LUTTKUS die *SECOFR-Rechte im OS/400 besitzt, kann diese Funktion jetzt nicht mehr von ihm ausgeführt werden.

Achtung: Sollten Sie einem Benutzer entzogene Rechte wieder zubilligen, so muss eine bestehende FTP-Sitzung des Benutzers zuerst beendet werden, bevor die neuen Richtlinien wirksam werden.

7.6.5.12

Seite 10

Der Domain Name Service (DNS)

Ein Merkmal des TCP/IP-Netzwerks besteht darin, dass sämtliche für den Transport von Datenpaketen erforderlichen Adressen numerisch gestaltet sind. Aus Gründen, die auf der Hand liegen hat man sich schon frühzeitig darum bemüht, sozusagen als Stellvertreter für numerische Adressen namensbezogene Adressen zu vergeben. In den frühen TCP/IP-Implementationen war für die Umsetzung von numerischen Adressen in Namen die Datei „HOSTS“ zuständig.

Die Verwendung der Datei HOSTS bringt sowohl Vorteile als auch Nachteile mit sich. Ein Vorteil besteht sicherlich in der einfachen Handhabung dieser Datei. Lediglich ein Texteditor ist notwendig, um die Datei zu bearbeiten. Die Einträge, die hinterlegt werden müssen haben einen einfachen und logischen Aufbau. Als gravierender Nachteil kann angesehen werden, dass die Datei HOSTS ein lokales Objekt jedes TCP/IP-Hosts ist. Jeder Host, der eine Adressumsetzung für sich durchführen soll, muss eine eigene Datei HOSTS besitzen, die darüber hinaus auch noch einen bestimmten Standort in der Verzeichnisstruktur eines jeden Hosts besitzen muss. Ein Netzwerkverzeichnis verbietet sich von vornherein für die Datei HOSTS.

Stellen Sie sich einmal vor, dass Sie 300 TCP/IP-Hosts zu administrieren haben, die jeder eine eigene Datei HOSTS besitzen. Wenn jetzt die Forderung besteht, dass alle diese Dateien den gleichen Inhalt haben, so ist die Vorstellung nicht abwegig, dass es irgendwann Asynchronitäten in diesem Konstrukt geben wird. Denken Sie an Murphy: was geschehen kann, wird auch irgendwann geschehen.

Aus diesem Mangel in der Verwendung der Datei HOSTS hat sich die Entwicklung des Domain Name Service (DNS) ergeben. Dieser Dienst ermöglicht Anwendungen, die miteinander über die Socket-Schnittstelle des TCP/IPs kommunizieren und Host-Namen einfach auswerten, indem sie eine Adressabfrage an einen zentralen Server senden.

3. Ergänzung 06/2005

7.6.6 Seite 2

Überblick über den DNS

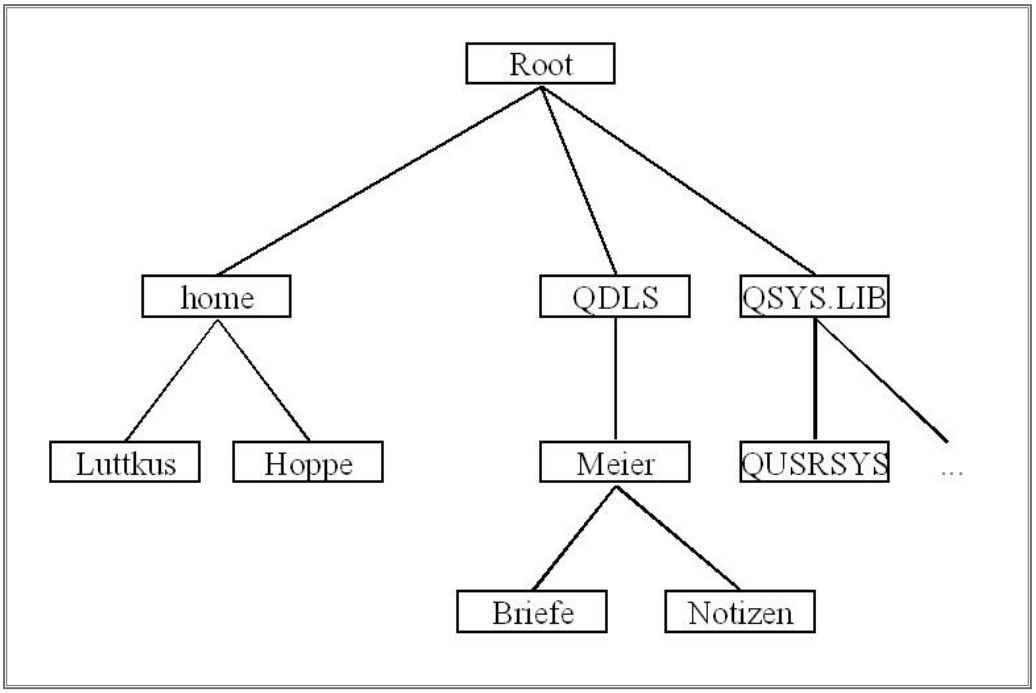
Der Domain Name Service, kurz DNS genannt, wurde bereits 1983 als Standard eingeführt und steht als Dienst des OS/400 seit Version 4 Release 2 den früheren iSeries- und heutigen iSeries-Anwendern zur Verfügung. Im Gegensatz zur flachen Struktur der Datei HOSTS, werden die Host-Namen vom Domain Name Service in einer hierarchischen Struktur verwaltet, die durchaus im Netzwerk verteilt verwaltet werden kann. Bevor ich jedoch in die Konfiguration des Domain Name Service der iSeries einsteige, möchte ich mit Ihnen noch einige Grundlagen des Domain Name Service betrachten. Diese Grundkenntnisse ermöglichen Ihnen dann, die Konfigurationsgänge besser zu verstehen.

7.6.6.1

Seite 2

Hierarchien

Formen hierarchischer Datenverwaltungen sind Ihnen allgemein sicherlich bekannt. Hierarchische Verzeichnisstrukturen werden von fast allen derzeit geläufigen Betriebssystemen verwendet – DOS, Windows, Netware, UNIX, OS/400 (seit V3R1). Diese, die Datenhaltung beschreibenden Strukturbäume, stehen immer auf dem Kopf und werden deshalb auch „inverse Bäume" (inverted trees) genannt.



Eine einfache Hierarchie

In einem inversen Baum repräsentieren die Schnitt- und Endpunkte (Kästen) der Struktur die Daten. Diese Punkte werden Knoten oder auch nodes genannt. Mit Hilfe eines voll qualifizierten Namens kann jeder Knoten im Baum eindeutig angesprochen werden. Der voll qualifizierte Name wird auch als Pfad (path) bezeichnet. Möchte ich mit Daten im Verzeichnis „Notizen" arbeiten, so spreche ich sie unter dem eindeutigen Pfad \QDLS\MEIER\NOTIZEN an.

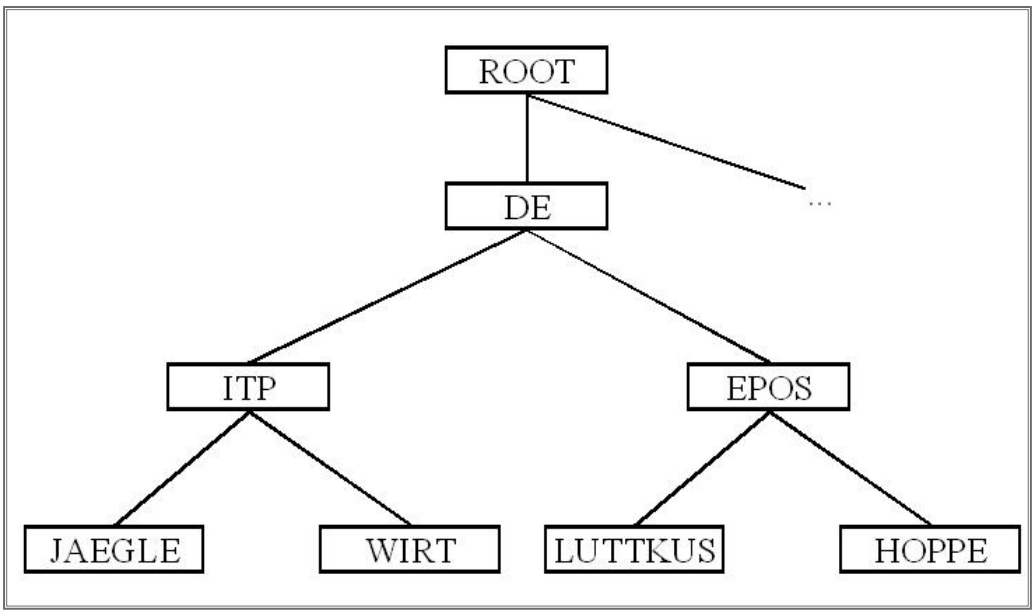
Eine wichtige Hierarchieregel sollten Sie nie außer Acht lassen! Knoten gleicher Hierarchieebene dürfen niemals einen identischen Namen haben. Es darf nicht unterhalb des Knotens MEIER einen zweiten Knoten mit dem Namen MEIER geben.

7.6.6.1.1

Seite 2

Der Domain Name Space

Die hierarchische Datenbank des Domain Name Service wird als „Domain Name Space“ bezeichnet. Jeder, der im Domain Name Space verzeichneten Hosts hat einen eindeutigen voll qualifizierten Namen.



Ein DNS-Baum

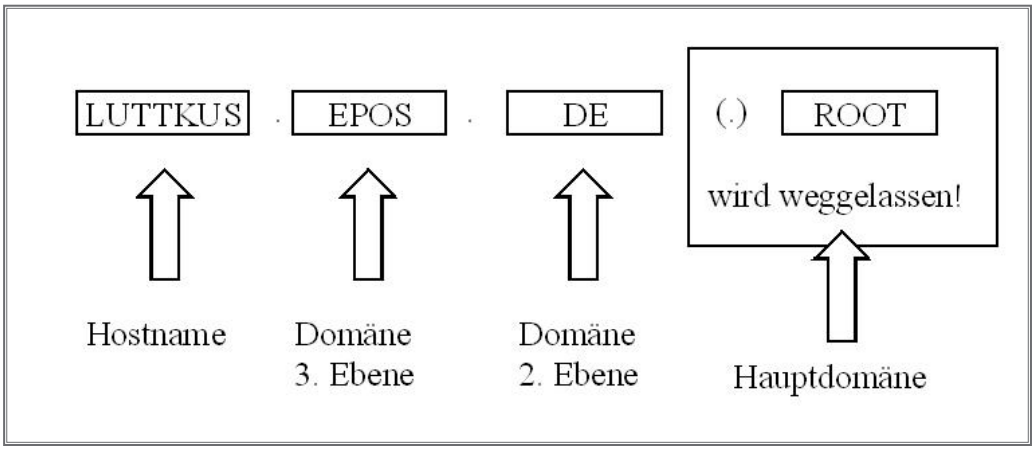
Das Root-Verzeichnis einer Domäne wird allgemein als Hauptdomäne bezeichnet. Jeder Knoten im Baum erhält einen Namen, der aus maximal 63 Zeichen bestehen kann. Jeder Knoten wird durch einen voll qualifizierten Namen beschrieben. Im Gegensatz zu einem Verzeichnisbaum beginnt die Beschreibung des voll qualifizierten Namens nicht mit dem Root (der Hauptdomäne), sondern mit dem Namen des in Frage kommenden Knotens und endet mit der Hauptdomäne. Die einzelnen Domänenknoten werden durch Punkte voneinander getrennt.

Im oben gezeigten Bild wäre ein Beispiel für eine voll qualifizierte Domäne folgende Angabe:

LUTTKUS.EPOS.DE

Eigentlich müsste noch ein Punkt als Trennzeichen für die Hauptdomäne angegeben werden, dieser wird jedoch im Allgemeinen weggelassen.

7.6.6.1.2
Seite 2



Auswertung des Namenseintrags

Der hier gezeigte Domänenname lässt folgende Interpretation zu:

- LUTTKUS ist der Name eine Systems (HOSTS) in der Domäne EPOS, die wiederum Bestandteil der nationalen Domäne DE ist.

Achtung:

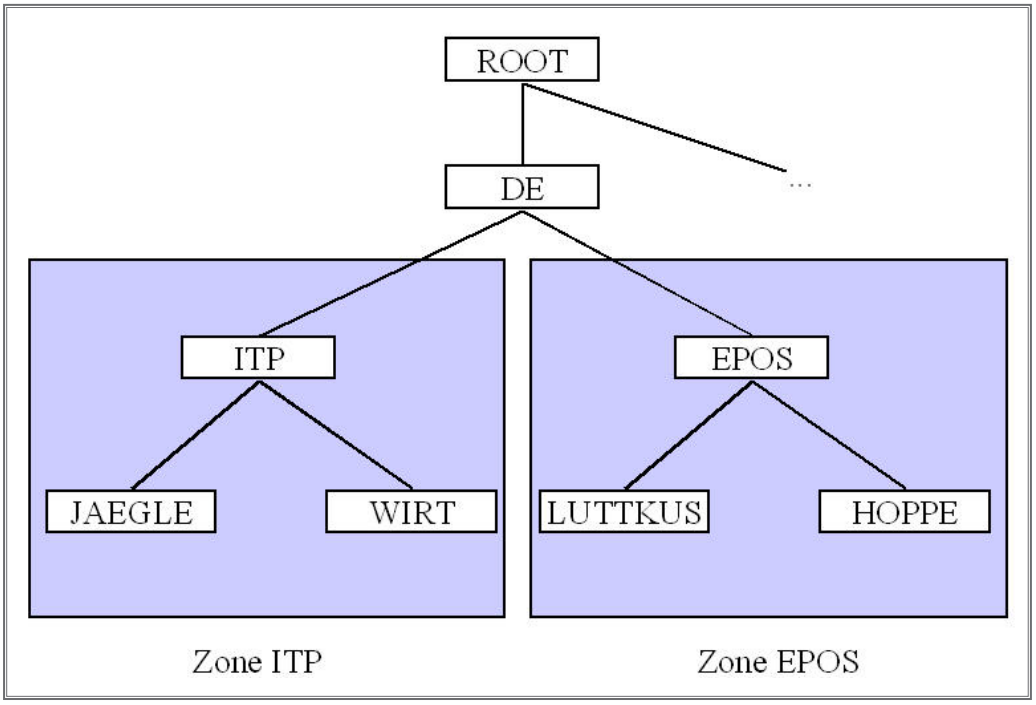
Rein theoretisch wäre auch folgende Interpretation zulässig. Die Domäne LUTTKUS ist Bestandteil der übergeordneten Domäne EPOS, die wiederum Bestandteil der nationalen Domäne DE ist.

Welche der genannten Interpretationen richtig ist, setzt genaue Kenntnisse über die jeweilige Domäne voraus und kann nicht dem Namen allein entnommen werden.

Die Zone

Der Domain Name Service wurde entwickelt, um das Internet insgesamt besser verwalten zu können. Dieses ist jedoch zu umfangreich, um von einem einzigen Punkt zentral verwaltet werden zu können. Deshalb hat man die Verwaltung von Subdomänen an untergeordnete Namens-Server übergeben.

Namens-Server (auch Domain Name Server) sind Programme, die einen Domain Name Space verwalten. Dabei kann der gesamte Verwaltungsbereich in so genannte Zonen unterteilt werden.



Aufteilung einer Domäne in Zonen

Eine Zone stellt die Untermenge eines Domain Name Space-Baums dar. Wobei dann ein Namens-Server für eine oder mehrere Zonen zuständig sein kann.

7.6.6.1.3

Seite 2

Warnung: Der Domänenbegriff

Ich möchte Sie dringend bitten, einen Sachverhalt zu berücksichtigen. Als Netzwerkadministratoren in einem heterogenen Netzwerk, welches unterschiedliche Systeme beinhaltet, werden wir öfter mit dem Begriff der Domäne konfrontiert, vor allen Dingen aber dann, wenn wir mit Microsoft Betriebssystemen arbeiten müssen.

Das Microsoft-Netzwerk verwendet ebenfalls den Begriff der Domäne, jedoch in einem komplett anderen Kontext als im TCP/IP-Netzwerk:

Im Microsoft-Netzwerk beschreibt eine Domäne eine bestimmte Anzahl von Systemen, die über eine gemeinsame Sicherheitsdatenbank verwaltet werden. Es kann also durchaus geschehen, dass Hosts, die in verschiedenen Microsoft Domänen verwaltet werden, Bestandteil einer TCP/IP-Domäne sind.

Fazit: Die Domänenbegriffe dieser beiden Netzwerkwelten haben nichts miteinander zu tun!

7.6.6.1.4

Seite 2

Der iSeries Domain Name Service

Ab OS/400 V4R2 ist auf dem System IBM AS/400 bzw. iSeries ein DNS-Server verfügbar.

Beabsichtigen Sie den Domain Name Service auf der iSeries zu nutzen so sollten Sie sicherstellen, ob die Komponente 31 des Betriebssystem (5722-SS1) bereits installiert ist. Sollte dies nicht der Fall sein, so muss diese Komponente nachinstalliert werden. Verwenden Sie hierfür den Befehl RSTLICPGM oder alternativ das LICPGM-Menü und dann die Option 11 „Lizenzprogramm installieren". Mit dem Befehl CHKPRDOPT sollten Sie sodann überprüfen, ob die Installation korrekt vollzogen wurde.

Alle vom DNS der iSeries erzeugten Konfigurationsdateien werden direkt im Integrated File System der iSeries abgelegt und zwar in folgendem Zweig:

/QIBM/UserData/OS400/DNS

Für die Konfiguration des Dienstes sind folgende Sonderberechtigungen notwendig:

*ALLOBJ

*IOSYSCFG

Um Berechtigungsproblemen aus dem Weg zu gehen, sollten Sie als ein Benutzer, der der Berechtigungsklasse *SECOFR angehört, auf der iSeries angemeldet sein.

7.6.6.2
Seite 2

Der iSeries Navigator

Die Installation und Konfiguration eines DNS-Servers kann komplex sein. Jedoch beinhaltet der iSeries Navigator des iSeries Access einige Wizzards und Assistenten, die die Anfangskonfiguration des iSeries-Domain Name Service stark vereinfachen.

Der iSeries Navigator stellt die einzige verfügbare Schnittstelle für den DNS-Server bereit. Es gibt derzeit keine CL-Befehle, die die komplette Konfiguration eines Domain Name Service aus der 5250-Bildschirmschnittstelle heraus ermöglichen.

Die Unterstützung des DNS-Servers durch iSeries Navigator bietet mehrere Vorteile:

- Der Assistent für DNS-Konfiguration von iSeries Navigator liefert einen einfachen Prozess für die ersten Konfigurations- und Arbeitsschritte zum fertigen DNS-Server. Das System erstellt automatisch die vielen Dateien, die für Konfiguration und Betrieb eines DNS-Servers erforderlich sind.
- Die Funktion „Domänendaten importieren“ von iSeries Navigator bietet eine Möglichkeit, vorhandene DNS-Server-Konfigurationsdateien in Dateien umzusetzen, die in der Anzeige „DNS-Konfiguration“ von iSeries Navigator verwaltet werden können.
- Außerdem können iSeries-Host-Tabelleneinträge in Dateien umgesetzt werden, die in der Anzeige „DNS-Konfiguration“ von iSeries Navigator verwaltet werden können.
- Normalerweise werden durch den Domain Name Service Hostnamen in numerische Adressen umgesetzt. iSeries Navigator stellt eine Methode für das automatische Aktualisieren von primären Domänendateien mit umgekehrter Zuordnung (IP-Adresse zu Host-Name) mit einem PTR-Ressourceneintrag bereit. Diese umgekehrte Richtung der Adressauflösung wird für einige TCP/IP-Dienste benötigt. Das System führt diesen Schritt automatisch – wenn gewünscht – bei der Eintragung eines neuen Host aus, wenn zu einer primären Domänendatei mit normaler Zuordnung (Host-Name zu IP-Adresse) ein Eintrag hinzugefügt wird.

7.6.6.2.1

Seite 2

Die Konfiguration des Domain Name Service

Für die Anfangskonfiguration unseres Domain Name Service vergegenwärtigen wir uns folgendes Szenario: Wir haben ein einfaches nicht allzu großes TCP/IP-Netzwerk. Bestandteil dieses Netzwerks sind 50 PC, die unter Windows laufen. Derzeit wird die Zugangsberechtigung zum Netzwerk nicht durch einen Windows Server gesteuert. Zentrales Serversystem in unserem Netzwerk ist eine iSeries. Diese iSeries wird als Applikations-Server – 5250-Anwendungen – und als File-Server für alle PC-Dateien eingesetzt. Diese Daten werden durch Office Anwendungen (Winword, Excel etc.) erzeugt und direkt im ASCII-Bereich des Integrated File Systems gehalten. Eine lokale Datenhaltung ist nicht vorgesehen. Bisher wurde hauptsächlich mit numerischen IP-Adressen gearbeitet, bei einigen Systemen wurden auch Hosttabellen eingesetzt. Um eine einfachere Administration zu ermöglichen, soll jetzt die iSeries als Domain Name Server eingesetzt werden.

Da alle PC-Systeme auf die iSeries als zentrales System via iSeries Access zugreifen, soll dieser Zugriff in einem ersten Arbeitsgang ermöglicht werden. Folgender voll qualifizierter Name wurde für die iSeries festgelegt:

EPOSIS01.EPOSGMBH.DE

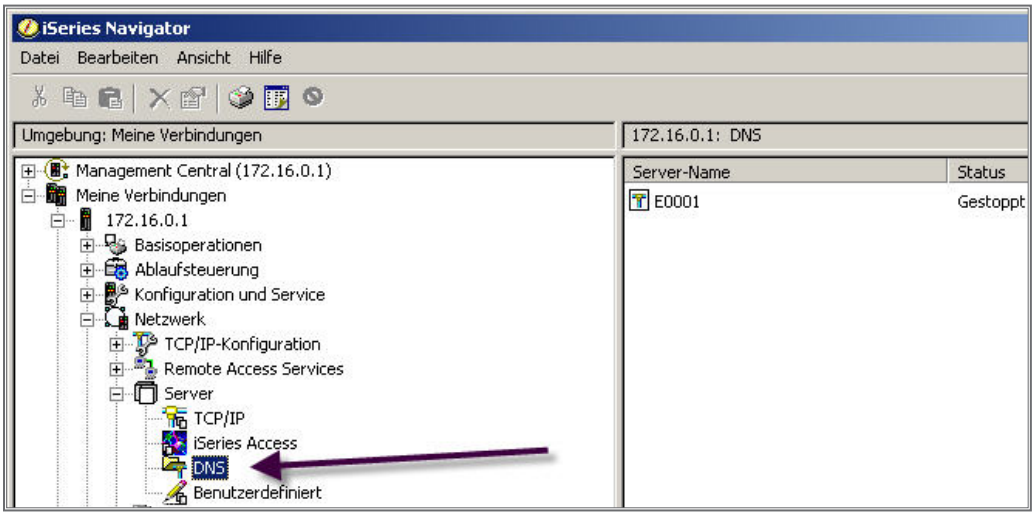
Die Schnittstelle für diesen Host wurde mit der IP-Adresse 172.16.0.1 gebunden.

Lassen Sie uns mit der Konfiguration beginnen, indem wir den iSeries Navigator (IN) vom Desktop unseres PC aus aufrufen.

Sobald der IN aufgerufen wurde, müssen Sie das Maschinensymbol für die iSeries expandieren. Öffnen Sie sodann den Knoten Netzwerk, unterhalb dieses Knotens finden Sie das Verzeichnis DNS.

Sollte dieses nicht der Fall sein, so müssen Sie die Betriebssystemkomponente 31 (s.o.) nachinstallieren.

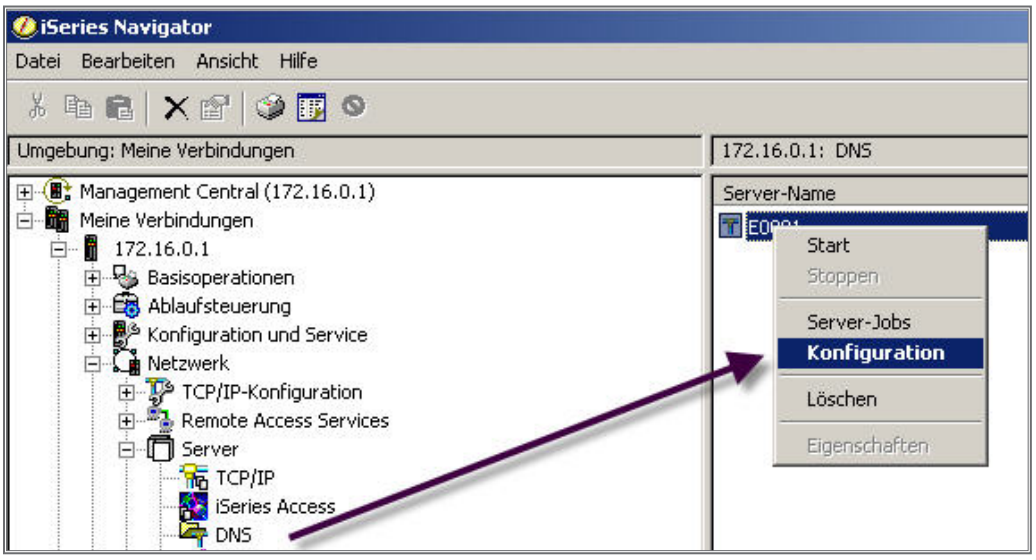
7.6.6.2.2
Seite 2



Der Domain Name Service ist installiert

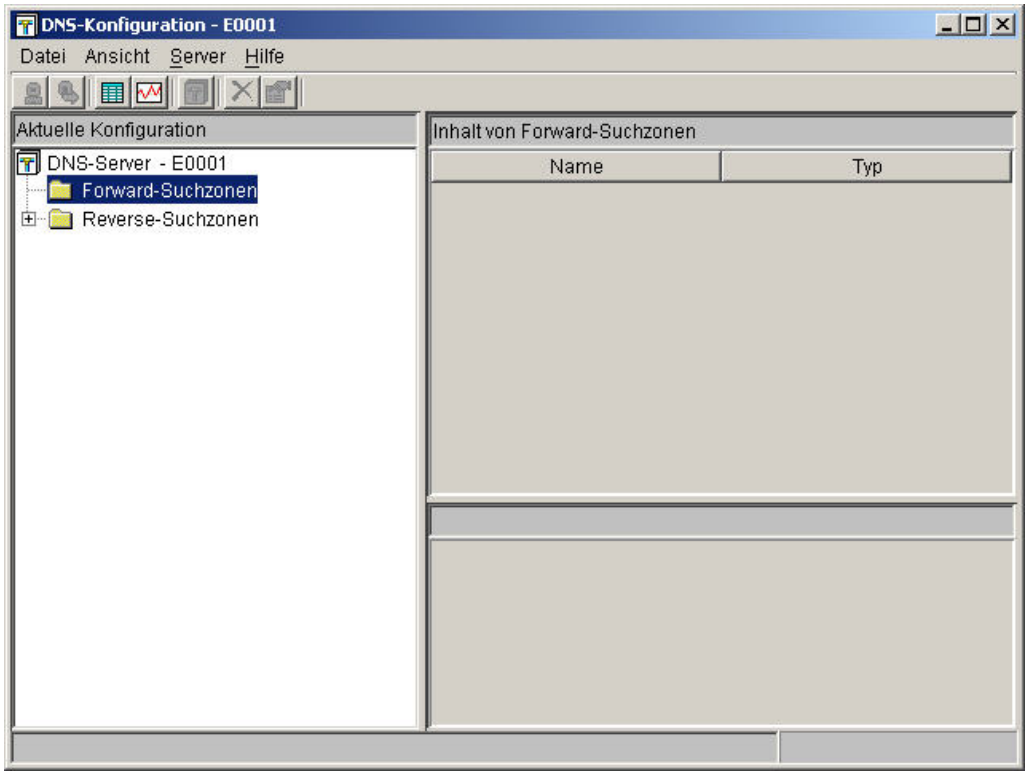
Wir sehen hier, dass der Domain Name Service vorhanden ist, er ist derzeit jedoch nicht gestartet. Ein Start des Dienstes ergibt auch keinen Sinn, denn bisher wurden auch keine Einträge in den Domain Name Space gemacht.

Ich markiere jetzt den Eintrag DNS mit der rechten Maustaste. Es öffnet sich folgendes Kontextmenü:



Beginn der Konfiguration

Durch Auswahl des Menüpunkts „Konfiguration“ beginnt der Konfigurationsdialog. Bedingt durch den Umstand, das der DNS-Dienst bisher weder aktiv noch konfiguriert war, wird jetzt automatisch der Konfigurationsassistent des iSeries Navigator aktiv, der uns praktisch durch den Konfigurationsdialog führt.

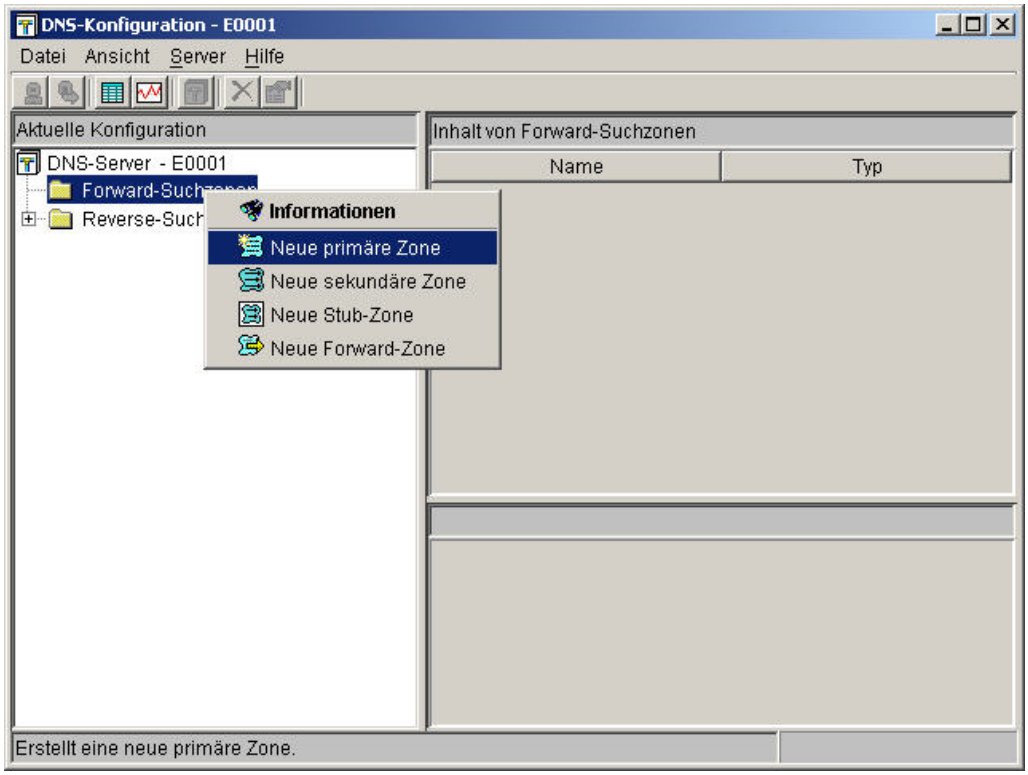


Der DNS-Administrator

Zuerst öffnet sich der DNS-Administrator, in dem alle Pflege- und Wartungsarbeiten bezüglich des DNS realisiert werden.

Um eine primäre Zone anzulegen, fordere ich mit der rechten Maustaste den Eintrag „Forward-Suchzonen“ im Administrator an und folgendes Kontextmenü erscheint.

7.6.6.2.2
Seite 4



Neue primäre Zone

Warum erzeugen wir zuerst eine primäre Zone?

Da dieses der erste DNS ist, der von uns definiert wird, muss hier das Menü „Primäre Zone“ gewählt werden. Ein primärer Namens-Server lädt die Informationen eines Berechtigungsbereichs (die Zonendaten!) von der Platte und hat die Berechtigung für den in der Plattendatei angegebenen Berechtigungsbereich.

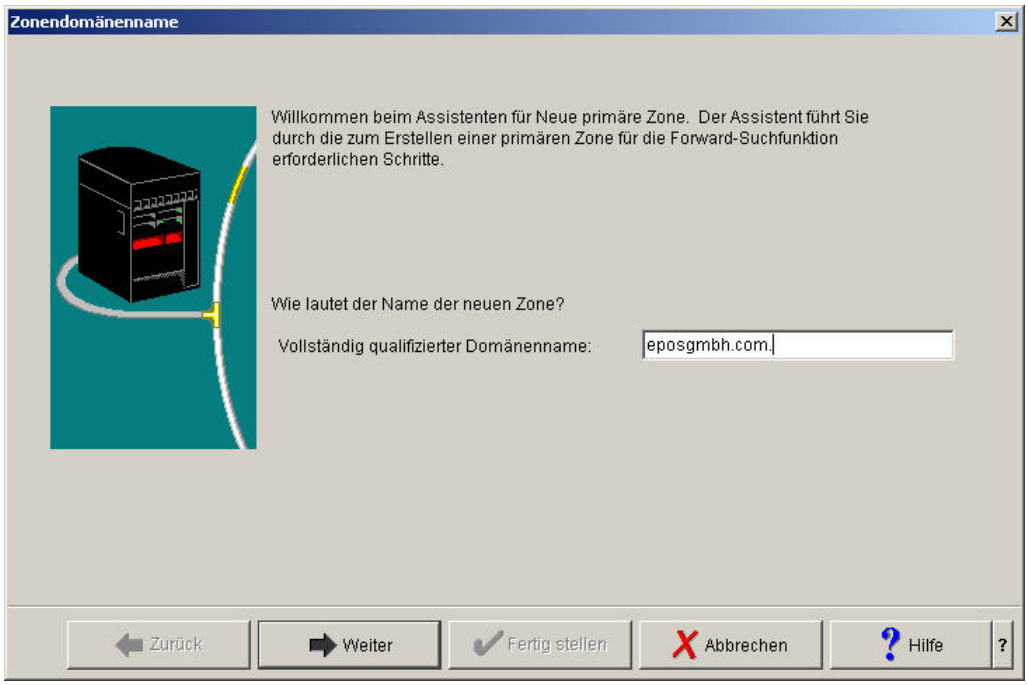
Ein sekundärer Namens-Server (sekundäre Zone) hat die Berechtigung für einen Berechtigungsbereich, erhält die Informationen zu seinem Berechtigungsbereich jedoch von einem primären Server. Dieser Prozess wird „Übertragung des Berechtigungsbereichs“ genannt. Um die Synchronisation zwischen sekundärem und primärem Server aufrechtzuerhalten, senden die sekundären Namens-Server regelmäßig (in der Regel alle drei Stunden) Anfragen an den primären Namens-Server. Die Übertragung des Berechtigungsbereichs wird erneut ausgeführt, wenn der primäre Server aktualisiert wird. Die Definition eines sekundären Namens-Servers ist eine Absicherung gegen den Ausfall des primären Domänen-Servers. Sollte der primäre Server ausfallen, so übernimmt der sekundäre Server dessen Aufgaben.

Ein Namens-Server, der keine Berechtigung für einen Berechtigungsbereich hat, wird als Cache-Only-Server (hier: Stub Zone) bezeichnet. Ein Cache-Only-Server erhält alle angeforderten Daten von primären oder sekundären Namens-Servern. Ein solcher Server benötigt mindestens einen Root-Server,

um auf einen Namens-Server zu zeigen, von dem er Informationen erhalten kann. Stub-Server sind spezielle sekundäre Zonen-Server.

Mit einem Mausklick starte ich den Dialog „Primäre Zone“.

In diesem Dialog wird erwartet, dass hier die TCP/IP-Domäne benannt wird, für den der DNS-Dienst aktiv sein wird.



Hinzufügen des Root-Server 1

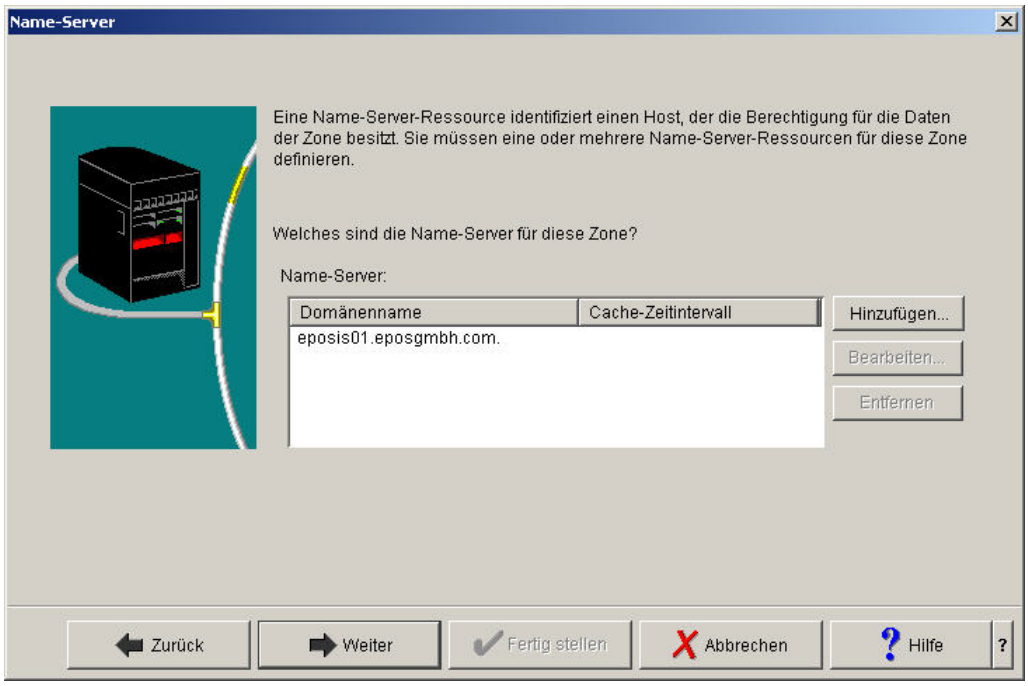
Der angezeigte Wert wird aus den TCP/IP-Netzwerkattributen der iSeries automatisch entnommen. Sollte hier kein Wert angegeben sein, so haben Sie in den Domänenattributen der iSeries (Menü CFGTCP à Auswahl 12) keinen entsprechenden Eintrag hinterlegt. Holen Sie diesen Arbeitschritt gegebenenfalls nach. Und beginnen Sie eine neue Konfiguration der primären Zone.

Achtung!

Beachten Sie, dass der Domänenname im oben gezeigten Bild mit einem Punkt abgeschlossen wird.

7.6.6.2.2
Seite 6

Betätigen Sie jetzt den „Weiter“-Knopf und folgendes Fenster öffnet sich:



Name-Server Eintrag

Auch hier sollte automatisch der voll qualifizierte Hostname (mit abschließenden Rott-Domänen-Punkt!) angezeigt werden. Wenn nicht, so fügen Sie den Namen durch „Drücke auf Hinzufügen…“ hinzu.

Wichtig ist folgende Form:

- Vollständig qualifizierter Host-Name:
  EPOSIS01.EPOSGMBH.COM.

Groß- und Kleinschreibung wird im DNS nicht unterschieden, so dass Sie entweder die Namen groß, klein oder gemischt schreiben können.

Statische oder dynamische Zone

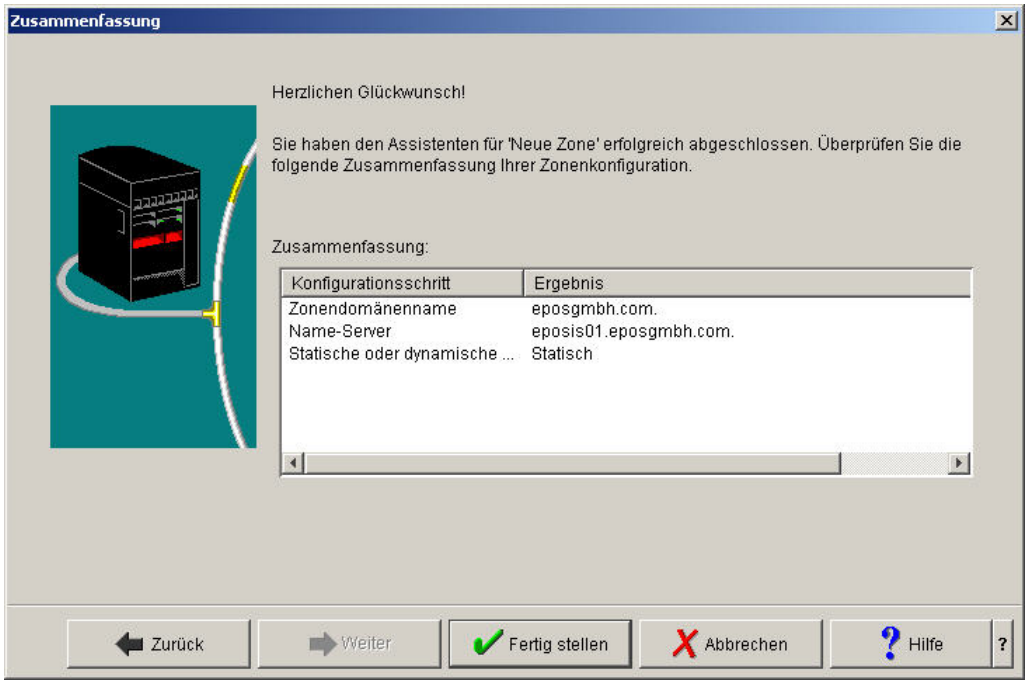
Im gezeigten Dialog wird abgefragt, ob Sie eine so genannte statische oder dynamische Zone erstellen wollen. Eine statische Zone erwartet, dass alle Konfigurationen bezüglich der Hosts, die später durch diese Zone verwaltet werden sollen, händisch in die Zone eingetragen werden.

In einer dynamischen Zone trägt sich ein Host automatisch in die Zone ein, wenn er das erste Mal in der Zone zu der er per Client Definition gehört, aktiv wird.

Wir beabsichtigen eine statische Zone zu nutzen.

Mit „Weiter“ werden diese Eingaben bestätigt. Daraufhin erscheint das Fenster „Zusammenfassung“:

7.6.6.2.2
Seite 8



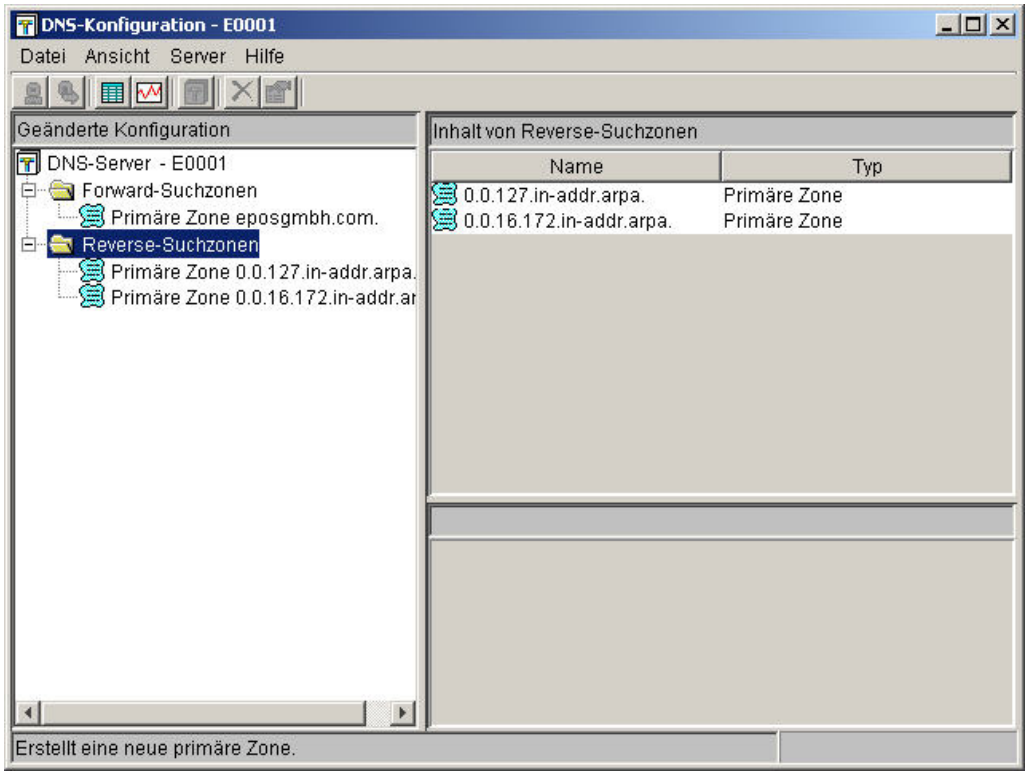
Zusammenfassung

Ich beende den Dialog mit „Fertig stellen“.

Achtung:

Wir haben jetzt lediglich definiert, welches sozusagen der gastgebende Server für den DNS-Dienst unseres Netzwerks ist. Wir haben jedoch noch keinen Eintrag hinterlegt, mittels dessen eine Namens- bzw. Adressauflösung getätigt werden kann. Diese Einträge müssen noch später erfolgen. Also wir sind noch lange nicht am Ende.

Als nächstes wird die Reverse-Suchzone erstellt.



Reverse-Suchzone

Ich markiere in der Anzeige des DNS den Verzeichniseintrag Reverse-Suchzone, mittels des Kontextmenü-Eintrags „Primäre Zone“ wird auch hier ein Wizzard tätig, der mich durch die Konfiguration führt.

Diese Dialoge entsprechen denen der Forward Zone. Deshalb werden die entsprechenden Dialoge hier nicht mehr gezeigt, da sie aus dem vorherigen bekannt sind.

In der Folge geht es jetzt darum neue Hosts in die Forward- und Reverse-Zone einzutragen.

Nachdem jetzt die Schnittstelle unserer iSeries unter dem Namen EPOSIS01.EPOSGMBH.COM angesprochen werden kann, widmen wir uns der Aufnahme weiterer Hosts in die Zonendatei des von uns konfigurierten DNS-Servers.

Doch lassen Sie uns zuerst die Frage stellen, welche Hosts überhaupt mit Hilfe des DNS-Dienstes angesprochen werden sollen. Grundsätzlich lässt sich folgendes sagen: Es macht eigentlich nur Sinn Zielhosts in die Zonendatei aufzunehmen. Verstehen wir unsere iSeries als zentralen Server in unserem Netzwerk, der Applikations-Server, File-Server, Kommunikations-Server etc. ist, so betrachten wir dieses System sicherlich als einen Zielhost, der uns zentral Ressourcen zur Verfügung stellt.

7.6.6.2.2
Seite 10

Meistens ist unser Netzwerk jedoch nicht so strikt zentralisiert. Wir haben z.B. einen oder mehrere Windows-Server in unserem Netzwerk, die für die Berechtigungssteuerung, den Backup von Berechtigungsdaten und das Versorgen mit Windows-Anwendungen zuständig sind. Möglicherweise tun dann auch noch ein oder mehrere Linux-Systeme ihren Dienst als Kommunikations-Server, dann haben wir auch noch einen RS6000 Rechner, auf dem Lotus Domino installiert ist usw. usf.

Die Folge davon ist, dass wir nicht nur unsere iSeries als Ziel-Host in unseren Domain Name Service eintragen müssen, sondern auch noch diverse andere Server-Systeme.

An dieser Stelle setzen wir jetzt ein. Stellen Sie sich folgenden Sachverhalt vor:

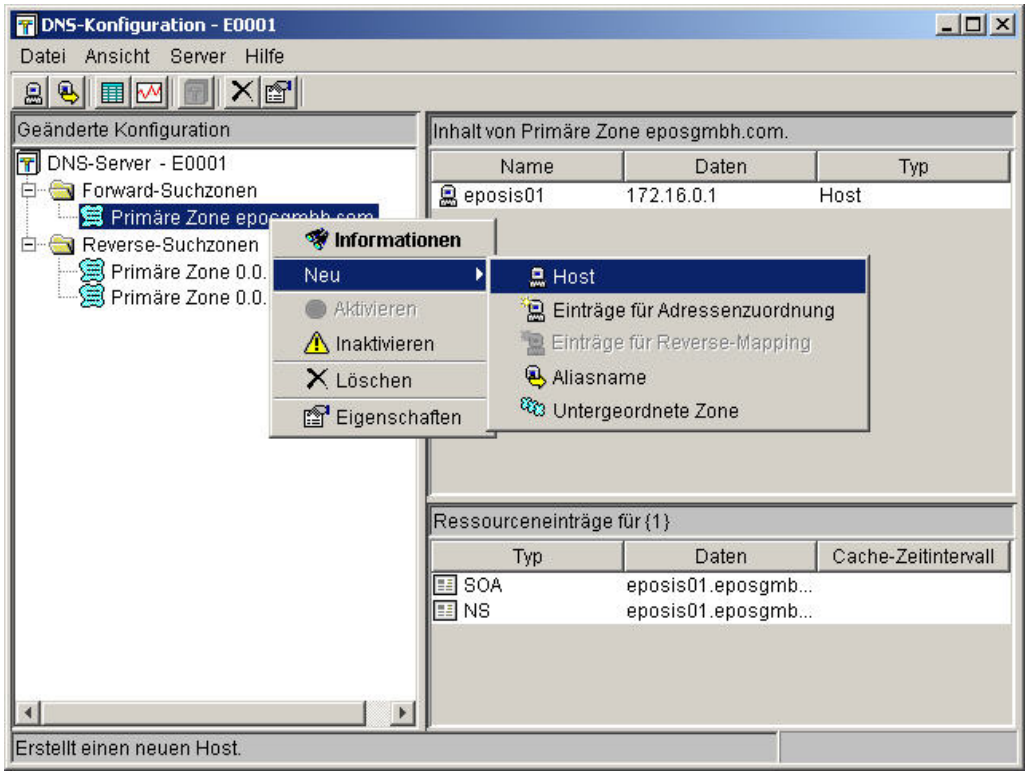
Neben dem iSeries System EPOSIS01 mit der IP-Adresse 172.16.0.1, existiert noch ein Windows-Server als Applikations-Server. Dieser Server versorgt das gesamte Netzwerk mit Windows-Applikationen wie Winword, Excel, Powerpoint etc. Auch dieses System wird zentral von allen Clients als Zielserver angesprochen und soll mit Hilfe des DNS-Dienstes auf der iSeries angesprochen werden können.

Folgende Werte/Eckdaten müssen uns bekannt sein:

- die IP-Adresse,
- der Name, der für den neuen Host zu verwenden ist.

Der Name des Windows-Servers wird von uns mit TSESERVER festgelegt, die IP-Adresse für unser Beispiel lautet 172.16.0.5.

Jetzt beginnt die Konfiguration. Rufen Sie den iSeries Navigator auf, gehen Sie über das Symbol „Netzwerke“ in den Bereich TCP/IP, dann in den Bereich DNS und öffnen Sie sodann den DNS-Dienst durch Doppelklick.



DNS Administrator – Hostkonfiguration

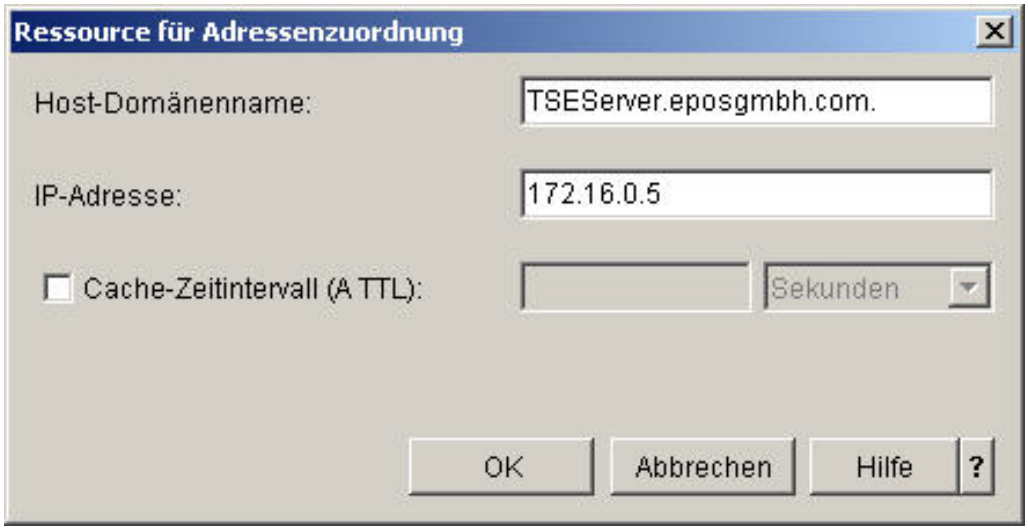
Nachdem sich der DNS-Administrator geöffnet hat, sehen Sie auf der rechten Seite des DNS-Explorers, die bereits in den DNS-Dienst aufgenommenen Hosts. In unserem Falle lediglich den Hostnamen und die IP-Adresse des bisher eingetragenen Host EPOSIS01 – 172.16.0.1.

Markieren Sie jetzt Forward-Suchzone und gehen Sie jetzt mit der rechten Maustaste in das Kontextmenü „Neu ➜ Einträge für Adresszuordnung“. Folgendes Fenster öffnet sich:

7.6.6.2.2
Seite 12

Neue Adresszuordnung

Klicken Sie jetzt auf „Hinzufügen“ und der eigentliche Konfigurationsdialog für die Aufnahme eines neuen Host in den DNS-Dienst der iSeries beginnt.



Adresswerte versorgen

In diesen Dialog tragen wir folgende Werte ein:

Host-Domänenname: TSEServer.eposgmbh.com.

Achtung!

Vergessen Sie nicht den Root-Domänen-Punkt. Auch hier gilt, dass nicht nach Groß- und Kleinschreibung unterschieden wird.

IP-Adresse: 172.16.0.5

Die anderen Werte interessieren uns an dieser Stelle nicht und deshalb wird der Dialog mit „OK“ bestätigt und beendet. Jetzt erscheint ein Fenster, was wir eigentlich schon kennen, jedoch beinhaltet dieses Fenster die gerade eingegebenen Daten:

Ein neuer Host wurde in den DNS aufgenommen

Wir sehen hier, dass dem Hostnamen TSESERVER der Name der Domäne einfach beigefügt wurde und welcher IP-Adresse dieser Hostname entspricht.

Eigentlich könnten wird jetzt schon den Dialog insgesamt beenden, Sie sehen jedoch, dass „Fertig stellen“ noch nicht aktiviert wurde. Was bleibt uns also noch zu tun?

Wir müssen entscheiden, ob der von uns erstellte Host nicht nur in die Forward-Suchzone, sondern auch in die Reverse-Suchzone eingetragen wird.

7.6.6.2.2
Seite 14

Reverse-Mappings erstellen 1

Im oben gezeigten Dialog markiere ich „Ja“, so dass der Wizzard zur Erstellung von Reverse-Mapping-Einträgen in der Reverse-Suchzone gestartet wird.

Warum?

Einige TCP/IP-Dienste betreiben nicht nur die Auflösung eines Hostnamens in eine IP-Adresse, sondern benötigen die Auflösung einer IP-Adresse in einen Hostnamen.

Hostname	Auflösung	IP-Adresse
EPOSAS01	→	172.16.0.1
EPOSAS01	←	1.0.16.172

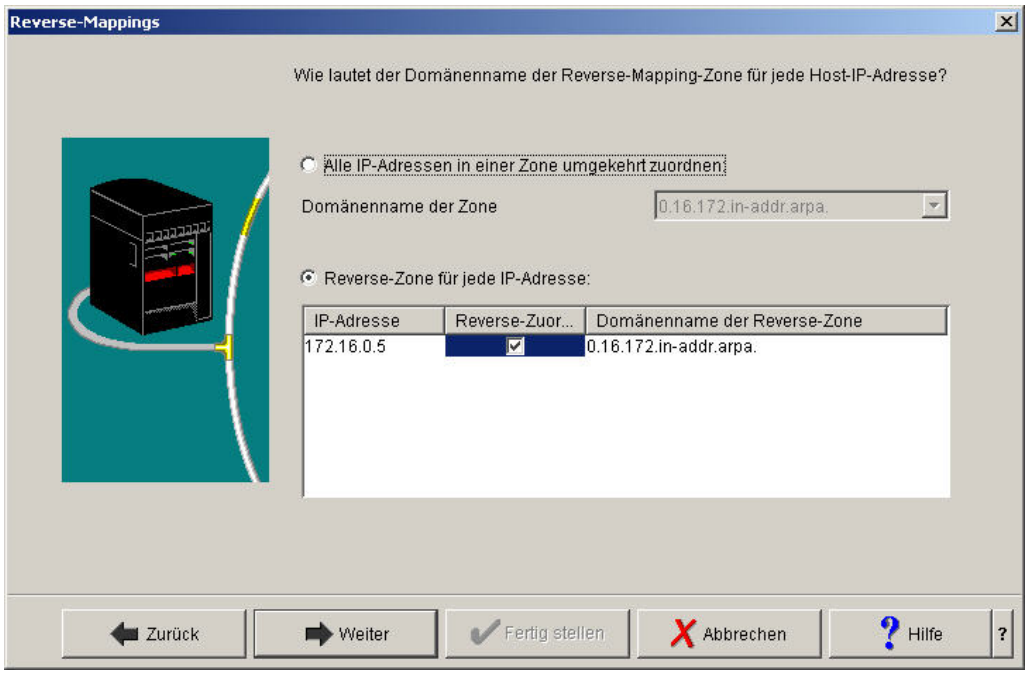
Ich habe diese Vorgänge für Sie noch einmal in der dargestellten kleinen Tabelle aufgezeichnet. Die erste Zeile beschreibt die Adressauflösung in eine IP-Adresse. Die zweite Zeile die Adresse einer IP-Adresse in einen Hostnamen. Achten Sie einmal genau auf die Schreibweise der IP-Adresse. Das ist kein Druckfehler. Die Auflösung der IP-Adresse beginnt mit der Host-ID der gesamten Adresse.

Damit diese umgekehrte Adressauflösung funktionieren kann, muss ein Eintrag für die umgekehrte Zuordnung einer Adresse hinterlegt werden. Diese „umgekehrten Einträge“ werden jedoch nicht in die gleiche Zonendatei geschrieben, wie die anfänglich von uns erzeugten Hosteinträge.

Für unseren DNS-Dienst haben wir eine Domäne EPOSGMBH.COM erzeugt. Diese enthält alle Hosteinträge für die Auflösung eines Namens in eine IP-Adresse.

Erstellen wir umgekehrte Einträge, so wird eine Domäne mit dem Namen 0.16.172.in.addr.arpa erzeugt. Der numerische Teil dieses Domänennamens ist variabel und hängt von der vergebenen IP-Adresse ab.

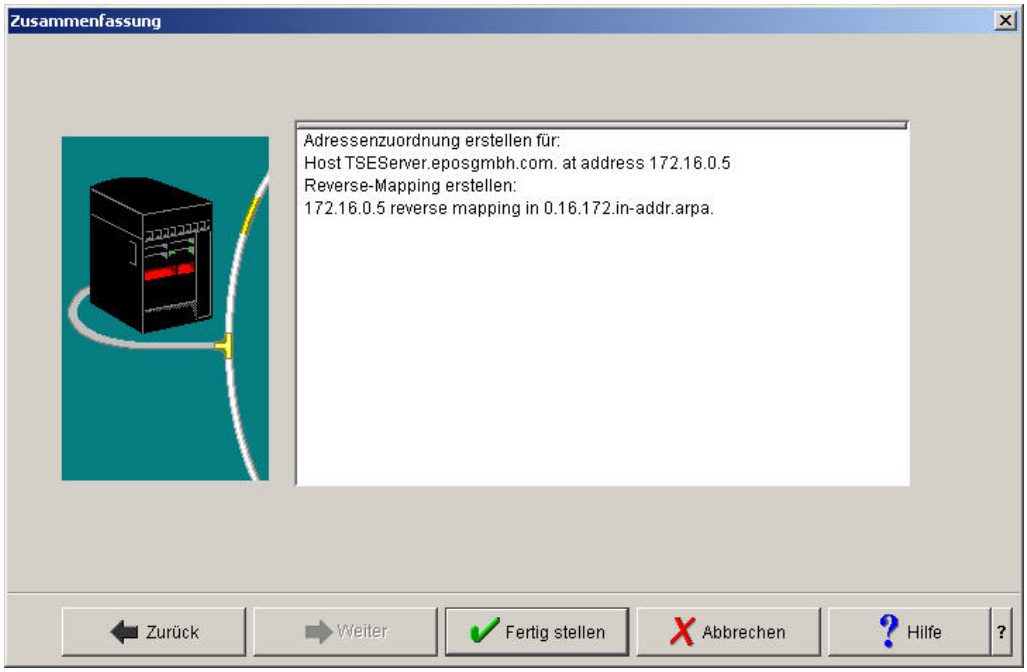
Ich klicke auf „Weiter“.



Reverse-Mappings erstellen 2

Hier bestätige ich lediglich die Default-Angaben und drücke „Weiter“.

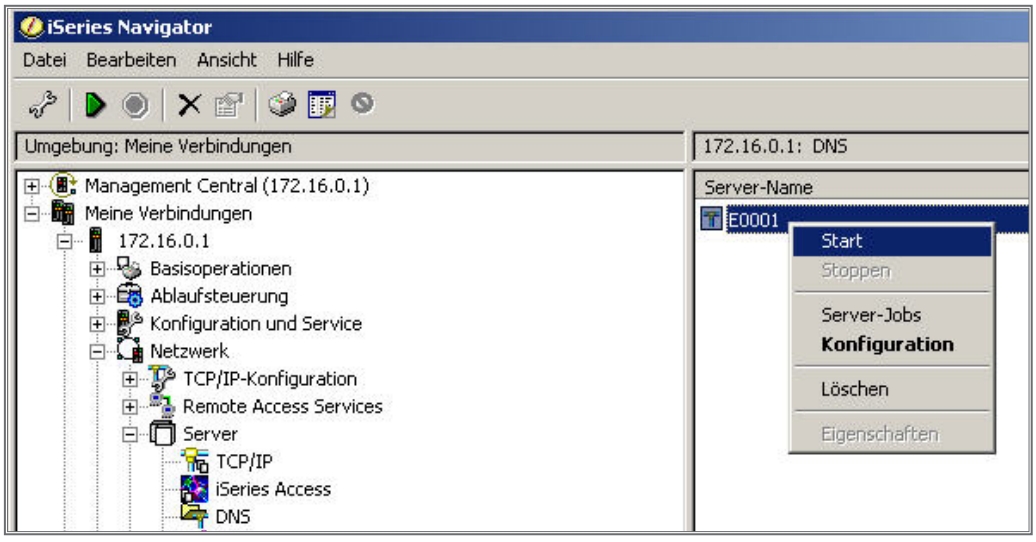
7.6.6.2.2
Seite 16



Reverse-Mappings erstellen 3

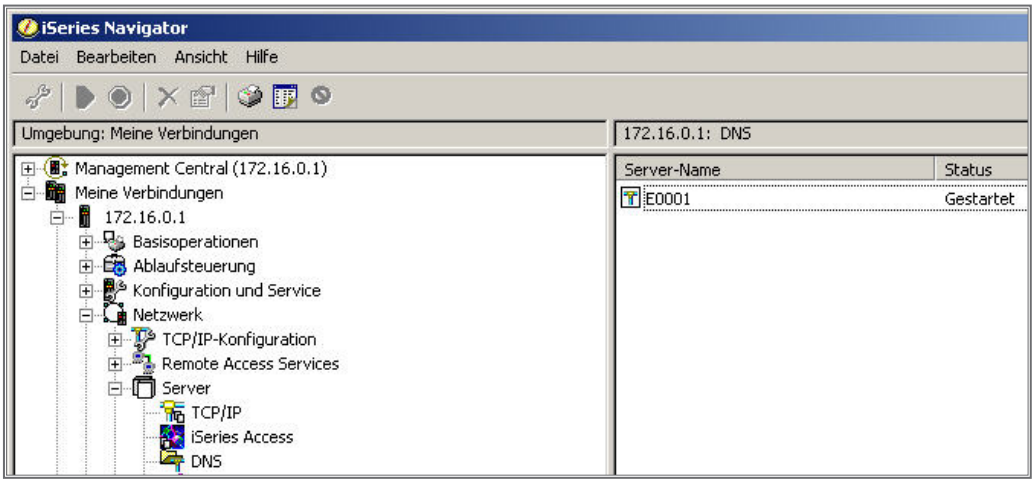
Nachdem wir im Register den Punkt „Umgekehrte Zuordnungssätze“ für diesen Host erstellt und den Dialog mit „Fertig stellen“ bestätigt haben, finden Sie dann auch im DNS-Administrator des DNS-Dienstes die entsprechenden Daten unter Reverse-Suchzone.

Um jetzt diese neuen Konfigurationswerte in den Serverbetrieb zu übernehmen, starten Sie den DNS-Dienst.



Der Server wird gestartet

Achten Sie darauf, dass der Server wirklich gestartet ist!



Der DNS-Server läuft!

Können Sie jedoch nicht den Serverdienst beenden, weil er von Clients benutzt wird, und ein Abstellen den gesamten Betrieb in Ihrem Unternehmen zusammenbrechen lassen würde, so gibt es noch den folgenden Weg:

2.) Öffnen Sie den DNS-Adminstrator und öffnen Sie das Menü „Server ➜ Aktualisieren“, sodann werden alle neu konfigurierten Hosteinträge in der DNS Zone aktiv.

Was haben wir bis jetzt erreicht? Dazu betrachten wir noch einmal die Einträge im DNS-Administrator.

Betrachten wir zuerst den linken Bereich des Fensters. Hier wird die aktuelle Konfiguration angezeigt. Wir sehen folgende Bestandteile der Konfiguration als Verzeichnisstruktur:

- Name des DNS Servers: E0001
- die Primäre Zone: eposgmbh.com.
- eine Reverse Zone: 0.0.127.in-addr-arpa
- eine weitere Reverse Zone eposgmbh.com.: 00.16.172. in-addr-arpa

Im Kontextmenü „Neu“, das wir bereits für die Eintragung eines neuen Host verwendet haben, findet sich noch ein weiterer interessanter Konfigurationsgang: Alias-Name.

Im Dialog „Alias-Name“ können für vergebene Hostnamen Alias-Namen hinterlegt werden.

Im Prinzip könnte man einfach sagen, dieses wäre ein Parameter für die Schreibfaulen unter uns. Durch den erzeugten Hosteintrag in der DNS-Datenbank wird unser Windows-Server unter dem Hostnamen TSESERVER.EPOSGMBH.COM anzusprechen sein. Es wäre doch schön, wenn wir alternativ einen Kurznamen verwenden könnten, wie z. B. TSE. Sollte das der Fall

sein, so würden wir hier jetzt durch den „Hinzufügen"-Dialog den Namen TSE eintragen und wir könnten diesen Windows-Server einfach mit dem Namen TSE einsprechen. Auch hinter diesem Namen stünde im übertragenen Sinne die IP-Adresse 172.16.0.1.

Aber nur Schreibfaulheit als Grund für die Existenz von Alias-Hostnamen anzuführen wäre nicht genug. Eigentlich steckt vielmehr folgender Sachverhalt dahinter: Wenn ein Serverdienst nicht die Berechtigungsstruktur des OS/400 nutzt, aber die im OS/400 angelegte Netzwerkschnittstelle verwendet, so muss zwischen den Serverdiensten der OS/400-Umgebung und denen des „Fremdserver" unterschieden werden. In diesem Fall **muss** mit Alias-Namen gearbeitet werden. Zu kompliziert? Lassen Sie uns ein einfaches Beispiel zur Erklärung heranziehen.

Nehmen wir einmal folgende Situation an:

Unsere iSeries ist im DNS der iSeries mittels des Hostnamens EPOSIS01.EPOSGMBH.COM erreichbar. Hinter diesem Hostnamen steht die IP-Adresse 172.16.0.1.

Wir installieren auf diesem iSeries-System den Lotus Domino-Server. Der Domino-Server erhält den Namen EPOSAS01. Da der iSeries-basierte Domino-Server ausschließlich über TCP/IP angesprochen werden kann, nutzt er in unserem Falle die gleiche Netzwerkkarte, wie die restlichen iSeries-Serverdienste.

Was ist daran so bemerkenswert? Wir haben einen Adressenkonflikt! Wenn jetzt z.B. ein iSeries Access Dienst mit Hilfe des Hostnamens EPOSAS01.EPOSGMBH.COM auf die Hostserver der iSeries zugreifen will, so wird er diese auch finden und mit ihnen arbeiten können. Soweit ist alles im grünen Bereich. Ist jetzt aber ein Lotus Notes Client auf demselben PC installiert, so wird er den Domino-Server mit dem Namen EPOSAS01 suchen. Der Notes Client versucht aber ebenfalls mittels TCP/IP die Verbindung herzustellen. Arbeitet der jedoch mit einem Namen, der nicht im DNS-Dienst der iSeries eingetragen wurde (EPOSAS01), so wird die Meldung zurückgegeben, dass keine Verbindung hergestellt werden konnte. Der Notes Client kann also seinen Domino-Server nicht finden.

Tragen wir jedoch den Namen EPOSAS01 als Alias-Namen für den Namen EPOSAS01.EPOSGMBH.DE ein, so kann auch der Lotus Notes Client den Domino-Server finden.

Windows Clients für DNS konfigurieren

Nachdem die Serverdienste konfiguriert wurden, müssen die Clients jetzt für den Betrieb mit dem DNS-Server der iSeries konfiguriert werden. Solange dieses nicht geschehen ist, kann kein Client auf den DNS-Dienst der iSeries zugreifen. Jedoch ist die Anbindung eines Windows-PC an den DNS-Dienst relativ einfach und unaufwendig.

Die gesamte Konfiguration erfolgt aus der Netzwerkumgebung des Windows-PC heraus. Ich markiere also, das Symbol „Netzwerkumgebung“ auf dem Desktop meines Windows-PC. Um diese Umkonfiguration der Netzwerkattribute realisieren zu können, habe ich mich lokal als Administrator angemeldet. Dieses ist notwendig, da es sich bei meinem Windows Client um eine XP Workstation handelt. Beachten Sie, dass für diese Arbeit nicht nur das Benutzerkonto Administrator genutzt werden kann. Es kann auch jedes beliebige andere Benuterzkonto genutzt werden, solange es über die lokalen Administrationsrechte verfügt.

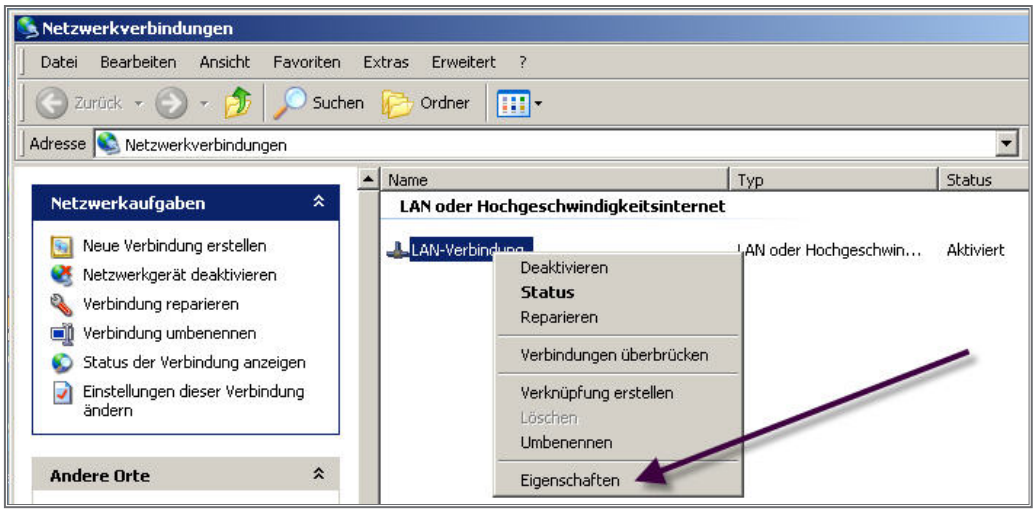
Die Windows-Netzwerkumgebung

Sodann drücken Sie die rechte Maustaste, markieren den Eigenschaften-Dialog und öffnen ihn durch Doppelklick.

7.6.6.3
Seite 2

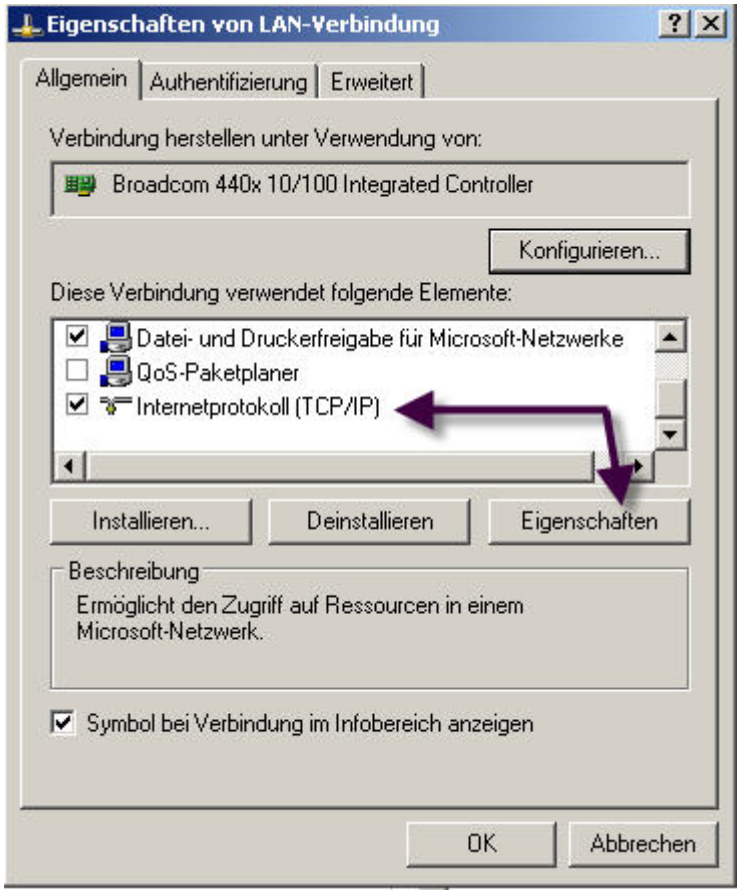
Die Eigenschaften der Netzwerkumgebung

Im Fenster „Netzwerkverbindungen“ markiere ich den Eintrag LAN-Verbindung und öffne mit der rechten Maustaste das Kontextmenü.



Die Eigenschaften der LAN-Verbindung 1

Auch hier öffne ich den Eigenschaften-Dialog.

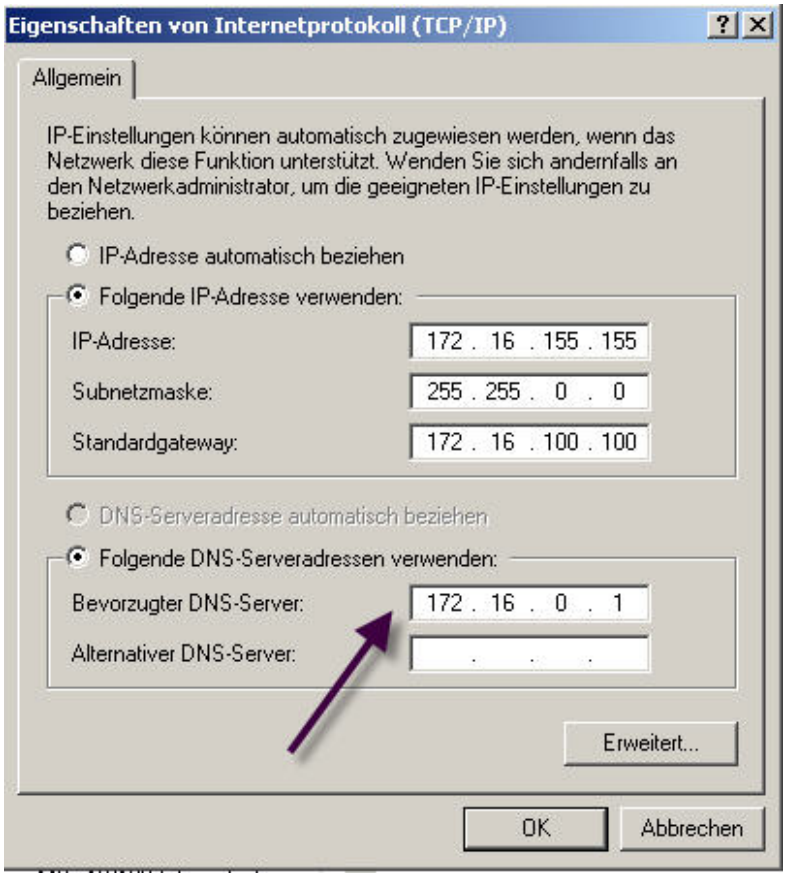


Die Eigenschaften der LAN-Verbindung 2

Sie sehen, dass für den hier betrachteten Client nur das TCP/IP-Protokoll zur Anwendung kommt. Durch einen Doppelklick auf den Eintrag „Internet-Protokoll“ öffne ich den Eigenschaften-Dialog.

Uns interessiert hier der aktivierte Punkt „Folgende DNS-Serveradressen verwenden“:

7.6.6.3
Seite 4



Bevorzugter DNS

Hier trage ich die IP-Adresse der iSeries ein. Damit ist die Konfiguration abgeschlossen.

Achtung! Arbeiten Sie mit einer sekundären Zone, die als Backup für Ihre primäre Zone dient, so sollten Sie die Adresse des sekundären DNS unbedingt in den Parameter „Alternativer DNS“ eintragen. Fällt Ihr primärer Server aus, so übernimmt der sekundäre, der ja die gleichen Namen auflöst wie der primäre automatisch die Namensauflösungsfunktion.

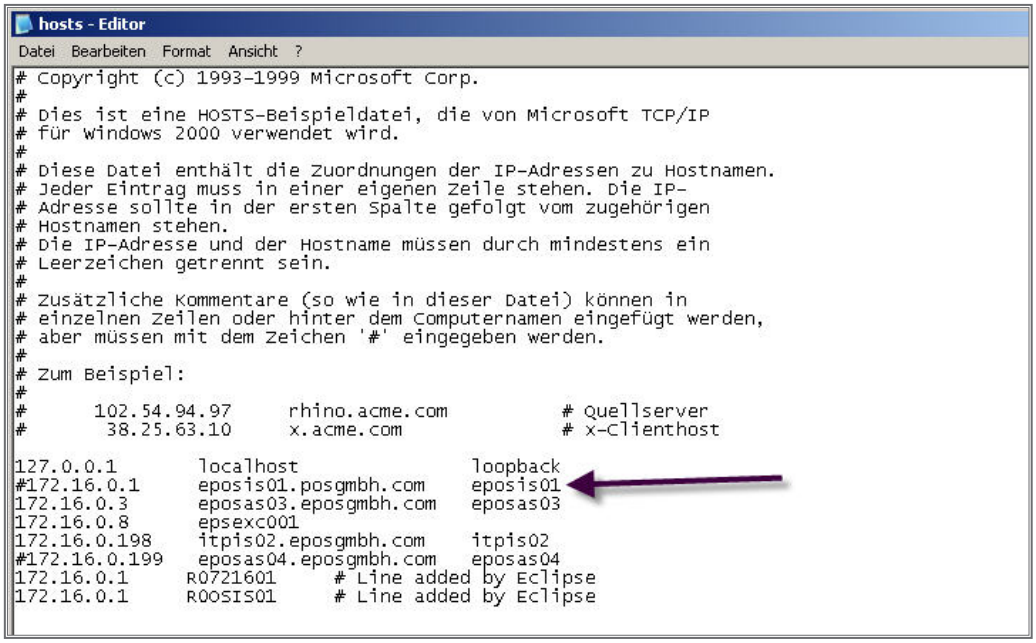
Achtung! Sollten Sie vor der Nutzung des DNS-Server bereits mit lokalen Hosttabellen gearbeitet haben, so sollten Sie beachten, dass Windows-Systeme grundsätzlich folgende Suchreihenfolge bei der Namens-Adressauslösung berücksichtigen.

- Zuerst in lokaler Hosttabelle suchen.
- Wird dort ein Eintrag gefunden, wird der Name in der betreffenden Adresse aufgelöst.
- Wird der Eintrag nicht gefunden, so wird, falls entsprechend konfiguriert, im DNS nachgesehen, ob hier der entsprechende Namen zu finden ist. Falls dieses der Fall ist, wird die IP-Adresse ermittelt.

Um eine aktive Hosttabelle zu deaktivieren gibt es zwei Möglichkeiten:

1.) Löschen Sie die Datei HOSTS. Dieser Schritt ist jedoch für meinen Geschmack etwas zu grobschlächtig. Vielleicht sind einige Server noch nicht Bestandteil des DNS-Dienstes. Sollten Sie also Ihre Datei HOSTS löschen, so verlieren Sie vorerst die Verbindung zu den betreffenden Servern.

2.) Deaktivieren Sie in der Datei HOSTS den Eintrag für die betreffende Adresse.



IP-Adresse deaktivieren

Ich habe hier den Hosttabelleneintrag dadurch deaktiviert, dass ich vor den Eintrag das Zeichen # (Raute) eingetragen habe. Dieses Zeichen macht die gesamte Zeile zu einer Kommentarzeile.

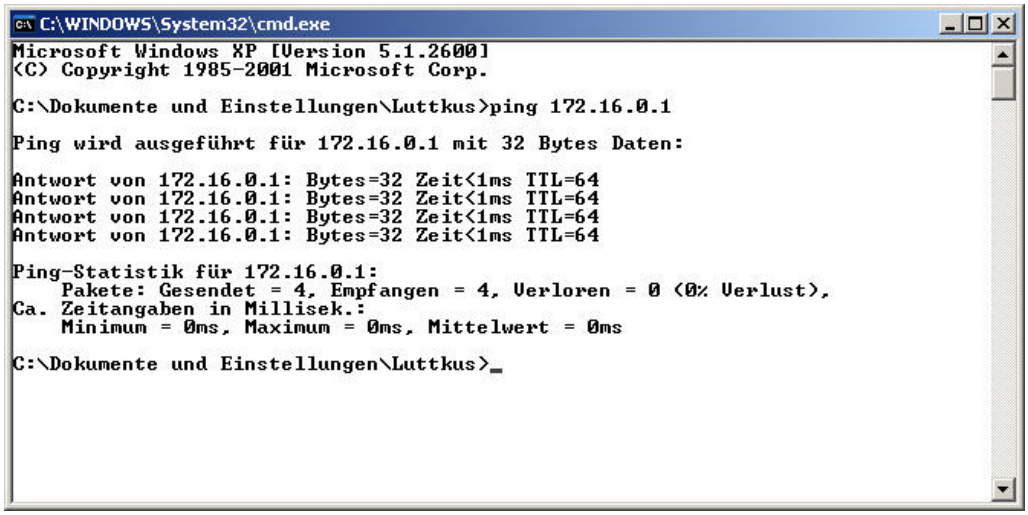
Sollten dann endlich alle Server in die DNS-Domäne übernommen worden sein, kann ich die Datei HOSTS immer noch löschen.

Nachdem die Konfiguration des Client beendet wurde, kommt jetzt der Moment der Wahrheit. Mit dem Befehl PING wird jetzt die Verbindung zur iSeries geprüft.

PING: mit qualifizierten Hostnamen

ping eposas01.eposgmbh.de

7.6.6.3
Seite 6



Erfolgreicher PING

Sie sehen das Ergebnis: Die iSeries antwortet, somit war unsere DNS-Konfiguration erfolgreich.

Das Dynamic Host Configuration Protocol (DHCP)

Seit OS/400 V4R2 ist der DHCP-Dienst Bestandteil der iSeries TCP/IP- Umgebung. Das Dynamic Host Configuration Protocol dient zum automatischen Konfigurieren von Workstations in einem TCP/IP-Netzwerk. D.h., dass nicht für jeden Host im TCP/IP-Netzwerk händisch eine IP-Adresse zu vergeben ist, also eine statische Adresszuordnung zu erfolgen hat, sondern das die Adressdaten von einem Serverdienst zur Verfügung gestellt werden. Die für die Adresszuordnung erforderlichen Daten sind in einem DHCP-Server abgelegt. Bei der Konfiguration des DHCP-Server wird ein Adresspool angelegt, aus dem sich die DHCP-Clients jeweils ihre Adressen herausziehen. Dieser Adresspool wird als Teilnetz vom iSeries DHCP bezeichnet. Verwechseln Sie den Begriff des DHCP-Teilnetz nicht mit dem Begriff des TCP/IP-Subnetzes (übersetzt: Teilnetz). Das DCHP-Teilnetz beschreibt lediglich die Menge aller Adressen, die sich in einem DHCP-Adresspool befinden. Grundsätzlich kann die Menge aller Adressen im Pool übereinstimmen mit der Menge aller Adressen in einem TCP/IP-Subnetz, jedoch ist dieses nicht zwingend erforderlich, bisweilen sogar nicht einmal wünschenswert.

Es gibt im Rahmen des DHCP zwei Verfahren, die die Adresszuordnung für einen DHCP-Client steuern:

1.) Dynamisches DHCP: Wird dieses Verfahren angewendet, so kann grundsätzlich ein beliebiger DHCP-Client auf die Dienste des DHCP-Server zugreifen. Soweit mit DHCP IP-Adressen dynamisch zugewiesen werden ist es kaum noch möglich, die Aktivitäten bestimmter Anwender oder Gruppen im Netzwerk gezielt zu konditionieren und zu überwachen. Außerdem ist es nicht mehr möglich, die Adressauflösung per DNS (DNS = Domain Name Service) durchzuführen. Mit OS/400 V5R1 ist dieses dynamische DHCP dann dahingehend erweitert worden, dass DHCP und DNS miteinander kommunizieren.

2.) Statisches DHCP: Wird dieses Verfahren angewendet, so muss von jedem Client, für den eine Adresse zur Verfügung gestellt werden soll, die MAC-Adresse – die auf der Netzwerkkarte befindliche Adapteradresse, Ethernet oder Token Ring – bekannt sein. Diese MAC-Adresse wird dann mit in den DHCP-Adresspool aufgenommen, so dass nur solchen Clients Adressen zugewiesen werden, deren MAC-Adressen im DHCP-Adresspool bekannt sind. Dadurch erhalten wir zumindest für die Überwachung von DHCP-Vorgängen eine größere Kontrolle.

4. Ergänzung 09/2005

7.6.7
Seite 2

Wichtig für iSeries DHCP: DHCP ist der technische Nachfolger von BOOTP (BOOTP = BootstrapProtocol), DHCP kann als Erweiterung von BOOTP betrachtet werden. DHCP ersetzt BOOTP aber nicht.

Beide Protokolle gehen von einem Client/Server-Modell aus. Der Server ist dabei ein Host mit der Datenbank, die die Konfigurationsdaten enthält. BOOTP wurde auf der iSeries für den Betrieb des IBM Netcomputers (NC) implementiert, und ermöglichte die statische Zuweisung von IP-Adressen.

Achtung!

Beide Dienste können nicht parallel auf der iSeries ausgeführt werden. Betreiben Sie Adresszuweisungen mittels BOOTP, so müssen Sie gegebenenfalls komplett auf DHCP umstellen! Aber keine Angst, im Rahmen des iSeries-DHCP wird Ihnen die Möglichkeit gegeben, die so genannte BOOTP-Tabelle in ein DHCP-Teilnetz zu importieren.

Bei der Zuordnung einer Adresse zu einem Client kommt es zu einem so genannten Adress-Leasing. Das Adress-Leasing bezeichnet die „Lebensdauer" einer Adresse. Diese kann sowohl in Sekunden, Minuten, Stunden, Tagen und Wochen vergeben werden. Grundsätzlich ist es darüber hinaus möglich, eine immer währende Leasing-Dauer zu vergeben.

Wenn die Adresse dynamisch zugewiesen wird, spricht man vom so genannten Adress-Leasing. Darunter ist die zeitlich befristete Zuweisung der IP-Adresse zu verstehen. Die Zuweisung endet vorzeitig, wenn der Client vor dem Ablauf der Leasingdauer die Verbindung abbaut und die IP-Adresse wieder freigegeben wird. Dazu versendet der Client eine DHCPRELEASE-Nachricht. Danach ist die Adresse wieder frei verfügbar.

Clients haben die Möglichkeit, beliebig oft eine Verlängerung der

Verleihdauer anzufordern. Ein Server ist von sich aus nicht in der Lage, einem Client eine zugewiesene Adresse zu entziehen.

Wichtig!

Die iSeries kann ein DHCP-Server sein, jedoch kein DHCP-Client. **Ergo:** Die iSeries kann anderen DHCP-Clients Adressen zuordnen, aber nicht sich selbst!

Das Booten mit DHCP

Der Boot-Vorgang läuft bei DHCP in vier Phasen ab. In der ersten Phase verschickt der Client mit Hilfe eines Broadcast eine Anforderung für eine IP-Adresse. Diese DHCPDISCOVER-Nachricht enthält die physikalische Adresse (Token Ring- / Ethernet-Adresse aus dem ROM der Netzwerkkarte) des Client. Darüber hinaus kann sie eine Reihe von Vorschlägen für die IP-Adresse und die Verleihdauer enthalten. Befindet sich der DHCP-Server nicht im lokalen Netzwerksegment, wird die Anforderung von BOOTP Forwarding Agents in andere Segmente weitergeroutet.

Dieses Stadium der Kommunikation zwischen Client und Server ist noch eine etwas genauere Betrachtung wert. Sieht man einmal von Diskless Workstations ab, die in EPROMs oder batteriegepufferten RAMs entsprechende Konfigurationsparameter bereithalten können, so muss zunächst davon ausgegangen werden, dass der Client seine IP-Adresse noch nicht kennt. Dies hat zur Folge, dass der DHCP-Server dem Client in der IP-Schicht keine Nachrichten zustellen kann. Damit die Kommunikation zwischen Client und Server zustande kommen kann, wird vom Client ein generisches UDP/IP-Paket erzeugt und zum Port 67 des Servers im lokalen Netzwerk-Segment geschickt.

Wenn ein DHCP-Server die in das lokale Netzwerk-Segment gesendete Anforderung empfängt, schlägt er in seiner internen Datenbank mit den Konfigurationsdaten die für die empfangene Adresse festgelegte Konfiguration nach.

In der zweiten Phase des Boot-Vorgangs bekommt der Client bei der in der Regel verwendeten dynamischen Zuweisung die erste freie IP-Adresse aus der Datenbank angeboten. Jeder Server, der über eine freie IP-Adresse verfügt, kann dem Client mit einer DHCPOFFER-Nachricht antworten. Der Server ist dafür verantwortlich sicherzustellen, dass die IP-Adresse nicht schon anderweitig verwendet wird.

In der dritten Phase des Boot-Vorgangs empfängt der Client die DHCPOFFER-Nachricht von einem oder mehreren Servern. Dabei bleibt dem Client die Wahl, auf mehrere Antworten zu warten und dann eine bestimmte auszuwählen oder die zuerst eingetroffene zu verarbeiten. Windows Clients akzeptieren im Allgemeinen die erste Nachricht. In jedem Fall muss er nun eine DHCPREQUEST-Nachricht versenden, die die Server-ID enthalten muss. Die DHCPREQUEST-Nachricht wird wieder per Broadcast an den DHCP-Server versandt.

In der vierten Phase des Boot-Vorgangs antwortet der ausgewählte DHCP-Server auf die DHCPREQUEST-Nachricht des Client. Er verschickt dazu eine DHCPACK-Nachricht, die die Konfigurationsparameter für den anfordernden Client enthält. Damit kann der Client konfiguriert werden.

7.6.7.1
Seite 2

Falls der Server in dieser Phase die Anforderung des Client nicht bedienen kann, schickt er als Antwort eine DHCPNACK-Nachricht ab. Damit wird dem Client mitgeteilt, dass die Anforderung gescheitert ist und gegebenenfalls wiederholt werden muss.

Konfiguration des iSeries DHCP-Server

Mit diesem Grundwissen ausgestattet können wir jetzt an die Konfiguration unseres auf der iSeries befindlichen DHCP-Server herangehen. Bevor wir uns aber an die Arbeit machen, müssen wir unser Umfeld noch ein wenig beleuchten. Wie sieht unser Netzwerk aus?

Net- ID:	172.16.0.0
Bereits vergebene Adressen für iSeries Systeme	172.16.0.1 172.16.0.2 172.16.0.3
Internet-Router, das Default Gateway für alle Clients	172.16.100.100
Primärer DNS-Server Sekundärer DNS-Server	172.16.0.3 172.16.0.4
Subnetzmaske	255.255.0.0
Domänenname	Eposgmbh.de

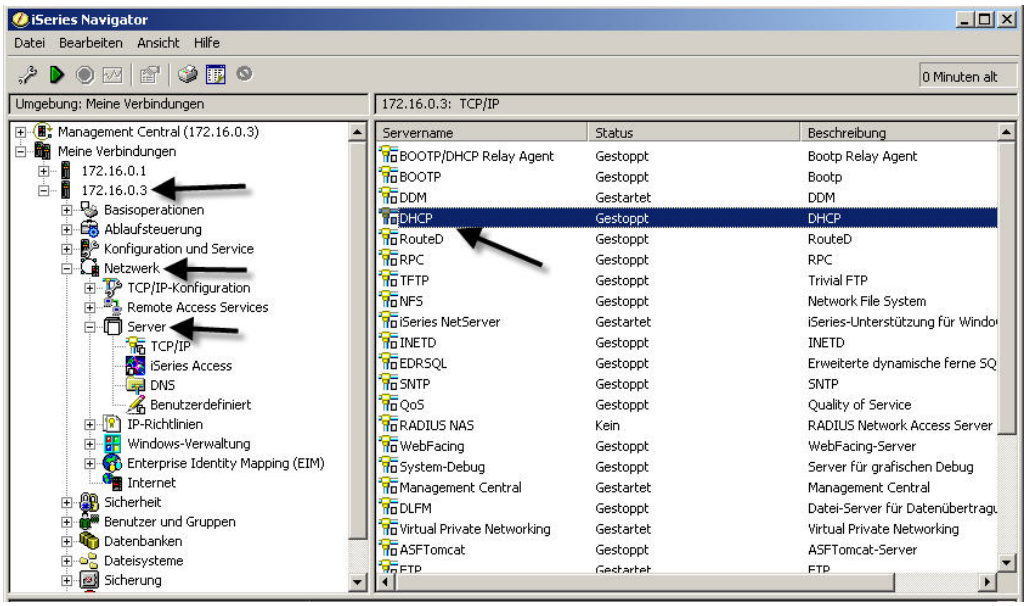
Die gesamte Konfiguration wird mit Hilfe des iSeries Navigator realisiert.

7.6.7.2
Seite 2

Aufruf der DHCP-Konfiguration

Ich starte also den iSeries Navigator und öffne folgenden Pfad im linken Listbereich des Operation-Navigator:

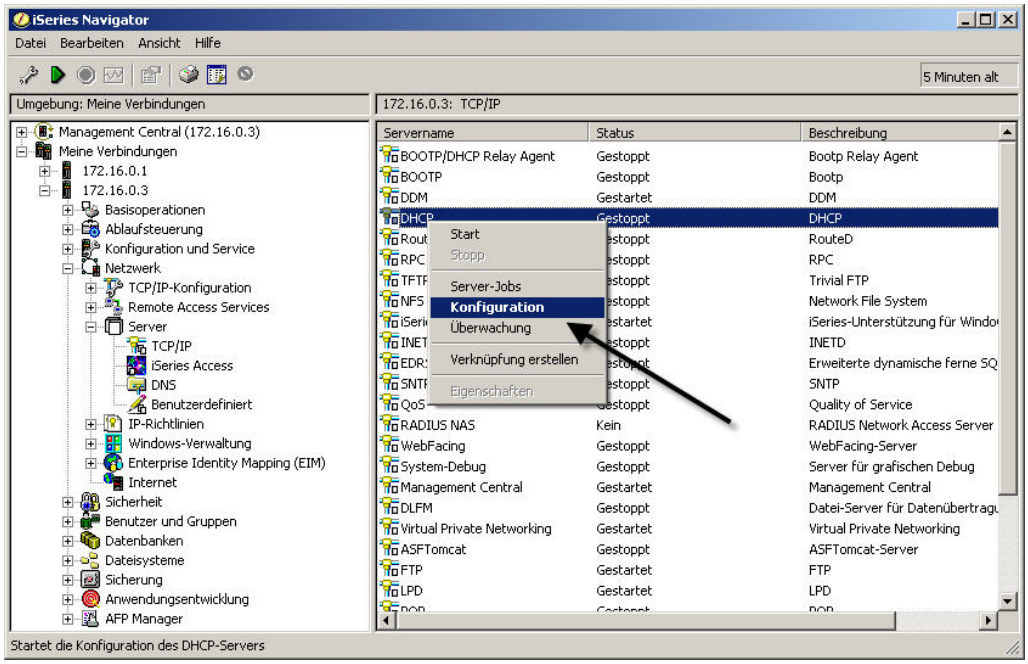
- Meine iSeries-Verbindungen
  - Netzwerk
  - Server
  - TCPIP



Der iSeries DHCP-Server

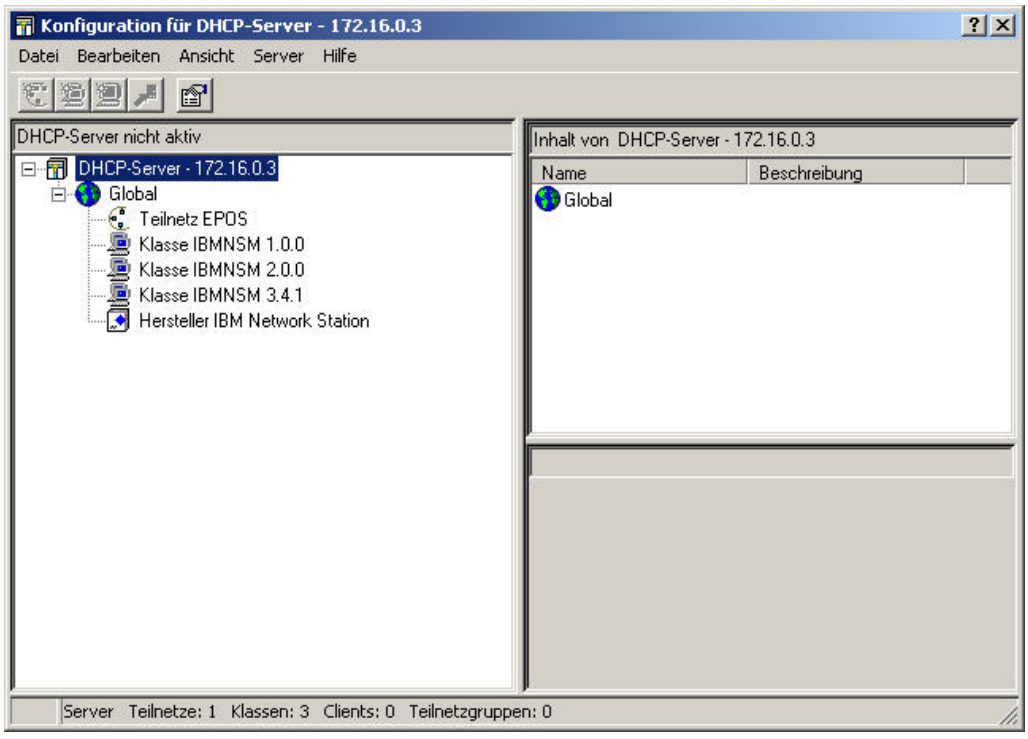
Auf der rechten Seite sehen Sie jetzt in der Höhe des Mauspfeils den derzeit noch gestoppten DHCP-Server. Damit der Server später seine Arbeit verrichten kann, muss er zuerst einmal konfiguriert werden. Ich markiere also den Eintrag DHCP und betätige die rechte Maustaste, und das angezeigte Kontextmenü öffnet sich.

7.6.7.2.1
Seite 2



Beginn der Konfiguration

Ich wähle die Menüoption „Konfiguration“. Wie der Dialog sich im Weiteren entwickelt hängt jetzt davon ab, ob Sie schon einmal diesen Konfigurationsdialog angefordert haben. Sofern das nicht der Fall ist, wird sich sofort der Konfigurationswizzard (Wizzard = Zauberer!) öffnen und seinen Begrüßungsdialog anzeigen. In dem hier verwendeten Beispiel wurde aber bereits ein DHCP-Server konfiguriert, so dass sich jetzt der DHCP-Manager öffnet.



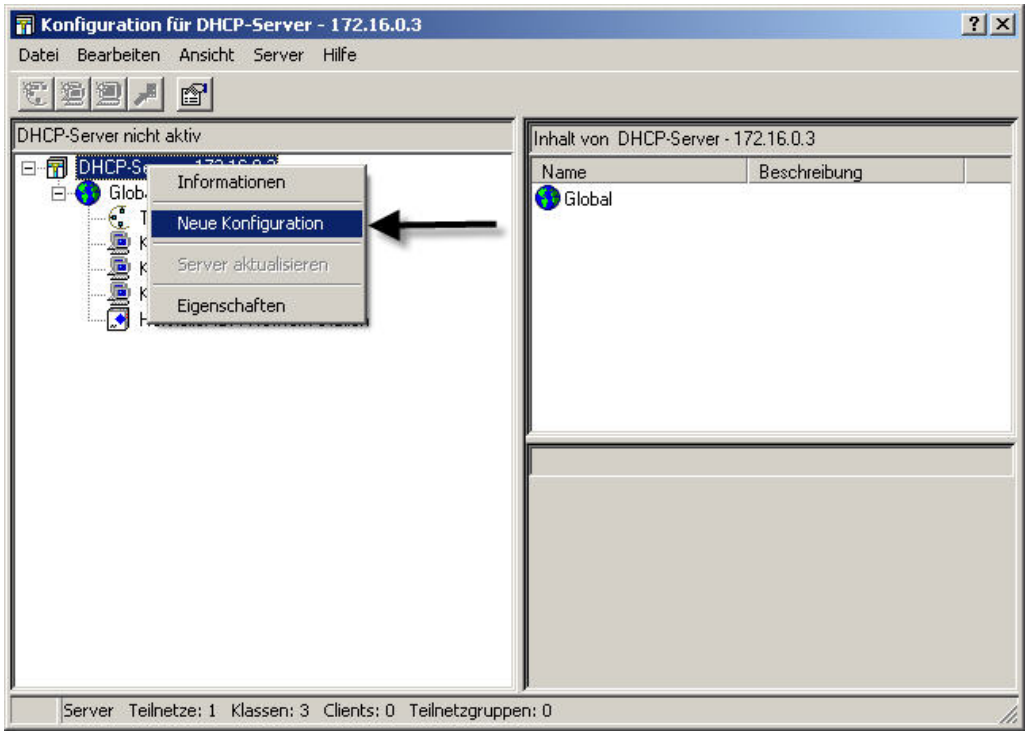
Der DHCP-Manager

Im linken Teil des Dialogs sehen Sie das gesamte DHCP-Netz. Es besteht aus folgenden Komponenten:

- DHCP-Server: sozusagen das Root-Verzeichnis des DHCP-Netzwerks
  - Global: Alle Komponeten des DHCP-Netzwerks.
    Dieses kann bestehen aus:
    - Teilnetzen: Adresspools
    - Klassen: Gerätespezifische Konfigurationen
    - Herstellerbezogenen Konfigurationen

Im oberen rechten Dialog des DHCP-Managers werden Detailkomponenten des globalen DHCP-Netzwerks angezeigt. Darunter im derzeit grauen Bereich, DHCP-Konfigurationsdaten für ein DHCP-Teilnetz. Da wir derzeit noch kein Teilnetz konfiguriert haben, sind die Detailanzeigen leer bzw. deaktiviert.

Lassen Sie uns also mit der Konfiguration eines DHCP-Teilnetzes beginnen. Ich markiere den Eintrag DHCP-Server – 172.16.0.3, drücke die rechte Maustaste und starte den Dialog „Neue Konfiguration“.

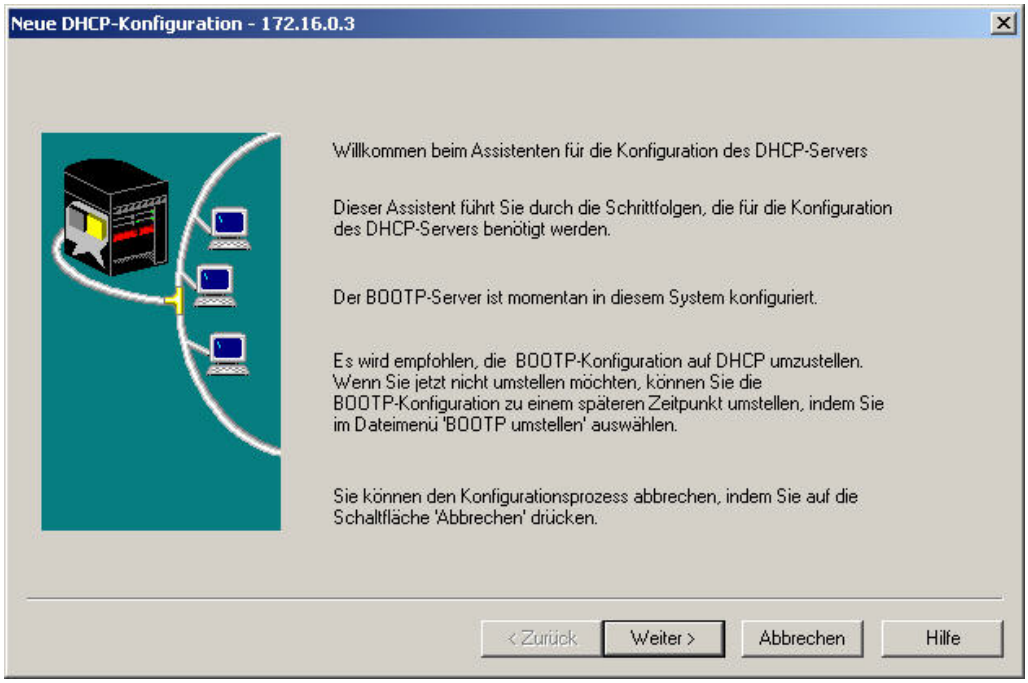


Die Konfiguration beginnt

Da bereits eine Konfiguration besteht werde ich gefragt, ob ich die bestehende Konfiguration wirklich löschen möchte. Diese Frage habe ich mit „OK“ beantwortet.

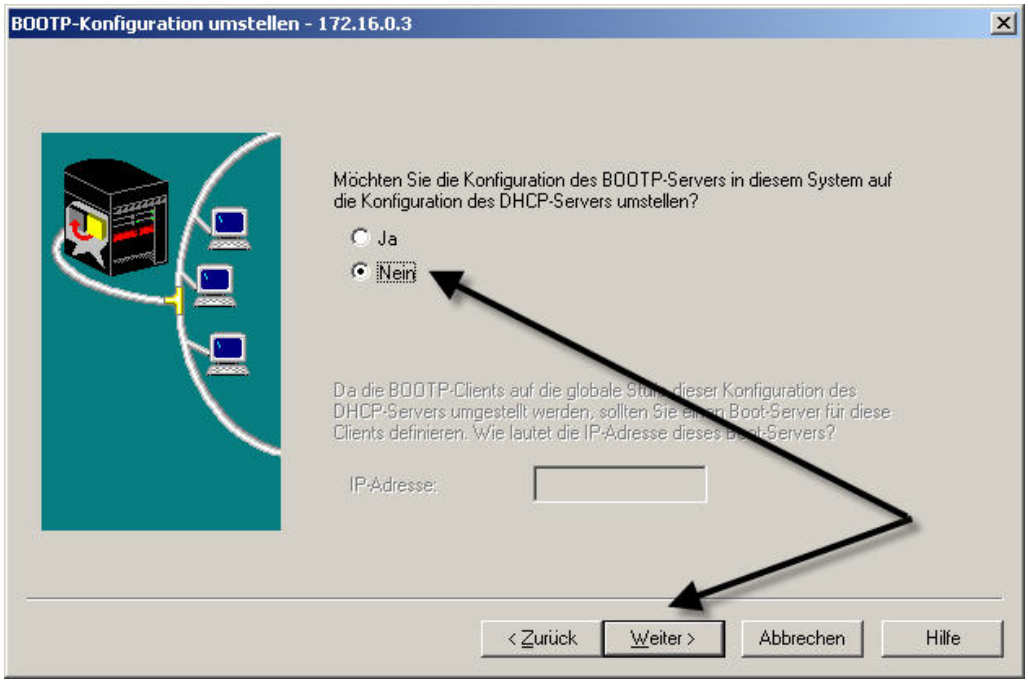
7.6.7.2.1
Seite 4

Danach wird der DHCP-Konfigurations-Wizzard gestartet. Wie ich oben schon sagte, wird der DHCP-Manager nur dann angezeigt, wenn es bereits eine bestehende Konfiguration gibt. Sollte das bei Ihnen nicht der Fall sein, so sehen Sie ohne den Zwischenschritt des DHCP-Managers den Willkommen-Dialog.



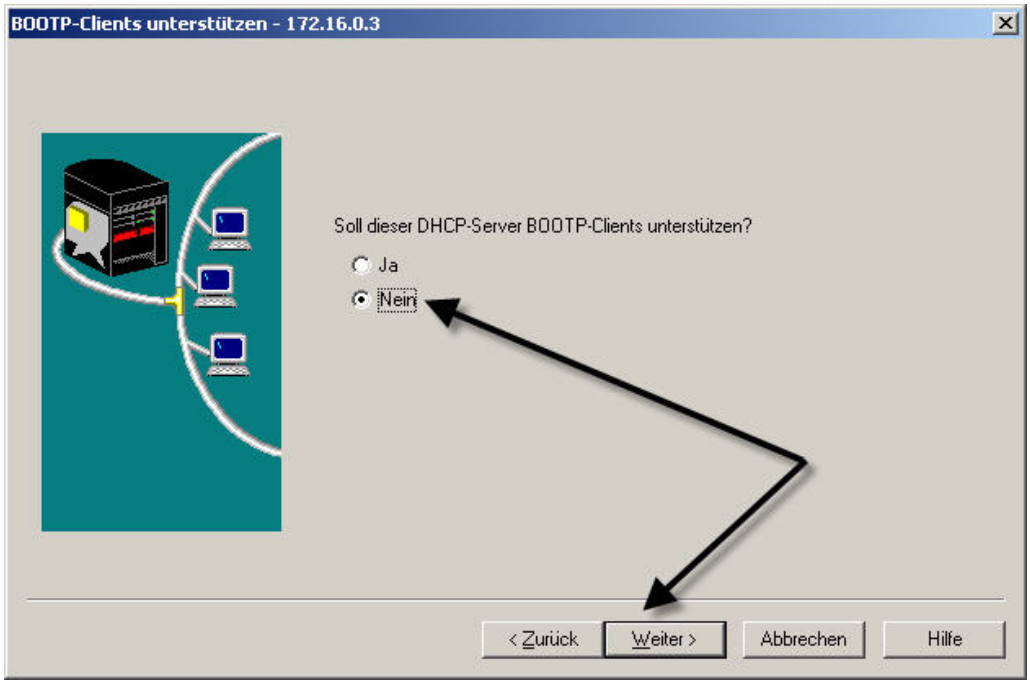
Der Willkommen-Dialog

In diesem Willkommen-Dialog sehen Sie in der Mitte die Mitteilung: „Der BOOTP-Server ist momentan in diesem System konfiguriert.“ usw. Diese Meldung ist typisch für ein System, auf dem bereits der BOOTP-Dienst in Betrieb war bzw. ist. Sollten Sie bisher nicht mit BOOTP gearbeitet haben, so entfällt dieser Abschnitt im Willkommen-Dialog und auch der folgende Dialog.



BOOTP-Konfiguration übernehmen

Im Dialog „BOOTP-Konfiguration umstellen“ besteht die Möglichkeit, die BOOTP-Konfiguration in den DHCP-Dienst zu importieren. Da auf meinem System der BOOTP-Dienst nicht gestartet ist und ich die BOOTP-Konfiguration nicht übernehmen will, deaktiviere ich den Umstellungsdialog, in dem ich den Radio-Button „Nein“ setze. Mit „Weiter“ komme ich dann in den nächsten Dialog.

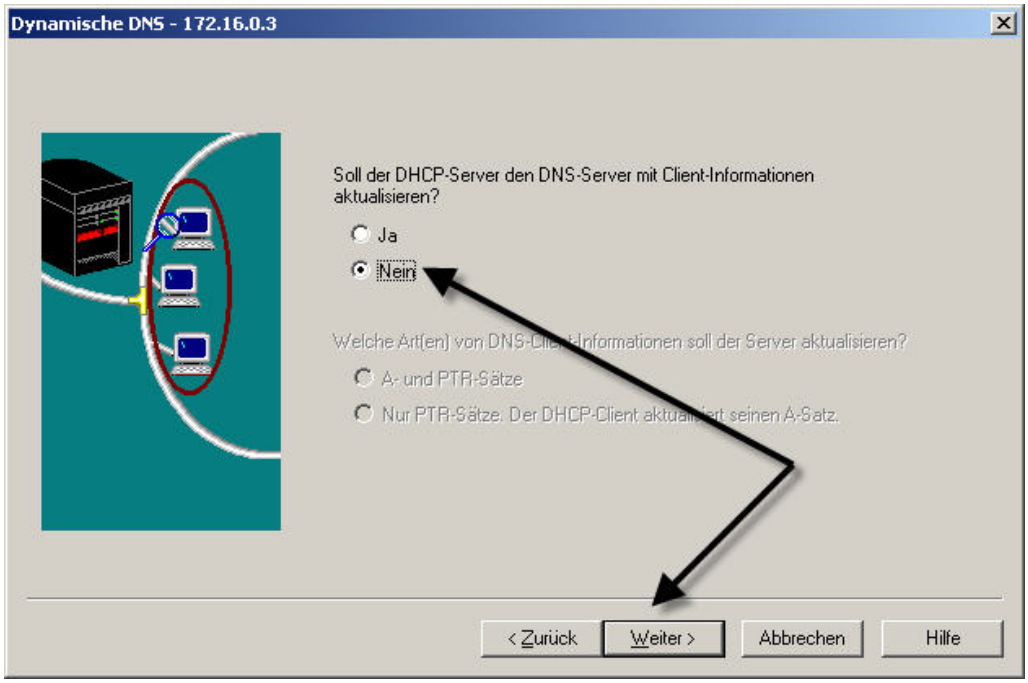


BOOTP-Clients unterstützen

7.6.7.2.1
Seite 6

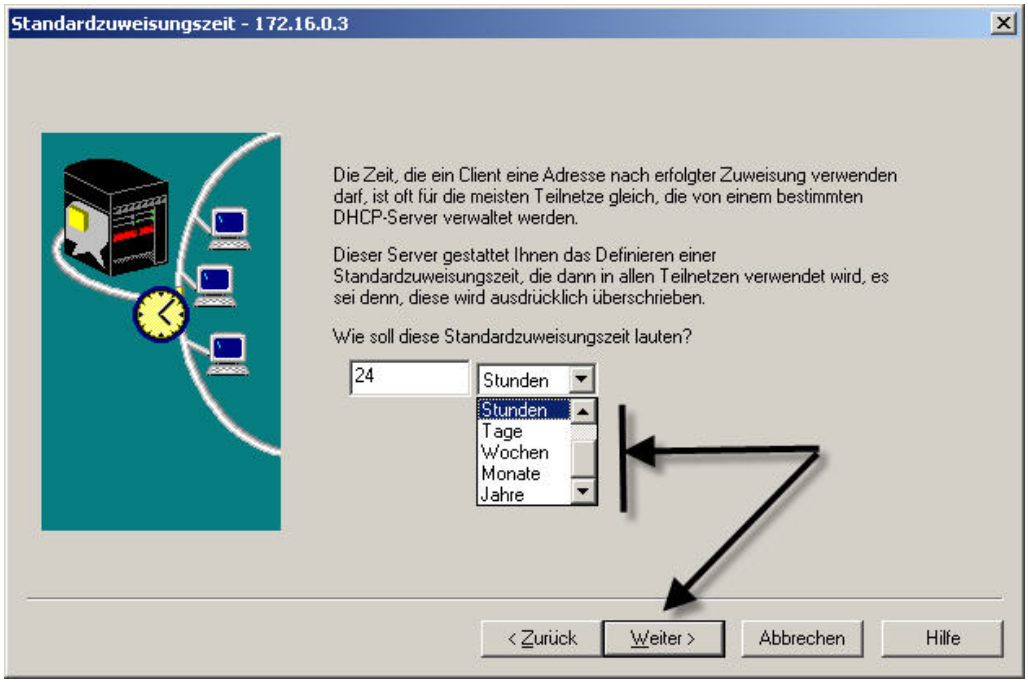
Da in meinem Netzwerk keine BOOTP-Clients unterstützt werden sollen, deaktiviere ich diese Funktion im Dialog „BOOTP-Clients unterstützen“ durch die Aktivierung des „Nein“-Radio-Buttons. Durch Drücken des Knopfes „Weiter“ komme ich in den Dialog „Standardzuweisungszeit“.

Im folgenden Dialog wir die Frage gestellt, ob der DHCP-Server mit einem dynamisch konfiguriertem DNS-Dienst kommunizieren soll.



Dynamic DNS?

Ich verneine diese Frage und beende den Dialog mit „Weiter“.



Standardzuweisungszeit

In diesem Dialog geht es darum, eine grundlegende Leasing-Dauer für den gesamten DHCP-Serverdienst zu konfigurieren. Derzeit steht in diesem Parameter ein Standardwert von 24 Stunden.

Lassen Sie uns hier ein wenig verharren. Es stellt sich jetzt hier die Frage, ob 24 Stunden eine gute oder eine nicht so gute Leasing-Dauer darstellt. Die Antwort ist, wie so oft in der EDV: „Es kommt darauf an."

Auf der einen Seite ist folgende Erkenntnis wichtig: Je kürzer die Leasing-Dauer für eine Adresse ist, desto mehr muss der Client um die Verlängerung der Leasing-Dauer bitten. Was wiederum bedeutet, dass durch die vermehrten Nachfragen das Netzwerk belastet wird. Wenn wir uns vorstellen, dass nicht nur ein oder zwei DHCP-Clients sich im Netzwerk befinden, so können wir uns sicherlich leicht ausrechnen, dass sich die Netzwerklast durch häufiges Nachfragen noch potenziert.

Auf der anderen Seite steht das Sicherheitsbedürfnis im Netzwerk: Je länger eine Adresse verleast wird, desto größer ist auch die Wahrscheinlichkeit, dass mittels dieser längerfristig vergebenen Adresse Angriffe auf das Netzwerk gestartet werden können.

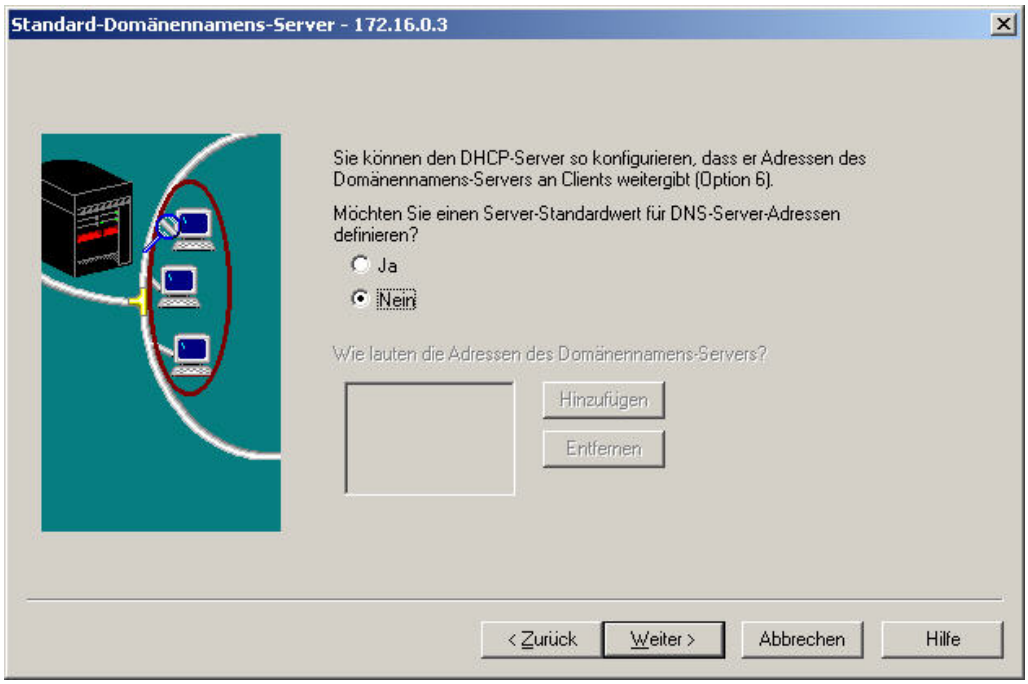
Hieraus folgt die Praxis, dass wir bei WAN-Verbindungen geneigt sind, mit relativ kurzen Leasing-Zeiten zu arbeiten – eine Leasing-Dauer von 60 bis 80 Sekunden ist hier nicht ungewöhnlich. Damit geht hier die Wahrscheinlichkeit gegen Null, dass ein remoter Client bei zwei Anmeldungen im Netzwerk die gleiche IP-Adresse zugewiesen bekommt.

7.6.7.2.1
Seite 8

Bei LAN-Verbindungen arbeiten wir hingegen mit längeren Leasing-Zeiten. Hier sind Werte im Tagesbereich nicht unüblich, wie z.B. 3 Tage. Hier ist die Wahrscheinlichkeit groß, dass ein Client jeden Tag aus dem Netz abgemeldet wird, so dass er bei jeder erneuten Anmeldung eine neue Adresse erhält. Bleiben Clients längere Zeit im Netz, so verursachen sie nur geringen DHCP-Traffic.

Ich belasse den Wert auf seinem Standard von 24 Stunden. Auf der Teilnetzebene wird mir darüber hinaus die Möglichkeit gegeben, diesen Wert zu überschreiben.

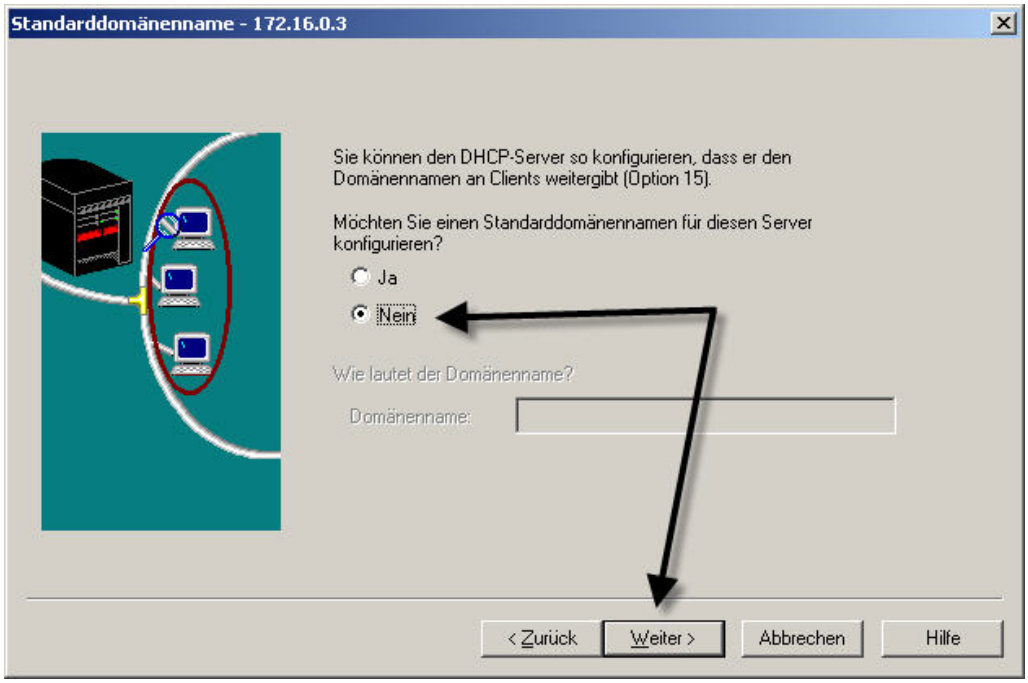
Der jetzt folgende Dialog bezieht sich wieder auf die Kommunikation zwischen DHCP und dynamischem DNS.



Adresse des dynamischen DNS-Dienstes

Da ich derzeit keinen dynamischen DNS betreibe, belasse ich es hier bei den Standard-Einstellungen.

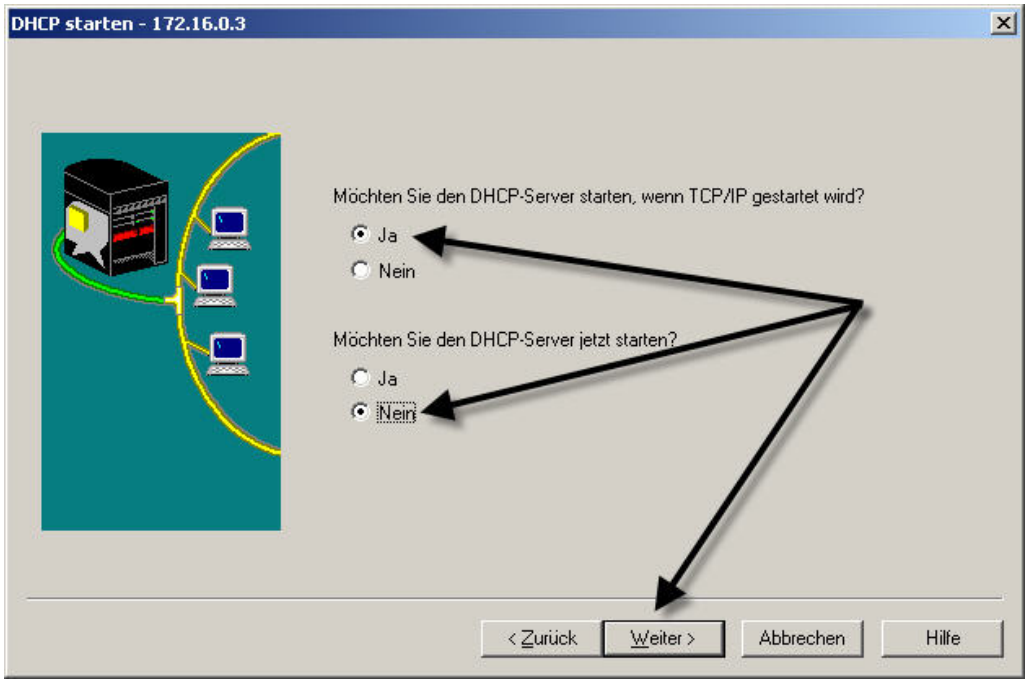
Im dem nächsten Fenster wird die Möglichkeit gegeben, dass der DHCP-Server insgesamt den Domänennamen an Clients weitergeben kann.



Weitergabe des TCP/IP-Domänennamens

Ich deaktiviere diese Möglichkeit auf Server-Ebene.

Im daraufhin folgenden Dialog wird die Möglichkeit gegeben, den Server zu starten.



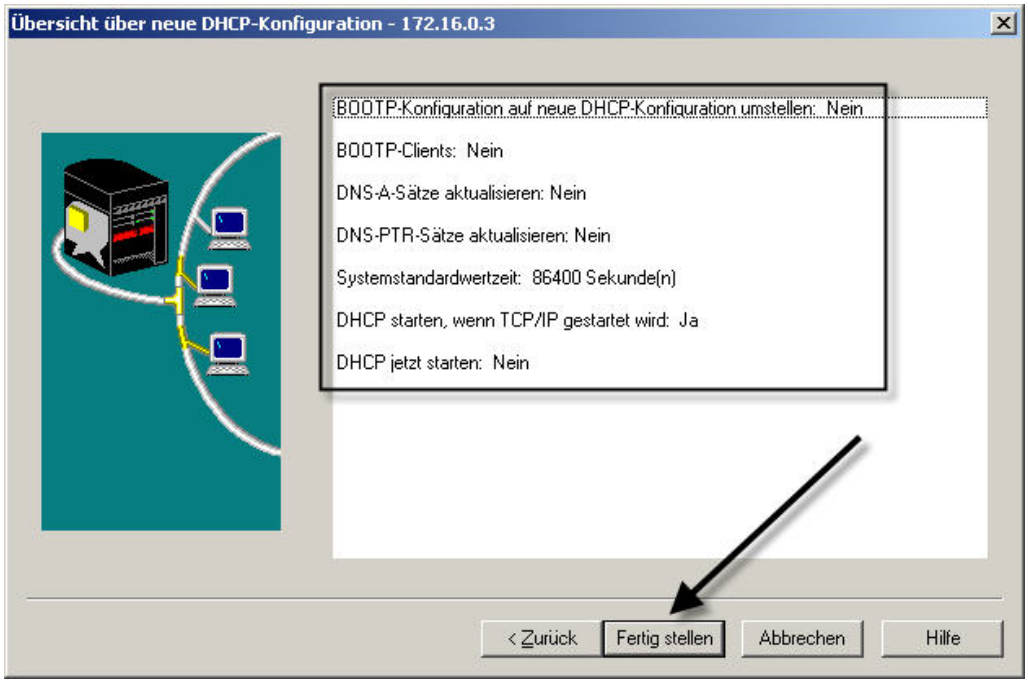
DHCP-Server jetzt starten?

7.6.7.2.1
Seite 10

Und zwar in zweifacher Hinsicht:

1. Soll der DHCP-Server bei jedem IPL der iSeries gestartet werden? Diese Frage beantworte ich mit „Ja“.
2. Soll der DHCP-Server jetzt gestartet werden? Diese Frage beantworte ich mit „Nein“, da ich für unsere Aufgabe bisher keine sinnvollen Daten – Adressbereiche für das zu verwaltende Subnetz – erzeugt habe.

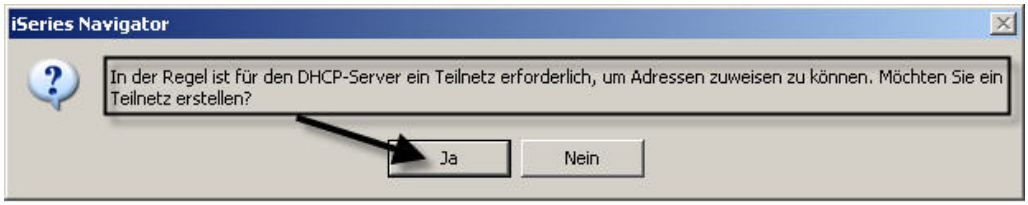
Danach erscheint eine Übersicht der bisher eingegebenen Daten.



DHCP-Server: Konfigurationsübersicht

Ich beende die Konfiguration mit „Fertig stellen“.

Da die bisherige Konfiguration soweit keinen Sinn macht, wird die Frage gestellt, ein DHCP-Teilnetz zu erstellen.



Teilnetz erstellen?

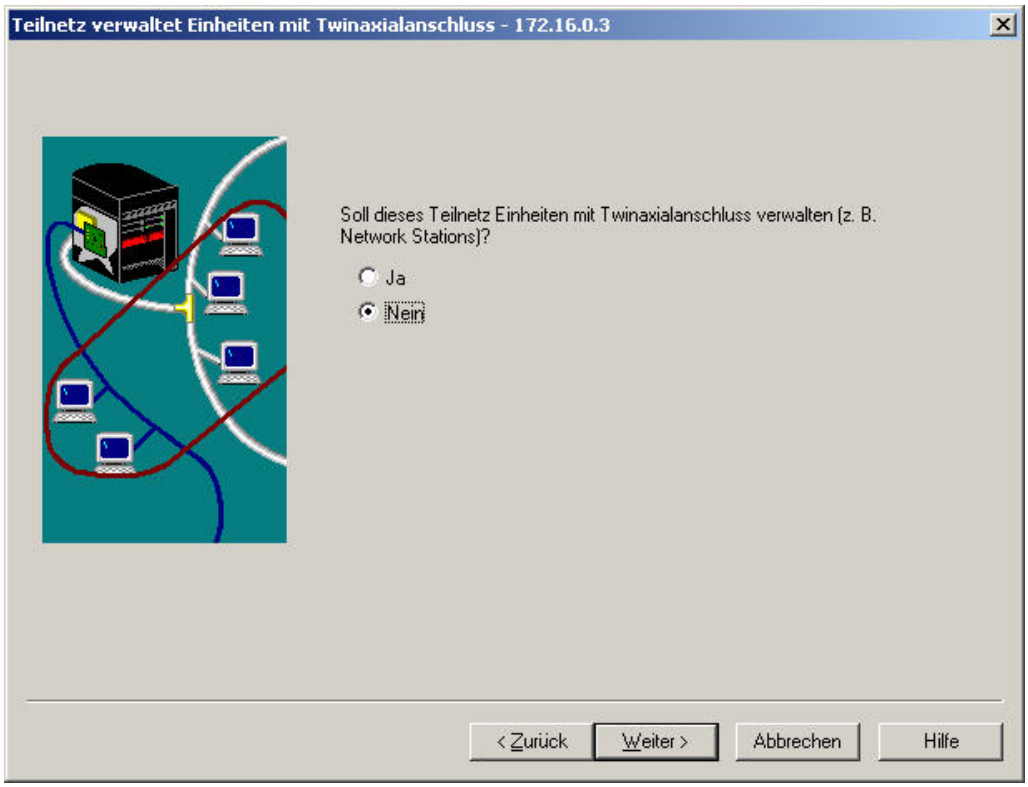
Mit dem Knopf „Ja“ geht es weiter in den Dialog „Neues Teilnetz erstellen“.

Wie in vielen Konfigurationsdialogen beginnt auch dieser mit einem Willkommen-Dialog.

Willkommen!

7.6.7.2.1
Seite 12

Das Fenster klicke ich einfach weiter und folgende interessante Frage wird gestellt:

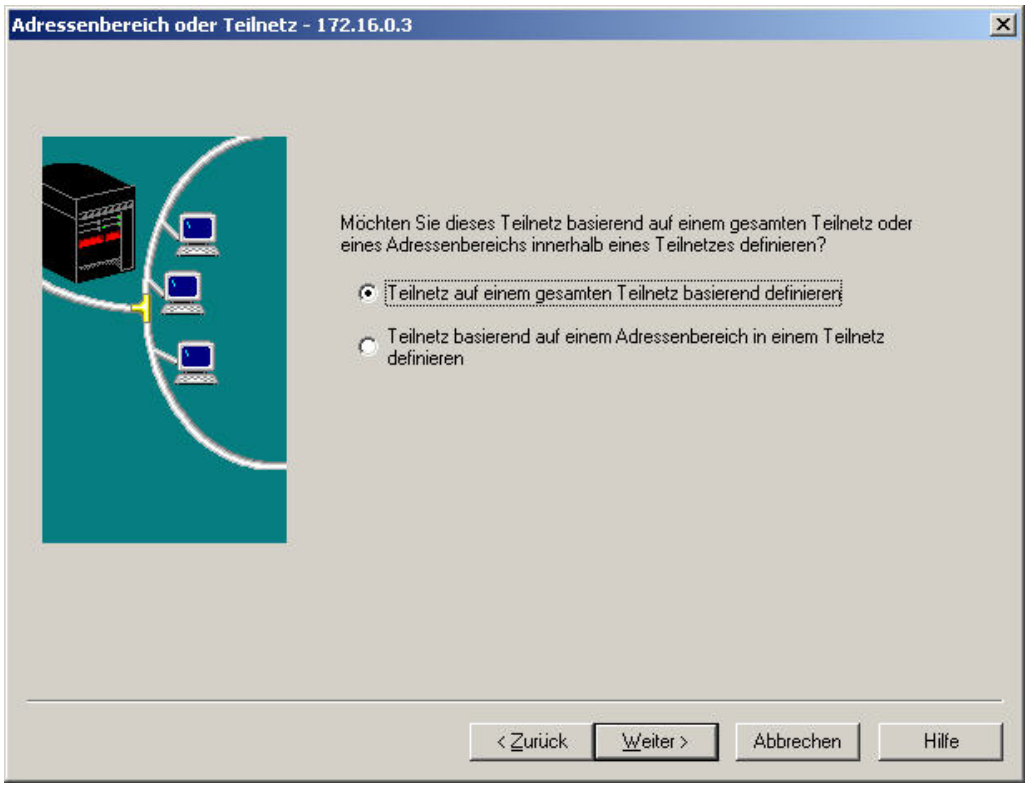


DHCP und Twinax?

„Soll dieses Teilnetz Einheiten mit Twinaxialanschluss verwalten?“

Machen Sie sich an dieser Stelle keine Hoffnung, dass Sie auf diesem Wege Ihre alten 3488-, 3477-, 3196/97-Bildschirme in das TCP/IP-Netzwerk integrieren könnten. Diese Option hebt ausschließlich auf den Umstand ab, dass bestimmte Hardwarekomponenten das so genannte TCP/IP over Twinax beherrschen. Dazu gehören aber leider nicht die alten Twinax-Terminals, sondern nur IBM-NCs (oder NCs anderer Hersteller) und PCs mit Twinax-Emulationskarten, die diese Technik unterstützen. Da es in meinem Netzwerk eine derartige Technik nicht gibt, deaktiviere ich diese Funktion.

Der nächste Dialog wird dann für die Konfiguration des Teilnetzes wirklich wichtig. Hier wird folgende Frage gestellt:



Was will uns dieser Dialog sagen?

„Möchten Sie dieses Teilnetz basierend auf einem gesamten Teilnetz oder eines Adressbereichs innerhalb eines Teilnetzes definieren?“ Was gibt es hierzu zu sagen? Erst einmal sieht man hier sehr schön die Ratlosigkeit des Übersetzers, der von Technikern vollkommen alleine gelassen wurde!

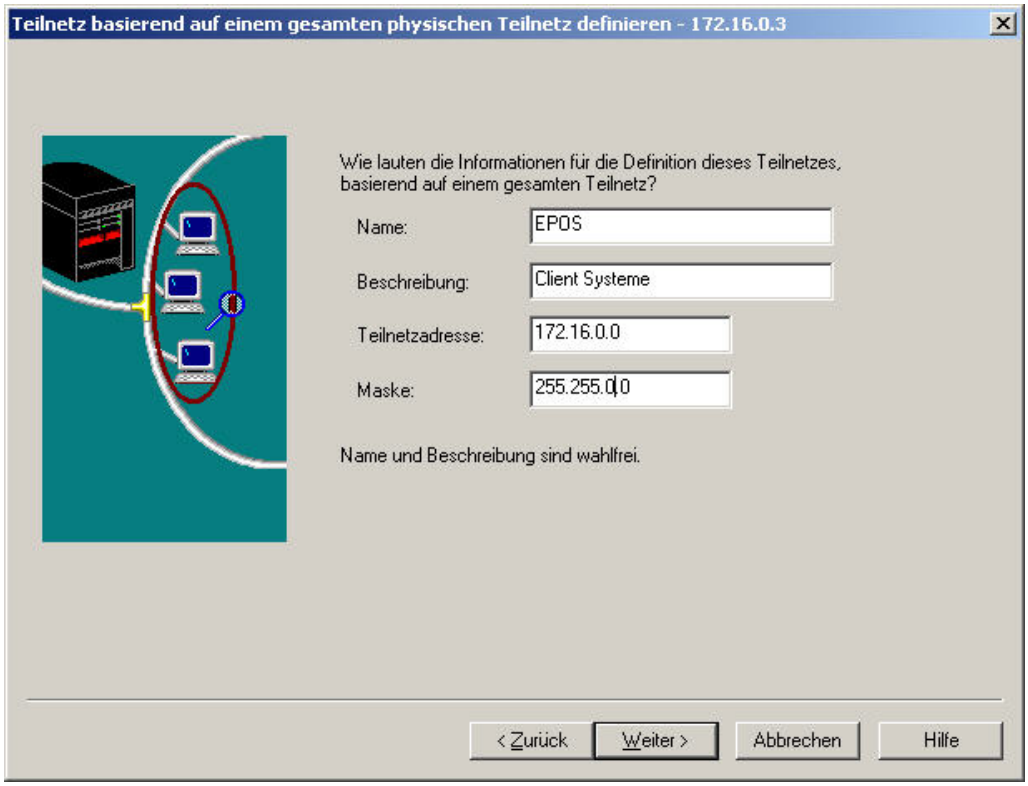
Welche Frage wird uns hier eigentlich gestellt? Ganz einfach!

Wollen wir für ein komplettes TCP/IP-Subnetz einen Adressenpool generieren? Wenn ja, so markieren wir den ersten Radio-Button.

Wollen wir jedoch einen Adressbereich, der durch eine Anfangsadresse und eine Endadresse gebildet wird, in den Adresspool aufnehmen, so markieren wir den zweiten Radio-Button.

7.6.7.2.1
Seite 14

Ich markiere zunächst den ersten Button, woraufhin sich folgender Dialog öffnet:



Adresspool für das Netz 172.16.0.0 definieren

Parameter

Name: Ein beliebiger Name für das DHCP-Teilnetz. Ich vergebe hier den Namen EPOS.

Beschreibung: Eine beliebige Zeichenfolge als Erläuterung.

Teilnetzadresse: Hier ist lediglich der Net-ID-Anteil der TCP/IP-Adresse einzutragen. Für mein Class-B-Netzwerk mit der Net-ID 172.16 trage ich hier den Wert 172.16.0.0 ein. Die beiden Nullen im Host-ID-Bereich der Adresse sind nur Platzhalter und können auch weggelassen werden.

Maske: Hier trage ich die Subnetzmaske ein, die für alle im Adresspool befindlichen Adressen mit an die Clients übergeben werden soll.

Eigentlich bin ich fertig und könnte weitergehen. Halt! Gefahr droht! Dieser Wert steuert außerdem, wie viele Adressen in den Adresspool aufgenommen werden. Da ich hier ein reines Class-B-Netzwerk habe, würden 65535 Adressen generiert werden. An sich eigentlich nichts Tragisches. Bauen wir jedoch einen derart großen Adresspool auf, so dankt uns das der DHCP-Server der iSeries mit einem Absturz! Würden wir uns den Joblog für den DHCP-Server ansehen, so würden wir eine Nachricht lesen können, in der steht, dass der DHCP-Server aufgrund von Speichermangel abnormal beendet wurde. D.h., dass wir unwissentlich die Maximalgröße des Adresspools überschritten haben.

Um diesen Fehler an dieser Stelle zu bereinigen, müssen wir also die Anzahl der Adressen, die in den Pool aufgenommen werden reduzieren. Dieses erreichen wir, indem wir die Subnetzmaske erweitern müssen.

Lasse ich die Teilnetzadresse unmodifiziert, also auf 172.16.0.0 und setze die Subnetzmaske auf 255.255.255.0 – wie viele Adressen werden dann in den Adresspool meines Teilnetzes aufgenommen? Genau 254 – nämlich die Adressen von 172.16.0.1 bis 172.16.0.254. Sollte Ihnen das geheimnisvoll vorkommen, so empfehle ich Ihnen die Abschnitte über Subnetze noch einmal zu lesen.

Gut, damit wäre unser Problem gelöst. Aber jeder DHCP-Client bekommt jetzt aus dem Adresspool eine Adresse folgender Struktur:

IP-Adresse z.B.: 172.16.0.5

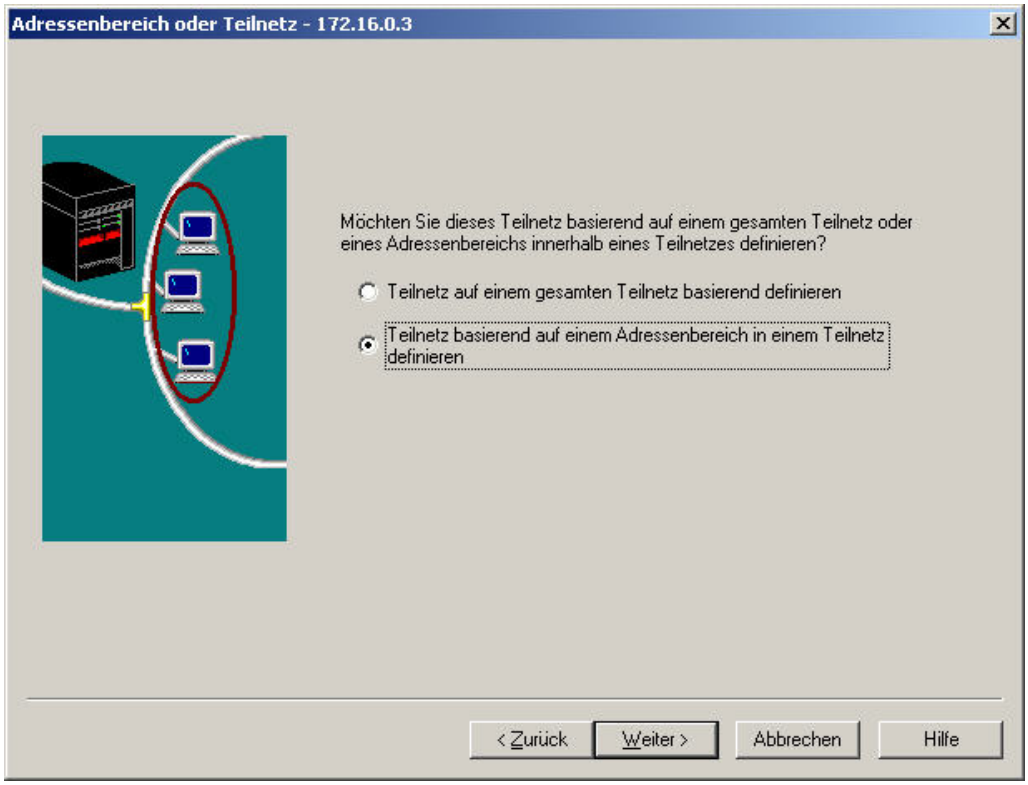
Subnetzmaske: 255.255.255.0

Damit können alle Clients, die Adressen vom DHCP-Server beziehen, untereinander kommunizieren, aber auf die iSeries mit der Adresse 172.16.0.1, Subnetzmaske 255.255.0.0 nicht.

Um nicht in diesem Problemumfeld unterzugehen, ändere ich meine Strategie hinsichtlich des Aufbaus eines Teilnetzes komplett und werde jetzt ein DHCP-Teilnetz dadurch definieren, dass ich eine Start- und Endadresse für mein Teilnetz angebe.

7.6.7.2.1
Seite 16

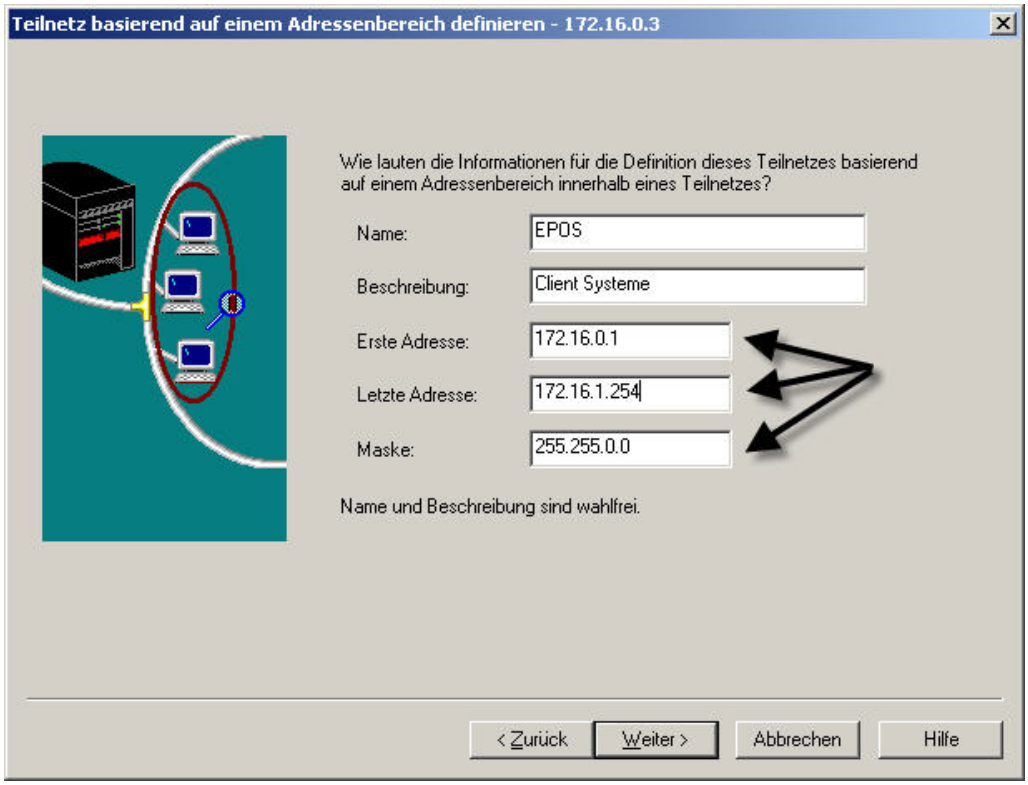
Ich verlasse also diesen Dialog mit „Zurück“ und komme wieder in den Dialog „Adressbereich oder Teilnetz“.



Adressbereich oder Teilnetz, das zweite Mal

Ich markiere hier die Auswahl „Teilnetz basierend auf einem Adressenbereich....“ und gehe dann weiter.

Ich sehe jetzt den Dialog „Teilnetz basierend auf einem Adressenbereich definieren“.



Einen Adressenbereich definieren

Parameter

Name: Hier trage ich den Namen EPOS ein. Dieser Name ist frei wählbar und erscheint im DHCP-Manager als Teilnetzname.

Beschreibung: Eine frei wählbare Zeichenfolge.

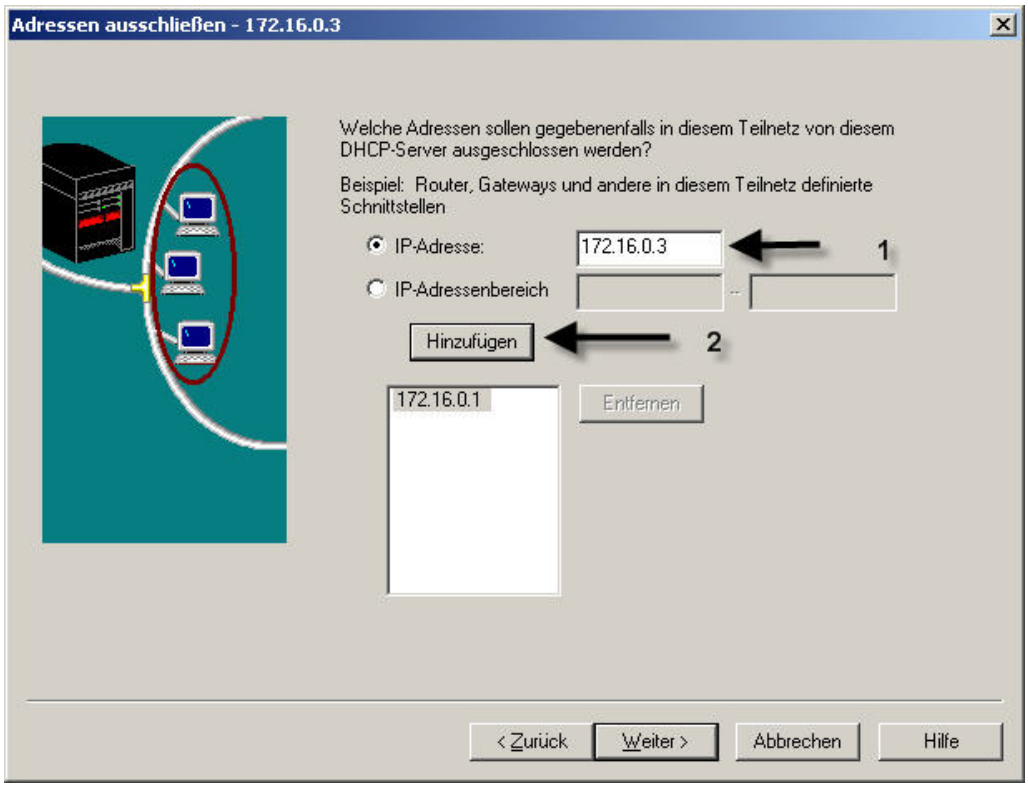
Erste Adresse: Die Anfangsadresse des Teilnetzes. Also die kleinste Adresse, die sich im Adresspool des Teilnetzes zuordnen lässt. Hier 172.16.0.1.

Letzte Adresse: Die Endadresse des Teilnetzes. Also die größte Adresse, die sich im Adresspool des Teilnetzes zuordnen lässt. Hier 172.16.1.254.

Maske: Die an die DHCP-Clients weitergereichte Subnetzmaske.

7.6.7.2.1
Seite 18

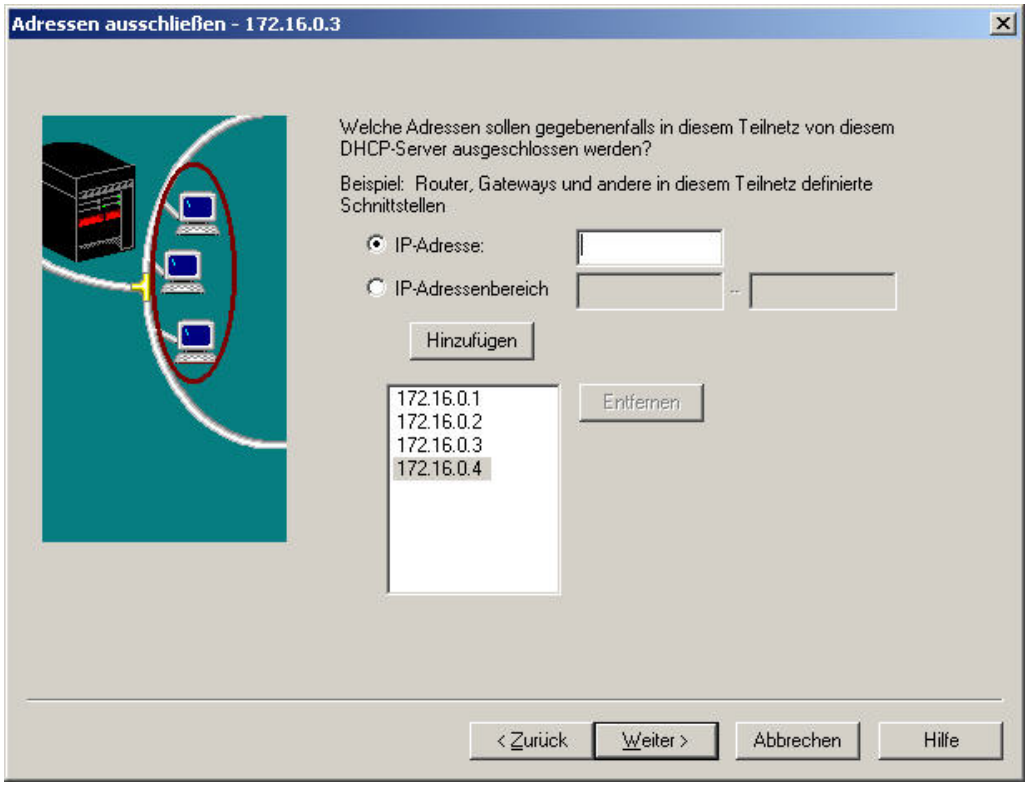
Im folgenden Dialog wird abgefragt, welche Adressen nicht im Adresspool des Teilnetzes vorgehalten werden sollen.



Adressen ausschließen (1)

Bedenken Sie, dass in meinem Netzwerk bereits mehrere Adressen statisch vergeben wurden. Werden diese Adressen in den Adresspool des Teilnetzes mit aufgenommen, so können diese bereits vergebenen Adressen erneut durch den DHCP-Server zugeordnet werden, was unweigerlich zum Adresskonflikt führt.

Ich drücke also im Dialog „Adressen ausschließen“ auf den Knopf „Hinzufügen“ und trage die Adresse 172.16.0.1 ein. Damit wird jetzt verhindert, dass diese Adresse in den Adresspool aufgenommen wird.

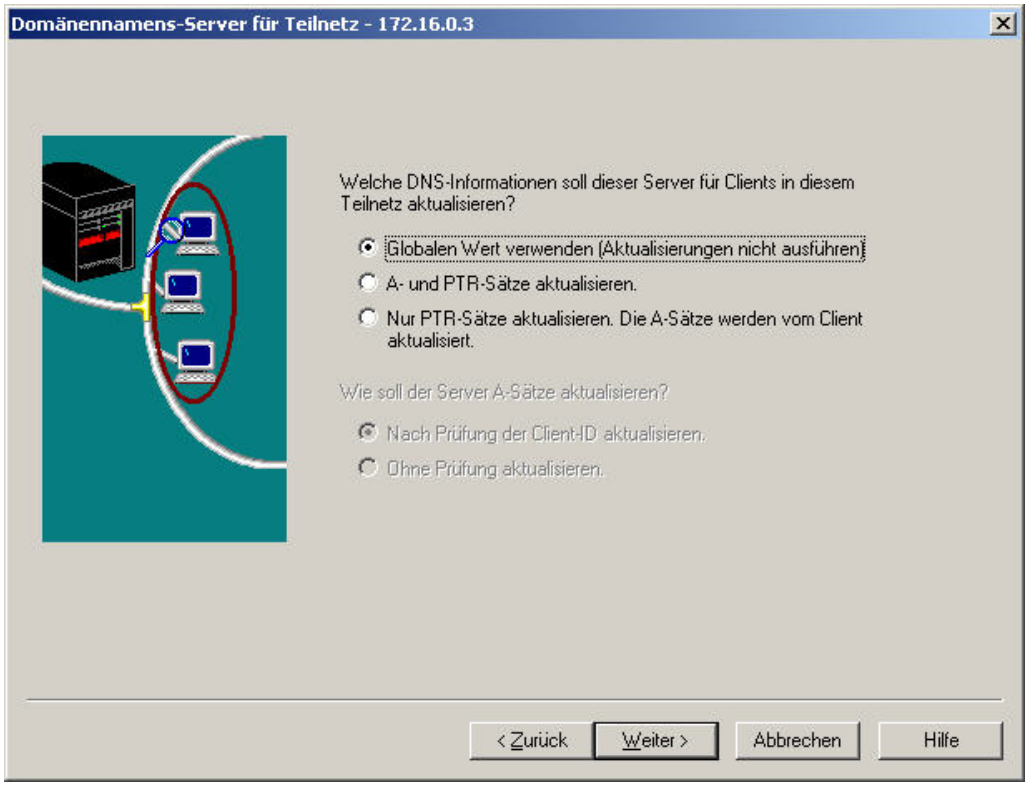


Adressen ausschließen (2)

Analog verfahre ich mit den Adressen 172.16.0.2, 172.16.0.3, 172.16.0.4, die alle Server-Funktionen im Netzwerk repräsentieren. Die Routeradresse 172.16.100.100 wird hier nicht berücksichtigt, da sie sich nicht im Adressbereich unseres Subnetzes befindet.

Der nächste Dialog bezieht sich wiederum auf die Kommunikation mit einem dynamischen DNS-Dienst.

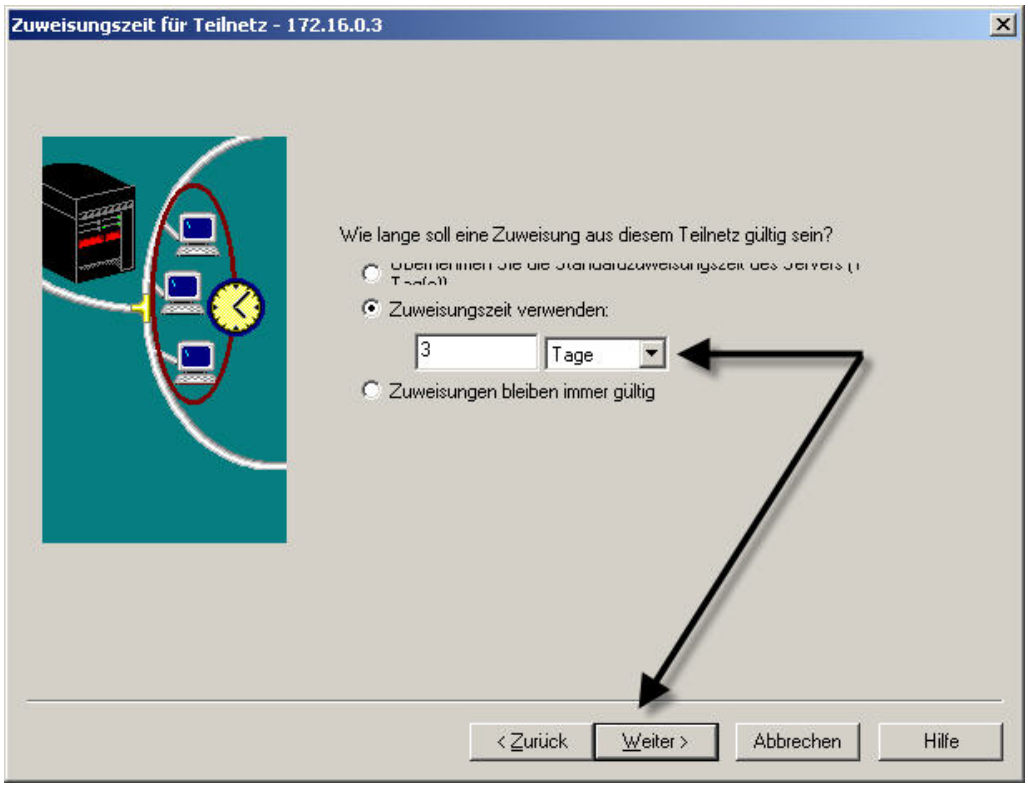
7.6.7.2.1
Seite 20



Aktualisierungen des dynamischen DNS steuern

Da ich mit statischem DNS arbeite, markiere ich den Wert „Globalen Wert verwenden (Aktualisierungen nicht durchführen)“ und setze mit „Weiter“ fort.

Im nächsten Dialog wird jetzt für dieses DHCP-Teilnetz die Standard Leasing-Zeit (Zuweisungszeit) des DHCP-Servers überschrieben.



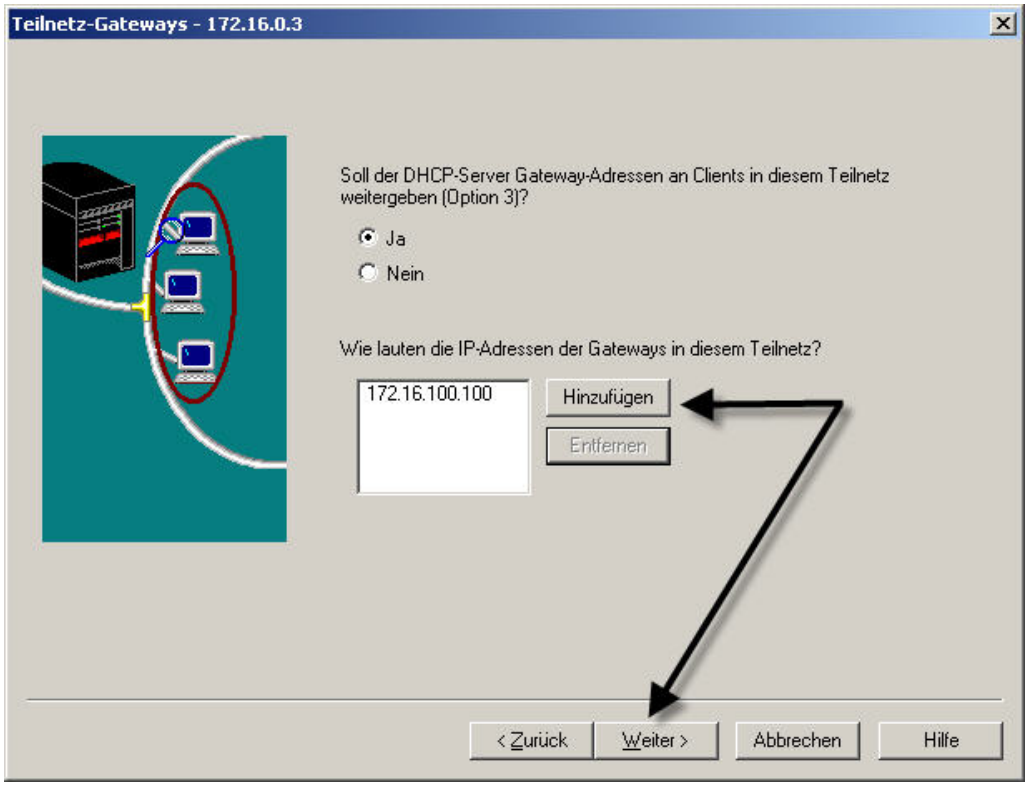
Zuweisungszeit für Teilnetz

Da ich davon ausgehe, dass dieses Teilnetz nur Clients im LAN bedient, verlängere ich die Leasing-Zeit nur für dieses Teilnetz auf 3 Tage. Dadurch erreiche ich, dass der Netzwerkverkehr insgesamt, der durch DHCP-Vorgänge bedingt ist, gering gehalten wird.

Da jeder Client meines Netzwerks Zugriff auf das Internet haben soll und diese Verbindungsaufnahme durch den Router mit der Adresse 172.16.100.100 gesteuert wird, muss ich dafür Sorge tragen, dass die lokale Adresse des Routers als Default Gateway in die lokale Netzwerkkonfiguration meines Windows PC aufgenommen wird.

Im Dialog „Teilnetz-Gateways“ sind daher folgende Werte einzutragen:

7.6.7.2.1
Seite 22

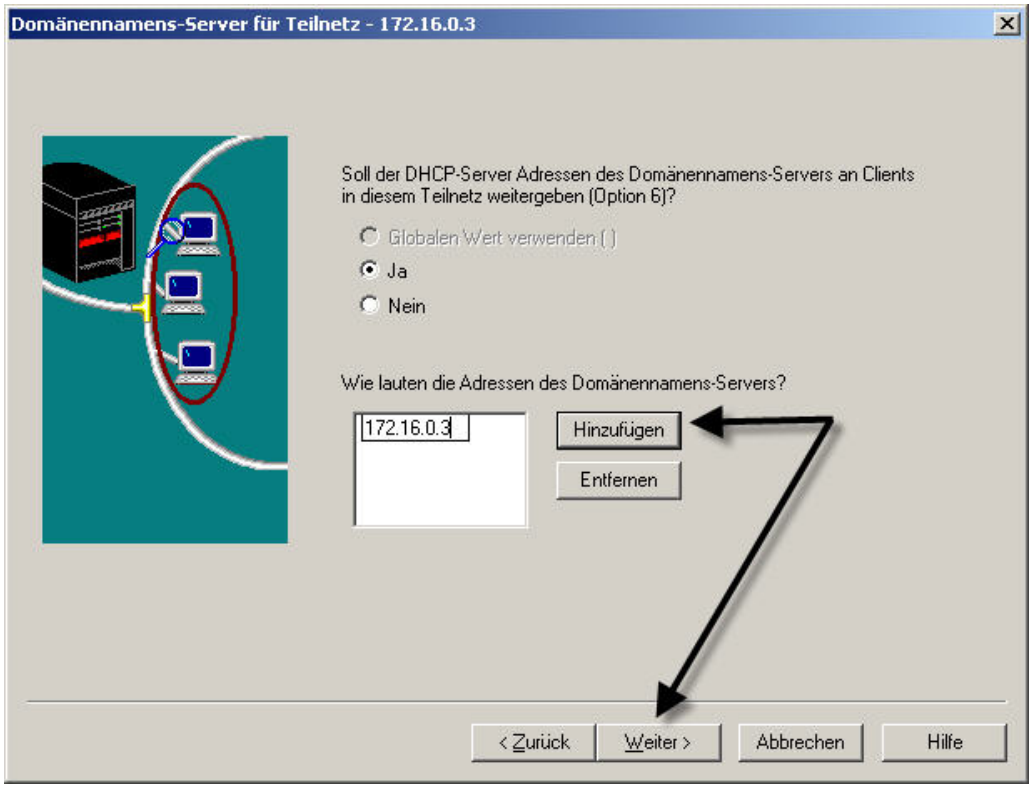


Teilnetz-Gateways

Zuerst einmal muss die Weitergabe von Gateway-Adressen generell aktiviert werden. Ich habe deshalb die Frage „Soll der DHCP-Server Gateway-Adressen an Clients in diesem Teilnetz weitergeben?“ mit dem Radio-Button „Ja“ markiert.

Mit Hilfe des „Hinzufügen“-Dialogs habe ich dann die Gateway-Adressen 172.16.100.100 eingetragen.

Da ich im lokalen Netzwerk einen eigenen DNS-Dienst bereitgestellt habe, sollen auch die DNS-Adressen an jeden Client weitergegeben werden.



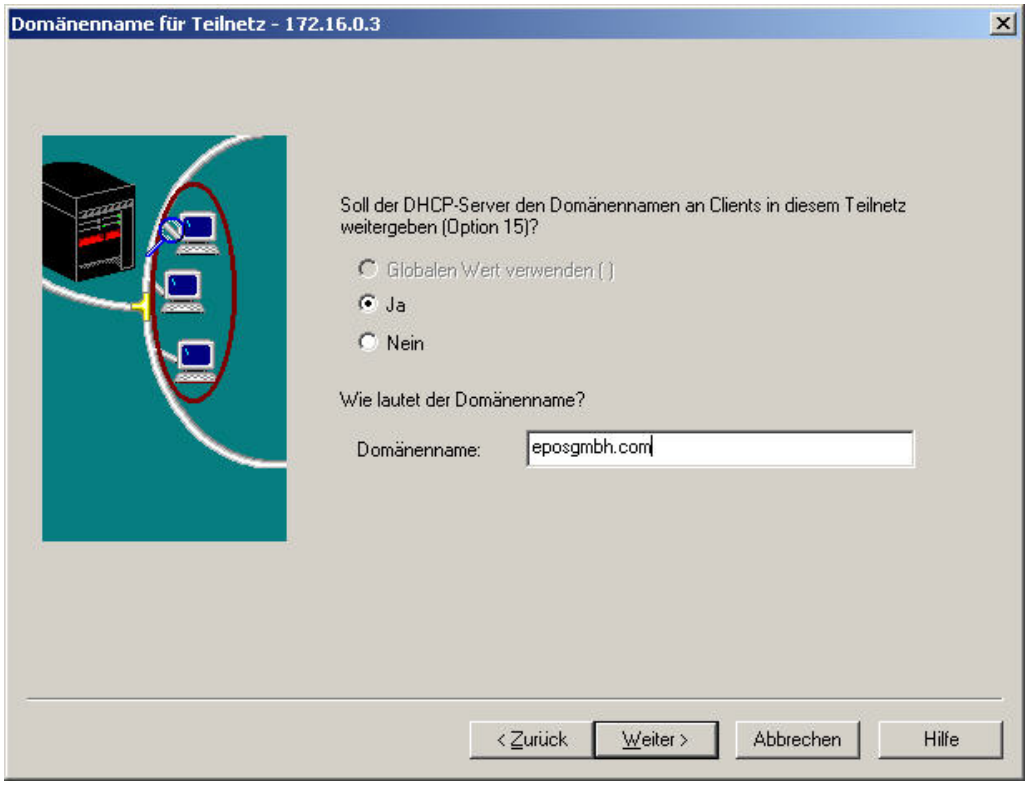
DNS für das Teilnetz

Deshalb aktiviere ich auch hier generell diese Funktion und trage die Adressen des primären DNS in das entsprechende Feld ein.

Weiterhin muss dann noch die Adresse des sekundären Domain Name Servers eingetragen werden.

7.6.7.2.1
Seite 24

Im folgenden Dialog beantrage ich dann, dass der Domänenname meiner TCP/IP-Domäne ebenfalls an die Clients übergeben wird.

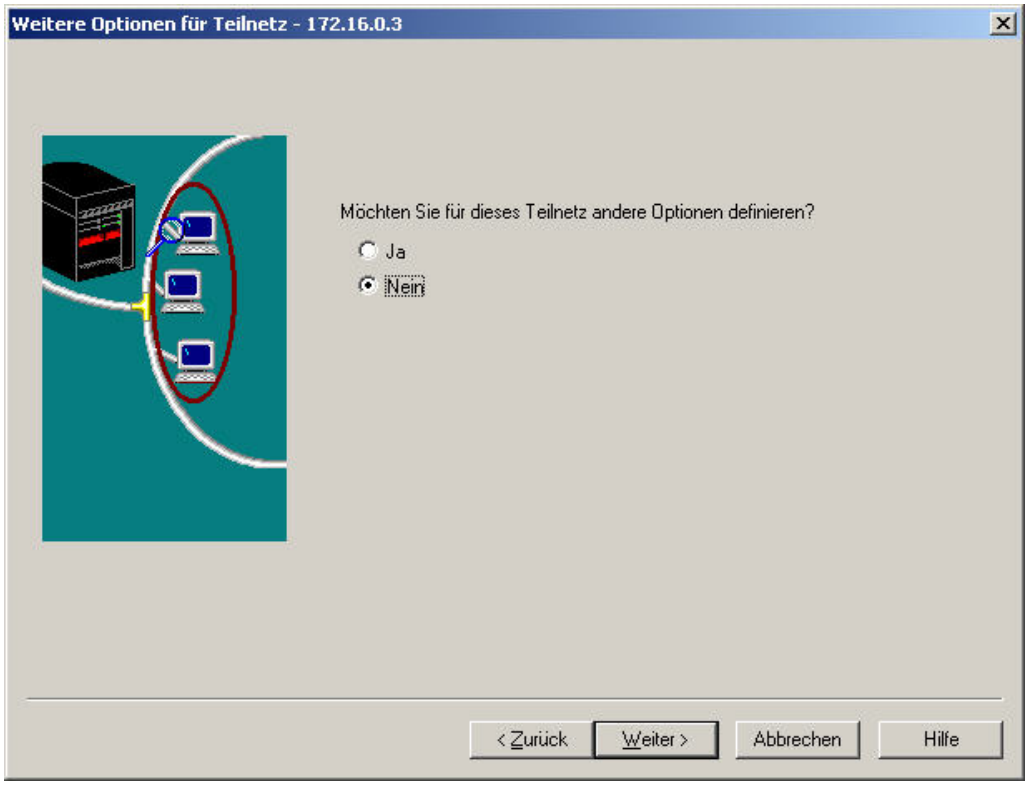


Domänenname für Teilnetz

Ich aktiviere diese Funktion und trage in das Feld „Domänenname“ den Wert EPOSGMBH.DE ein. Dieser Wert muss mit dem im DNS hinterlegten Domänennamen übereinstimmen.

Aktivieren Sie diese Funktion nicht, erleiden Sie keine schwerwiegenden funktionalen Einbußen. Haben Sie diese Funktion aktiviert, so trägt der DHCP-Server den Namen der Domäne in die Netzwerkumgebung des Client ein, so dass entfernte Systeme mit einem Kurznamen aufgerufen werden. Ist im DNS z. B. die Adresse 172.16.0.1 mit dem Hostnamen EPOSAS01.EPOSGMBH.DE hinterlegt worden, so können Sie von dem betreffenden Client ebenfalls mit dem Kurznamen EPOSAS01 auf die Adresse 172.16.0.1 zugreifen. Der fehlende Domänennamen wird dann durch den Client einfach ergänzt.

Im nächsten Dialog können dann noch weitere zusätzliche Funktionen generell abgefragt werden.

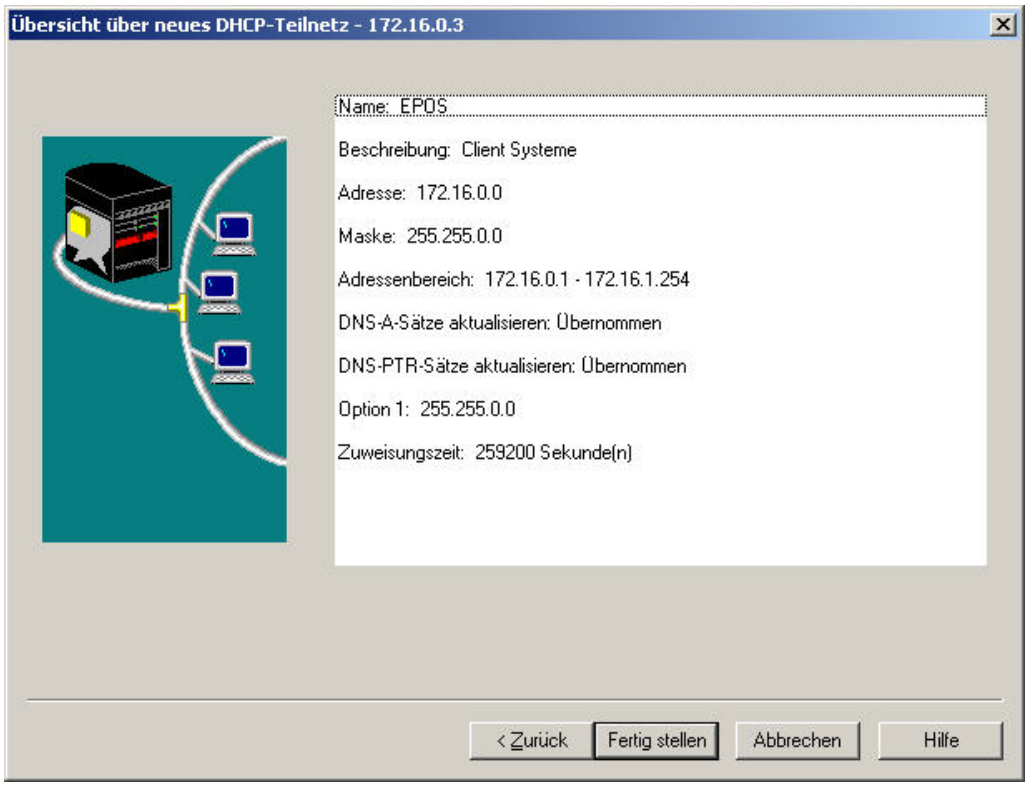


Weitere Optionen für Teilnetz

Ich verzichte auf diese Möglichkeit. Mit dem „Weiter"-Knopf beende ich diesen Dialog.

7.6.7.2.1
Seite 26

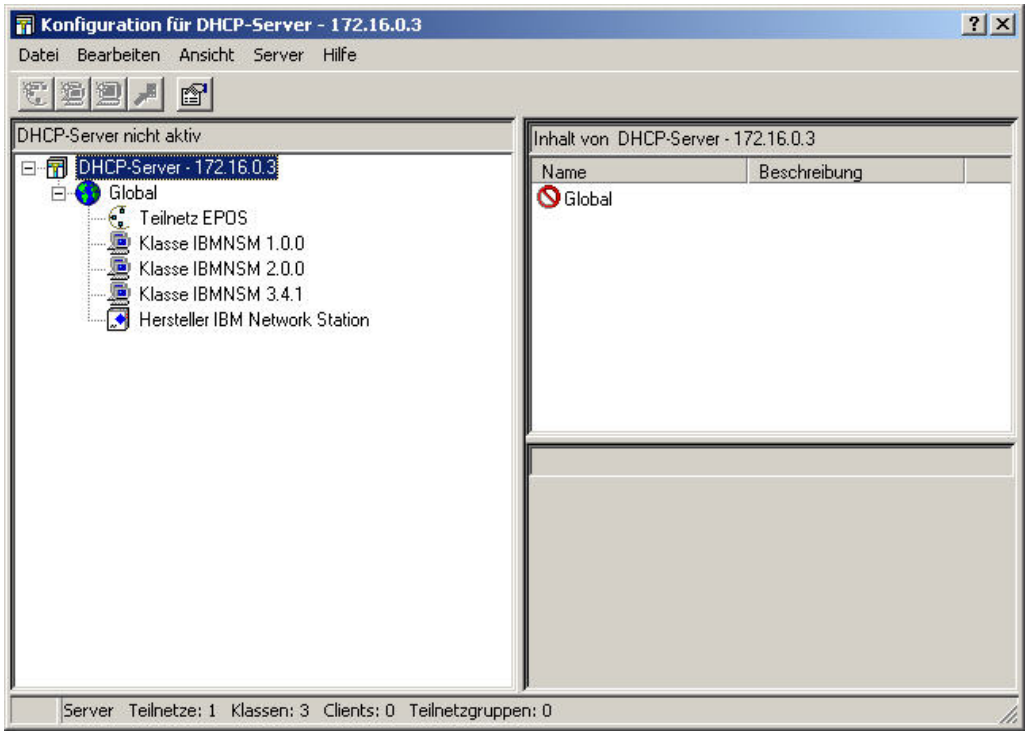
Erneut wird mir eine Übersicht angezeigt.



Subnetz: Übersicht

Diese Übersicht bezieht sich auf das gerade eben definierte DHCL-Subnetz. An dieser Stelle habe ich die Möglichkeit alle gesetzten Parameter noch einmal Revue passieren zu lassen, um zu überdenken, ob die Konfiguration korrekt ist. Mit „Fertig stellen“ wird danach der gesamte Konfigurationsdialog beendet.

Jetzt erscheint (gegebenenfalls wieder) der DHCP-Manager.

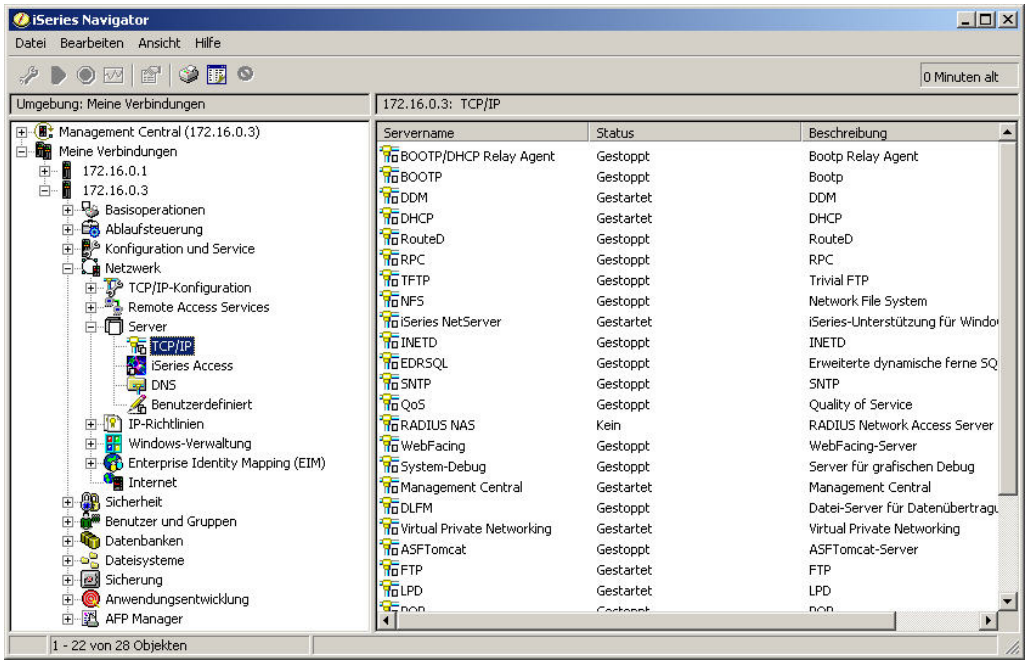


Der DHCP-Manager mit Teilnetz

Sie sehen, dass das Teilnetz EPOS, welches wir eben definiert haben angezeigt wird.

Ich beende den DHCP-Manager und gelange im iSeries Navigator wieder in die Auflistung aller TCP/IP-Server.

7.6.7.2.1
Seite 28



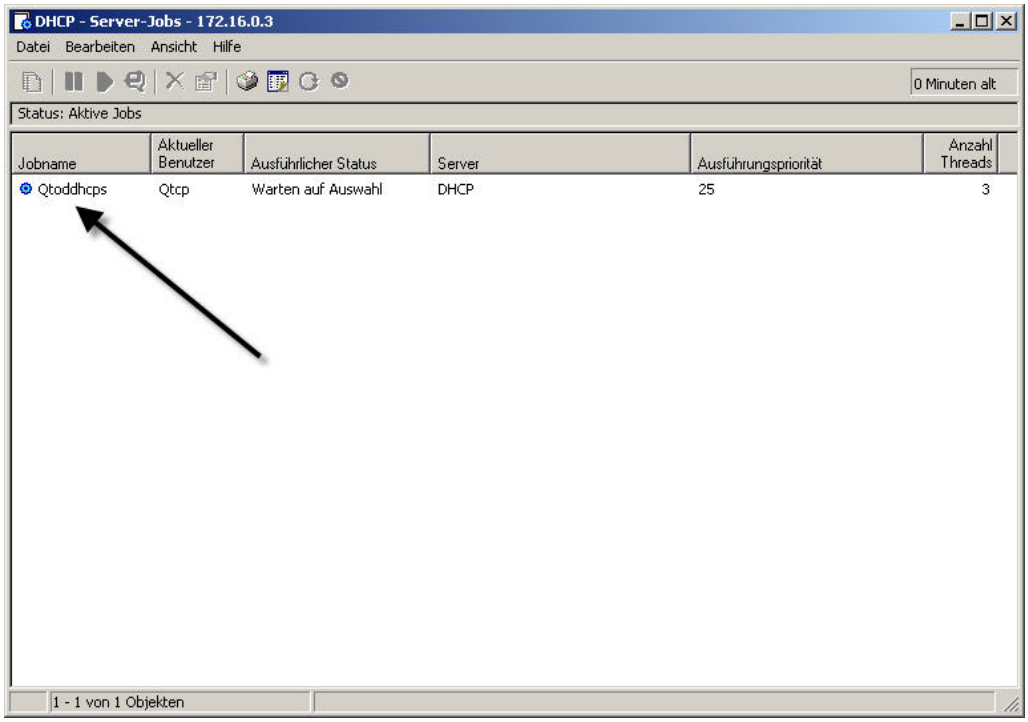
Der DHCP-Server ist gestartet

Es ist unschwer zu erkennen, dass der DHCP-Server jetzt gestartet wurde und von DHCP-Clients ab jetzt genutzt werden kann.

Bevor wir uns jedoch an die Konfiguration eines DHCP-Client begeben, wollen wir uns zuerst ansehen, welche Serverjobs der DHCP-Server benötigt, um seine Funktionen auszuführen. Ich markiere den DHCP-Server mit der Maus und drücke dann die rechte Maustaste. Folgendes Kontextmenü erscheint.

Das DHCP-Kontextmenü

In diesem Kontextmenü wähle ich die Option „Server-Jobs“ aus und gelange in den nun folgenden Dialog:



QTODDHCPS – Der DHCP Serverjob

Wir sehen hier, dass lediglich ein Serverjob, QTODDHCPS, für die Ausführung der DHCP-Dienste im OS/400 zuständig ist. Dieser Job läuft im Subsystem QSYWRK.

Sollten Sie hier Einträge sehen, deren Jobtyp auf dem Wert „Ausgabe“ steht, so handelt es sich hier um Jobprotokolle abnormal beendeter QTODDHCPS-Jobs. Dies deutet auf einen Fehler in der Konfiguration hin.

7.6.7.2.1

Seite 30

Konfiguration des Windows DHCP-Client

Die Konfiguration des Windows DHCP-Client ist verglichen mit der Konfiguration des Server kinderleicht. Auf Ihrem Windows Desktop finden Sie das Symbol Netzwerkumgebung.

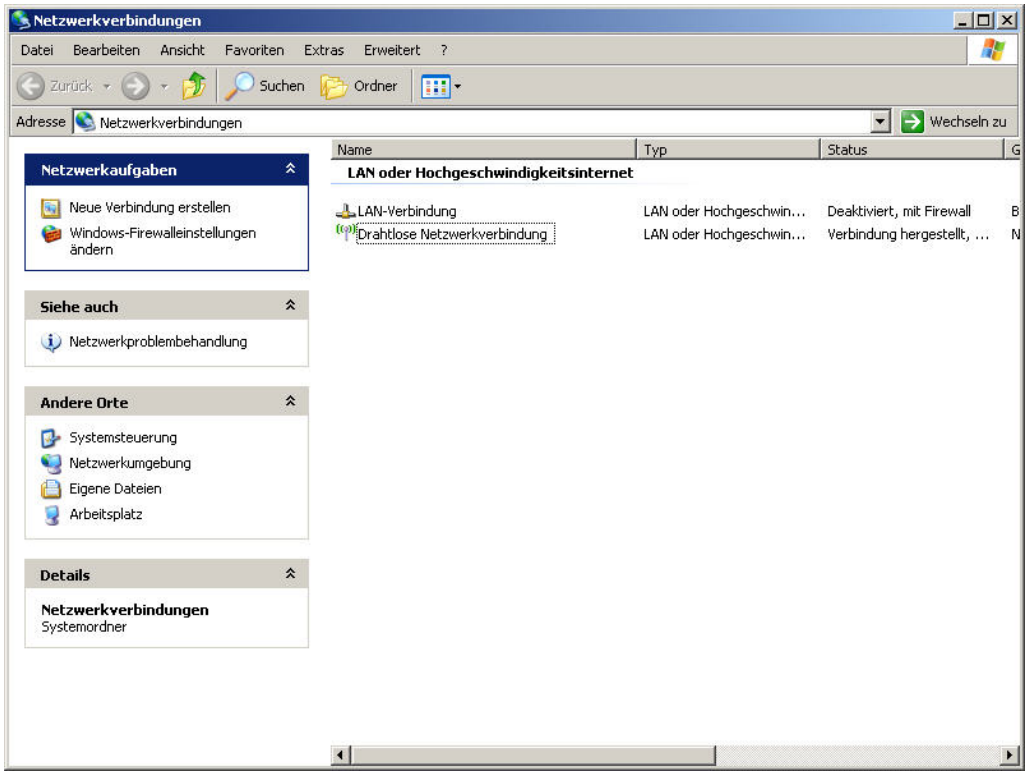
Die Windows Netzwerkumgebung

Markieren Sie dieses Symbol mit der rechten Maustaste. Jetzt erscheint folgendes Kontextmenü.

Die Netzwerk-Eigenschaften bearbeiten (1)

Hier wählen Sie den Menüpunkt „Eigenschaften“ aus und gelangen sodann bei Windows 2000 bzw. XP in die Anzeige der „Netzwerkverbindungen“.

7.6.7.3
Seite 2



Netzwerkverbindungen

In der Anzeige meines Arbeitsplatzes sehen Sie zwei Netzwerkverbindungen:

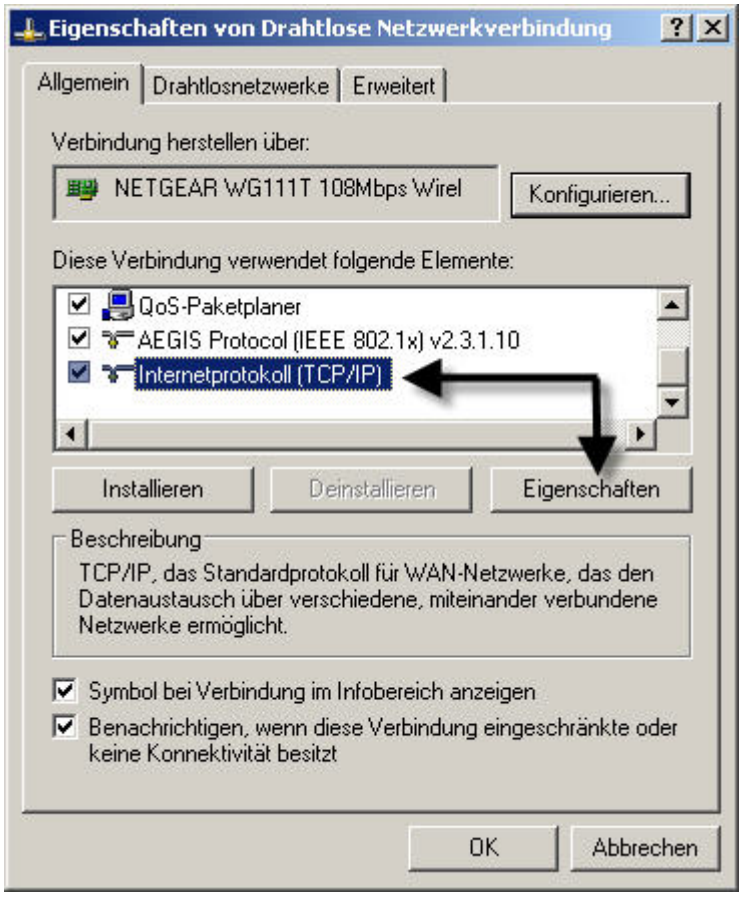
1. LAN-Verbindung: Diese ist deaktiviert und wird daher von mir nicht berücksichtigt.

2. Drahtlose Netzwerkverbindung: Diese ist aktiviert und muss von mir konfiguriert werden.

Ich markiere die Verbindung zwei mit der rechten Maustaste und folgendes Kontextmenü wird angezeigt.

Verbindungseigenschaften

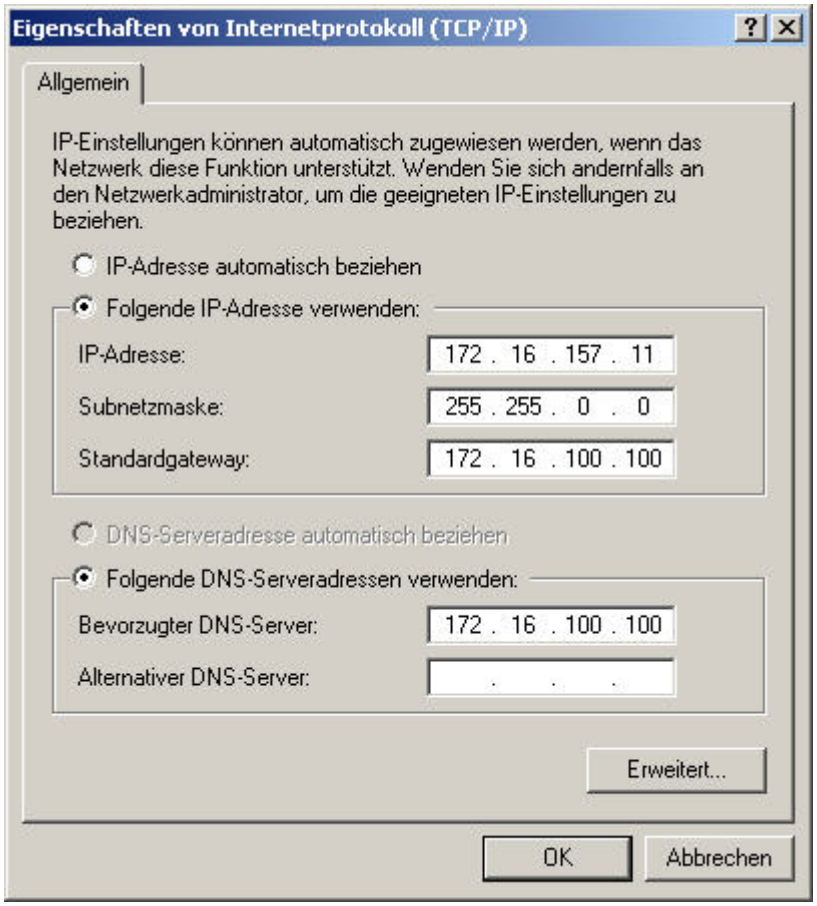
Jetzt öffnet sich folgender Dialog:



Die Verbindungseigenschaften bearbeiten (1)

In diesem Dialog gehen Sie in den Eintrag „Internetprotokoll (TCP/IP)“, markieren den Eintrag für das TCP/IP-Protokoll und betätigen den Knopf „Eigenschaften“.

7.6.7.3
Seite 4



Die Verbindungseigenschaften bearbeiten (2)

Im gezeigten Dialog markiere ich jetzt den Parameter „IP-Adresse von einem DHCP-Server beziehen“ und bin mit der Konfiguration des Client fertig!

Zwei wichtige Dateien

Nach der Konfiguration des iSeries DHCP-Server werden im Integrated File System der iSeries neben anderen zwei wichtige ASCII-Dateien abgelegt, deren Inhalt wir uns hier ansehen werden.

Beide Dateien befinden sich im Pfad /QIBM/UserData/OS400/DHCP.

7.6.7.4

Seite 2

Die Datei DHCPSD.CFG

Diese Textdatei enthält sämtliche Konfigurationsdaten des soeben konfigurierten DHCP-Server.

```
appendDomainName YES(1.)
leaseTimeDefault 86400 seconds(2.)
##### IBM_AS400_DESCR "49424D204E6574776F726B2053746174696F6E202D20
546F6B656E2052696E67"
class "IBMNSM 1.0.0"
{
 option 67 /QIBM/ProdData/NetworkStation/kernel
}
##### IBM_AS400_DESCR "49424D204E6574776F726B2053746174696F6E202D20
45746865726E6574"
class "IBMNSM 2.0.0"
{
 option 67 /QIBM/ProdData/NetworkStation/kernel
}
##### IBM_AS400_DESCR "49424D204E6574776F726B2053746174696F6E202D20
5477696E617869616C"
class "IBMNSM 3.4.1"
{
 option 67 /QIBM/ProdData/NetworkStation/kernel
}
vendor "IBM Network Station" hex"41 53 34 30 30"
#
##### IBM_AS400_DESCR "436C69656E742053797374656D65"
subnet 172.16.0.0 255.255.0.0 172.16.0.1-172.16.1.254
(alias="EPOS" (3.)
{
 client 0 0 172.16.0.1 (4.)
 client 0 0 172.16.0.2
 client 0 0 172.16.0.3
 client 0 0 172.16.0.4
 client 0 0 172.16.1.1
 option 51 259200
 option 3 172.16.1.1
 option 6 172.16.0.3
 option 15 eposgmbh.de
}
#
numLogFiles 10
logFileSize 1000
logFileName dhcpsd.log(5.)
logItem SYSERR
logItem OBJERR
```

7.6.7.4.1
Seite 2

```
logItem PROTERR
logItem WARNING
logItem EVENT
logItem ACTION
supportBootP NO
supportUnlistedClients YES(6.)
statisticSnapshot 1
leaseExpireInterval 1 minute
pingTime 1 second
reservedTime 5 minutes
usedIPAddressExpireInterval 1000 seconds
```

1. Domänenname an Client übergeben
2. Standard Leasing-Zeit
3. Definition des DHCP-Teilnetzes
4. – Option 51, ausgeschlossene Adressen
   – Option 3, Leasing-Zeit des Teilnetzes
   – Option 6, Adresse des DNS-Server
   – Option 15, der zu übergebende Domänenname
5. Log-Datei für DHCP-Vorgänge
6. Dynamisches DNS verwenden

Die Datei DHCPSD.LOG

In der Datei DHCPSD.CFG haben wir gesehen, dass Vorgänge des DHCP-Server in der Datei DHCPSD.LOG protokolliert werden. Zur Fehlerdiagnose kann diese Datei neben dem Joblog des DHCP-Server herangezogen werden.

Ich werde an dieser Stelle nicht das gesamte DHCP-Log sezieren, jedoch einige wichtige Elemente hervorheben.

```
05/23 13:15:19 : INFO:  .. log_initialize: *   NEW LOG FOLLOWS
```

Am 23.05.2000 wurde der Server gestartet, die Logging-Funktion wurde initialisiert.

```
05/23 13:15:19 : SYSERR: .. log_initialize: Logging ENABLED
05/23 13:15:19 : OBJERR: .. log_initialize: Logging ENABLED
05/23 13:15:19 : PROTERR:.. log_initialize: Logging ENABLED
05/23 13:15:19 : WARNING:.. log_initialize: Logging ENABLED
05/23 13:15:19 : EVENT: .. log_initialize: Logging ENABLED
05/23 13:15:19 : ACTION: .. log_initialize: Logging ENABLED
```

Folgende Ereignisse werden protokolliert:

- SYSERR Systemfehler
- OBJERR Objektfehler
- PROTERR Protokollfehler
- WARNING Warnungen
- EVENT Alle generellen Ereignisse
- ACTION Spezifische Anforderungsoperationen.

```
05/23 13:16:33 : ACTION: .... processDISCOVER: Address
```

Ein DHCP-Client wurde entdeckt.

```
172.16.250.1 has been reserved
```

Für diesen Client wurde die Adresse 172.16.250 reserviert.

7.6.7.4.2
Seite 2

Sie sehen anhand dieser einfachen Beispiele, dass sich mit Hilfe dieser Datei die Funktion des DHCP-Server sehr differenzierend bewerten lässt, so dass Ihnen hiermit ein wirksames Diagnose-Tool zur Verfügung steht.

Single Signon (SSO) zwischen System i und Windows Benutzern

Um ein SSO zu erreichen, stehen unterschiedliche technische Verfahren zur Verfügung. Die IBM hat sich für ein Verfahren ausgesprochen, dass von folgenden Grundsätzen ausgeht:

- Benutzer sind im bestehenden System und / oder Applikationen bereits vorhanden.
- Diese Benutzer sollen nicht umkonfiguriert und / oder gelöscht werden.
- Ein SSO soll in mehreren Schritten einführbar sein und soll noch nicht am SSO teilnehmende Benutzer nicht in Frage stellen.

Weiterhin ist zu berücksichtigen, dass der Anmeldungsprozess aus zwei Phasen besteht:

1. Authentifizierung: Es wird geprüft, ob ein Benutzer bekannt ist.
2. Autorisierung: Damit ein Benutzer in einem System / einer Applikation arbeiten kann, müssen ihm Rechte vom System bzw. der Applikation zugebilligt werden.

Von diesen Komplexen ausgehend, werden zwei Protokolle zur Realisation herangezogen:

- Kerberos für die zentrale Authentifizierung
- LDAP für das Zuordnen des Kerberos-Benutzers zu den bereits vorhandenen Benutzern der verschiedenen Systeme und / oder Applikationen

Um eine zentrale Kerberos-basierte Authentifizierung zu realisieren, brauchen wir einen Kerberos-Server, der uns in Form des Windows 2000 bzw. 2004 Servers vorliegt. Arbeiten wir in einem Microsoft Active Directory (ADS), so wird der gesamte Windows-Authentifizierungsprozess bereits durch Kerberos gesteuert. Deshalb muss in einem derartigen Umfeld lediglich das System i in die Windows-Domäne, die mit einem Kerberos-Realm gleichzusetzen ist, integriert werden.

Folgende Arbeitsschritte sind auszuführen, um ein SSO zwischen Windows ADS und System i i5/OS zu realisieren:

1. Network Authentication Services (Kerberos) im i5/OS konfigurieren
2. Windows-Benutzer und Principal erstellen, der i5/OS-Dienste repräsentiert
3. Enterprise Identity Mapping (EIM) konfigurieren
4. EIM-Identifier erstellen und Benutzer assoziieren
5. PC5250 für Single Signon konfigurieren

7.6.8
Seite 2

Voraussetzungen i5/OS

Damit die Kerberos-Konfiguration im i5/OS erfolgreich verläuft, müssen folgende Voraussetzungen erfüllt sein:

Lizenzprogramme, die unter V5R4 installiert sein müssen:

```
5722SS1  V5R4M0   Host-Server
5722SS1  V5R4M0   Qshell
5722SS1  V5R4M0   CCA Cryptographic Service Provider
5722XE1  V5R4M0   IBM eServer iSeries Access für Windows
5722NAE  V5R4M0   Network Authentication Enablement
```

Lizenzprogramme, die unter V5R2 und V5R3 installiert sein müssen:

- 5722SS1 Option 12 OS/400 - Host Servers
- 5722SS1 Option 30 OS/400 - QShell Interpreter
- 5722AC3 Crypto Access Provider 128-bit for AS/400
- 5722CE3 Client Encryption 128-bit
- 5722DG1 IBM HTTP Server for iSeries
- 5722XE1 iSeries Access for Windows

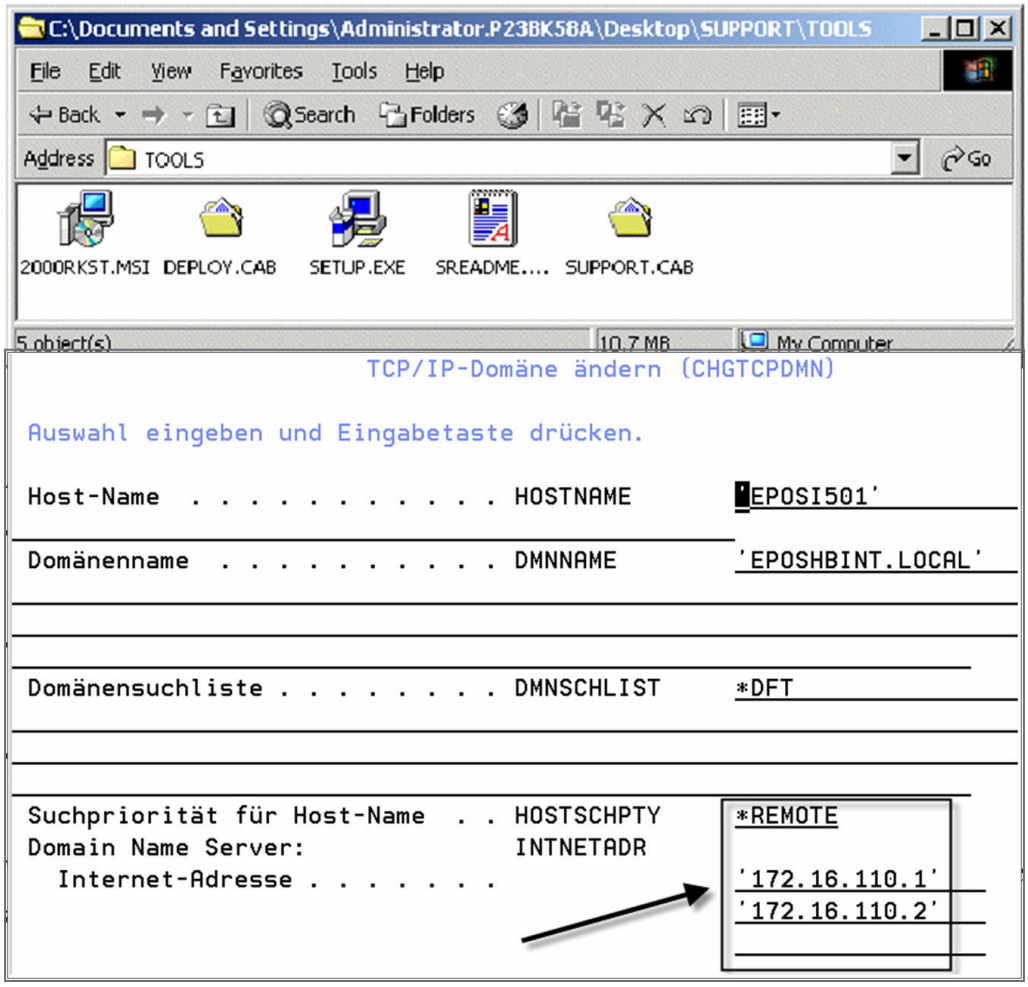
7.6.8.1

Seite 2

Voraussetzungen Windows 2000/2003

Folgende Software muss auf Ihren Windows Domain Controller installiert worden sein:

```
Windows 2000/2003 Installations CD:  LW:\Support\Tools
```



Achten Sie darauf, dass im Parameter DMNNAME der Name der Windows-Domäne eingetragen wird.

Achten Sie weiterhin darauf, dass in den Parameter INTNETADR die Adresse der Domain Name Server Ihres Netzwerks eingetragen werden.

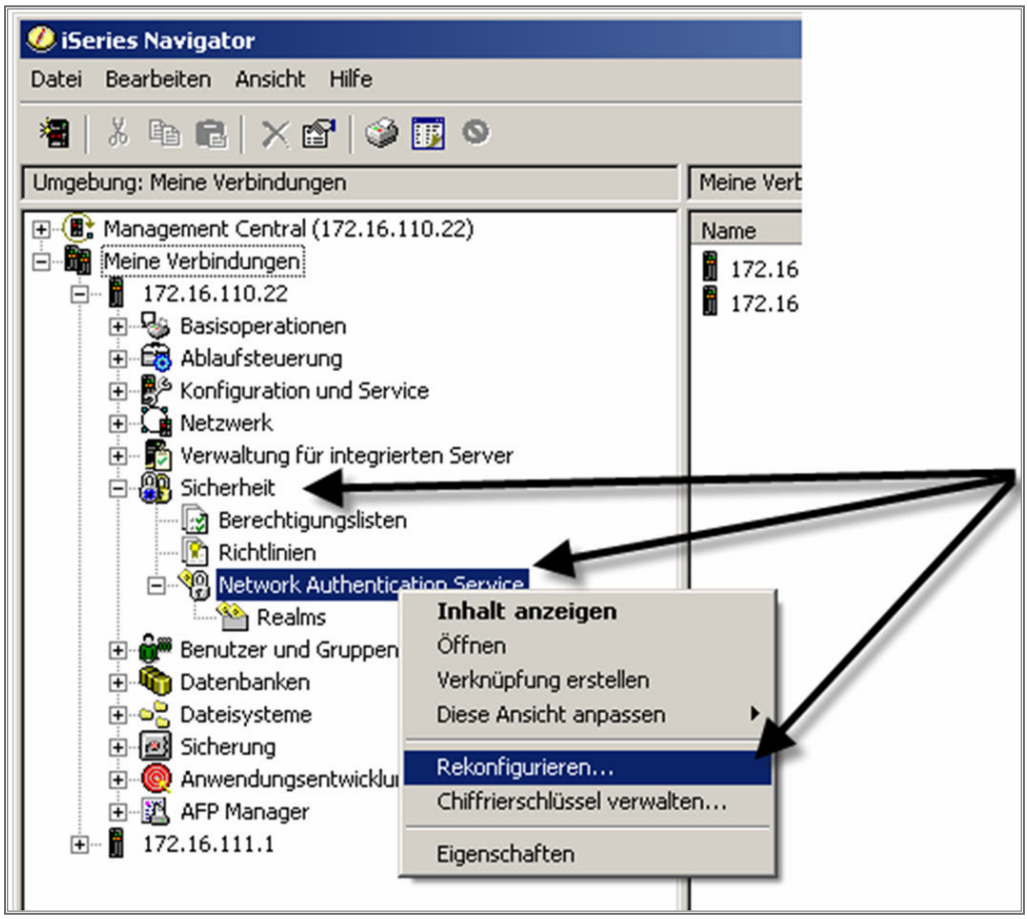
Ändern Sie die Suchpriorität gegebenenfalls in *REMOTE, um Konflikte mit der Hosttabelle Ihres Systems zu vermeiden.

7.6.8.2

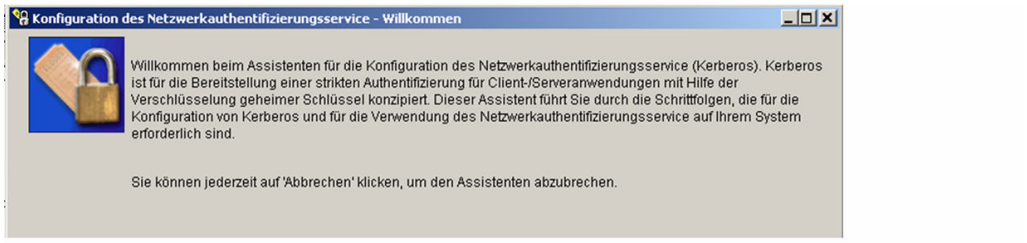
Seite 2

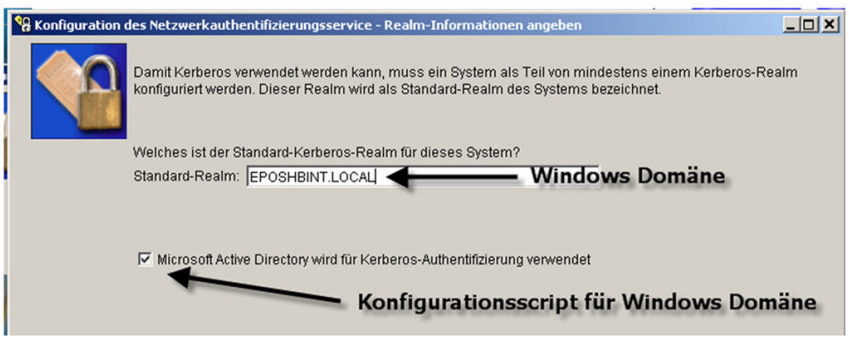
Kerberos-Konfiguration

Jetzt kann die Kerberoskonfiguration im i5/OS beginnen. Öffnen Sie im System i Navigator (SysNav) den im Bild gezeigten Pfad:



Folgender Dialog öffnet sich:

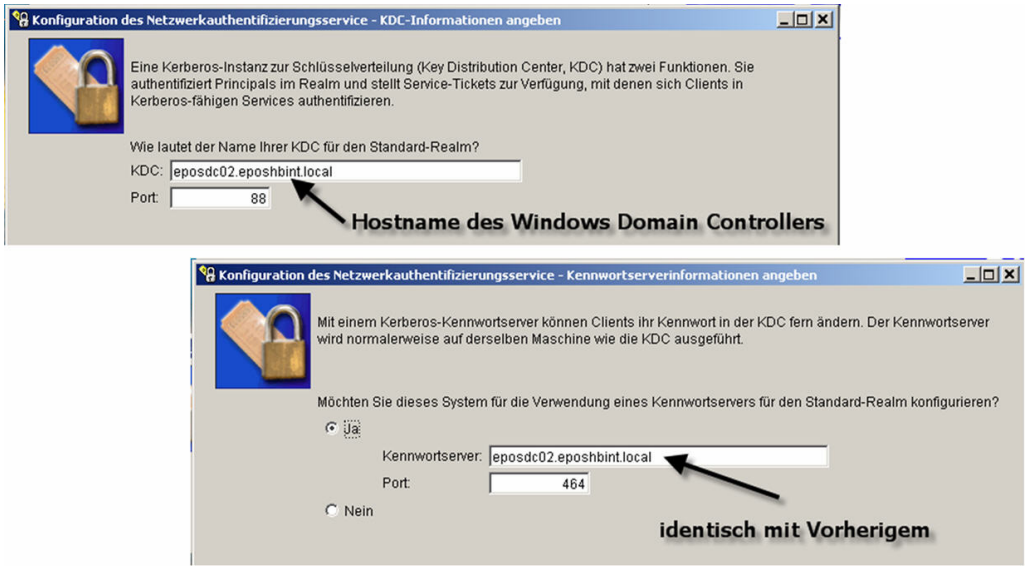




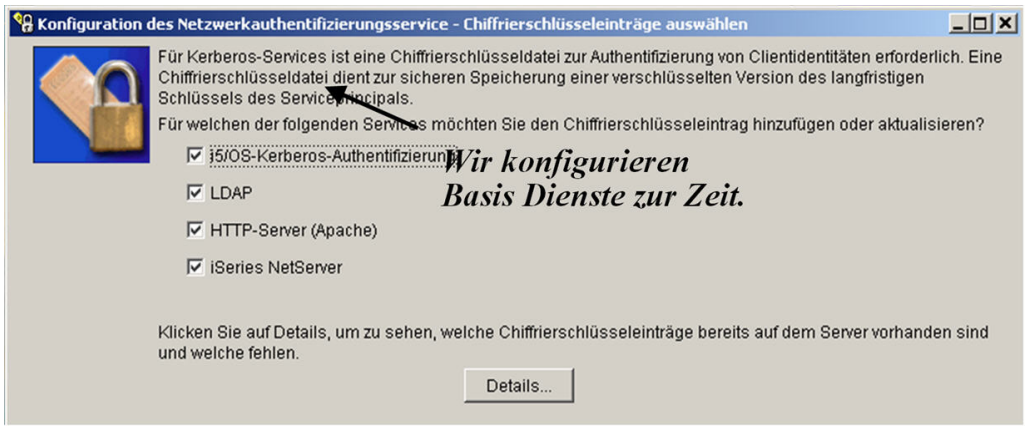
7.6.8.3
Seite 2

Tragen Sie in den Parameter „Standard-Realm“ den Namen der Windows-Domäne ein.

Durch Markieren des Parameters „Microsoft …“ bewirken Sie, dass für die Windows-ADS-Konfiguration ein Script erstellt wird.



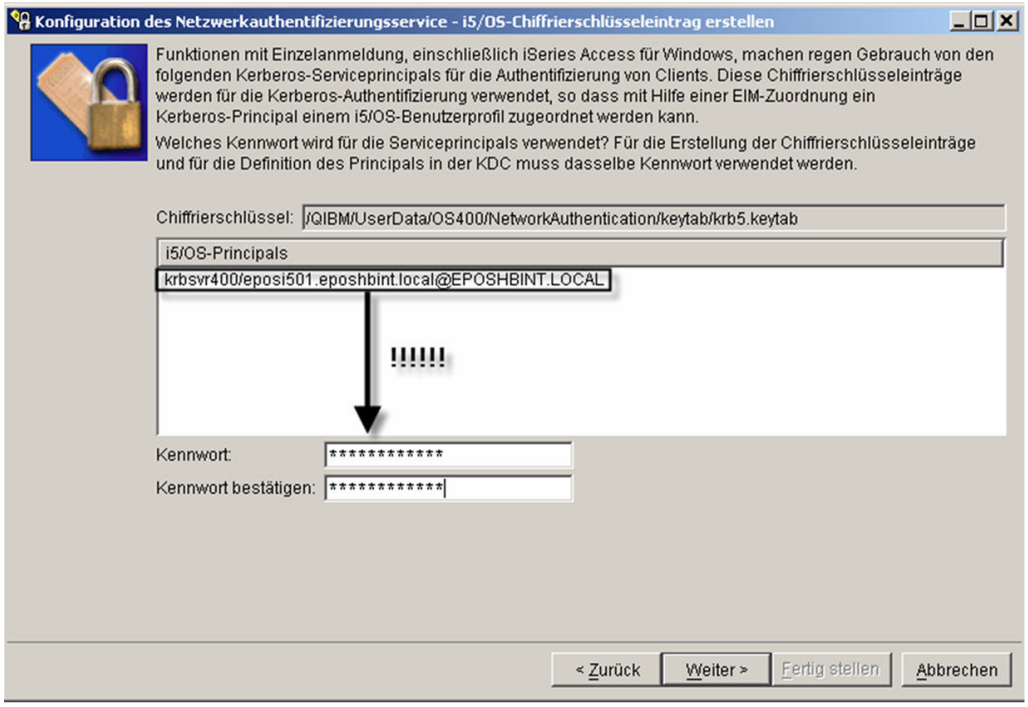
In den nächsten beiden Dialogen tragen Sie den voll qualifizierten Hostnamen Ihres Windows-Domaincontrollers ein.

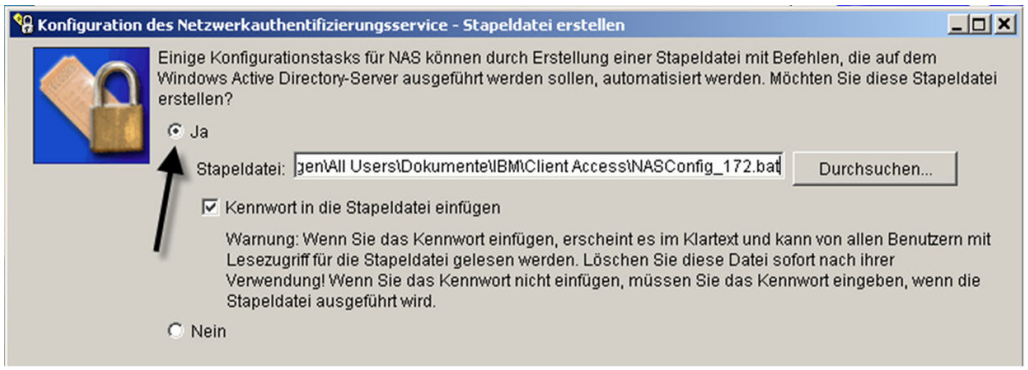


Markieren Sie sodann, für welche i5/OS-Dienste eine Kerberos-basierte Authentifizierung ermöglicht werden soll.

Daraufhin wird der erzeugte Kerberos-Prinzipal für das i5/OS angezeigt. Ein genaues Abbild dieses Prinzipals wird später auf Ihrem Windows-Domaincontroller zu installieren sein.

Die Prinzipale müssen durch ein Kennwort geschützt werden. Achten Sie darauf, dass das Kennwort den Regeln der Windows-ADS gemäß vergeben wird.





In diesem Dialog geben Sie durch Aktivierung von „Ja“ an, dass Sie ein Konfigurationsscript für die Kerberoskonfiguration unter Windows erstellen wollen.

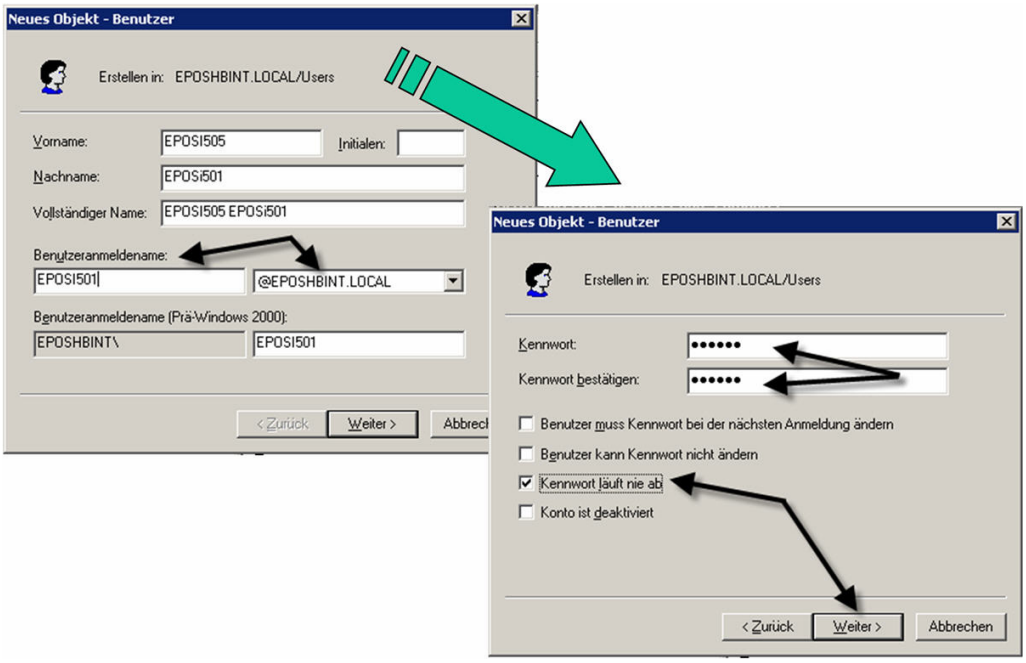
Damit ist die System i-seitige Konfiguration zunächst beendet.

Das oben gezeigte Konfigurationsscript wird dann auf dem Windows-Domaincontroller ausgeführt.

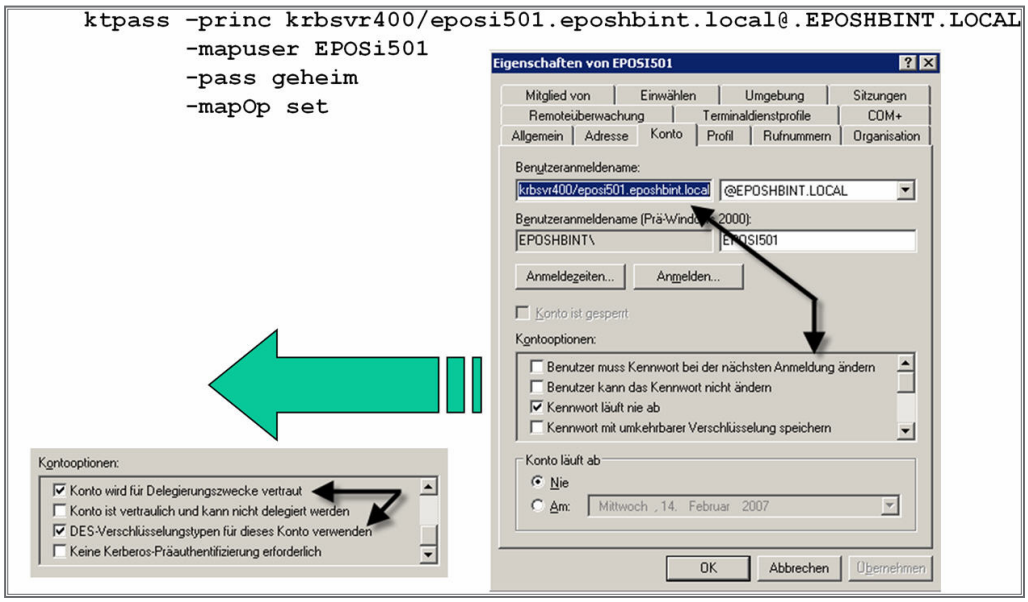
Die Ausführung des Scripts bewirkt, dass in Windows neue Benutzer erzeugt werden, die als Stellvertreter der System i-Dienste anzusehen sind.

7.6.8.3
Seite 4

Hier sehen Sie einen derartigen Dienstestellvertreter:



Die gezeigten Werte sind durch das Script erzeugt worden. Aktivieren Sie „Kennwort läuft nie ab“.



Das Konfigurationsscript hat oben gezeigten Befehl ausgeführt, der ein Abbild des i5/OS-Prinzipals in Windows erstellt.

Markieren Sie in den Kontooptionen die gezeigten Parameter.

Damit ist die Windows-Konfiguration beendet.

Gehen Sie wieder auf Ihr System i und öffnen Sie die Qshell mit dem Befehl QSH:

Bevor Sie den gezeigten Befehl eingeben, stellen Sie sicher, dass im IFS unterhalb des Verzeichnisses „Home“ ein Verzeichnis mit dem Namen Ihres System i Benutzerprofils existiert. Wenn nicht, erstellen Sie es jetzt.

Dann geben Sie den Befehl „kinit“ wie gezeigt ein.

Endet der Befehl ohne eine Nachricht, so haben Sie die Kerberos-Konfiguration korrekt durchgeführt.

Hier einige Fehlermeldungen und deren Ursachen:

CWBSY1012 - Kerberos principal not found on server...

Kein Principal-Eintrag vorhanden im KDC. Alternativ: Namensauflösungsproblem

CWBSY1017 - rc=608 Kerberos credentials not valid on server...

Iseries betrachtet das Ticket als nicht für sich gültig. Namensauflösungsproblem. iSeries kann eigenen Namen nicht korrekt auflösen.

CWBSY1017 - rc=612 Kerberos credentials not valid on server...

Die iSeries lehnt das Ticket ab. Wir haben ein Password-Problem. Vermutlich ist das geheime Kennwort falsch.

Damit ist die Kerberoskonfiguration abgeschlossen.

7.6.8.3
Seite 6

Nachdem die Authentifizierung mittels Kerberos realisiert wurde, wenden wir uns jetzt der EIM-Authentifizierung zu.

Folgende Arbeitsschritte sind bereits in Teil 7.6.8 ausgeführt worden, um ein SSO zwischen Windows ADS und System i i5/OS zu realisieren:

1. Network Authentication Services (Kerberos) im i5/OS konfigurieren
2. Windows-Benutzer und Prinzipal erstellen, der i5/OS-Dienste repräsentiert.

Folgende Funktionen müssen noch konfiguriert werden.

3. Enterprise Identity Mapping (EIM) konfigurieren
4. EIM-Identifier erstellen und Benutzer assoziieren
5. PC5250 für Single Signon konfigurieren

Die Punkte 3 bis 4 werden in diesem Beitrag erläutert.

Enterprise Identity Mapping (EIM) konfigurieren

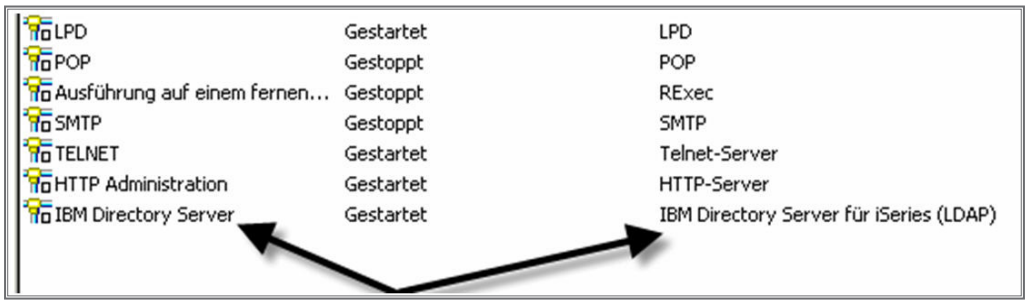
Wenn wir uns den Anmeldeprozess vorstellen, so sehen wir, dass die Authentifizierung gegen die Windows ADS läuft. Der auf dem System i um Zugang bittende Dienst hat vom ADS ein Kerberos-Ticket bekommen, das den Namen des Kerberos/Windows-Benutzer enthält. i5/OS vertraut diesen Angaben.

Jetzt beginnt die Authentifizierung, in der i5/OS dem Dienst Rechte zubilligen will, dieses jedoch nicht kann, da der Kerberos/Windows-Benutzer i5/OS unbekannt ist.

Das EIM stellt eine Funktion dar, die es möglich macht, den Kerberos-Benutzer – in unserem Fall den Windows-Benutzer – dem System i Benutzer zuzuordnen.

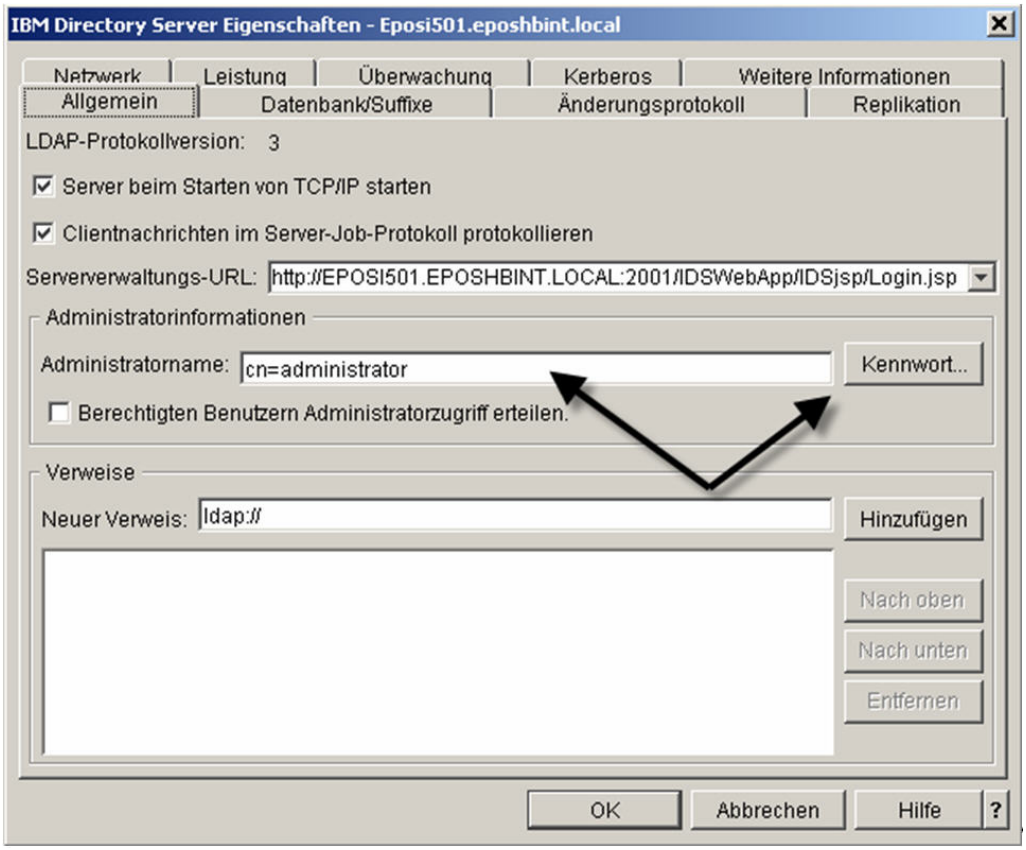
EIM ist eine Funktion des i5/OS LDAP Servers. Damit EIM richtig funktionieren kann, muss der LDAP-Server gestartet und funktionsfähig sein.

Sie sollten zuerst den Zustand des LDAP-Servers in i5/OS prüfen. Sehen Sie sich die TCP/IP-Server im System i Navigator an.



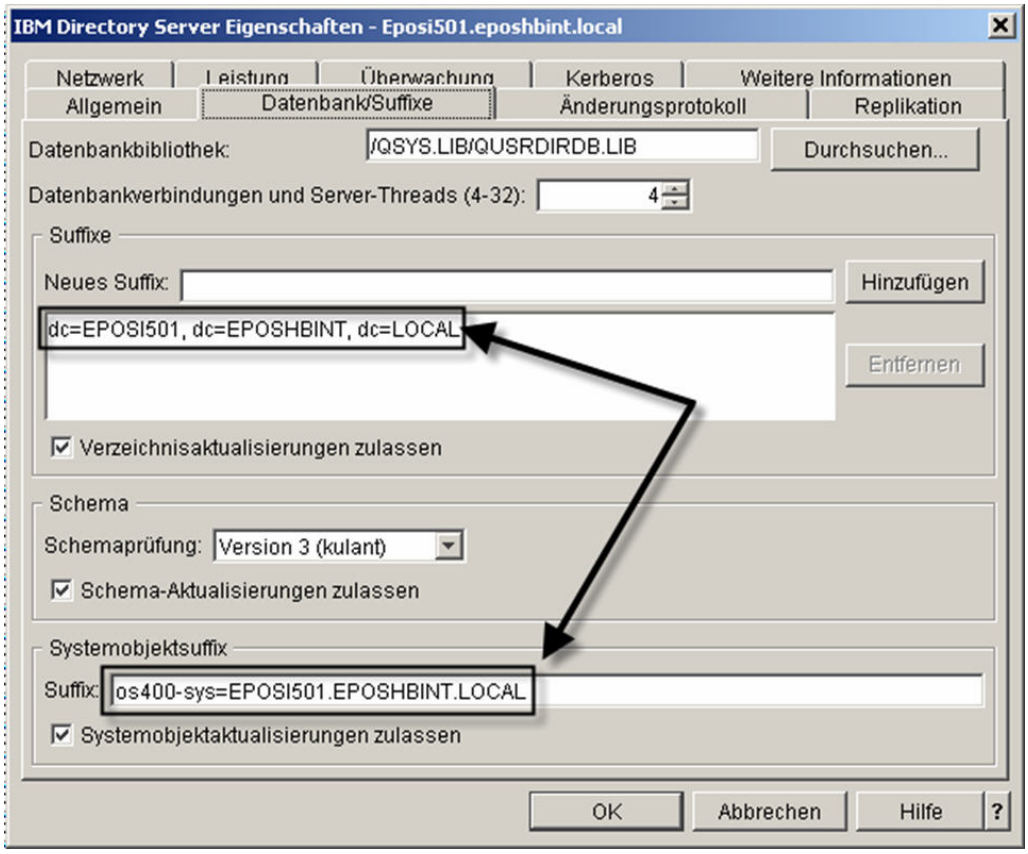
7.6.8.4
Seite 2

Sehen Sie sich die Eigenschaften des Servers an.



Falls Sie das Kennwort des LDAP-Administrators nicht kennen, sollten Sie es in Erfahrung bringen bzw. ihm ein neues geben.

Wechsel Sie in das Register „Datenbank/Suffixe“.



Achten Sie darauf, dass in „Suffixe“ der i5/OS TCP/IP-Domänenname steht. Gleichen Sie diesen Eintrag gegebenenfalls mit dem Inhalt von CFGTCP ➜ Auswahl 12 ab.

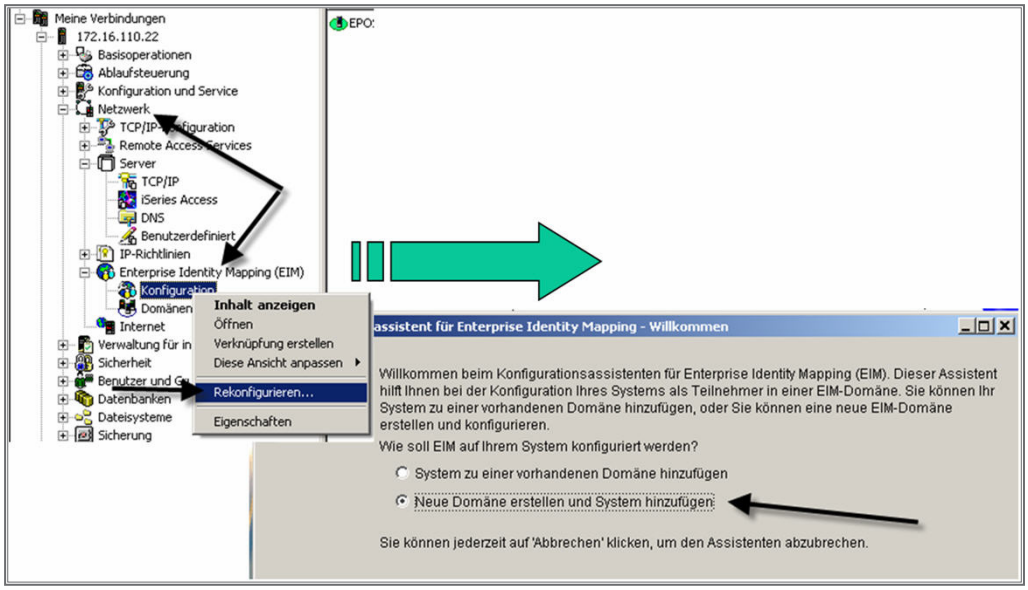
Das Gleiche gilt für den Parameter „Suffix“ in „Systemobjektsuffix“.

Starten Sie gegebenenfalls den LDAP-Server erneut.

Jetzt muss die sogenannte EIM-Domäne erstellt werden.

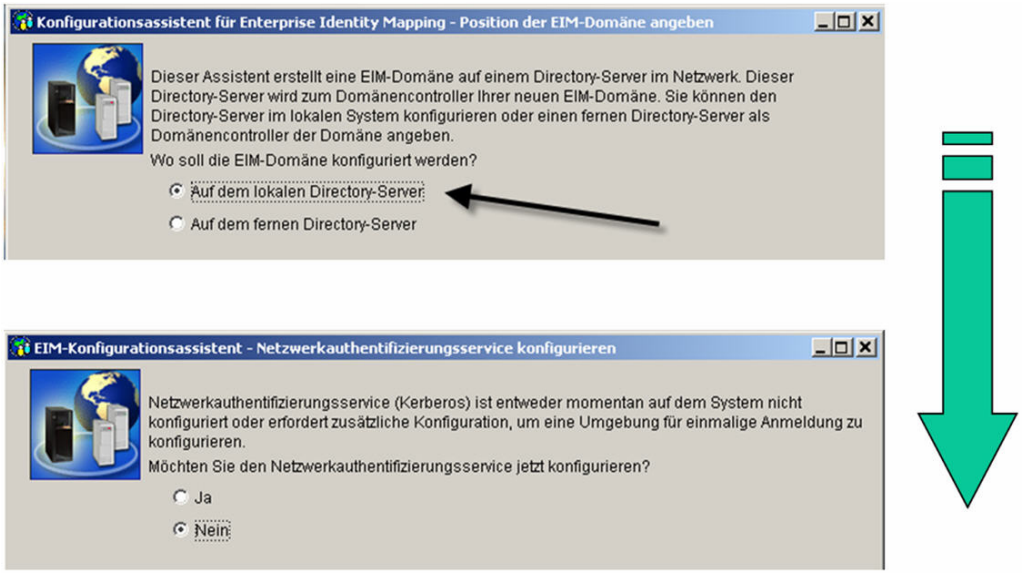
7.6.8.4
Seite 4

Öffnen Sie im System i Navigator den Pfad Netzwerk ➜ EIM ➜ Konfiguration ➜ Rechtsklick.



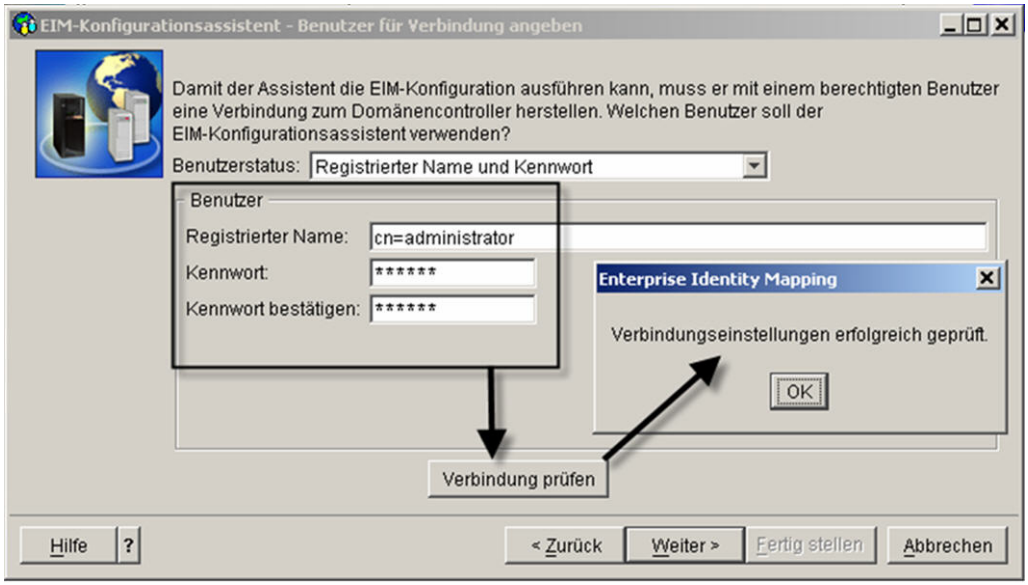
Im Dialog „Konfigurieren“ bzw. „Rekonfigurieren“ wählen Sie „Neue Domäne erstellen ...“

Daraufhin werden nacheinander folgende Dialoge gezeigt.

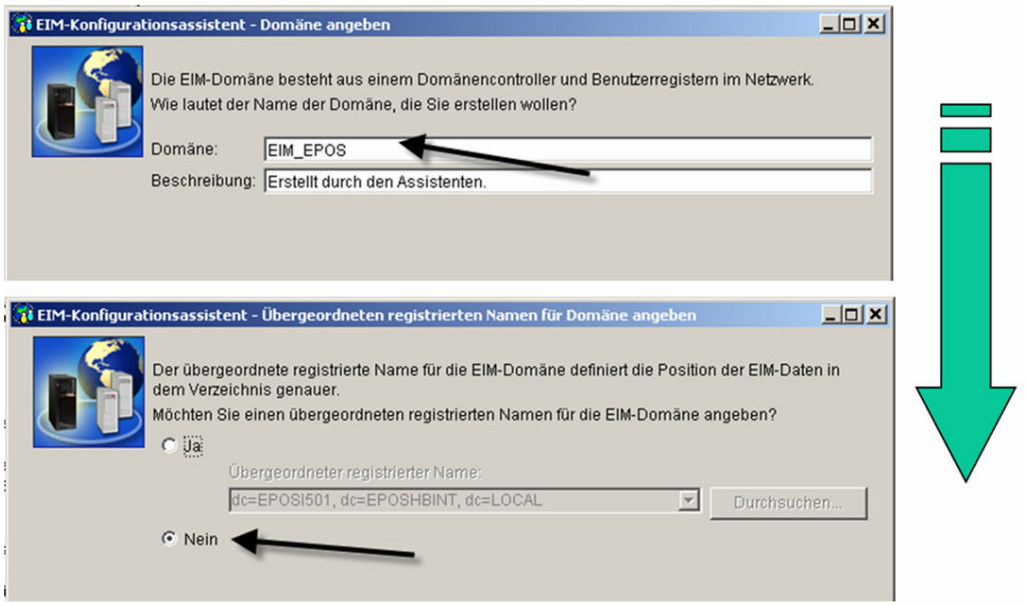


Erstellen Sie einen EIM-Domänencontroller auf dem lokalen Directory-Server.

NAS werden Sie nicht konfigurieren, da Ihre Kerberos-Konfiguration bereits ausgeführt wurde.

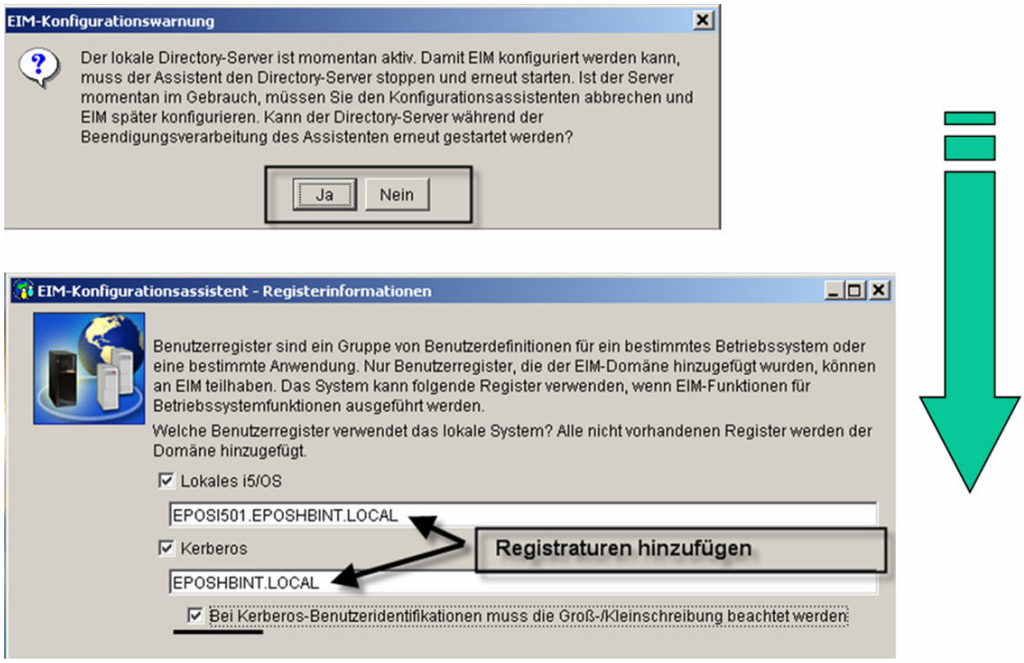


Melden Sie sich danach am LDAP-Server an und klicken auf weiter.



Geben Sie der EIM-Domäne einen Namen. Den nächsten Dialog beantworten Sie mit „Nein“.

7.6.8.4
Seite 6

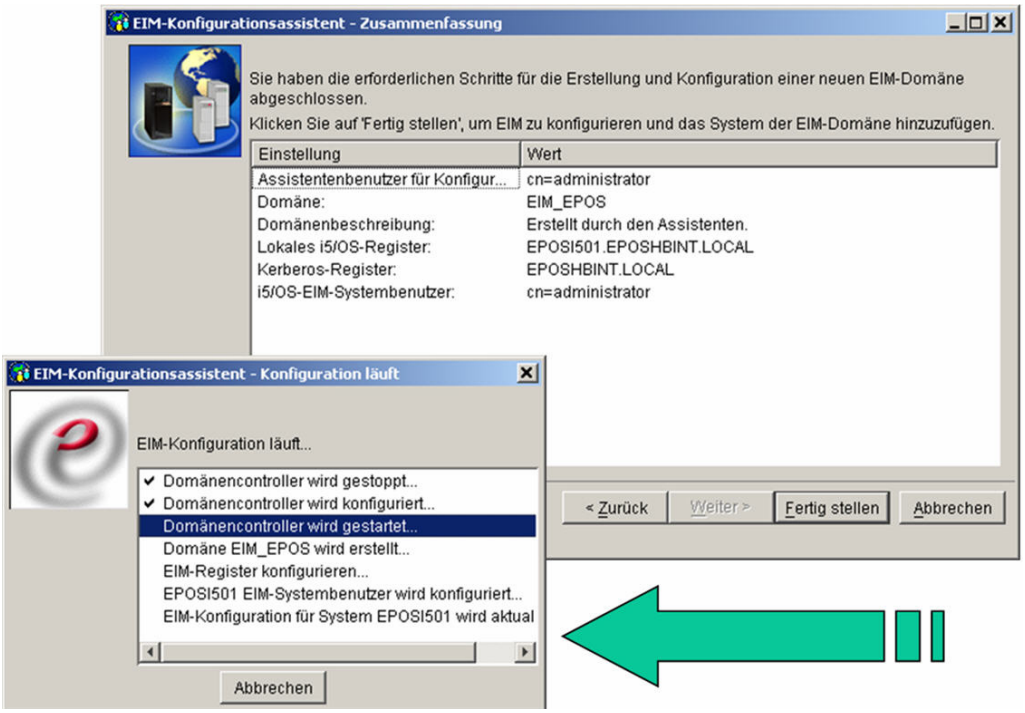


Den ersten nachfolgenden Dialog beantworten Sie mit „Ja“.

Im zweiten Dialog werden die sogenannten Benutzerregistraturen dem EIM-Controller hinzugefügt. Die gezeigten Parameter sollten bereits aufgrund der bisherigen Konfiguration mit den korrekten Werten belegt sein.

Wichtig!

Aktivieren Sie den Parameter „Bei Kerberos-Benutzeridentifikation muss Groß-/Kleinschreibung beachtet werden“.



Damit ist die EIM-Basiskonfiguration beendet.

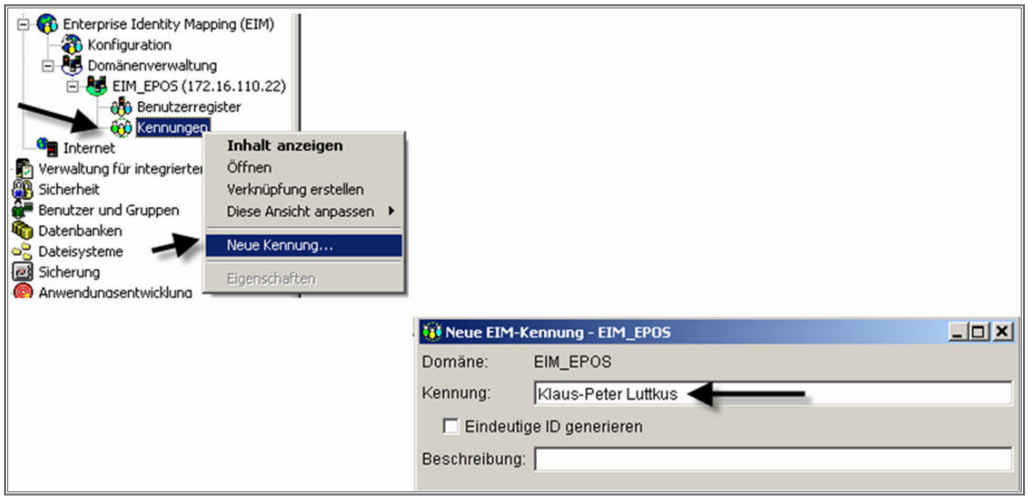
7.6.8.4

Seite 8

EIM-Identifier erstellen und Benutzer assoziieren

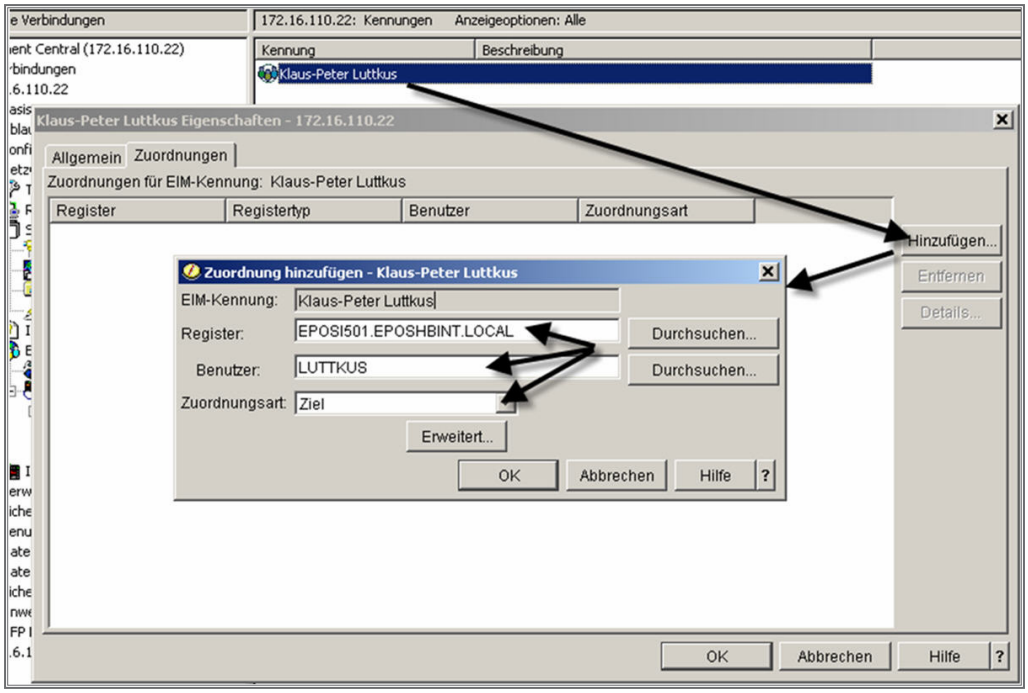
Jetzt muss die Benutzerzuordnung (assoziieren) konfiguriert werden. Dazu muss für jeden Benutzer zuerst einmal ein EIM-Identifier angelegt werden.

Öffnen Sie im System i Navigator den Pfad Netzwerk ➜ EIM ➜ Domänenverwaltung ➜ Kennungen.



Hier vergeben Sie einen Namen für die EIM-Kennung eines Benutzers. Es hat sich als sinnvoll herausgestellt, hier den realen Namen eines Benutzers als Kennung zu verwenden.

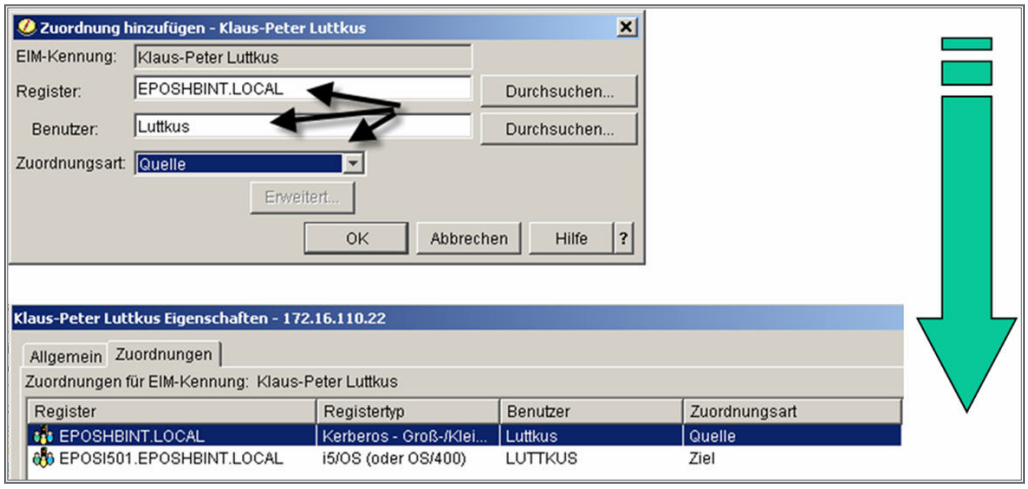
Jetzt müssen die Zuordnungen definiert werden.



Ein Rechtsklick auf die soeben erzeugte Kennung plus „Hinzufügen“ bringt Sie in den Zuordnungsdialog. Hier tragen Sie zuerst den Namen des System i-Benutzerprofils ein mit der Zuordnungsart „Ziel“.

7.6.8.5
Seite 2

Dann ordnen Sie der EIM-Kennung den Kerberus-Benutzer hinzu.



Unten sehen Sie den komplettierten Zuordnungen-Dialog.

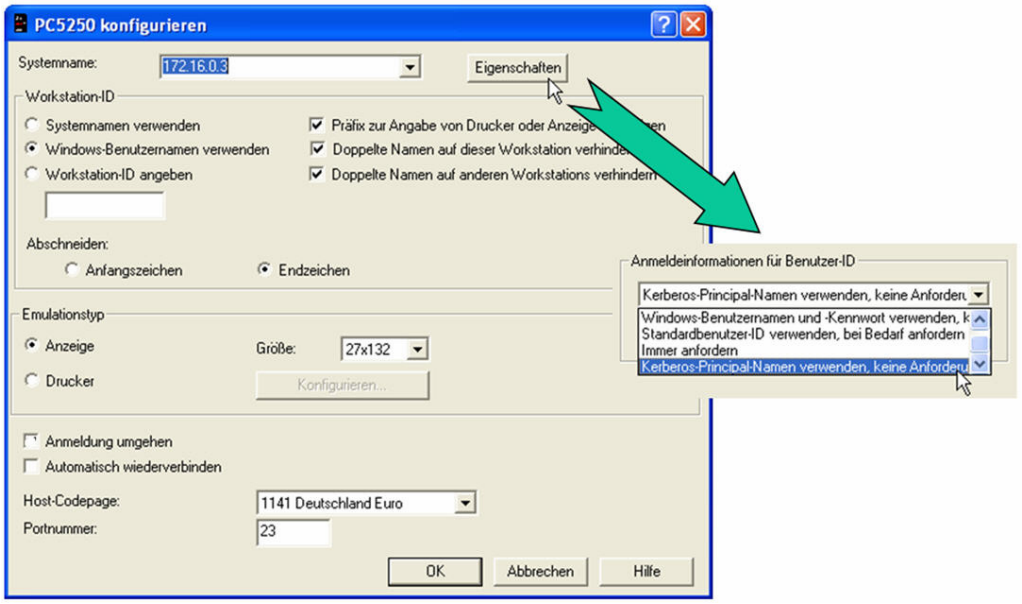
Wichtig!

Achten Sie peinlich genau auf die Schreibweise der Benutzernamen bezüglich Groß-Kleinschreibung.

Damit ist die EIM-Konfiguration für einen Benutzer fertig gestellt.

PC5250 für Single Signon konfigurieren

Im letzten Schritt muss PC5250 des System i Access für Windows für Kerberos konfiguriert werden.



Stellen Sie in den Eigenschaften der PC5250-Konfiguration den Parameter „Anmeldeinformationen für Benutzer-ID“ auf den Wert „Kerberos-Prinzipalnamen verwenden…“

Damit der Single Signon auch wirklich komplett durchgeführt wird, sollte der i5/OS – Systemwert QRMTSIGN auf *VERIFY gesetzt werden.

Prüfen Sie sodann die Konfiguration durch Starten der Bildschirmemulation.

7.6.8.6

Seite 2

Mögliche Fehler und deren Ursache

CWBSY1012 - Kerberos principal not found on server...

Kein Principal-Eintrag vorhanden im KDC. Alternativ: Namensauflösungsproblem

CWBSY1017 - rc=608 Kerberos credentials not valid on server...

Iseries betrachtet das Ticket als nicht für sich gültig. Namensauflösungsproblem. iSeries kann eigenen Namen nicht korrekt auflösen.

CWBSY1017 - rc=612 Kerberos credentials not valid on server...

Die iSeries lehnt das Ticket ab. Wir haben ein Password-Problem. Vermutlich ist das geheime Kennwort falsch.

CWBSY1018 - Kerberos credentials could not be mapped to user...

Die Kerberos Authetifizierung ist grundsätzlich in Ordnung. Jedoch wurde der Windows Benutzer im EIM fehlerhaft assoziiert.

7.6.8.7

Seite 2

IBM Systems Director Navigator for i

Mit dem Betriebssystem IBM i V6R1 hat IBM den designierten Nachfolger der Client-Anwendung System i Navigator (in der Folge iNAV) ausgeliefert. Der IBM Systems Director Navigator for i ist eine Webanwendung für die System i-Verwaltung, dessen Funktionen denen des System i Navigators ähneln.

Jedoch ist die Implementation beider Programme komplett unterschiedlich: Wird der System i Navigator auf einem Windows PC installiert, so wird der IBM System Director Navigator for i (in der Folge als SDN bezeichnet) im internen Anwendungsserver des i/OS betrieben.

Eine manuelle Installation des IBM System Director Navigator for i ist nicht notwendig. i/OS V6R1 wird bereits mit einer lauffähigen Installation des SDN ausgeliefert.

Derzeit sind im SDN jedoch nicht alle Funktionen des iNAV implementiert, so dass ein Parallelbetrieb beider Programme derzeit anzuraten ist.

Welche Funktionen sind derzeit im SDN implementiert?

- **System** (Systemstatus, Systembedienernachrichten, Systemprotokoll, Plattenstatus, Kennwort ändern und vieles mehr)
- **Basisoperationen** (Nachrichten, Drucker, Druckausgabe und vieles mehr)
- **Ablaufsteuerung** (aktive Jobs, Serverjobs, Ausgabewarteschlangen, aktive Subsysteme und vieles mehr)
- **Konfiguration und Service** (Systemwerte, Zeitmanagement, Platteneinheiten, Plattenpools und vieles mehr)
- **Netzwerk** (TCP/IP-Server, Leitungen, DNS-Server und vieles mehr)
- **Verwaltung integrierter Server** (alle virtuellen Platten, fernen Systeme, Domänen und vieles mehr)
- **Sicherheit** (Berechtigungslisten, Cryptographic Services Key Management, Erkennung von unbefugtem Zugriff und vieles mehr)
- **Benutzer und Gruppen** (Benutzer, Benutzer erstellen, Gruppen und vieles mehr)
- **Datenbanken** (Performance Monitors, Diagnosezentrale, Datenbanken und vieles mehr)
- **Journalverwaltung** (Journale, Journal erstellen und vieles mehr)
- **Leistung** (Daten untersuchen und vieles mehr): Der Performance Advisor – eine Funktion, die im iNAV *nicht* zur Verfügung steht.
- **Dateisysteme** (Integrated File System, Dateifreigaben, Dateifreigabe erstellen und vieles mehr)

19. Ergänzung 6/2009

7.7
Seite 2

- **Internetkonfigurationen** (Digital Certificate Manager, System i Navigator-Tasks im Web und vieles mehr)
- **Cluster Resource Services** (Liste der Knoten anzeigen, Cluster löschen und vieles mehr)

Verglichen mit iNAV – welche Funktionen sind derzeit im SDN *nicht* implementiert?

- **Management Central**

Voraussetzungen

Wie bereits beschrieben, ist eine manuelle Installation von SDN nicht erforderlich, jedoch sollten folgende Lizenzprogramme für den fehlerfreien Betrieb von SDN installiert sein:

Betriebssystemfunktionen:
5761SS1 12 Host-Server
5761SS1 30 Qshell
5761SS1 33 Portable App Solutions Environment

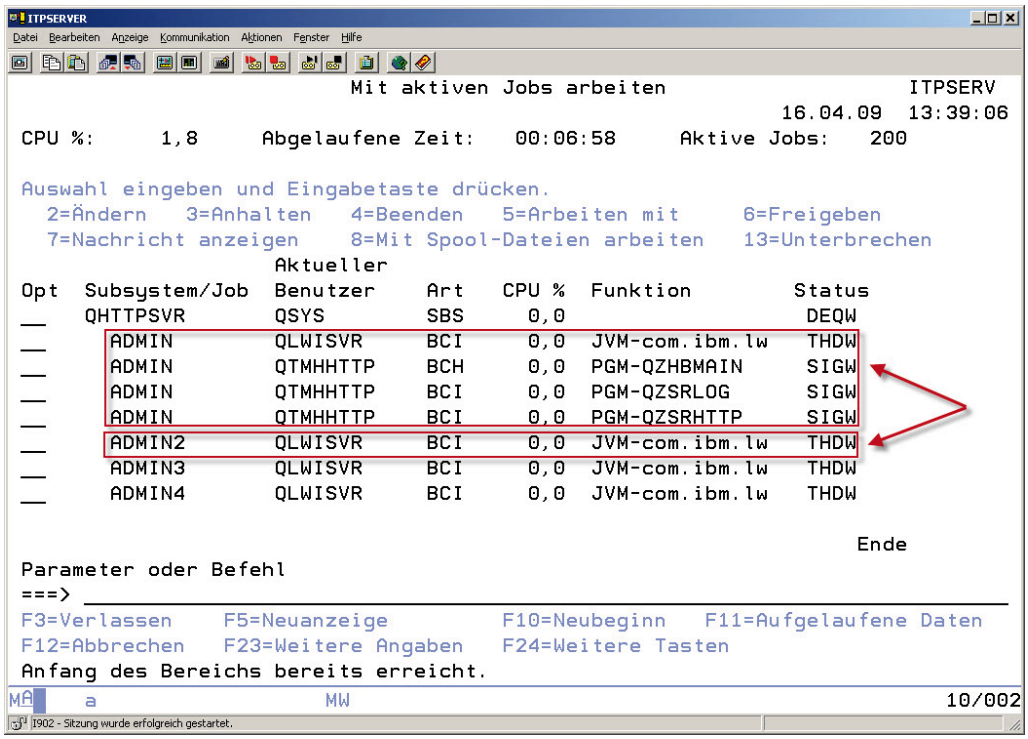
JAVA-Umgebung
5761JV1 *BASE IBM Developer Kit für Java
5761JV1 8 J2SE 5.0 32 Bit
5761JC1 *BASE IBM Toolbox for Java

Web Server
5761DG1 *BASE IBM HTTP-Server für i5/OS

Weiterhin sollten der administrative Webserver des i/OS gestartet sein. Falls dieses nicht automatisch auf Ihrem System geschieht rufen Sie folgenden Befehl auf:

```
STRTCPSVR *http HTTPSVR(*ADMIN)
```

Folgende Jobs sollten dann im Subsystem QHTTPSVR erscheinen:



SDN – Serverjobs

7.7.1
Seite 2

ADMIN stellt folgende Funktionen zur Verfügung:

- Web GUI
- Web Navigator
- LDAP-Administration
- IPP-Administration

ADMIN2 stellt folgende Funktion zur Verfügung:

- SDN
- Secure Perspective
- Cluster Services
- HASM

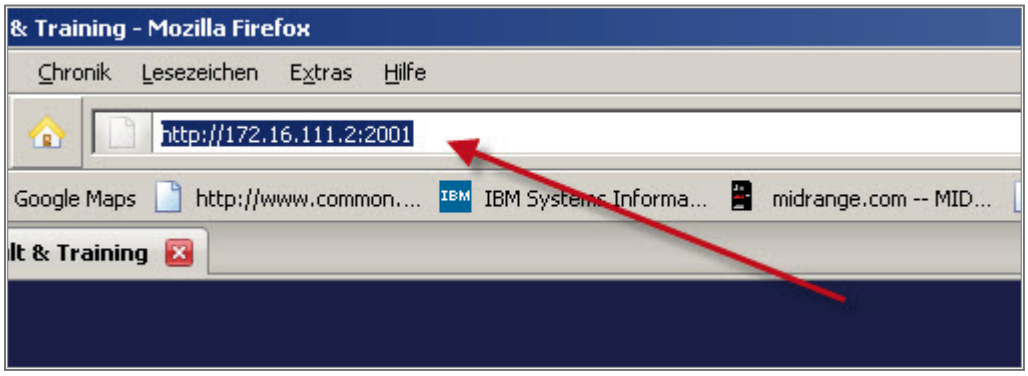
Stellen Sie sicher, dass die oben bezeichneten Jobs fehlerfrei laufen.

Folgende Web-Browser werden unterstützt:

- Microsoft Internet Explorer 6.0 oder höher
- Mozilla Firefox 1.0 oder höher

Start von SDN

Um SDN zu starten, geben Sie einfach folgende Adresse in Ihrem Browser ein:



SDN – Adresse

Die aufzurufende Adresse entspricht folgendem Muster:

http://<IP-Adresse bzw. Hostname Ihres System:Port

Übersetzt auf das von mir gezeigte System:

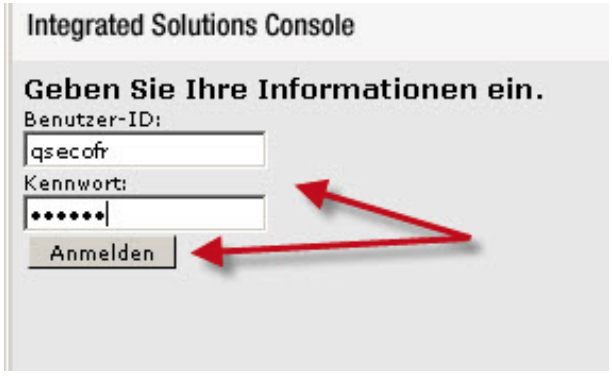
http://172.16.111.1:2001

oder

http://itpserver.eposhbint.local:2001

Beachten Sie, dass der Port 2001 zwingend vorgeschrieben ist.

Jetzt öffnet sich der Anmeldungsdialog des SDN:



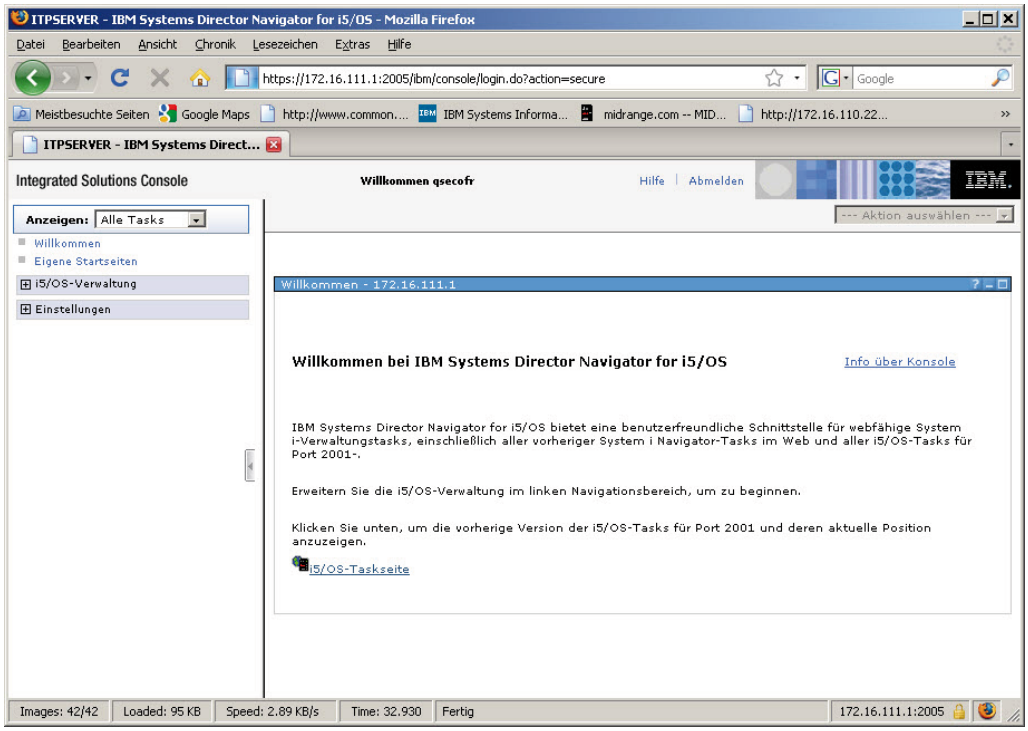
SDN – Anmeldung

Beachten Sie, dass sich der jeweilige Benutzer nur einmalig via SDN anmelden kann. Konkret bedeutet das: Ich melde mich wie in oben gezeigtem Dialog abgebildet als Benutzer QSECOFR an der sogenannten Integrated Solutions Console (ISC) an. Nach der erfolgreichen Anmeldung kann sich kein zweiter Administrator QSECOFR mit Hilfe der ISC am System anmelden.

7.7.2
Seite 2

Es ist jedoch möglich sich mittels eines anderen Benutzerprofils, das die SECOFR-Rechte besitzt, an der ISC anzumelden.

Nach einiger Zeit sehen Sie folgenden Dialog in Ihrem Browser.



ISC – Oberfläche

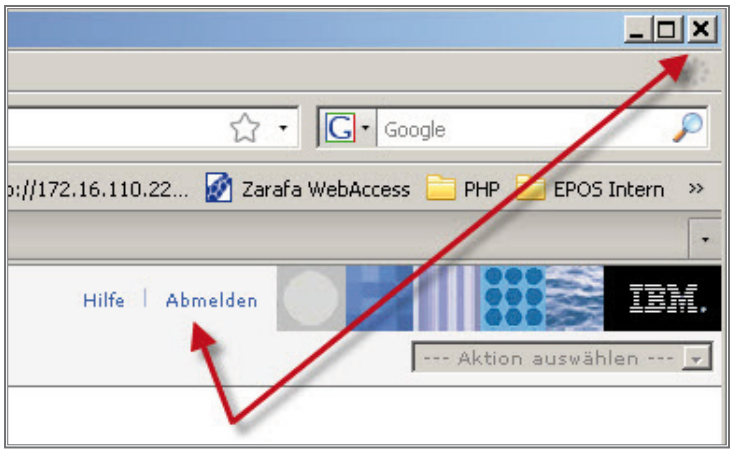
Achtung!

Ist Ihre SDN-Sitzung für 30 Minuten inaktiv, so wird die Sitzung beendet und Sie müssen sich erneut anmelden.

Oberfläche

Ich möchte in diesem Abschnitt einige grundsätzliche Funktionsweisen der ISC klären, ohne auf bestimmte administratorische Fähigkeiten einzugehen.

Die Abmeldung aus der ISC erfolgt mit Hilfe des Links Abmelden.



Abmelden

Verwenden Sie statt des Abmelden-Links das x im Browser-Fenster, so schießen Sie quasi Ihre Sitzung ab.

Erfolgt daraufhin eine erneute Anmeldung als Benutzer QSECOFR, so erhalten Sie eine Meldung, dass bereits eine Sitzung für den Benutzer QSECOFR besteht und dass jetzt zuerst die bestehende Sitzung abnormal beendet wird, bevor die neue aktuelle Sitzung erzeugt werden kann.

Navigationsleiste

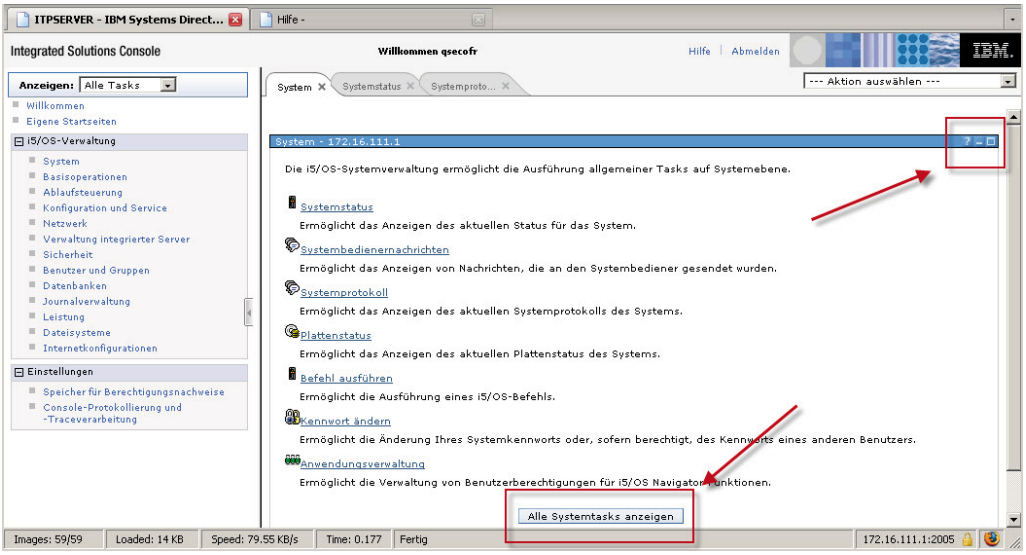
7.7.3
Seite 2

Hier werden die in der Konsole verfügbaren Tasks aufgelistet. Die Tasks sind in Organisationsknoten gruppiert, die bestimmten Taskkategorien entsprechen, z. B. Server oder Anwendungen. Die Organisationsknoten können in mehreren Ebenen verschachtelt sein.

In der Navigationsstruktur werden nur die Tasks angezeigt, auf die Sie Zugriff haben. Wenn Sie in der Navigationsstruktur auf eine Task klicken, wird im Arbeitsbereich eine Seite mit einem oder mehreren Modulen zum Ausführen der Task angezeigt.

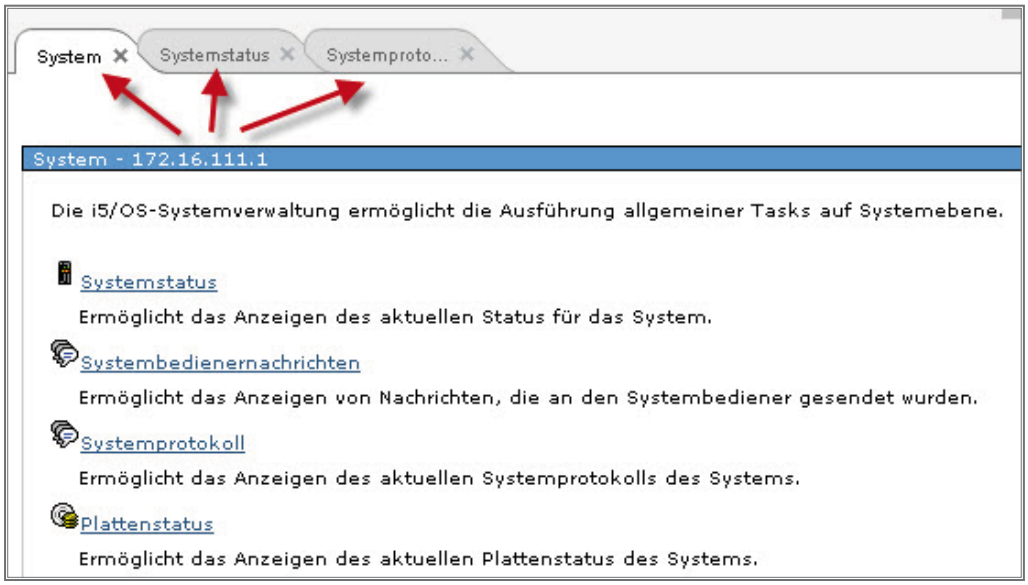
Zwischen Navigations- und Arbeitsbereich gibt es eine Steuerungsmöglichkeit zum Ausblenden der Navigation, so dass für den Arbeitsbereich mehr Platz zur Verfügung steht. Die Steuerung funktioniert wie ein Schalter. Mit einem einzigen Klick lässt sich die Navigation ein- oder ausblenden.

Nach dem Aufrufen einer Seite, wird der Inhalt der Seite im Arbeitsbereich angezeigt. Solange Sie keine Seite aufrufen, wird im Arbeitsbereich die Einführungsseite angezeigt. Eine Seite enthält ein oder mehrere Konsolenmodule zum Ausführen von Operationen. Jedes Konsolenmodul besitzt eigene Steuerelemente zur Navigation. Auf einigen Seiten befindet sich ein Steuerelement zum Schließen der Seite und Zurückkehren zur Einführungsseite.



Arbeitsbereich

Es wurde eine Seitenleiste hinzugefügt. Sie ermöglicht das Arbeiten mit Konsolenmodulen auf verschiedenen Seiten, ohne die Seite schließen zu müssen oder ungespeicherte Daten zu verlieren.



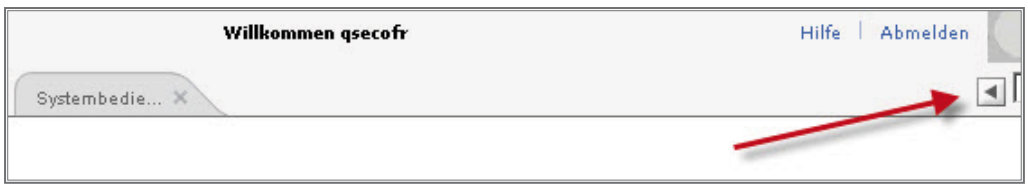
Seitenbereich

Beispiel: Wenn Sie mit einer Anwendung auf Seite A arbeiten, können Sie eine Anwendung auf einer anderen Seite öffnen. Dort können Sie Informationen zu einer Ressource sammeln, die Sie zum kompletten Ausfüllen des Formulars auf Seite A benötigen, ohne die dort bereits eingegebenen aber noch ungespeicherten Daten zu verlieren. Es können mehrere Seiten gleichzeitig geöffnet werden, während nur eine der geöffneten Seiten im Vordergrund sein kann (*aktuelle Seite*).

Mit den Registerkarten auf der Seitenleiste können Sie zwischen den offenen Seiten hin und her navigieren oder eine bestimmte Seite schließen. Auf der Seitenleiste können bis zu fünf Registerkarten angezeigt werden. Öffnen Sie aber mehr als fünf Seiten, weist die Seitenleiste Pfeile für horizontales Schieben auf, damit auf die anderen Registerkarten zugegriffen werden kann.

Mehr als fünf Seiten geöffnet

Klicke ich jetzt auf den bezeichneten Pfeil, so wird folgender Dialog angezeigt.



Die 6. Seite

7.7.3 Seite 4

Ein erneuter Klick auf das jetzt umgekehrte Pfeil-Symbol bringt mich zurück auf die ersten fünf Seiten.

Auf der Leiste befindet sich auch eine Liste zur Auswahl von Aktionen.

Aktionen

Folgende Aktionen stehen zur Auswahl:

- Seite schließen
  Schließt die aktuelle (im Vordergrund stehende) Seite.
- Zu eigenen Startseiten hinzufügen
  Fügt die aktuelle Seite zur Liste der Seiten, die bei der Anmeldung an der Konsole automatisch gestartet werden, hinzu.
- Geöffnete Seiten verwalten
  Öffnet ein Modul, mit dem Sie auf geöffnete Seiten zugreifen oder diese schließen können. Diese Anwendung ist besonders dann nützlich, wenn Sie sehr viele Seiten offen haben, die dann mit Hilfe der Registerkarten auf der Seitenleiste zu schwierig zu verwalten sind.
- Portletaktualisierung konfigurieren
  Dient zur Anpassung Ihrer persönlichen Aktualisierungseinstellungen für Konsolenmodule auf einer Seite, die Portletaktualisierungen unterstützen.
- Seite in neuer Registerkarte öffnen
  Bei Seiten, die mehrere Instanzen unterstützen (bekannt als *Multitaskknoten*), öffnet diese Aktion eine weitere Kopie der aktuellen Seite.

Das Verhalten der Registerkarten auf der Seitenleiste hängt vom Typ des Knotens ab, der zum Starten der Task verwendet wird.

Einzeltaskknoten
Ein Navigationsknoten, in dem pro Sitzung nur eine Instanz im Arbeitsbereich geöffnet werden kann. Wenn Sie in der Konsolennavigation auf einen Einzeltaskknoten klicken, wird die Task im Arbeitsbereich geöffnet und für die Seite eine Registerkarte hinzugefügt. Ist die Seite bereits geöffnet, wird sie zur aktuellen, im Vordergrund stehenden Seite.

Multitaskknoten
Ein Navigationsknoten, in dem pro Sitzung mehrere Instanzen im Arbeitsbereich geöffnet werden können. Jede neue Instanz lässt sich von einem Portlet aus oder über die Auswahl von „Seite in neuer Registerkarte öffnen“ aus der Dropdown-Liste für Aktionen starten.

Bei Multitaskknoten zeigt die Beschriftung der Registerkarte für die Seite an, dass mehrere Instanzen der Seite offen sind. Beispiel: Lautet der Seitenname für einen Multitaskknoten Ausgaben und sind drei Instanzen der Seite geöffnet, sind die Registerkarten in folgender Reihenfolge beschriftet:

- Ausgaben
- Ausgaben (1)
- Ausgaben (2)

Wenn bereits eine oder mehrere Instanzen eines Multitaskknotens im Arbeitsbereich geöffnet sind, wird beim Klicken auf den Knoten von der Konsolennavigation aus keine neue Instanz des Knotens geöffnet. Vielmehr wird die Knoteninstanz, auf die zuletzt zugegriffen wurde, zur aktuellen Seite.

Knoten für ausgeblendeten Knoten
Ein Navigationsknoten, von dem aus eine Seite gestartet wird, für die es auf der Seitenleiste keine Registerkarte gibt. Solange ein ausgeblendeter Knoten geöffnet ist, bleibt die Seitenleiste inaktiviert. Für Seiten, die von einem Knoten mit ausgeblendeter Task gestartet wurden, ist Folgendes nicht möglich: Auf der Seitenleiste stehen keine Aktionen zur Auswahl. Sie können nicht zur Liste der Startseiten hinzugefügt werden. Sie können nicht über ‚Geöffnete Seiten verwalten‘ geschlossen werden bzw. es kann darüber nicht auf sie zugegriffen werden.

19. Ergänzung 6/2009

7.7.3
Seite 6

Oberfläche an eigene Bedürfnisse anpassen

Aus dem bisher Gesagten ergibt sich, dass einzelne Funktionen im SDN als Tasks bezeichnet werden, dass sie in der Navigatorleiste angezeigt werden und dass logisch zusammengehörende Funktionen auf einer Seite zusammengefasst, gruppiert werden.

Sowohl auf der Task-Ebene als auch auf der Seiten-Ebene können wir die Oberfläche des SDN individualisieren. Ich werde mich zuerst mit der Seiten-Ebene beschäftigen. Im Jargon des SDN heißt das, ich möchte eine eigene Startseite erstellen.

7.7.3.1

Seite 2

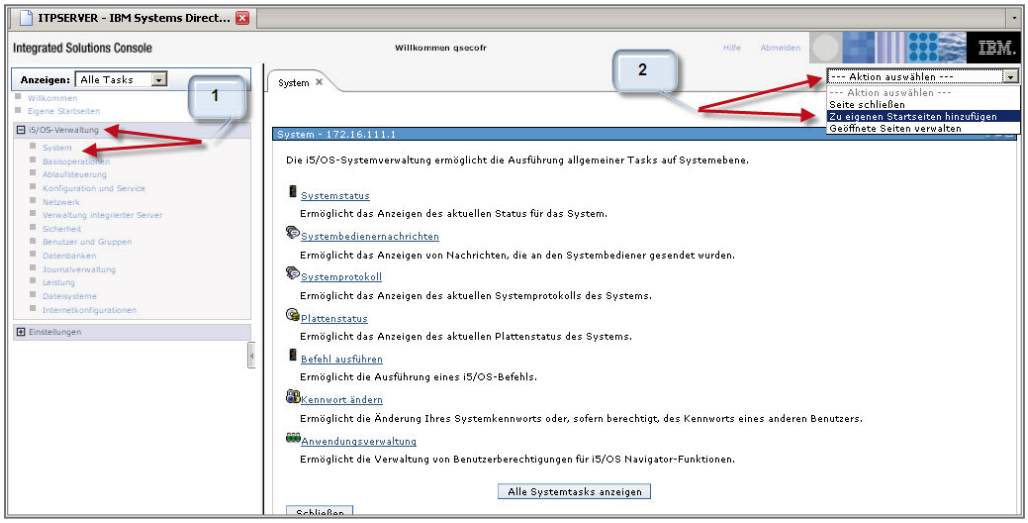
Eigene Startseite

Lassen Sie uns über folgendes Szenario nachdenken. Als Systemadministrator meines Power-Systems rufe ich folgende Seiten häufig ab:

- System
- Basisoperationen
- Ablaufsteuerung
- Benutzer und Gruppen

Diese einzelnen Seiten möchte ich auf einer eigenen Seite, sofort nach Start des SDN abrufen können.

Nachdem ich mich am SDN angemeldet habe, öffne ich mit Hilfe der Navigatorleiste die Seite im Verzeichnis i5/OS Verwaltung → System (1).



Eigene Startseite erstellen

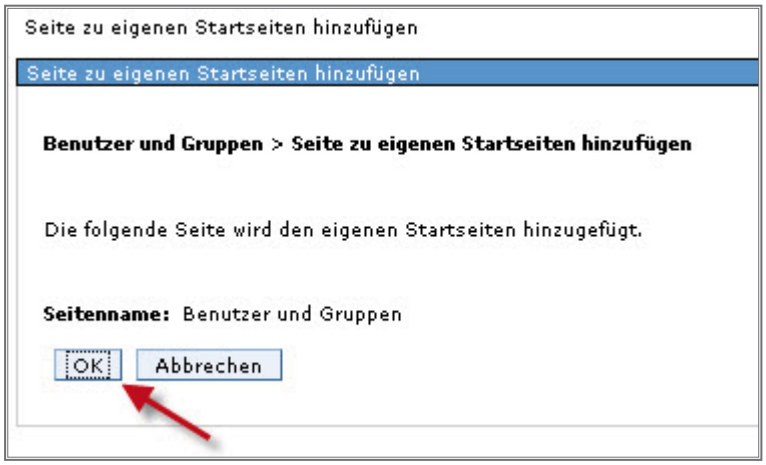
Danach öffne ich die (2) Aktionen auswählen Combobox und markiere den Eintrag Zu eigenen Seiten hinzufügen.

Gleichermaßen verfahre ich mit den Seiten

- i5/OS Verwaltung → Basisoperationen
- i5/OS Verwaltung → Ablaufsteuerung
- i5/OS Verwaltung → Benutzer und Gruppen

7.7.3.1.1

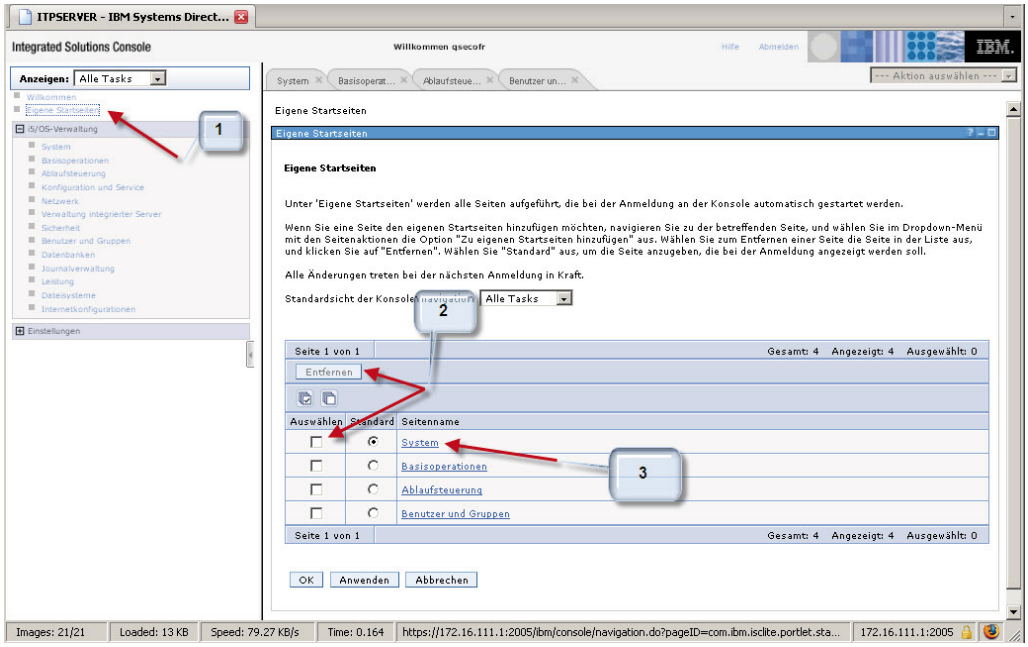
Seite 2



Hinzufügen bestätigen

Bevor die betreffende Seite der eigenen Startseite hinzugefügt wird, muss der Vorgang in dem oben angezeigten Dialog bestätigt werden.

Ich wechsle in der Navigatorleiste in den Eintrag Eigene Startseiten(1)



Eigene Startseite bearbeiten

In dem gezeigten Bild wird die eigene Startseite bearbeitet:

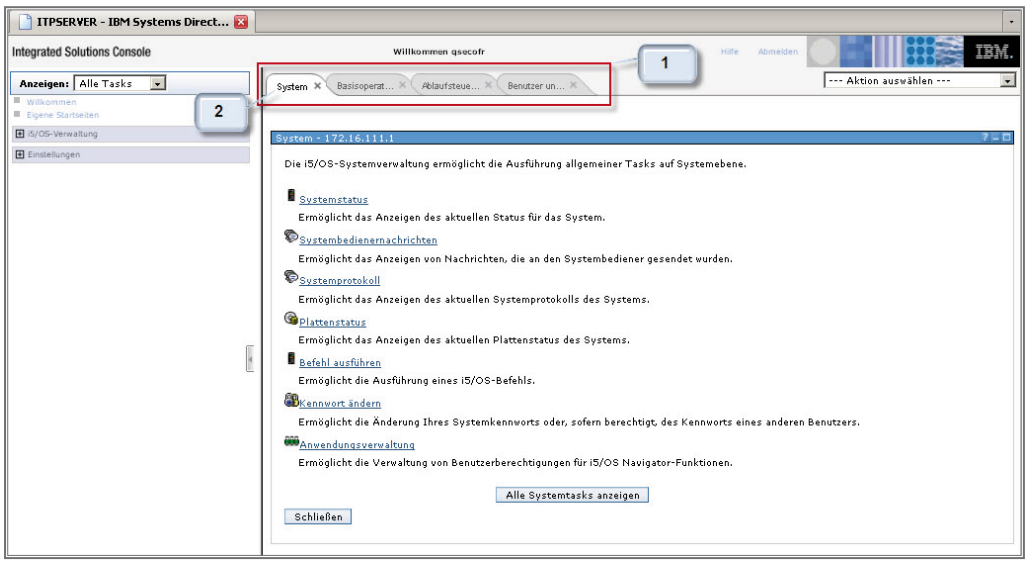
(2) Indem Sie einen oder mehrere Einträge in der Spalte Auswählen markieren und dann auf den Knopf Entfernen klicken, löschen Sie Einträge in Ihrer eigenen Startseite.

(3) Markieren Sie eine Seite in der Spalte Standard, so wird diese Seite sofort nach dem Start des SDNs angezeigt. Ich markiere hier den Eintrag System.

Ich bestätige den Dialog mit OK.

Jetzt melde ich mich vom SDN ab und starte ihn sofort wieder.

Jetzt sehen Sie die soeben individualisierte Startseite.



Die eigene Startseite

(1) Im Arbeitsbereich sehen Sie die einzelnen Register, der auf der Startseite befindlichen Funktionsseiten:

- System
- Basisoperationen
- Ablaufsteuerung
- Benutzer und Gruppen

(2) Weiterhin sehen Sie die sofort aktivierte Standardseite:

- System

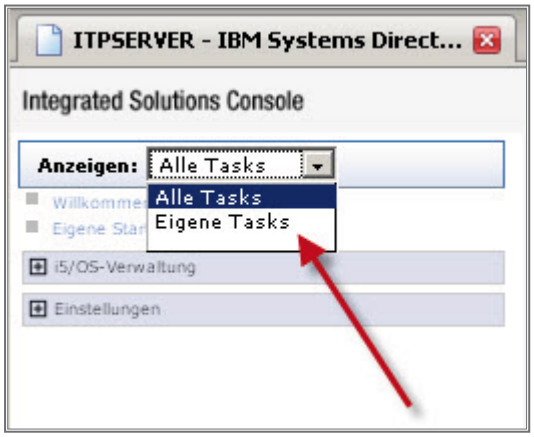
Die Individualisierungsfunktion Eigene Startseiten öffnet die bezeichneten Seiten im Arbeitsbereich, so dass wir schnell mit Hilfe der Register auf die einzelnen Seiten zugreifen können. Das Funktionsverzeichnis in der Navigatorleiste bleibt jedoch komplett bestehen.

7.7.3.1.1

Seite 4

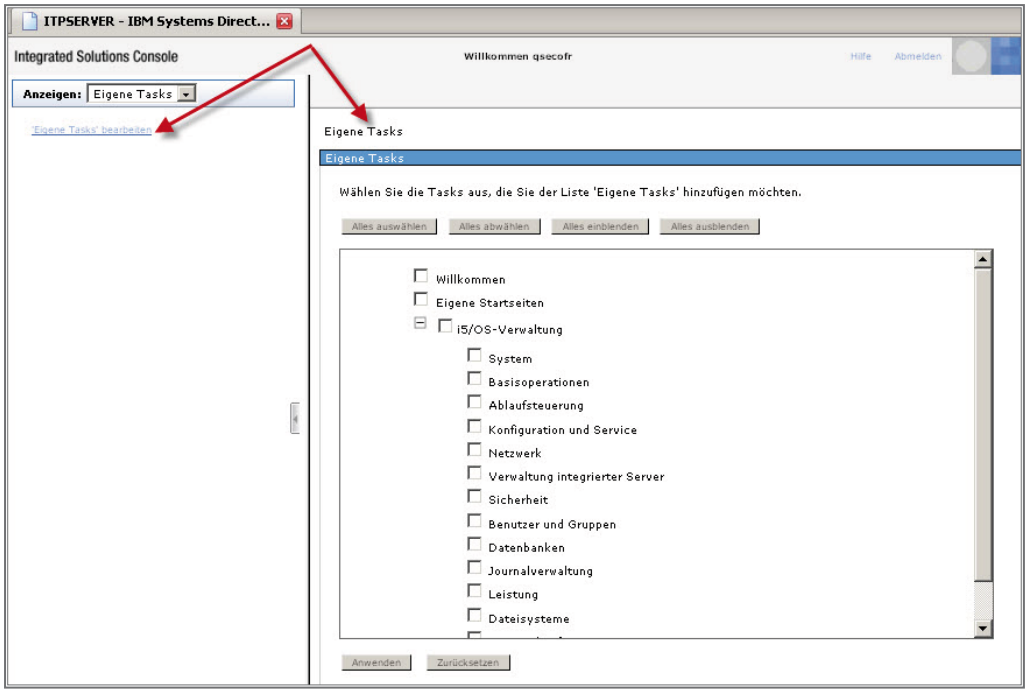
Eigene Tasks

Eine zweite Form der Individualisierung der SDN-Oberfläche besteht darin, das Funktionsverzeichnis in der Navigatorleiste zu bearbeiten.



Eigene Tasks

Um das Funktionsverzeichnis zu bearbeiten, markieren Sie in der Combobox Anzeigen, auf der linken Seite des SDN-Dialogs, den Eintrag Eigene Tasks.



Eigene Tasks bearbeiten

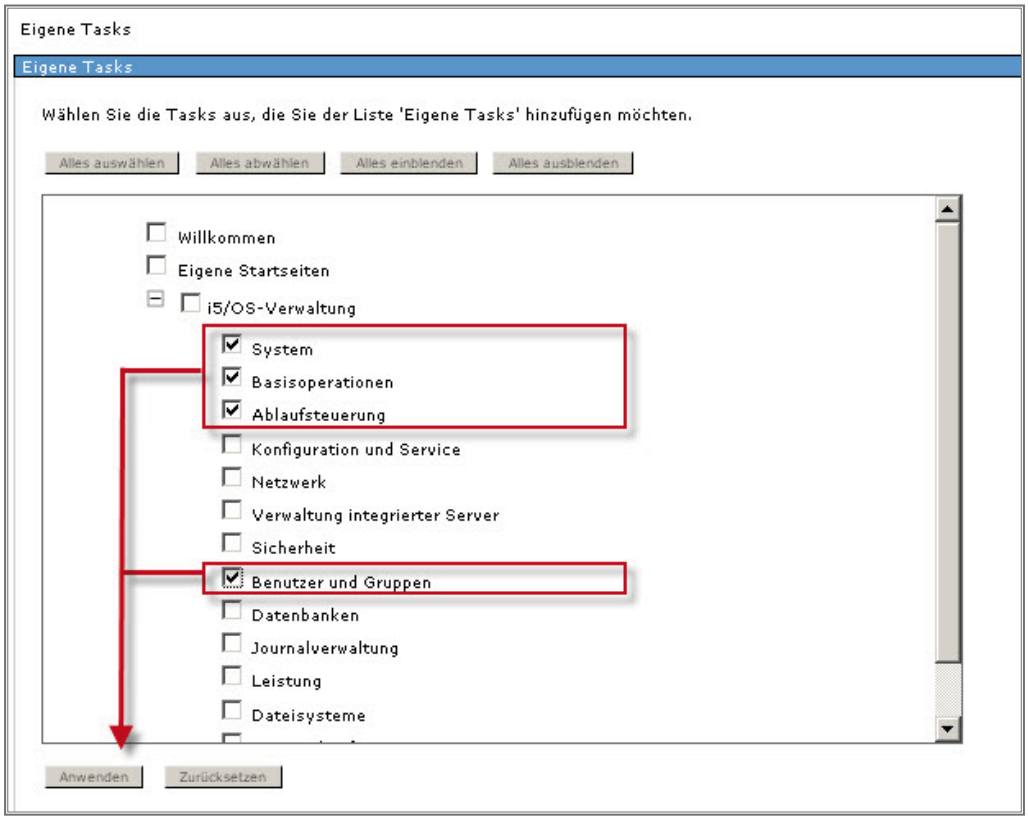
Sie sehen, dass jetzt in der Navigatorleiste nur ein einziger Eintrag vorhanden ist: Eigene Tasks bearbeiten. Ein Klick auf diesen Eintrag öffnet im Arbeitsbereich den Dialog Eigene Tasks.

Sie erkennen dort die Abbildung aller Tasks des SDN, so wie sie bis vor Kurzem noch in der Navigatorleiste angezeigt wurden.

7.7.3.1.2
Seite 2

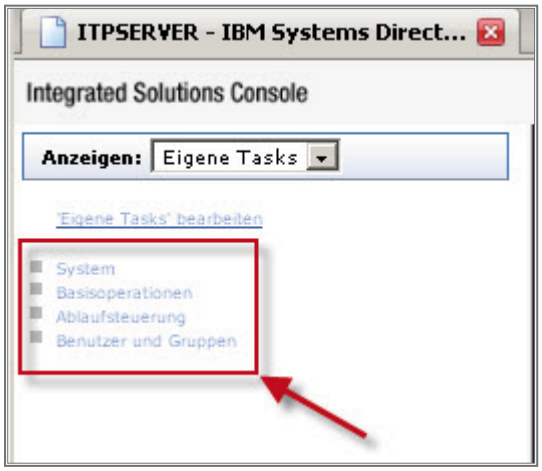
Ich gehe jetzt vom gleichen Beispiel aus wie in dem Abschnitt Eigene Startseite und wähle durch Markieren folgende Tasks aus:

- System
- Basisoperationen
- Ablaufsteuerung
- Benutzer und Gruppen



Eigene Tasks auswählen

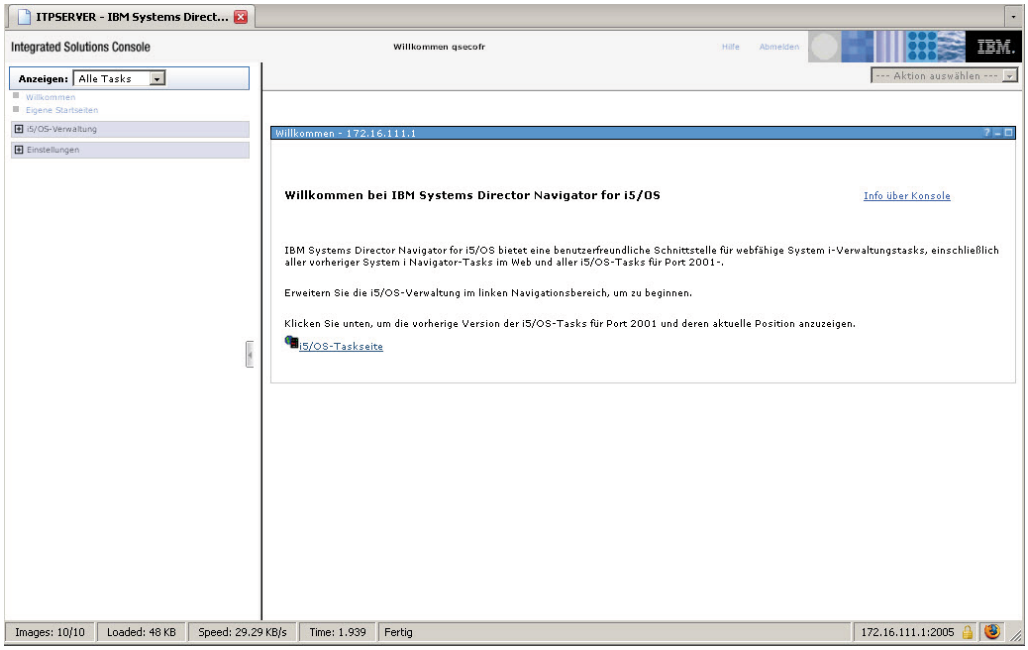
Ich markiere die entsprechenden Einträge und klicke auf den Knopf Anwenden.



Eigene Tasks

In der Navigatorleiste sehen Sie die soeben ausgewählten Tasks.

Ich melde mich vom SDN ab und sofort wieder an.

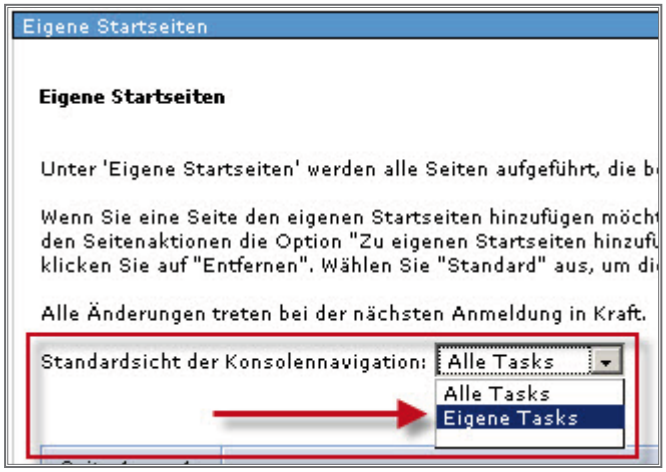


Enttäuschung!

Sie sehen, dass die eigenen Tasks nicht angezeigt werden. Die Ursache für dieses Verhalten liegt in der Combobox Anzeigen, denn diese zeigt den Eintrag Alle Tasks an und nicht den Eintrag Eigene Tasks.

7.7.3.1.2
Seite 4

Um das Verhalten der Combobox Anzeigen zu ändern, wechsle ich in die Task Eigene Startseite.

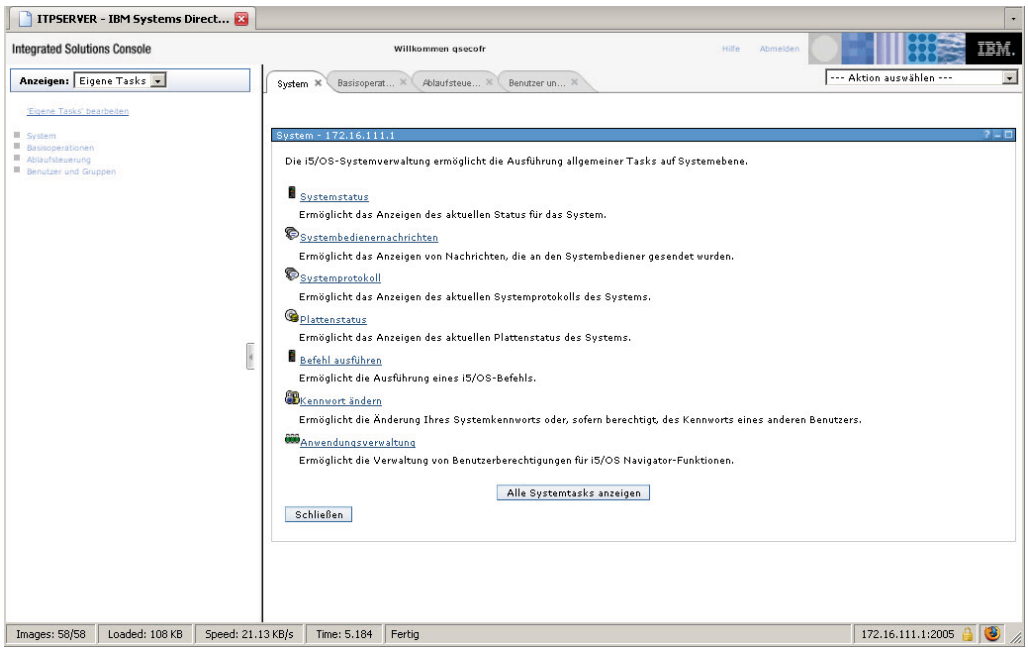


Konsolennavigation ändern

Im Dialog Eigene Startseiten ändere ich den Parameter Standardsicht der Konsolennavigation auf den Wert Eigene Tasks.

Ich bestätige diese Änderung mit OK.

Danach melde ich mich vom SDN ab und sofort wieder an.



Der individualisierte SDN

Benötigen Sie weitere Funktionen, so können Sie die Sicht Eigene Tasks sofort spontan umstellen, indem Sie in der Combobox Anzeigen den Eintrag Alle Tasks aktivieren.

Das alte Menü i5/OS-Task

Haben Sie bereits mit dem alten Web-Menü i5/OS-Task gearbeitet,

Das Menü i5/OS – Task

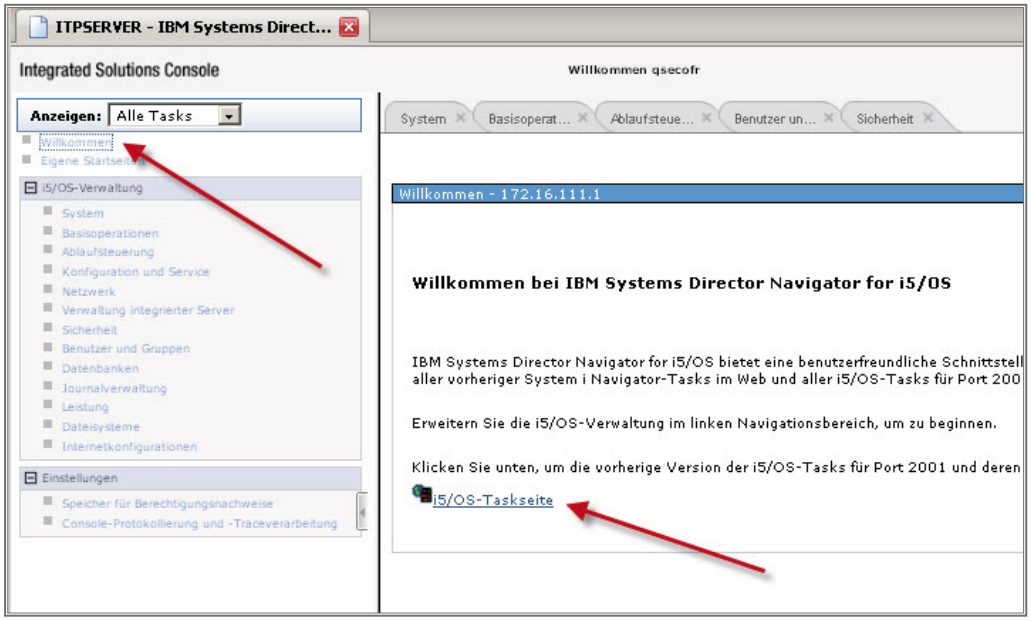
so werden Sie sicherlich bemerkt haben, dass dieses alte Menü über die gleiche URL aufgerufen wird wie der SDN.

http://172.16.111.1:2001

Mit Hilfe dieses Menüs konnten Sie u. a. Applikationsserver und Webserver konfigurieren, LDAP administrieren, IPP-Drucker erstellen etc. pp.

7.7.3.1.3
Seite 2

Dieses Menü existiert immer noch und wird jetzt direkt aus dem SDN heraus aufgerufen.

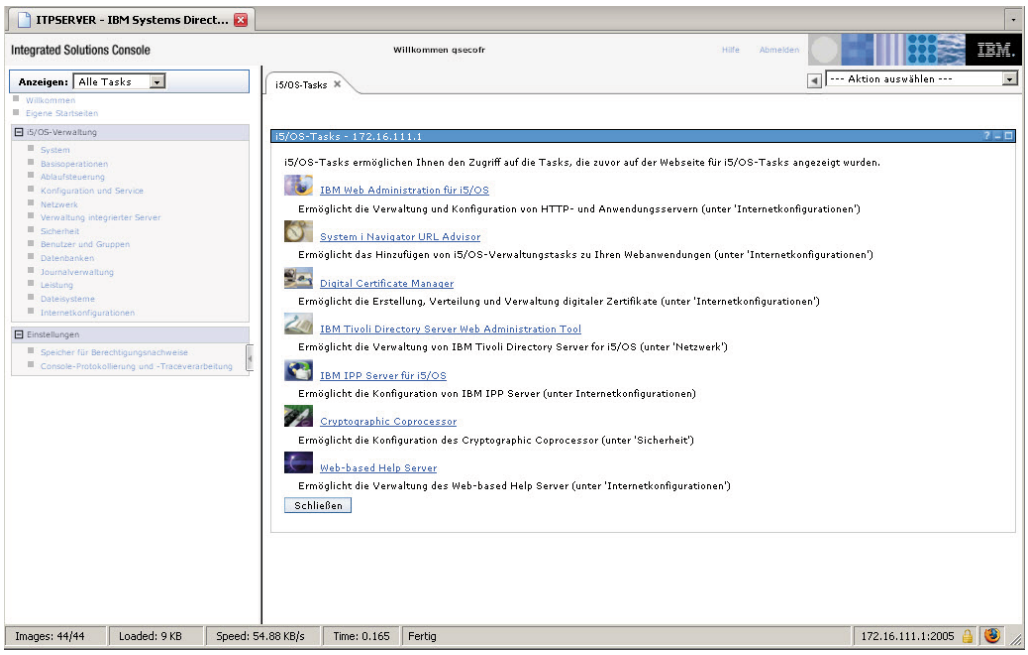


Das Menü i5/OS – Task finden

Öffnen Sie die Task Willkommen.

In der Seite Willkommen im Arbeitsbereich finden Sie sodann den Link i5/OS-Taskleiste.

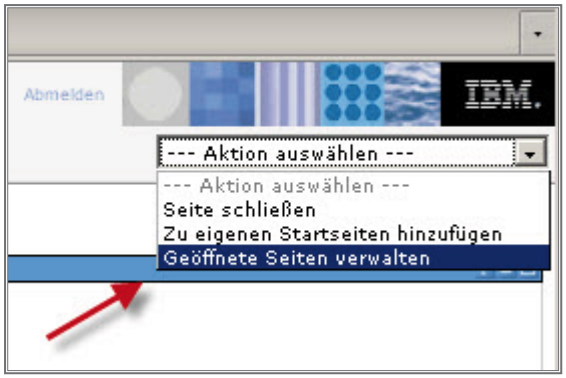
Ein Klick auf diesen Link bringt Sie sodann in das bezeichnete Menü.



Das Menü i5/OS – Task im SDN

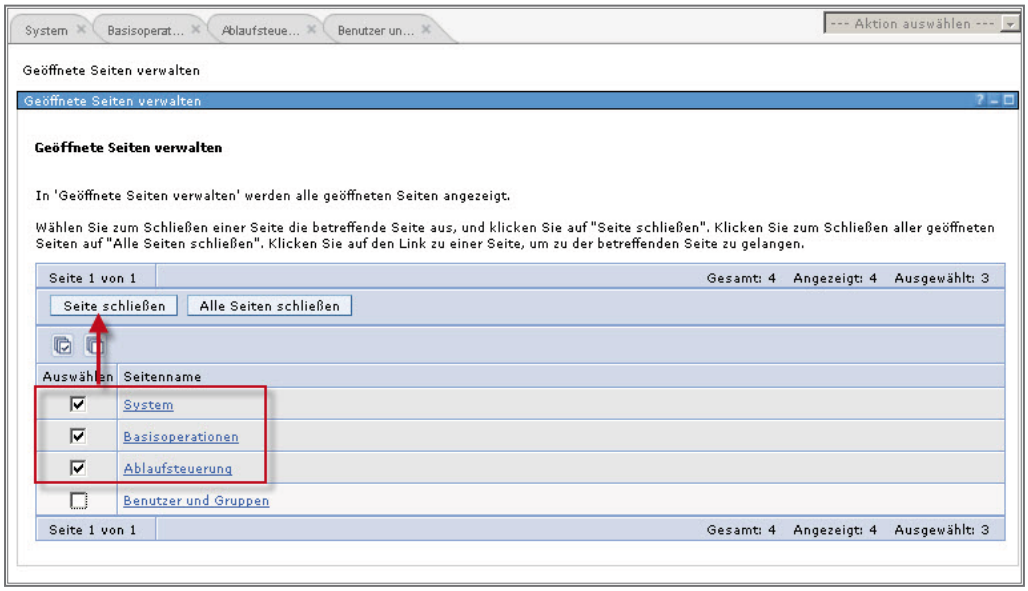
Geöffnete Seiten verwalten

Haben Sie im Laufe einer Arbeitssitzung eine ganze Reihe von Seiten geöffnet, so besteht oftmals der Wunsch alle oder mehrere Seiten „auf einen Schlag“ aus dem Arbeitsbereich zu entfernen.



Task Geöffnete Seiten verwalten aufrufen

Um diese Aufgabe zu erledigen, markieren Sie in der Combobox Aktion Auswählen den Eintrag Geöffnete Seiten verwalten.



Geöffnete Seiten verwalten

Markieren Sie die zu schließenden Seiten.

Klicken Sie auf den Knopf Seiten schließen.

Um alle geöffneten Seiten zu schließen, klicken Sie einfach auf den Knopf Alle Seiten schließen, ohne eine Seite zu markieren.

7.7.3.1.4

Seite 2

i5/OS – Verwaltung

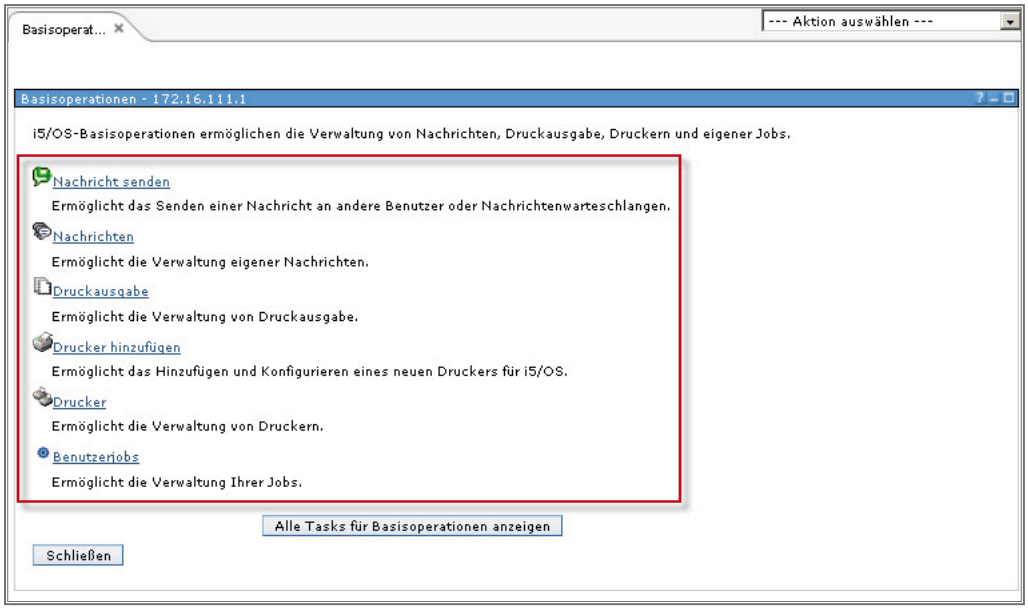
In diesem Abschnitt werde ich Ihnen die derzeit zur Verfügung stehenden Funktionen innerhalb des SDNs zeigen. Es geht hierbei jedoch nicht um die detaillierte Beschreibung von Administrationsaufgaben, sondern mehr um die Darstellung welche Administrationsaufgaben Sie mit Hilfe des SDNs realisieren können.

19. Ergänzung 6/2009

7.7.4
Seite 2

Basisoperationen

Der Bereich der Basisoperationen umfasst den bedienernahen Teil des SDN. Hier finden wir Systembedienungs-Funktionen, die tagtäglich von Administratoren ausgeführt werden.



Basisoperationen

Die Basisoperationen beinhalten folgende Funktionen

- Nachrichten senden
- Nachrichten – Mit Nachrichten arbeiten
- Druckausgabe – Mit Druckausgaben arbeiten
- Drucker hinzufügen
- Drucker – Mit Druckern arbeiten
- Benutzerjobs – Mit Benutzerjobs arbeiten

Die in der Folge gezeigten Funktionen können im Prinzip als Alternativen für die i/OS CL-Befehle SNDMSG (Send Messages), SNDBRKMSG (Send Break Message), WRKMSG (Work with Messages), WRKSPL (Work with Spoolfiles), CRTDEVPRT (Create Printing Device), WRKWTR (Work with Writers) und WRKUSRJOB (Work with User Jobs) angesehen werden.

7.7.4.1

Seite 2

Nachrichten senden

Sie werden ganz richtig vermuten, dass sich hinter dieser Funktion eine Art graphischer SNDMSG (Send Message)-Befehl verbirgt. Aber lassen Sie uns an dieser Stelle ein wenig über die Wirkungsweise der Nachrichtenversendungsbefehle des i/OS nachdenken.

Im System existieren zwei Nachrichtenversendungsbefehle, die dem Benutzerumfeld zuzuordnen sind:

- SNDMSG (Send Message)
- SNDBRKMSG (Send Breakmessage)

Beide Befehle haben Eigenarten, die ihren Einsatz einschränken können. Der Befehl SNDMSG sendet eine Benutzernachricht an einen Benutzer, dabei ist es unerheblich, an welchem Bildschirm sich der Benutzer angemeldet hat. Die Nachricht erreicht die Nachrichtenwarteschlange des Benutzerprofils auf alle Fälle.

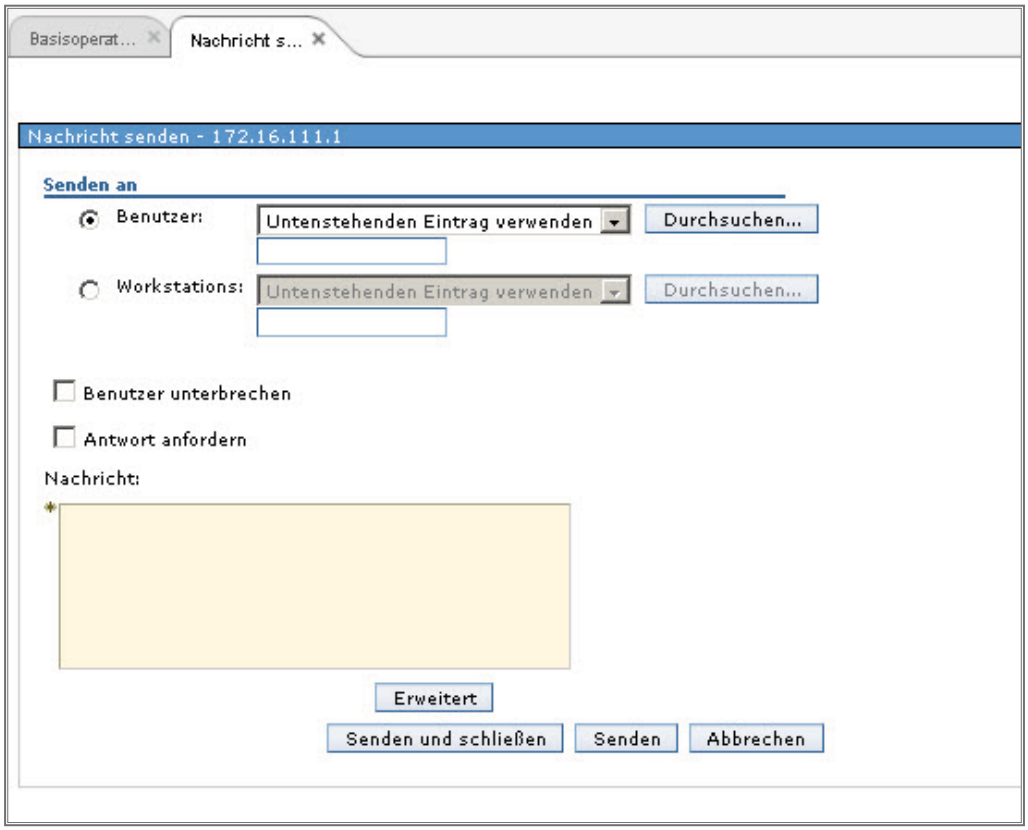
Jedoch hat die Sache einen Haken. Soll der adressierte Benutzer die Nachricht unbedingt lesen und möglicherweise darauf antworten, so kann der Nachrichtenversender dies nicht garantieren, denn wie die Nachricht vom Empfänger wahrgenommen wird, hängt von der Einstellung der Nachrichtenwarteschlange des Empfängerbenutzerprofils ab. Wenn die Nachrichtenwarteschlange des Empfängerprofils sich nicht im Modus *BREAK befindet, so führt das Eintreffen der Nachricht bestenfalls zu einem Signalton (Modus *Notify) oder wird schlimmstenfalls überhaupt nicht wahrgenommen (Modus *Hold).

Soll ein Anwender des Systems eine Nachricht unbedingt lesen, so bietet sich als nächster Systembefehl der Befehl SNDBRKMSG an. Wie der Name schon sagt, sendet dieser Befehl eine Durchbruchnachricht. Das heißt, dass der Bildschirm des Empfängers auf alle Fälle die Nachricht anzeigt.

Aber auch hier gibt es einen Haken. Im Standard schickt der Befehl SNDBRKMSG seine Nachricht an die Nachrichtenwarteschlange eines Bildschirms – nicht an einen Benutzer. Sende ich also eine Nachricht an einen Bildschirm, so kann es mir passieren, dass der Bildschirm zwar die Nachricht sofort beim Eintreffen anzeigt. Ob aber der gewünschte Benutzer wirklich vor diesem Bildschirm sitzt, das kann ich nicht garantieren.

An diesem Punkt setzt die Funktion Nachrichten senden des SDN ein.

7.7.4.1.1
Seite 2



Nachrichten senden

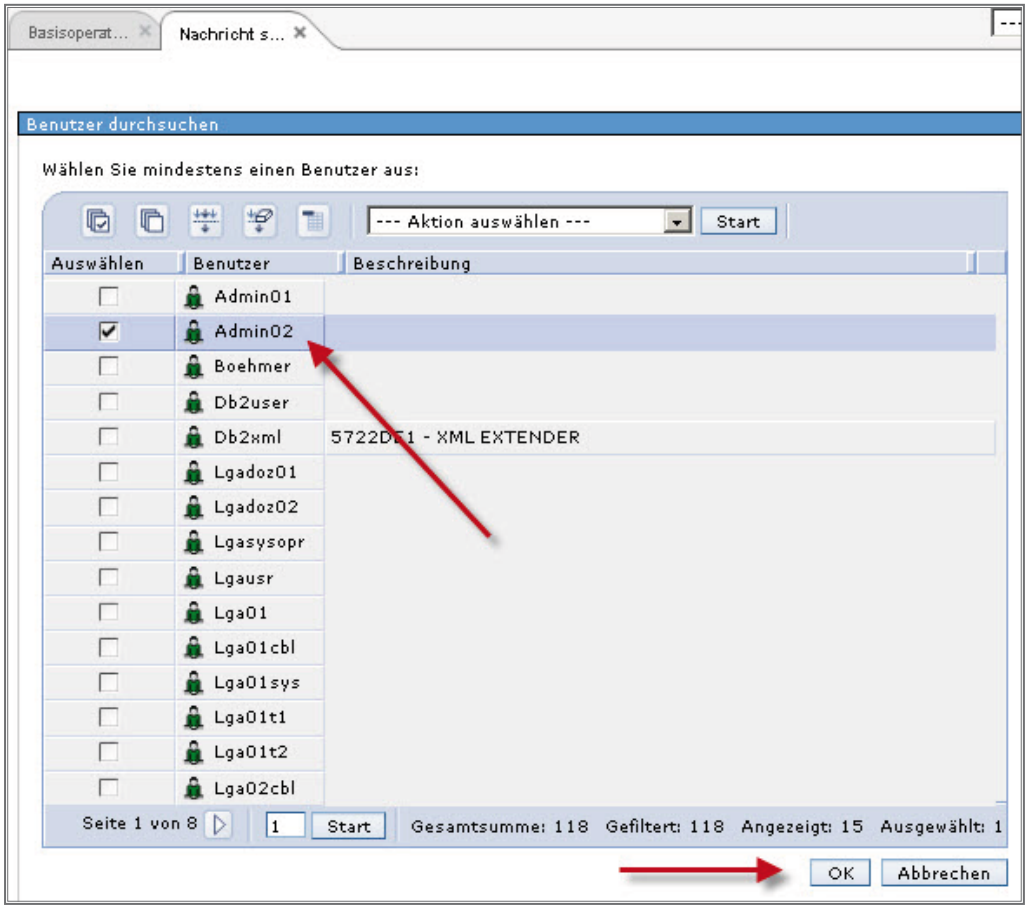
Folgende Parameter steuern diesen Dialog:

Benutzer:
Beabsichtigen Sie die Nachricht an Benutzer zu senden, so können Sie den Namen eines bekannten Benutzerprofils in das Dialogfeld Benutzer direkt eintragen. Darüberhinaus stehen Ihnen über das Pulldownmenü in diesem Dialogfeld vordefinierte Benutzer bzw. Gruppen zur Auswahl.

Vordefinierte Benutzer und Gruppen

Wählen Sie an dieser Stelle aus, ob Sie eine Nachricht an alle Benutzer senden wollen oder nur an die derzeit am System angemeldeten Benutzer oder an den Systembediener (QSYSOPR).

Alternativ ist es auch möglich den Namen eines Systembenutzers mit dem Durchsuchen Dialog ausfindig zu machen.



Nach Benutzern suchen

Sie sehen hier den Benutzer ADMIN02, dem eine Nachricht gesendet werden soll.

Datenstationen:
Neben der Funktion der Benutzernachrichten können dann aber auch Nachrichten an Datenstationsnachrichten-Warteschlangen abgesetzt werden.

Datenstationsnachrichten

Auch hier ist es möglich die Nachricht direkt an einen Bildschirm zu senden, sofern der Name der Bildschirmeinheit bekannt ist. Als vordefinierte Gruppenadresse steht hier nur die Möglichkeit an, allen Datenstationen eine Nachricht zu senden.

7.7.4.1.1
Seite 4

Auch an dieser Stelle im Dialog wird wiederum die Möglichkeit gegeben, einen bestimmten Bildschirm per Durchsuchen zu ermitteln.

Achtung!

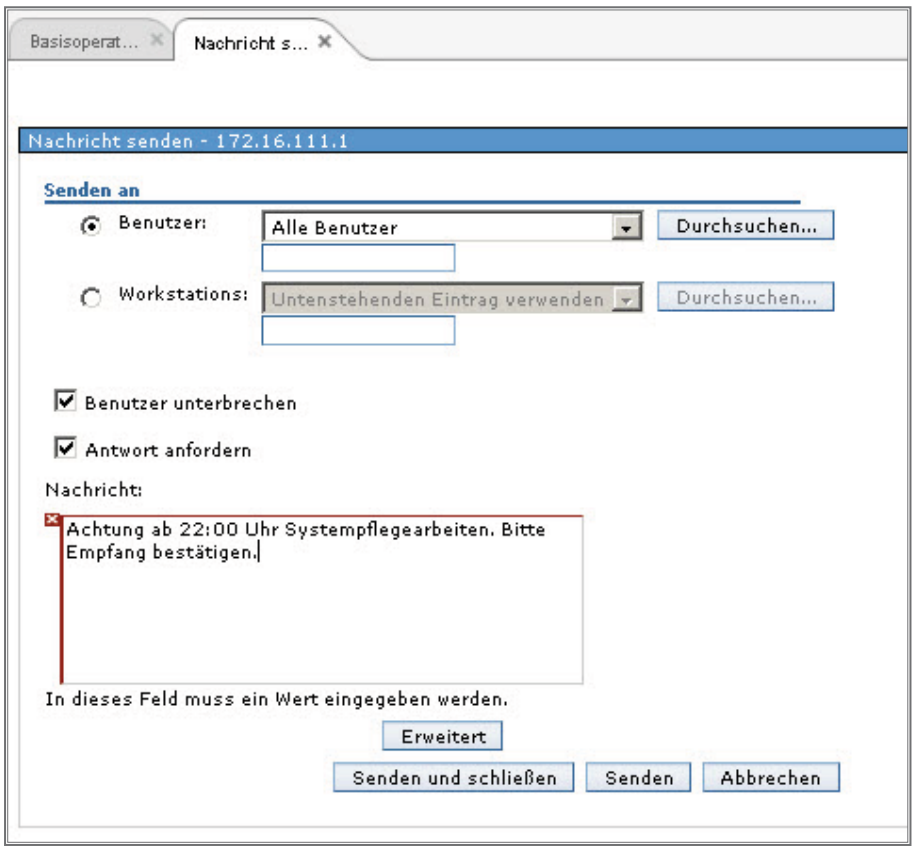
Sowohl der Durchsuchen Dialog für Benutzer und für Datenstationen haben einen Nachteil. Diese Funktionen können nicht zwischen aktiven Benutzern/Datenstationen und „Leichen" unterscheiden. Nehmen wir als Beispiel die Suchfunktion für Datenstationen: Ich als Administrator dieses Systems weiß, dass die beiden Datenstationen DOZENTM1 und ED physisch schon einige Zeit nicht mehr Bestandteil meines Systems sind. Diese Datenstationseinheiten wurden bei Versuchen, System i Access für Windows-Bildschirme unter TCP/IP zu konfigurieren, erstellt. Danach wurden die Bildschirmeinheiten umbenannt. Der Administrator hat jedoch vergessen diese nicht mehr benötigten Einheiten zu löschen (Schimpf und Schande über den Administrator...). Sollten wir also eine Nachricht an diese Bildschirme senden, so landen sie im Nirvana.

Benutzer unterbrechen:
Dieser Parameter gilt sowohl für Benutzernachrichten als auch für Datenstationsnachrichten. Diese Funktionalität stellt eine erhebliche Erweiterung der Funktion der Systembefehle der iSeries dar.

Anwort anfordern:
Benötigen Sie eine Antwort auf Ihre Nachricht, so markieren Sie diesen Parameter durch einen Haken.

Lassen Sie uns ein kleines Beispiel entwickeln. Sie beabsichtigen um 22 Uhr mit Systempflegearbeiten zu beginnen. Darüber wollen Sie alle Benutzer im System informieren. Sie wollen sicherstellen, dass alle Benutzer diese Nachricht sofort bei Eintreffen lesen. Außerdem erwarten Sie von jedem Benutzer, dass er Ihnen eine Antwort als Bestätigung, dass er die Nachricht gelesen hat, sendet.

Folgende Eingaben müssten realisiert werden:



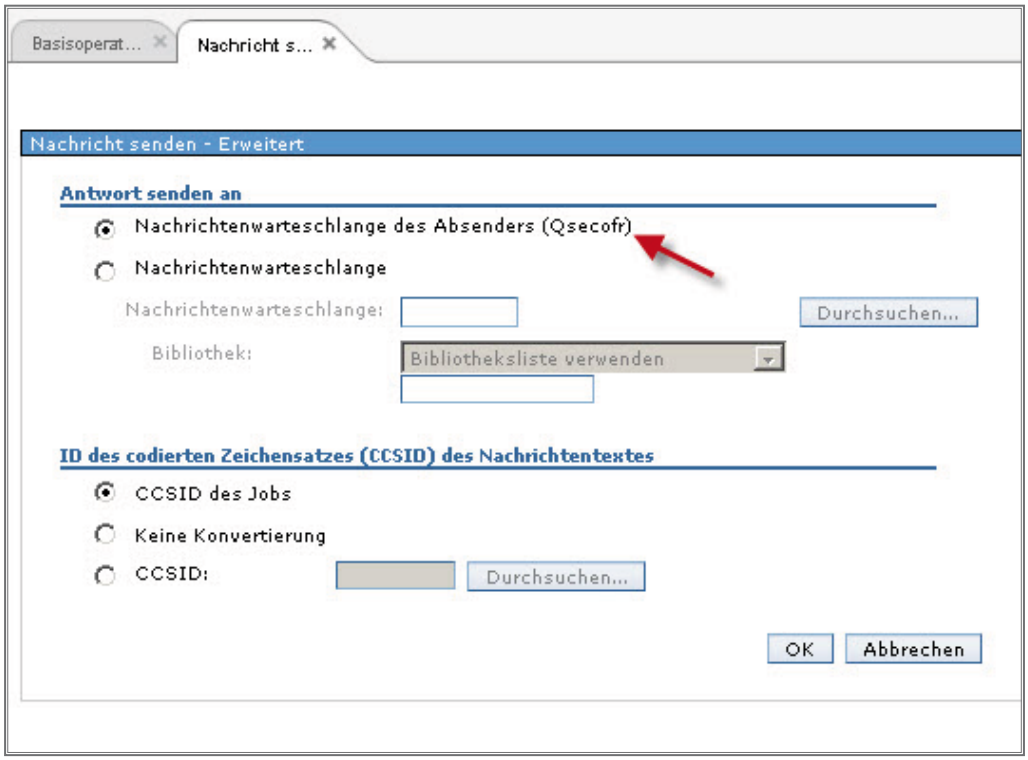
Durchbruchnachricht an alle Benutzer

Sehen Sie sich die Parameter des Dialogs Nachrichten senden an:

- Benutzer: Alle Benutzer
- Benutzer unterbrechen: aktiviert
- Antwort anfordern: aktiviert
- Nachricht: Der zu sendende Nachrichtentext

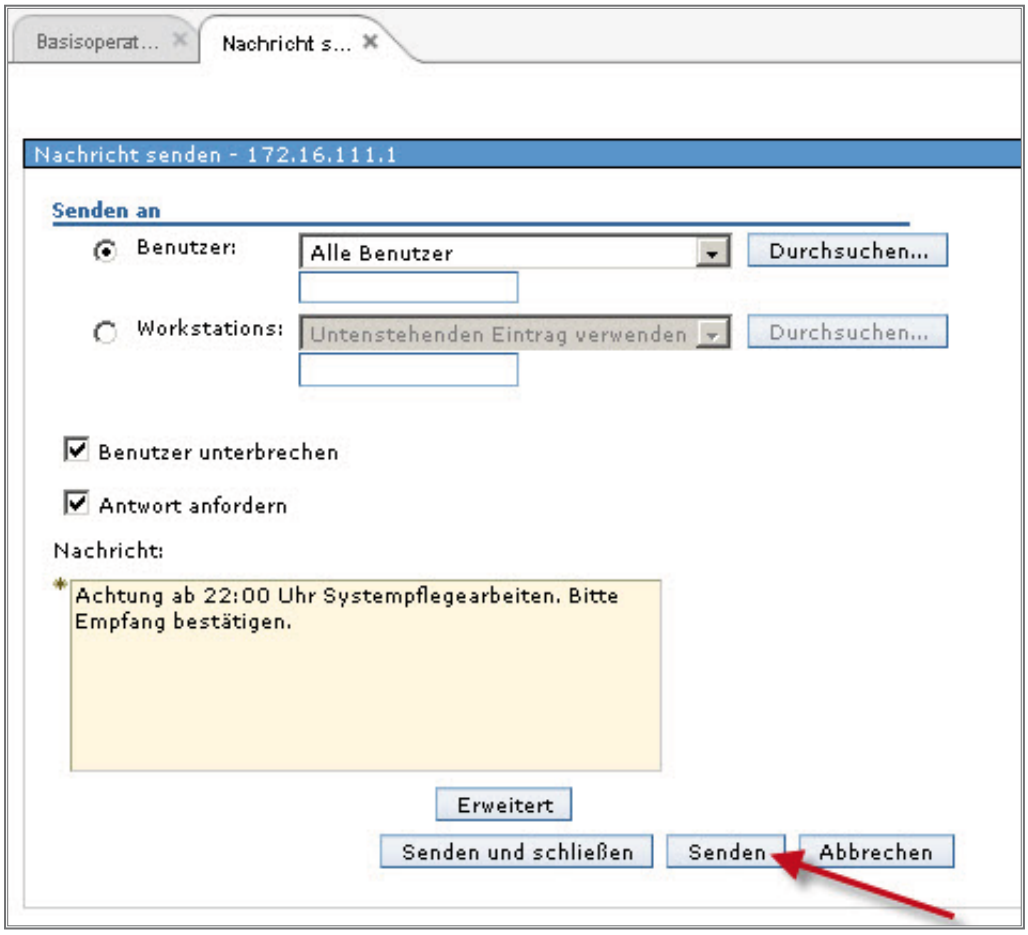
Das würde eigentlich reichen. Sie haben jetzt noch die Möglichkeit durch Drücken des Buttons Erweitert zu definieren, an welchen Benutzer die Bestätigungsnachrichten zu senden sind.

7.7.4.1.1
Seite 6



Die erweiterten Nachrichtenattribute

Sie sehen hier, dass alle Antworten auf die gesendete Nachricht an das Benutzerprofil QSECOFR gesendet werden sollen.



Nachricht senden

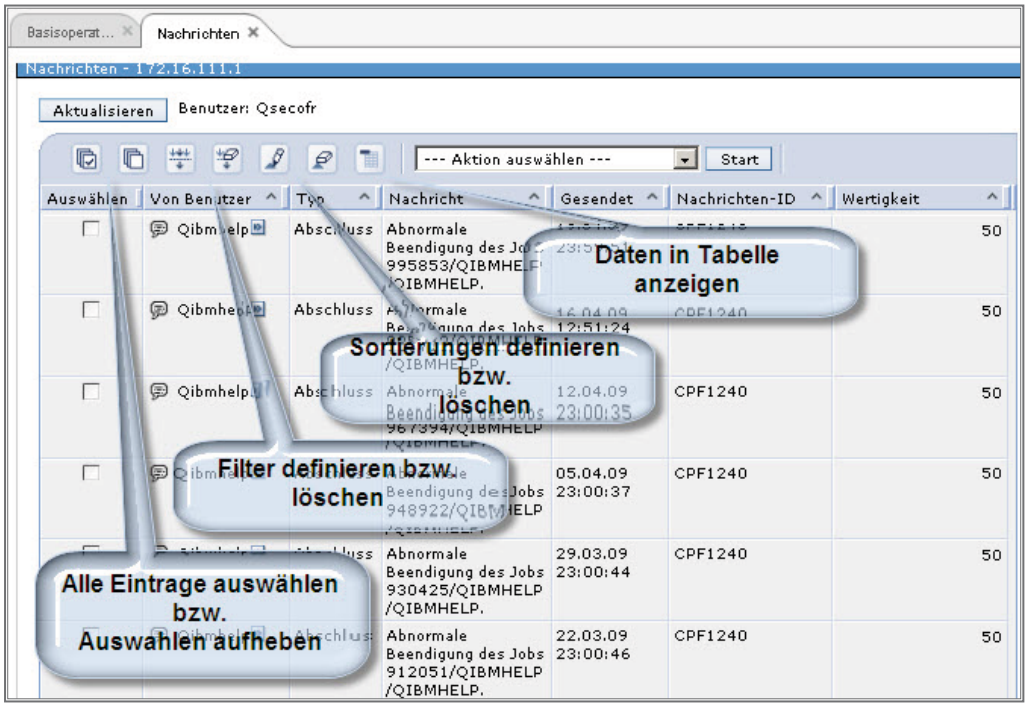
Nachdem der Erweitert-Dialog mit OK bestätigt und verlassen wurde, bleibt mir nur noch die Möglichkeit, die Nachricht zu versenden.

7.7.4.1.1

Seite 8

Nachrichten

Folgen Sie dem Link Nachrichten, so sehen Sie im Arbeitsbereich die Nachrichtenanzeige.



Nachrichtenanzeige

Wir sehen hier, dass für den Benutzer QSECOFR eine Reihe von Nachrichten eingegangen sind.

In der Anzeige sehen Sie einige typische Bedienungselemente, die Sie später in fast allen tabellarischen Anzeigen wiedertreffen:

Zwei Schaltflächen, mit denen Sie alle gezeigten Einträge für eine weitere Bearbeitung markieren, bzw. deren Markierung entfernen.

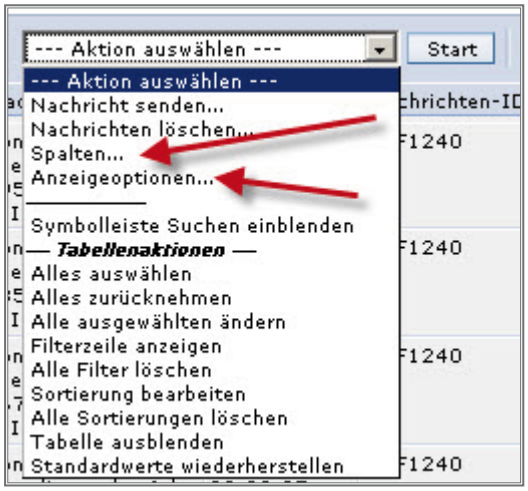
Eine weitere Schaltfläche, mit der Sie die sogenannte Filterfunktion aktivieren, so dass Sie die angezeigte Datenmenge reduzieren. Mit dem gleichen Schalter deaktivieren Sie diese Funktion.

Des Weiteren finden Sie zwei Knöpfe, mit deren Hilfe Sie die angezeigten Daten sortieren können, bzw. die Sortierung aufheben.

Ein letzter Knopf, mit dem Sie die angezeigten Daten aus dem Dialog entfernen.

7.7.4.1.2
Seite 2

Eine Combo Box, um mit ausgewählten Einträgen zu arbeiten.

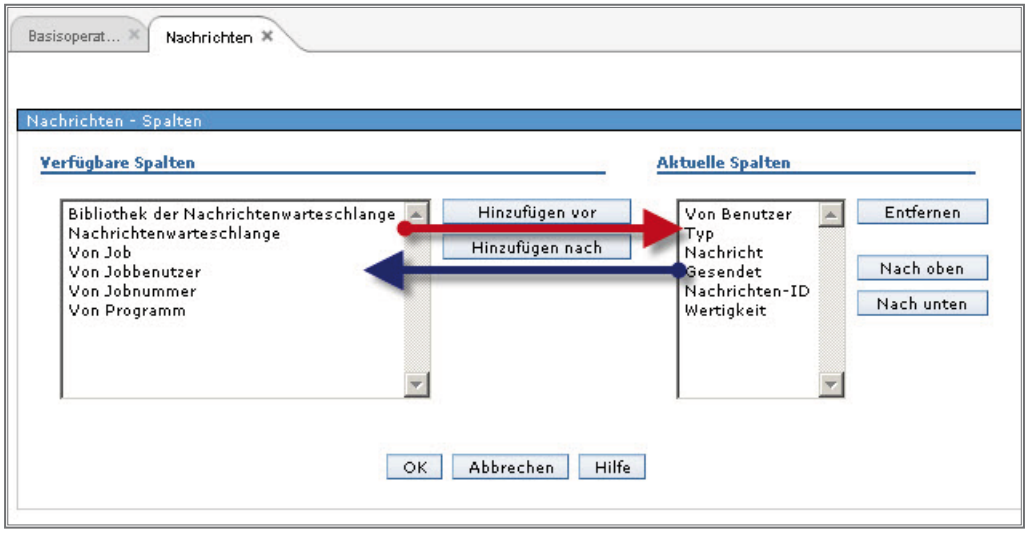


Aktionen

Die meisten der auszulösenden Funktionen sprechen für sich, so dass ich auf eine weitere Erläuterung verzichte. Ich beschränke mich an dieser Stelle lediglich auf die Funktionen

- Spalten… und
- Anzeigeoptionen… .

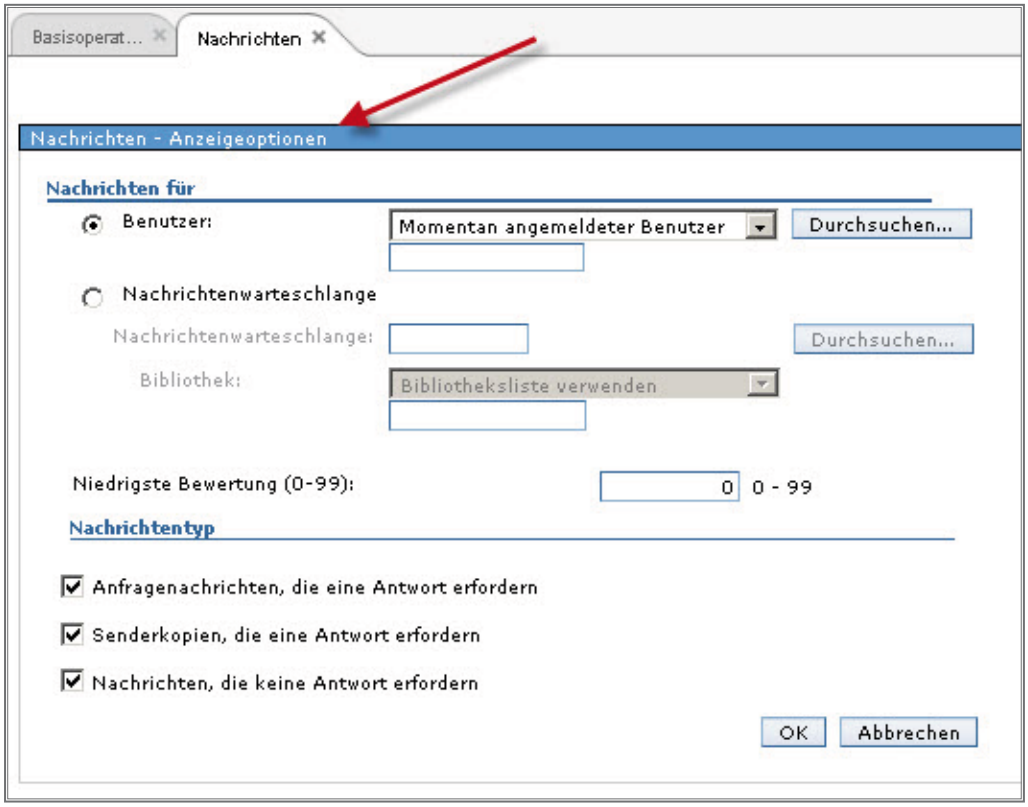
Nachdem ich Spalten… ausgewählt und auf Start geklickt habe, öffnet sich folgender Dialog im Arbeitsbereich:



Anzeige definieren

Ziehen Sie hier die derzeit nicht angezeigten verfügbaren Spalten mit Hinzufügen vor bzw. Hinzufügen nach in die aktuelle Anzeige.

Nachdem ich Anzeigeoptionen… ausgewählt und auf Start geklickt habe, öffnet sich folgender Dialog im Arbeitsbereich:



Anzeigeoptionen

Der oben dargestellte Dialog steuert die generelle Wirkungsweise der Funktion Nachrichten. Die dargestellten Parameter sprechen für sich und entsprechen weitgehend den Parametern des i/OS-Befehls DSPMSG, der Ihnen sicherlich bekannt ist.

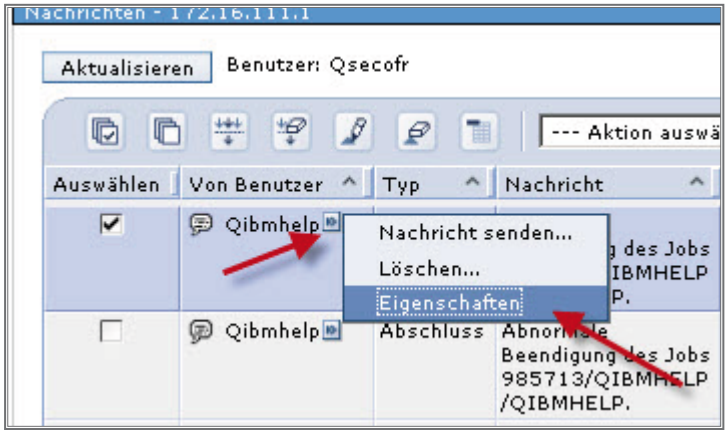
Wir kehren in die Nachrichtenanzeige zurück.

Die gezeigten Nachrichten sind vom Typ Abschluss – also Beendigungsnachrichten. Am Nachrichtentext erkennen wir, dass sich die Nachrichten sowohl auf normal als auch abnormal beendete Jobs im System beziehen.

Der hier teilweise sichtbare Nachrichtentext wird als Text der ersten Ebene (first level) bezeichnet. Aber als erfahrener Anwender kennen Sie diese Anzeige ja aus dem Befehl DSPMSG (Display Message).

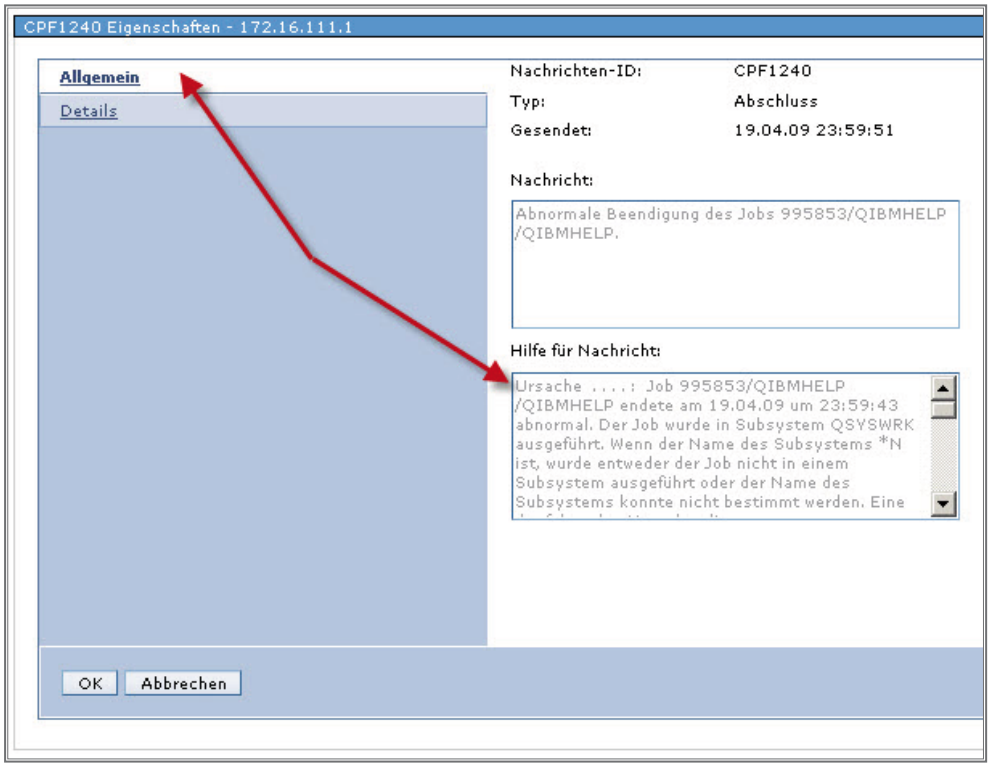
7.7.4.1.2
Seite 4

In der Spalte Benutzer finden Sie hinter dem Benutzernamen eine Schaltfläche. Klicken Sie diese an, so erscheint ein Popup-Menü.



Eigenschaften

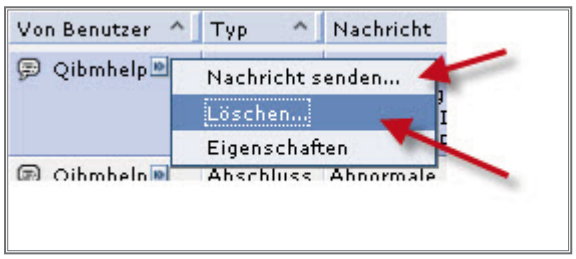
Klicken Sie auf den Eintrag Eigenschaften, so erscheint der komplette Nachrichtentext.



Allgemeine Eigenschaften

Dieser beinhaltet dann auch den Nachrichtentext der zweiten Ebene.

Das Nachrichten-bezogene Popup-Menü enthält zwei weitere Einträge:

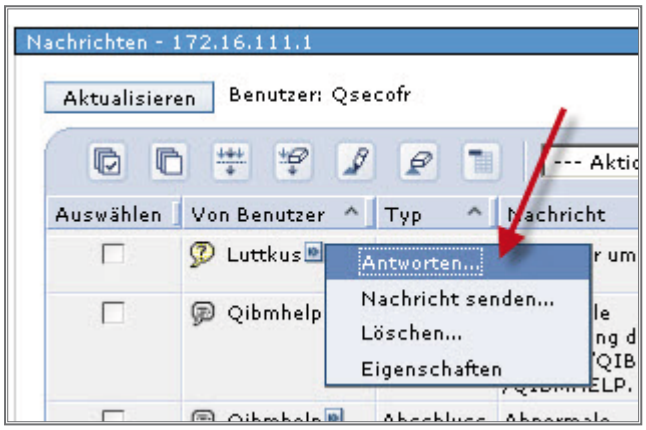


Allgemeine Menüeinträge 1

Nachrichten senden: Diese Funktion ist lediglich ein weiterer Link auf die bereits in Kapitel 31.4.1.1 besprochene Task.

Löschen: Mit dieser Funktion löschen Sie die markierte Nachricht.

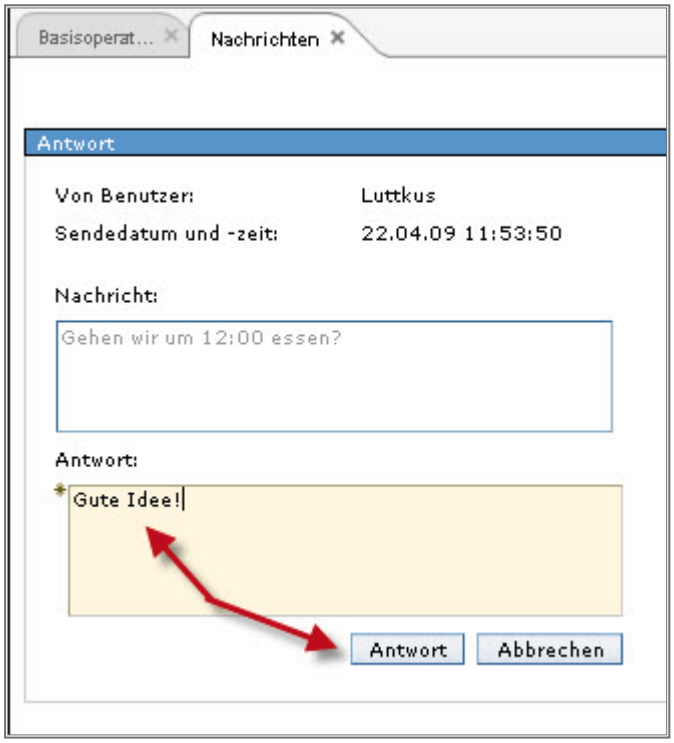
Erhalte ich eine Anfragenachricht, auf die eine Antwort erwartet wird, so erweitert sich das Popup-Menü.



Menüeinträge 2

7.7.4.1.2
Seite 6

Sie erhalten jetzt die Möglichkeit dieser Nachricht zu antworten.

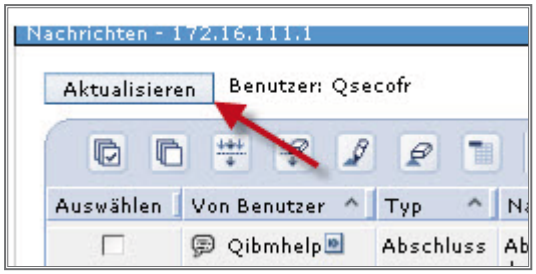


Anfragenachricht beantworten

Um die Betrachtung dieses Funktionsbereichs abzuschließen, möchte ich Ihnen noch einen Hinweis geben. Alle Anzeigen des SDN sind insgesamt als statisch zu betrachten. Anzeigen werden zu folgenden Zeiten aktualisiert:

Wenn wir eine Detailbearbeitung eines Funktionsbereichs öffnen.

Wenn wir die Aktualisieren-Funktion-Ansicht auslösen.

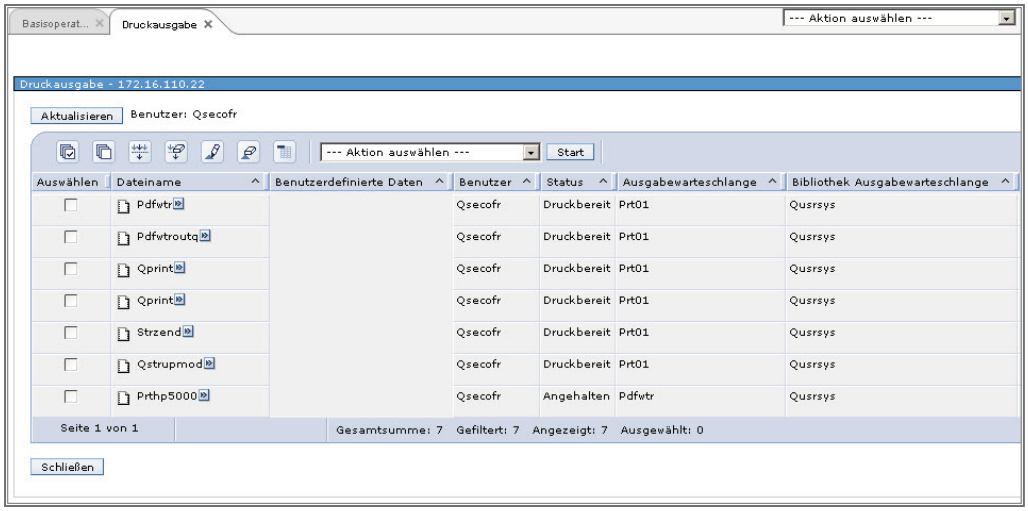


Anzeige aktualisieren

Gerade die Statik kann aber bei der Nachrichtenfunktion zu Nachteilen führen. Stellen Sie sich einmal vor, Sie bekommen die wichtige Nachricht von einem Benutzer, dass er mit der Bedienung seines Druckers nicht zurechtkommt. Weil Sie die Aktualisierungsfunktion nicht auslösen, wird diese wichtige Nachricht nicht angezeigt, so dass Sie dem Benutzer die erwartete Hilfe nicht gewähren können. Ein Konflikt ist vorprogrammiert.

Druckausgabe

Wenden wir uns jetzt dem Funktionsbereich Druckausgaben zu.



Mit Druckausgaben arbeiten

Um mit einer Druckausgabe zu arbeiten,

Druckausgabe bearbeiten

klicken Sie auf das oben bezeichnete Symbol.

Spooldateien Kontextmenü

Ich befinde mich jetzt in der Anzeige der Druckausgaben für dieses System, habe eine Spooldatei markiert und das Kontextmenü geöffnet.

7.7.4.1.3

Seite 2

Folgende Menüpunkte sind aktiviert:

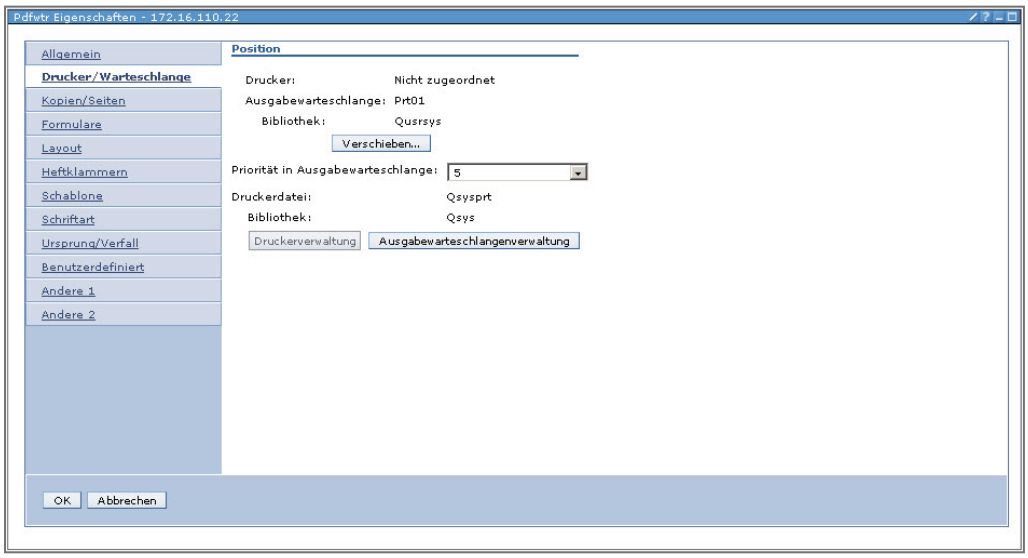
- Öffnen
- Anhalten
- Als nächstes drucken
- Senden
- Verschieben
- Löschen
- Umsetzen in PDF
- Exportieren
- Erweitert
- AFP-Viewer installieren
- Eigenschaften

Eigenschaften

7.7.4.1.3.1
Seite 1

Ich beginne mit dem Eigenschaften-Dialog. In diesem Dialog können die Attribute einer Spooldatei geändert werden. Auf dem System steht zu diesem Zweck der Befehl CHGSPLFA (Change Spoolfile Attributes) zur Verfügung.

Ich möchte auch hier nur schwerpunktbezogen einige Register betrachten, deren Parameter im Systembetrieb öfter benötigt werden. Ich öffne zunächst das Register Drucker/Warteschlange.



Register Drucker/Warteschlange

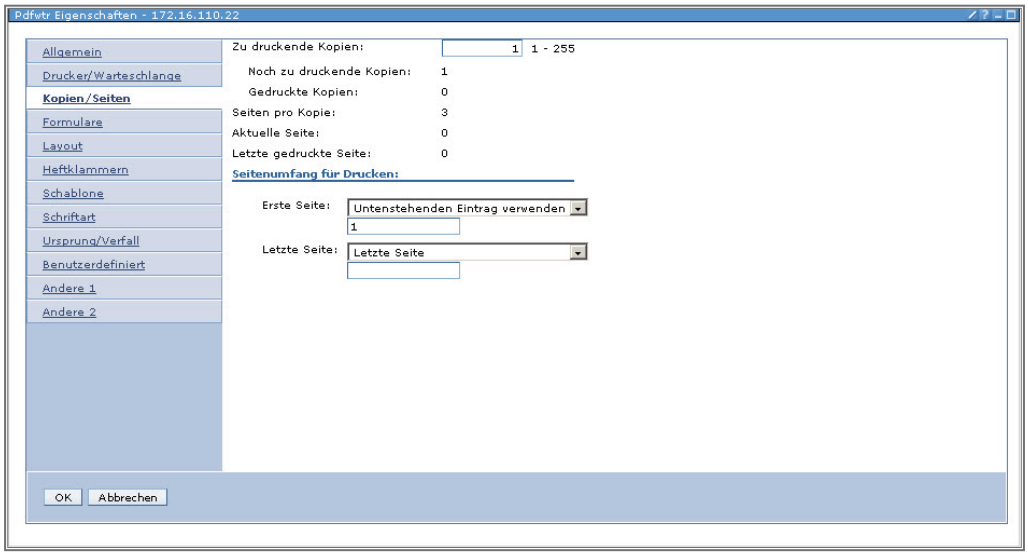
Folgende Parameter werden hier eingestellt:

- Verschieben: Ordnen Sie mit Hilfe dieses Dialogfeldes einen anderen Drucker zu. Mittels des Dialogs Verschieben... können Sie interaktiv einen Systemdrucker auswählen.
- Warteschlangeverwaltung: Diese Auswahl entspricht einem WRKOUTQ für die Druckdatei.
- Priorität für Ausgabewarteschlange: Hier verändern Sie die Priorität der Ausgabe und können damit sowohl einen Druck in der Warteschlange vorziehen als auch in der Rangfolge weiter nach hinten stellen

Im Register Kopien/Seiten stellen Sie die Anzahl der Kopien ein, die Sie für eine bestimmte Ausgabe erstellt wissen wollen. Darüberhinaus können Sie Teilbereiche einer Druckdatei auswählen, um diese dann auszugeben.

Im Ausgabeparameter Noch zu druckende Kopien wird Ihnen der Fortschritt des Druckens angezeigt. Beabsichtigen Sie z. B. fünf Kopien einer Ausgabe zu drucken, und haben bereits zwei Exemplare gedruckt, so werden Sie in diesem Feld den Wert drei (3) sehen. Das ist die Anzahl der noch zum Druck anstehenden Ausgabeexemplare.

7.7.4.1.3.1 Seite 2

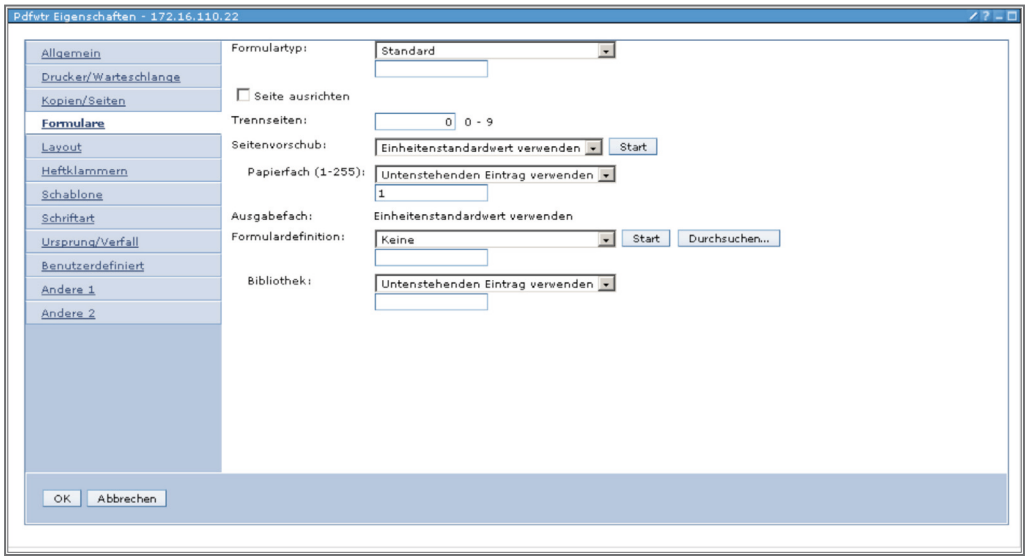


Register Kopien/Seiten

Weiterhin können Sie hier einstellen, ob alle Seiten gedruckt werden sollen (das ist die Default-Einstellung) oder ob nur bestimmte Seitenbereiche gedruckt werden sollen.

Um diese Einstellungen zu realisieren, nutzen Sie die Parameter Erste Seite und Letzte Seite.

Das Register Formulare beschließt die Betrachtung der Spooldateiattribute.



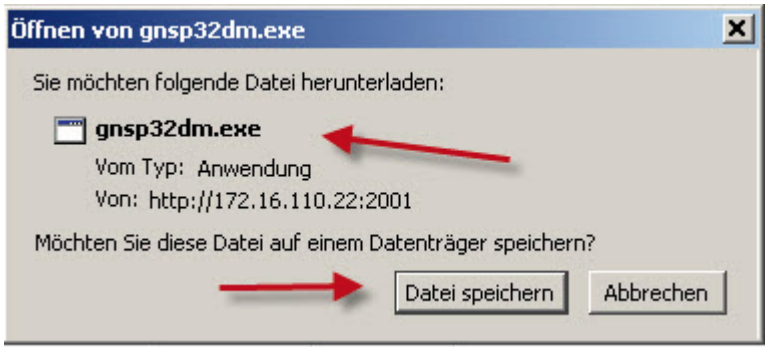
Register Formulare

Sehen Sie sich bitte diese Parameterwerte an und Sie werden den Sinn der Parameter aufgrund Ihrer bisherigen Erfahrungen mit dem CHGSPLFA–Befehl erkennen.

AFP-Viewer installieren

Bevor Sie erstmalig eine Spooldatei öffnen, sollten Sie AFP-Viewer Plug-In installieren.

Klicken Sie auf den entsprechenden Kontextmenüeintrag und folgender Dialog öffnet sich:

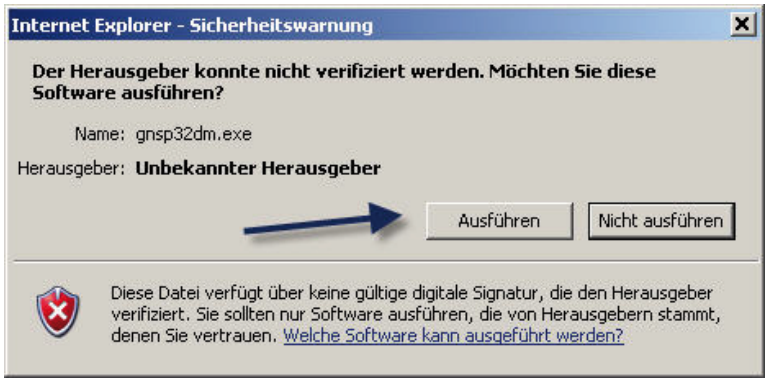


AFP-Viewer herunterladen

Mit diesem Dialog laden Sie die Datei gnsp32dm auf Ihren Desktop herunter.

Danach führen Sie die Datei gnsp32.exe auf Ihrem Desktop aus.

Eventuell erscheint eine Sicherheitsmeldung der Windowsumgebung.



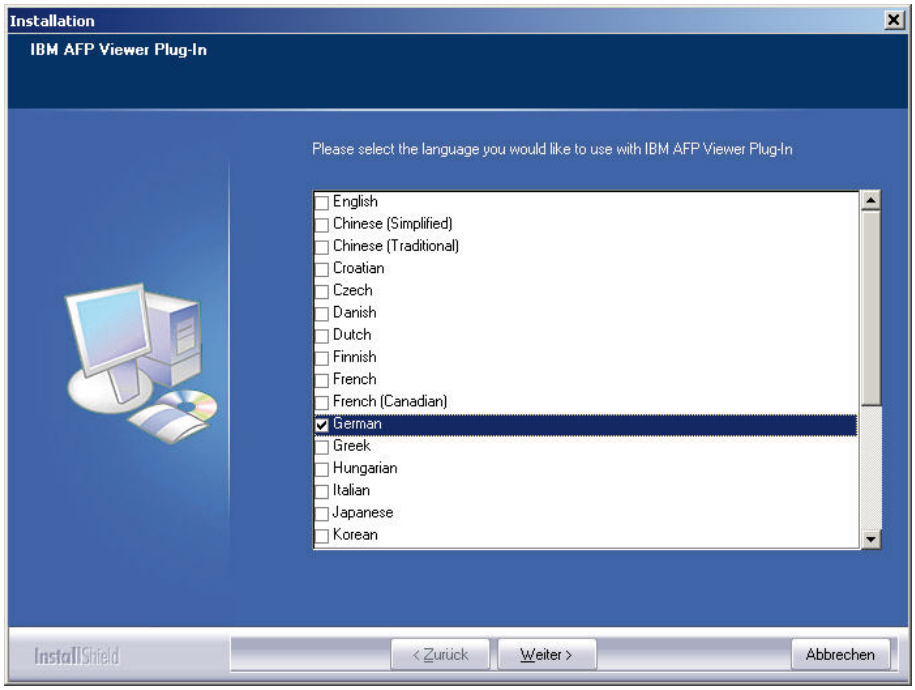
Sicherheitswarnung

Gegebenenfalls quittiere ich diesen Dialog mit Ausführen.

Danach beginnt die eigentliche Installation des Viewer Plug-ins.

7.7.4.1.3.2

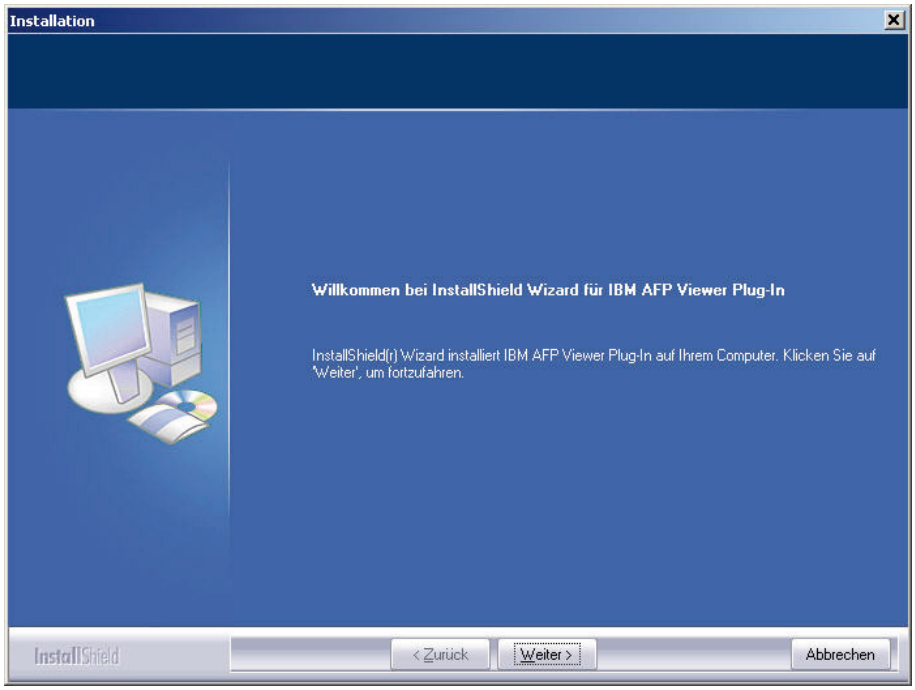
Seite 2



Sprachauswahl

Ich wähle hier German als Sprache und klicke auf Weiter.

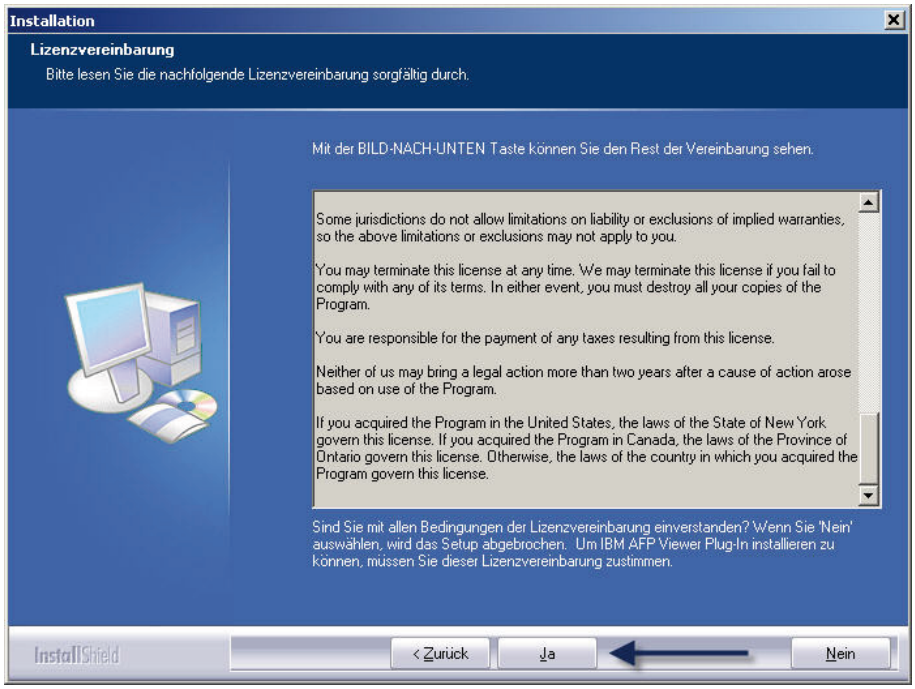
Ich werde jetzt auf Deutsch willkommen geheißen.



Willkommen

Diesen Dialog beende ich mit Weiter.

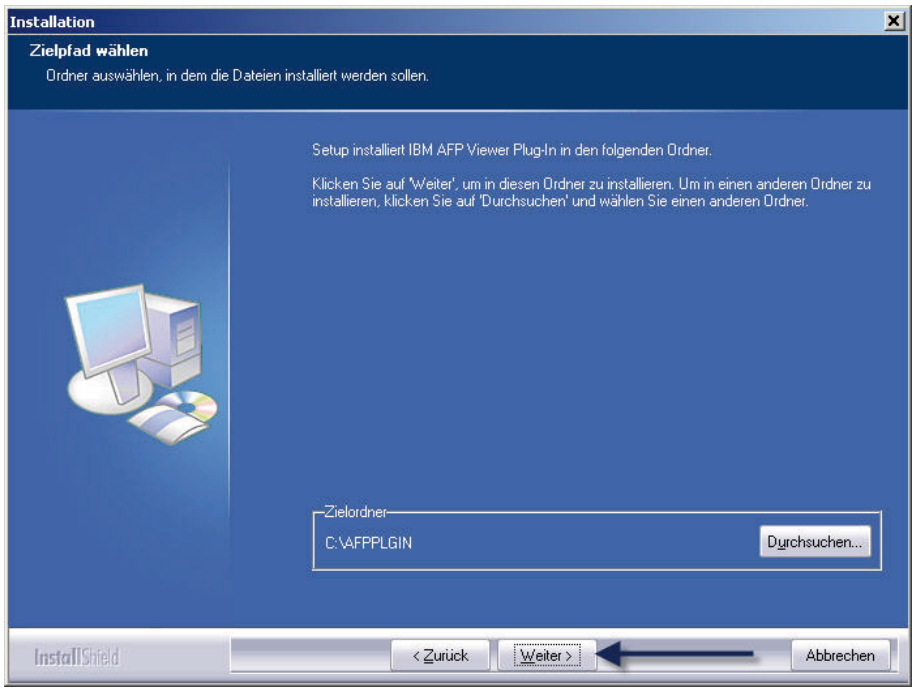
Danach werden lizenzrechtliche Informationen angezeigt.



Lizenzbedingungen

Ich akzeptiere die Lizenzbedingungen mit Ja.

Danach wird nach dem Installationsverzeichnis auf meinem Arbeitsplatz gefragt.

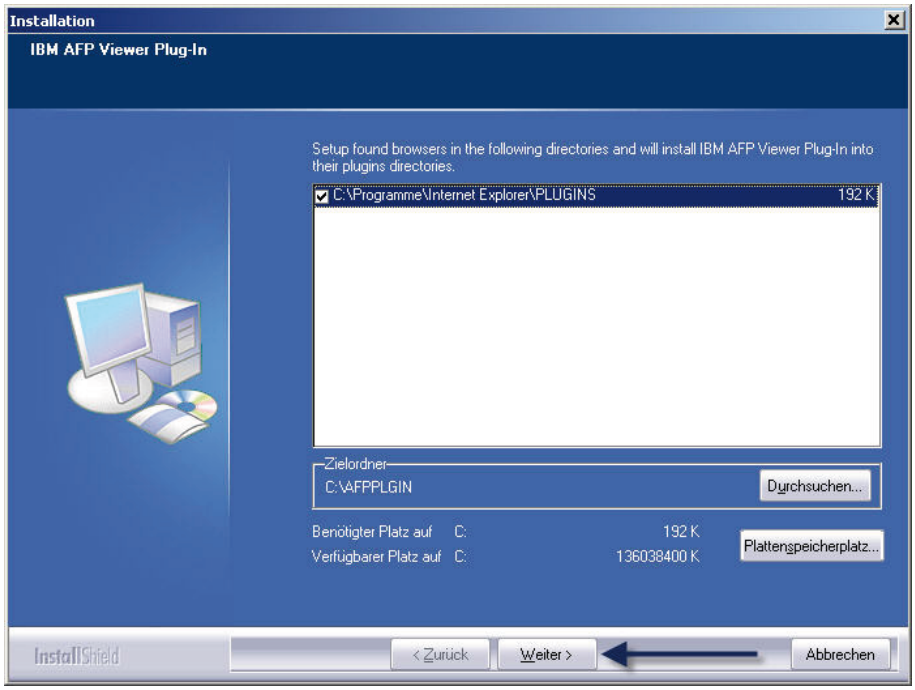


Installationsverzeichnis

7.7.4.1.3.2
Seite 4

Ich belasse es bei den Standardeinstellungen und beende den Dialog mit Weiter.

Im nächsten Fenster wird angezeigt, ob und welcher Plug-in-fähige Browser auf meinem Arbeitsplatz installiert wird.



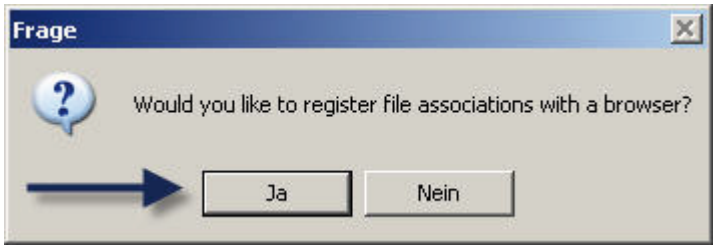
Browser Erkennung

Sie sehen, dass auf meinem Arbeitsplatz der Internet Explorer von Microsoft gefunden wurde.

Langsam nähert sich der Installations-Dialog dem Ende.

Es kommen lediglich drei Fenster.

1.) Sollen AFP-Dateien mit dem Viewer assoziiert werden?



Assoziationen erlauben

Ich bestätige die Angabe mit Ja.

2.) Ist diese Arbeit korrekt beendet worden?

Assoziationen sind registriert

Ich bestätige diese Angabe mit OK.

3.) Die Installation des Plug-ins ist beendet.

Auch diesen Dialog bestätige ich mit OK.

Achtung!

Verwenden Sie Internetexplorer Version 6 oder älter, so führt der Aufruf des Plug-Ins zum Absturz des Browsers!

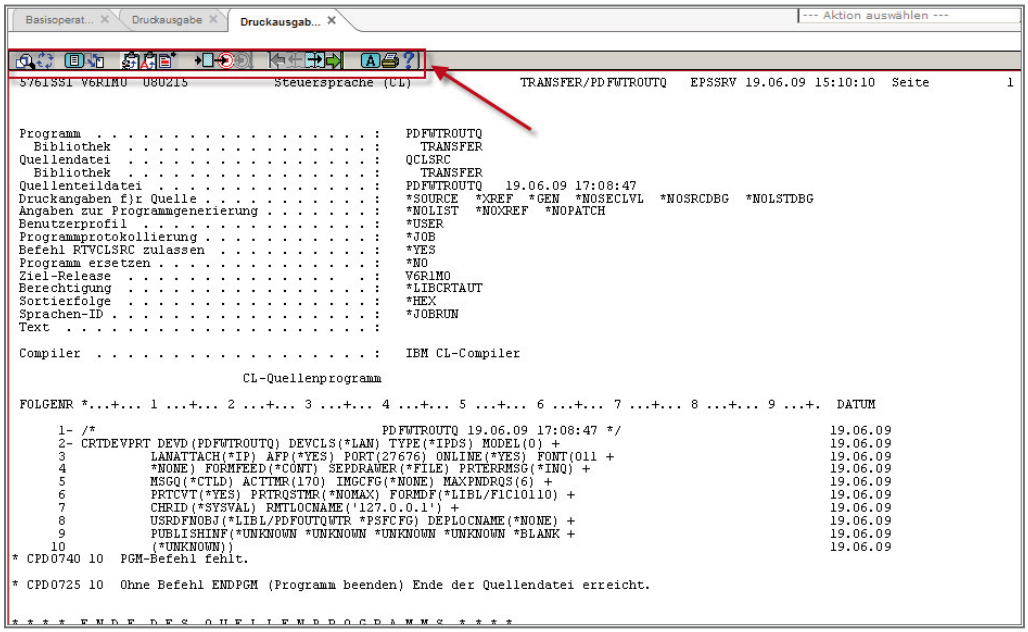
7.7.4.1.3.2

Seite 6

Öffnen

7.7.4.1.3.3
Seite 1

Der Kontextmenüpunkt Öffnen ruft den AFP-Viewer auf, der Spooldateien im graphischen Umfeld anzeigt.



Der AFP-Viewer

Kurz gesagt ermöglicht diese Komponente des System i Access alle Funktionen, die auch der Systembefehl DSPSPLF (Display Spool File) zeigt.

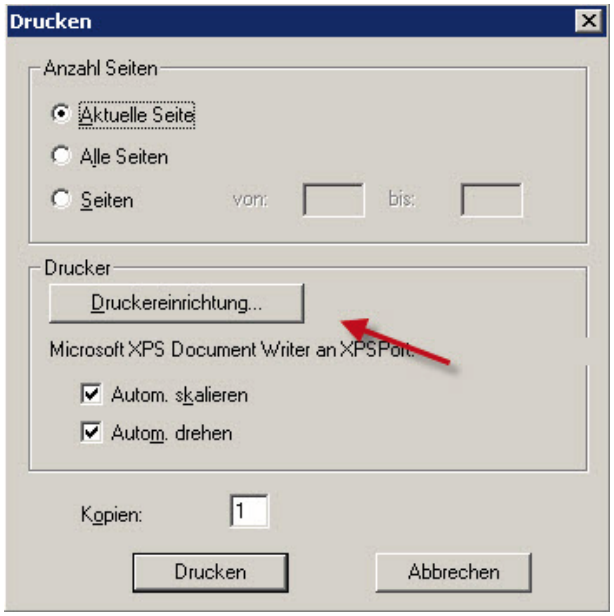
Es muss jedoch gesagt werden, dass der AFP-Viewer über die Funktionen des DSPSPLF-Befehls hinausgeht. So ist der Viewer nicht nur ein Ersatz für den bereits zitierten Anzeigebefehl.

Darüber hinaus werden Funktionen realisiert, die die Weitergabe von Spooldateien an PC-Anwendungen via Clipboard ermöglicht.

Eine der meines Erachtens besten Funktionen des AFP-Viewers besteht darin, Spooldateien direkt über einen lokalen Drucker auszugeben. Um eine Druckausgabe vorzubereiten, klicken Sie einfach auf das Druckersymbol im AFP-Viewer.

7.7.4.1.3.3

Seite 2



Drucker auswählen

Sie sehen, dass hier der Standarddrucker meines Systems ausgewählt wurde. Mit dem Button Druckereinrichtung können Sie dann einen anderen lokalen Drucker auswählen. Diese Auswahl bestätige ich mit OK.

Danach klicke ich auf Drucken und der Druck wird ausgeführt.

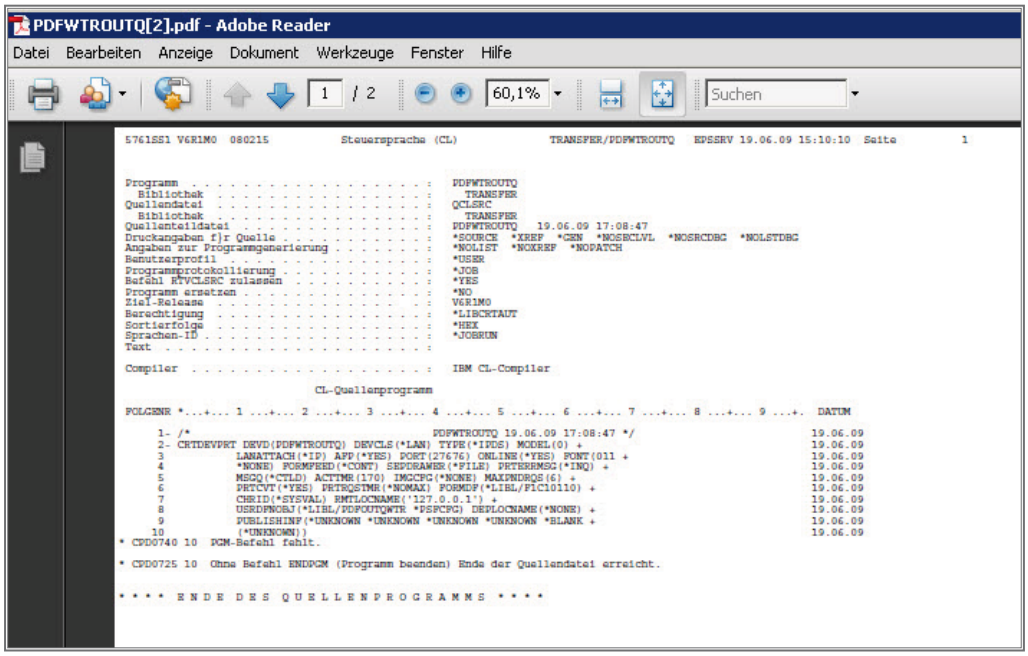
Mit Hilfe dieser Druckfunktion lassen sich aber auch aus jeder iSeries Spool-Datei PDF-Dokumente erzeugen. Hierzu benötigen Sie entweder den Acrobat Distiller oder ein kostenfreies Produkt wie Win2PDF, um PC-Ausgaben ins PDF-Format zu konvertieren, oder den Acrobat PDFwriter, der Bestandteil des kostenpflichtigen Produkts Acrobat ist. Ich werde win2pdf zum Erstellen einer PDF-Datei benutzen.

Ich wechsle hierfür wieder in die Druckereinrichtung.

Win2pdf einrichten

Ich markiere den Eintrag Win2PDF und bestätige die Auswahl mit OK.

Danach klicke ich auf Drucken.

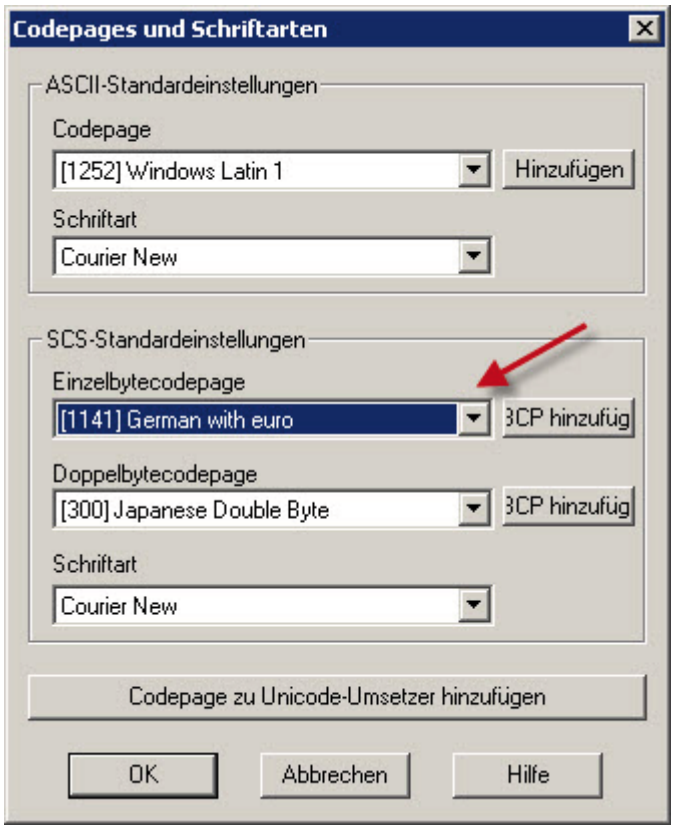


Anzeige der Spool in Acrobat

Zum Schluss noch zwei Tipps, mit denen Sie die Anzeigen des AFP-Viewers verbessern können. Mir ist auf einer ganzen Reihe von PCs aufgefallen, dass die Umlautdarstellung des AFP-Viewers nicht korrekt erfolgt. Dieser Fehler lässt sich leicht beheben.

7.7.4.1.3.3
Seite 4

Klicken Sie im AFP-Viewer auf das Symbol A – Codepages und Schriftarten. Danach sehen Sie folgenden Dialog:



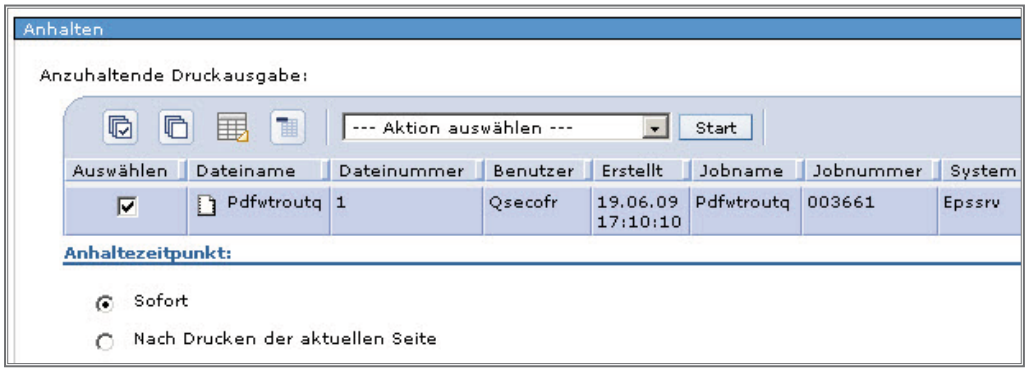
Weitere Vorgaben

Tragen Sie in den Parameter Einzelbytecodepage den Wert 1141 – Deutsch mit Euro – ein. Beenden Sie den Dialog mit OK. Beenden Sie danach den AFP-Viewer. Beim nächsten Öffnen des Viewers werden die Umlaute korrekt angezeigt.

Diverse

Einige der Kontextmenüpunkte sprechen in Ihrer Funktionalität für sich selbst. Deshalb werde ich an dieser Stelle nur wenige Erläuterungen anbringen.

Anhalten…



Druckausgabe anhalten

Hält eine im Druck befindliche Ausgabe sofort oder nach Beendigung der gerade zu druckenden Seite an.

Als nächstes drucken

Setzt die Ausgabepriorität einer Druckausgabe so hoch, dass sie sofort gedruckt wird.

Senden

Druckausgabe an fernes System übergeben

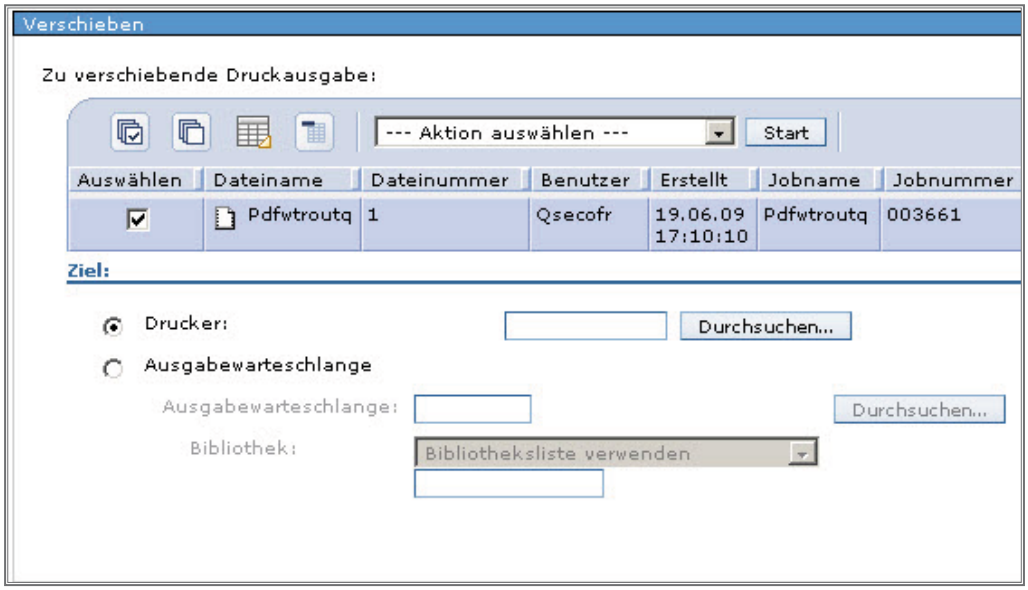
Mit dem Punkt Senden übergeben Sie eine Druckausgabe an ein fernes System. Zu diesem Zweck stehen zwei Protokolle zur Verfügung.

- TCP/IP
- SNA

Die Auswahl TCP/IP entspricht dem Systembefehl LPR bzw. SNDTCPSPLF. Mit dieser Auswahl lassen sich Druckdateien nicht nur an ferne Power i-Systeme versenden, sondern auch an jedes andere System, auf dem der TCP/IP-LPD-Dienst implementiert und ausgeführt wird.

7.7.4.1.3.4
Seite 2

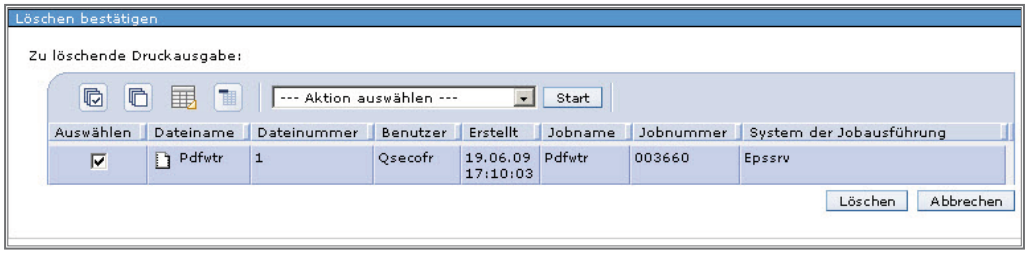
Verschieben



Druckausgabe verschieben

Mit Hilfe dieses Punkts transferieren Sie an eine andere Ausgabewarteschlange des eigenen Systems.

Löschen…



Druckausgabe Löschen

Mit Hilfe dieser Option können Sie eine Druckausgabe aus dem System entfernen.

Exportieren…

Eine Druckdatei des Systems wird in eine ASCII-Datei konvertiert und kann auf Ihrem Desktop gespeichert werden.

Erweitert

Erweiterte Druckbearbeitung

Unter diesem Punkt stehen Ihnen zwei Verarbeitungsoptionen zu Verfügung:

- Ausgabewarteschlangen verwalten
- Drucker verwalten

Bearbeiten Sie die Ausgabewarteschlange, in der sich der aktuelle Druck befindet.

Arbeiten Sie mit den Druckern dieses Systems.

Zu diesen Punkten werde ich später mehr erläutern.

7.7.4.1.3.4

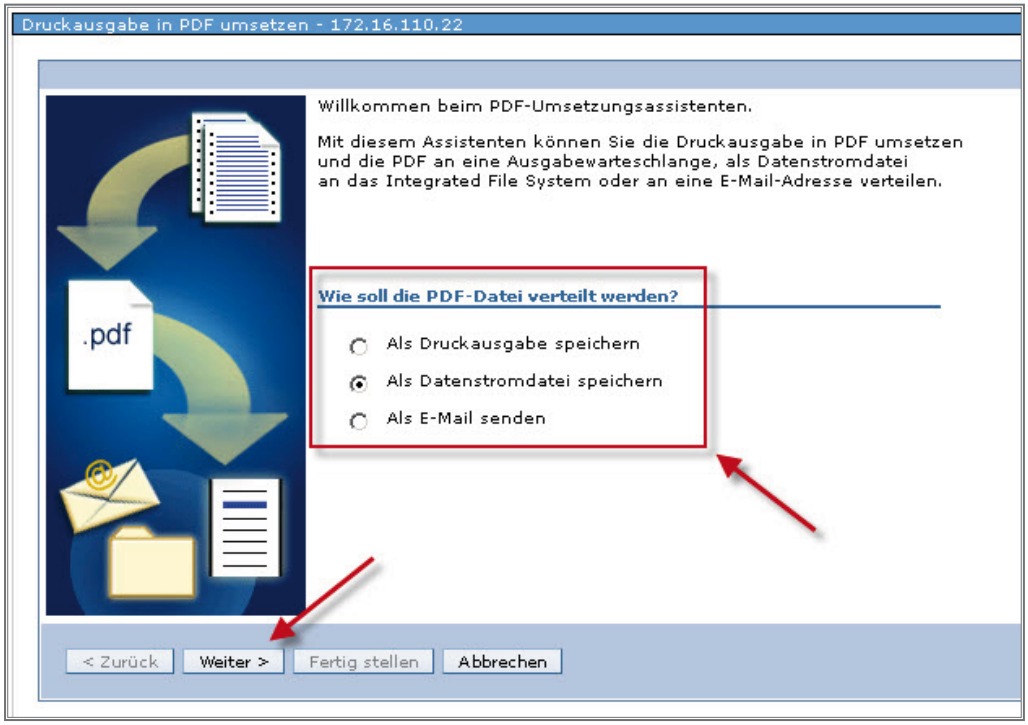
Seite 4

Umsetzen in PDF

Um die Funktion Umsetzen in PDF einsetzen zu können, benötigen Sie das Lizenzprogramm IBM Infoprint Server. Mit dem Infoprint Server erzeugen Sie im System sogenannte PDF-Drucker, die dann im i/OS die native Konvertierung von Spooldateien in PDF bewirken.

Ist der Infoprint Server nicht installiert, so ist diese Verarbeitungsoption funktionslos.

Starten Sie diese Verarbeitung, so öffnet sich ein Wizard, mit dem Sie ein PDF für markierte Druckausgaben erstellen können.



PDF-Datei im IFS erzeugen

Auf der ersten Seite des Wizards bestimmen Sie, in welcher Form das PDF verwertet werden soll.

Als Druckausgabe speichern: Die erzeugte PDF-Ausgabe wird in eine Ausgabewarteschlange des Systems gespoolt. Diese Ausgabewarteschlange bedient einen nativen PDF-Drucker (Hardware).

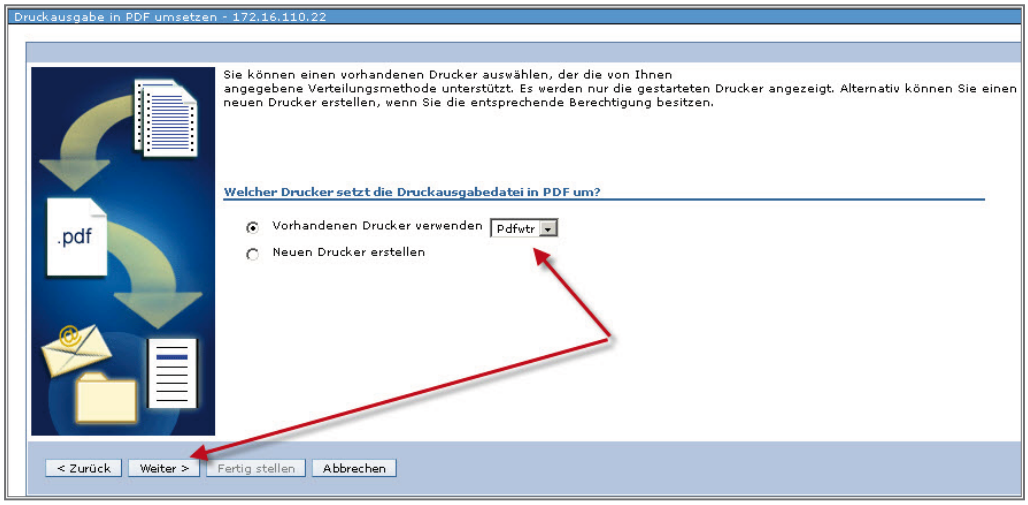
Als E-Mail senden: Der Punkt spricht für sich. Wichtig ist jedoch, dass für diese Funktion der TCP/IP SMTP-Dienst konfiguriert und aktiviert werden muss.

Als Datenstromdatei speichern: Das PDF-Dokument wird im IFS gespeichert.

7.7.4.1.3.5
Seite 2

Ich habe diese Verarbeitungsoption ausgewählt.

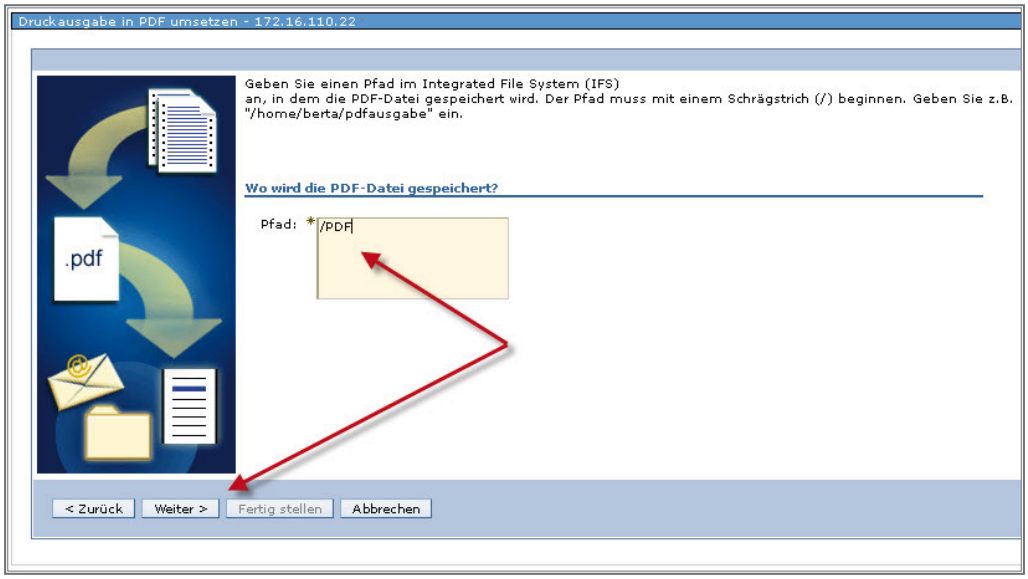
Im nun folgenden Dialog kann ich entweder einen bereits konfigurierten PDF-Drucker verwenden oder aber einen neuen erzeugen.



PDF Drucker

Ich verwende hier den bereits vorhandenen Drucker PDFWTR.

In welchem IFS-Verzeichnis soll das zu erzeugende Dokument gespeichert werden?



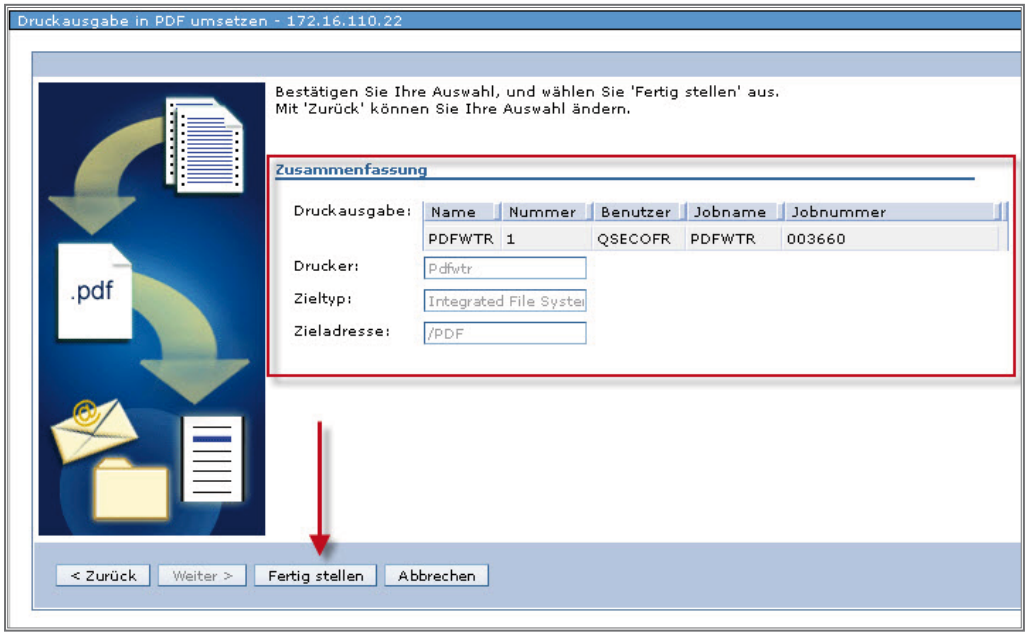
PDF Dokument speichern

Ich werde das zu erzeugende PDF-Dokument im bereits bestehenden Verzeichnis /PDF erstellen.

Wichtig!

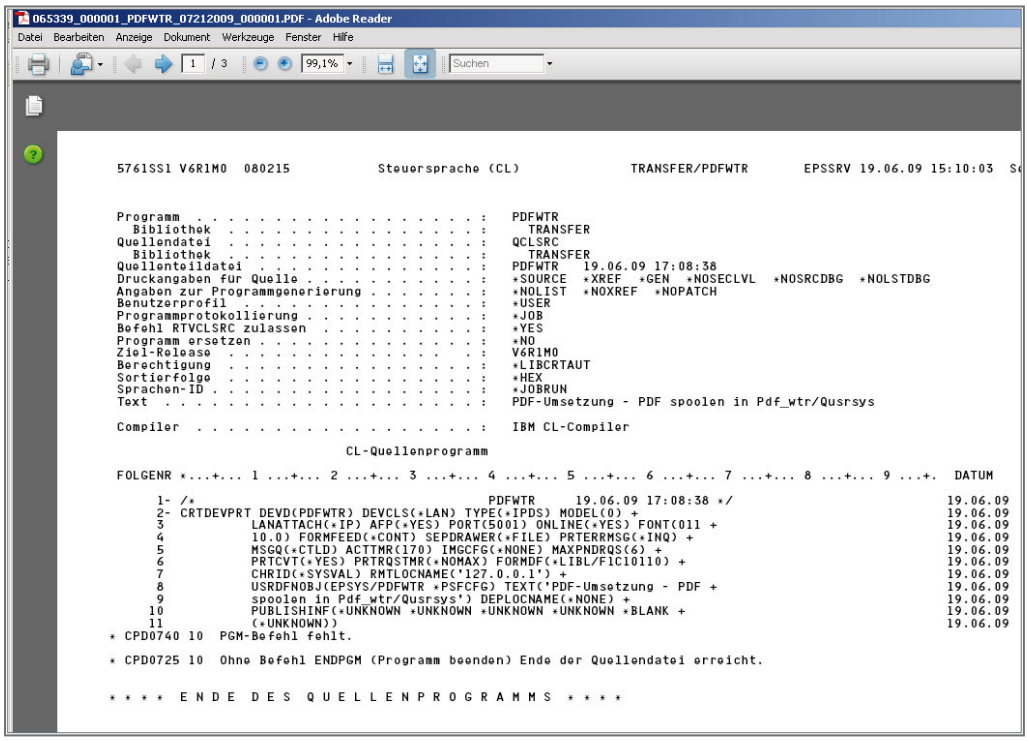
Achten Sie auf die korrekte Pfad-Notierung mit / (Schrägstrich)!

Die bis hierhin getätigten Eingaben werden in einer Zusammenfassung angezeigt.



Zusammenfassung

Ich beendige den Dialog mit Fertig stellen.



PDF-Dokument

7.7.4.1.3.5

Seite 4

Drucker hinzufügen

Im Funktionsbereich Druckausgabe ist ein Assistent enthalten, der die Konfiguration von Netzwerkdruckern sehr erleichtert.

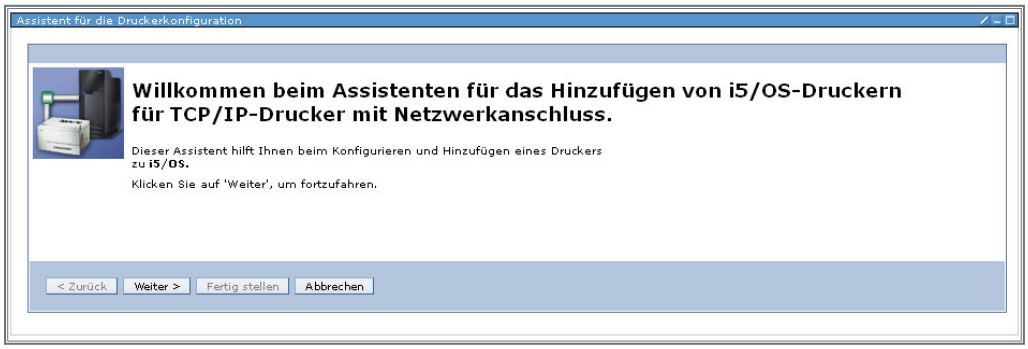
Rekapitulieren wir kurz: Das System i kann Netzwerkdrucker mit folgenden Technologien bzw. Protokollen ansprechen:

- Als Druckereinheit über direktes TCP/IP
- Als remote Ausgabewarteschlange über LPR/LPD
- Als http-Client über das IPP-Protokoll
- Als Drucker an einem Print-Server

Alle diese Konfigurationsvarianten unterstützt der Drucker-Konfigurations-Assistent.

Ich werde jetzt nicht alle diese Varianten besprechen, sondern werde mich auf die Konfiguration eines direkt angeschlossenen HP5000N Druckers beschränken.

In den Basisoperationen wähle ich den Menüpunkt Drucker hinzufügen. Jetzt erscheint der Willkommensdialog des Konfigurations-Assistenten.



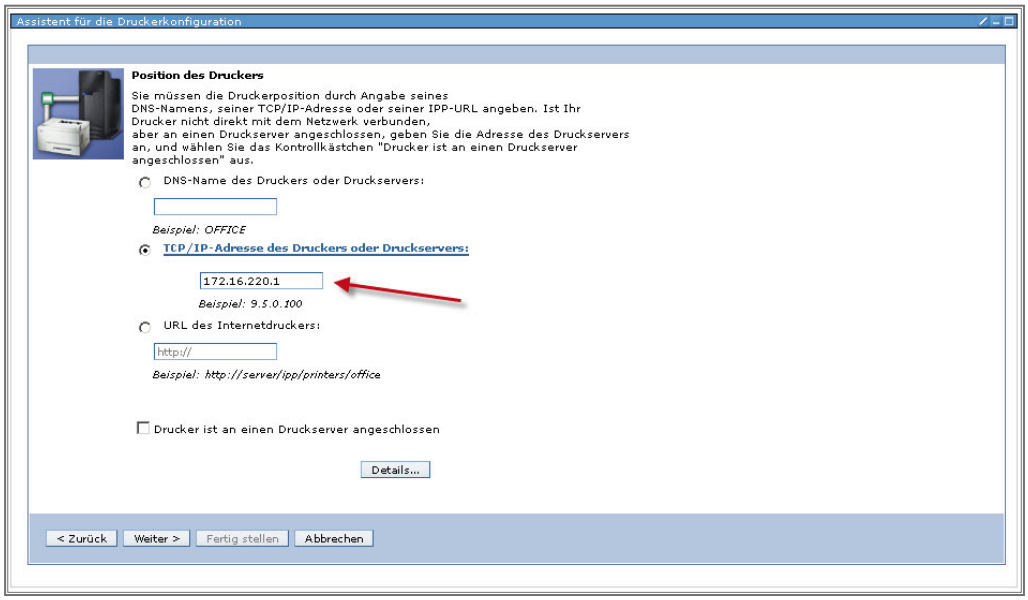
Willkommen

Diesen Dialog beende ich mit Weiter.

Im Bild Position des Druckers können die oben angesprochenen Konfigurationsvarianten ausgewählt werden.

7.7.4.1.3.6

Seite 2



Eingabe der Adresse

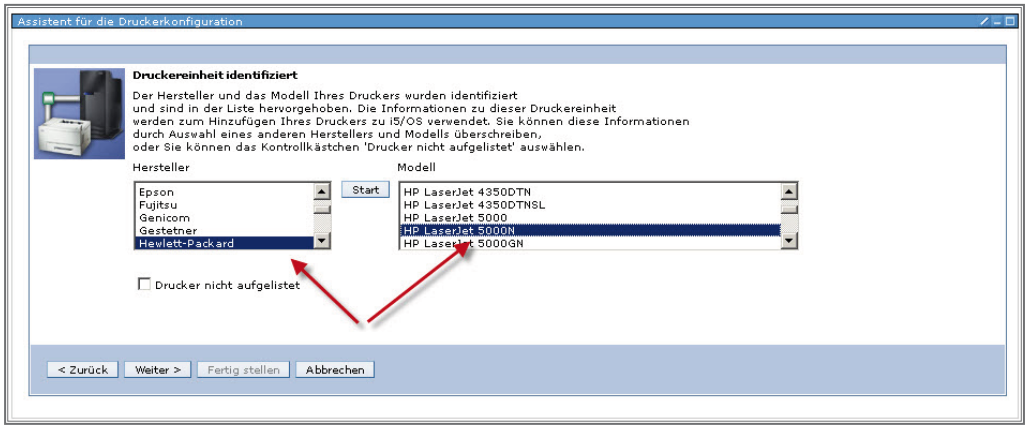
Zuerst einmal wird die Möglichkeit geboten, direkt die IP-Adresse bzw. den Hostnamen des Druckers bzw. Druckservers in die entsprechenden Parameter einzutragen.

Ich aktiviere hier die Option IP-Adresse des Druckers…

Verwände ich den Parameter URL des Internetdruckers, so würde ich einen IPP-fähigen Drucker ansprechen.

Die Aktivierung des Parameters Drucker ist an einen Druckserver angeschlossen erklärt sich selbst.

Ich bestätige die Eingaben mit Weiter.



Druckertyp wählen

Jetzt wähle ich den Hersteller und das der Aufgabenstellung entsprechende Druckermodell ein.

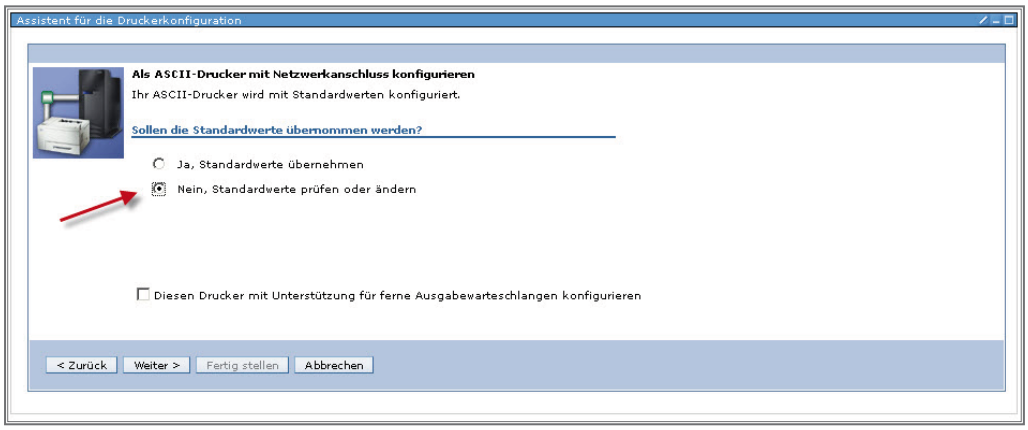
Hier: Hewlett-Packard und HP LaserJet 5000N

Ich beende die Auswahl mit Weiter.

IPDS oder nicht IPDS

Da ein HP Laserjet 5000N das IPDS-Protokoll nativ nicht unterstützt, wähle ich diese Möglichkeit ab.

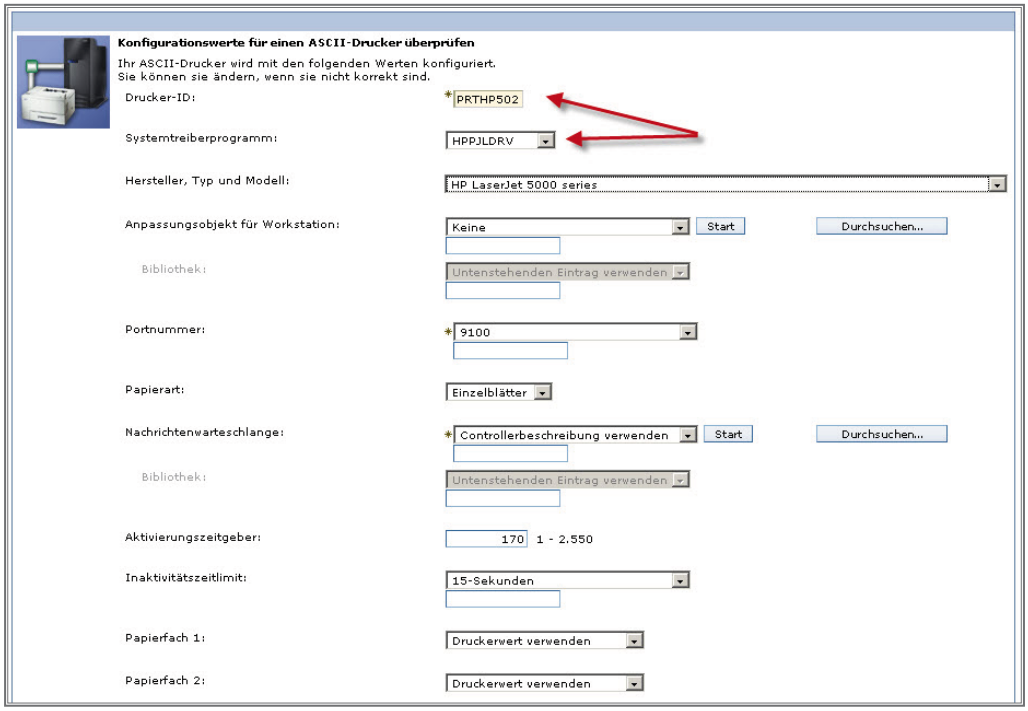
Mit Weiter verlasse ich den Dialog.



Standardwerte übernehmen?

7.7.4.1.3.6
Seite 4

Im danach folgenden Bild verzichte ich darauf, die Standardwerte für diese Druckkonfiguration zu übernehmen, und gehe weiter.

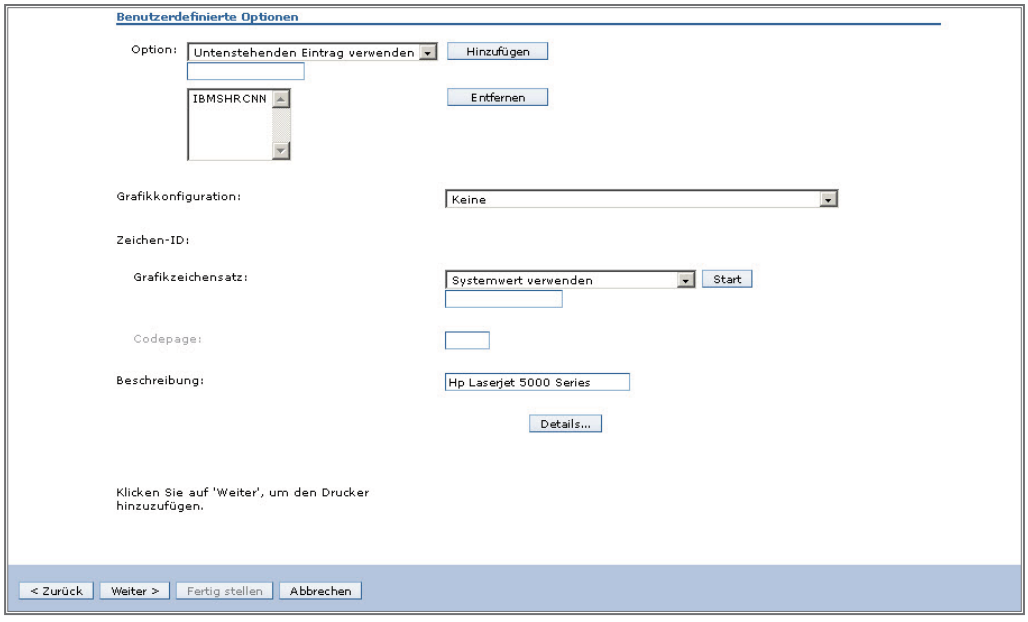


Detailkonfiguration 1

Im Dialog Konfigurationswerte für ASCII-Drucker… trage ich jetzt den Namen des zu erstellenden Netzwerkdruckers ein.

Hier: PRTHP502

Ich ändere den Systemtreiber von IBMPJLDRV in HPPJLDRV.



Detailkonfiguration 2

Alle weiteren gezeigten Parameterwerte belasse ich auf ihren Standardwerten.

Ich beende den Dialog mit Weiter.

Testseite drucken?

Die Frage, ob eine Testseite gedruckt werden soll, beantworte ich mit Ja.

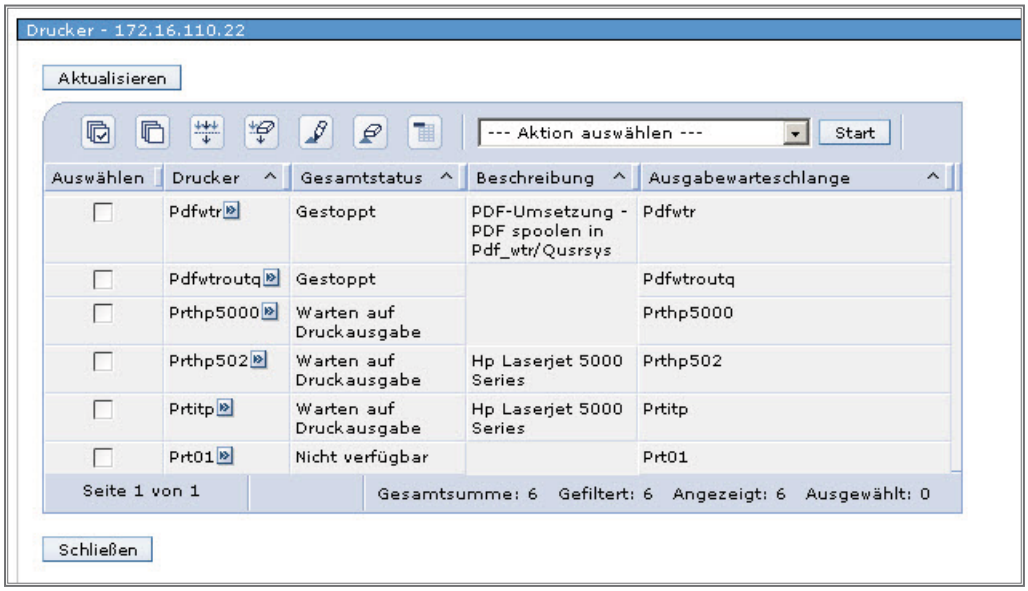
Dann beende ich die Konfiguration mit Fertig stellen.

7.7.4.1.3.6

Seite 6

Mit Druckern arbeiten

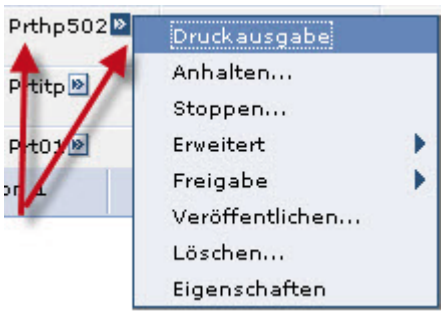
Der Funktionsbereich Drucker ermöglicht die Steuerung von Systemdruckern.



Drucker

Sie sehen in der obigen Hardcopy alle Systemdrucker, die am System 172.16.110.22 angeschlossen sind.

Ein Klick auf das Pfeilsymbol öffnet ein Kontentextmenü für den gewählten Drucker.



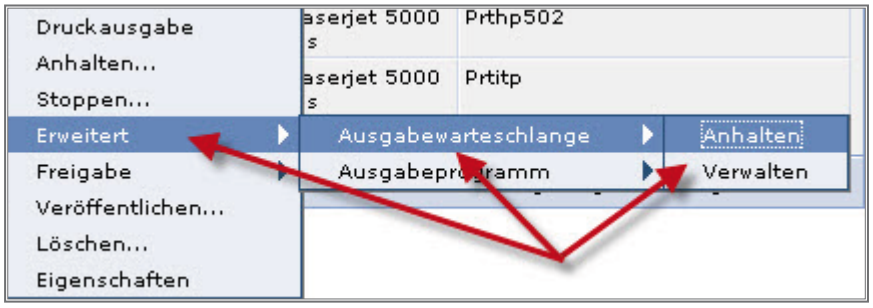
Kontextmenü Drucker

Dieses Menü stellt folgende Verarbeitungsmöglichkeiten bereit:

- Druckausgabe
  Anstehende Ausgaben des Druckers anzeigen
- Freigabe
  Druckerfreigabe für das Microsoft Netzwerk erstellen
- Veröffentlichen
  Drucken an einen LDAP-Server publizieren

7.7.4.1.4
Seite 2

- Anhalten
  Den Drucker anhalten
- Stoppen
  Das Druckausgabeprogramm beenden
- Eigenschaften
  Druckereigenschaften ändern/anzeigen
- Löschen…
  Eine Druckeinheit löschen.

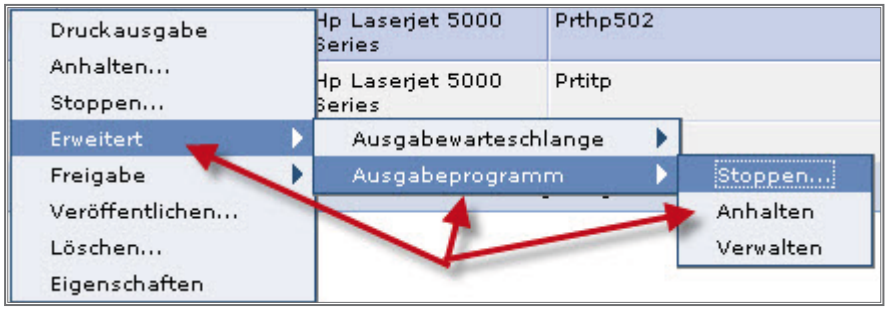


Kontextmenüeintrag Erweitert 1

Der Kontextmenüeintrag Erweitert… stellt mir zwei Verarbeitungsbereiche zur Verfügung.

Mit der Ausgabewarteschlange des ausgewählten Druckers arbeiten:

- Anhalten: Druckausgabewarteschlange anhalten. Dieses entspricht einem HLDOUTQ.
- Verwalten: Mit der Ausgabewarteschlange arbeiten. Dieses entspricht einem WRKOUTQ für die betreffende Warteschlange.



Kontextmenüeintrag Erweitert 2

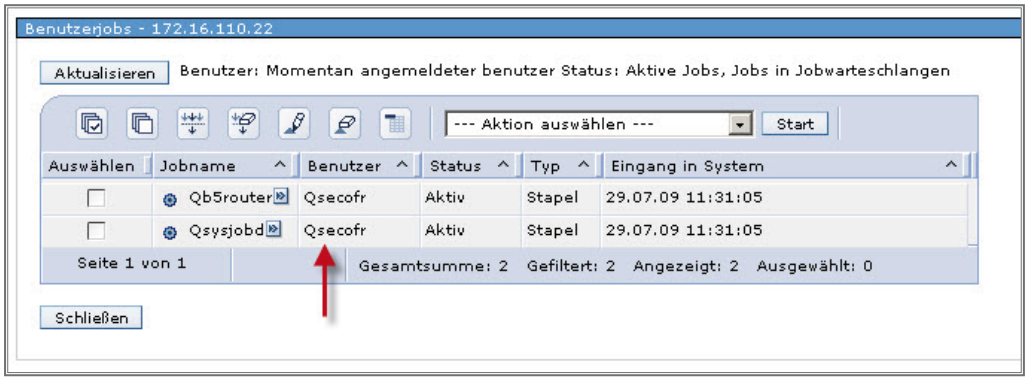
Mit der Druckausgabeprogramm des ausgewählten Druckers arbeiten:

- Stoppen: Drucker beenden (ENDWTR)
- Anhalten: Drucker anhalten (HLDWTR)
- Verwalten: Mit dem Drucker arbeiten

Mit Benutzerjobs arbeiten

Die Funktion Mit Benutzerjobs arbeiten entspricht weitgehend den Funktionen des CL-Befehls WRKUSRJOB. Dieser Befehl dient im OS/400 zur Suche nach im System befindlichen Jobs. Verwechseln Sie diese Funktion nicht mit einem WRKACTJOB (Mit aktiven Jobs arbeiten). Zwar sind viele Anzeigen dieser Funktion ähnlich, doch die komplexen Steuerungsfunktionen eines WRKACTJOB fehlen an dieser Stelle.

Anfänglich werden die Jobs des angemeldeten Benutzers angezeigt.

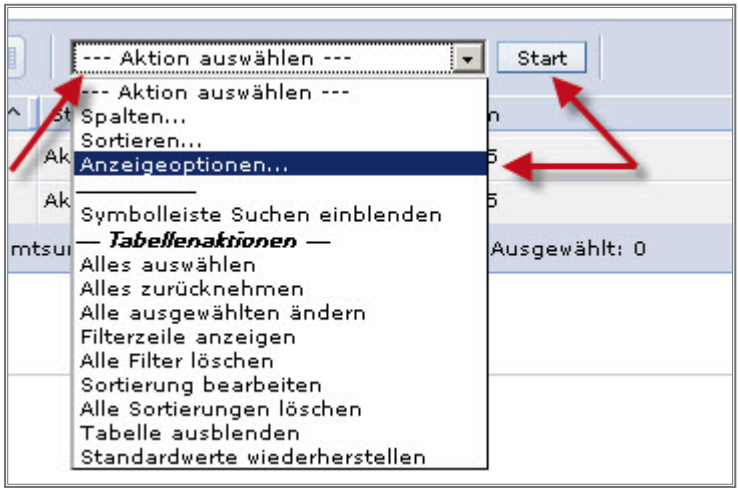


Standardanzeige

In diesem Fall handelt es sich um die Jobs des Benutzers QSECOFR.

Als kleine Fallstudie möchte ich nach allen Jobs des Benutzers Luttkus auf dem System suchen. Die Suche betrifft sowohl Jobs, die in einer Jobwarteschlange auf Verarbeitung warten, solche, die derzeit aktiv vom System bearbeitet werden, und Jobs, deren Ausgaben auf ihre Ausgabe in einer Ausgabewarteschlange warten.

Wie gehe ich vor? Zuerst einmal muss ich Sorge dafür tragen, dass ausschließlich die Jobs für den Benutzer Luttkus angezeigt werden.

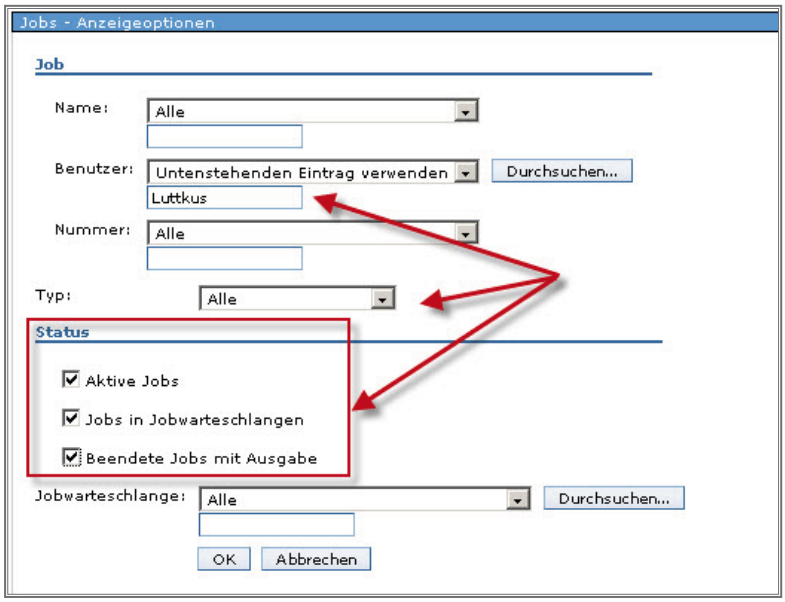


Aktion auswählen

7.7.4.1.5
Seite 2

Zu diesem Zweck öffne ich die Combo Box Aktion auswählen, wähle sodann Anzeigeoptionen und klicke auf Start.

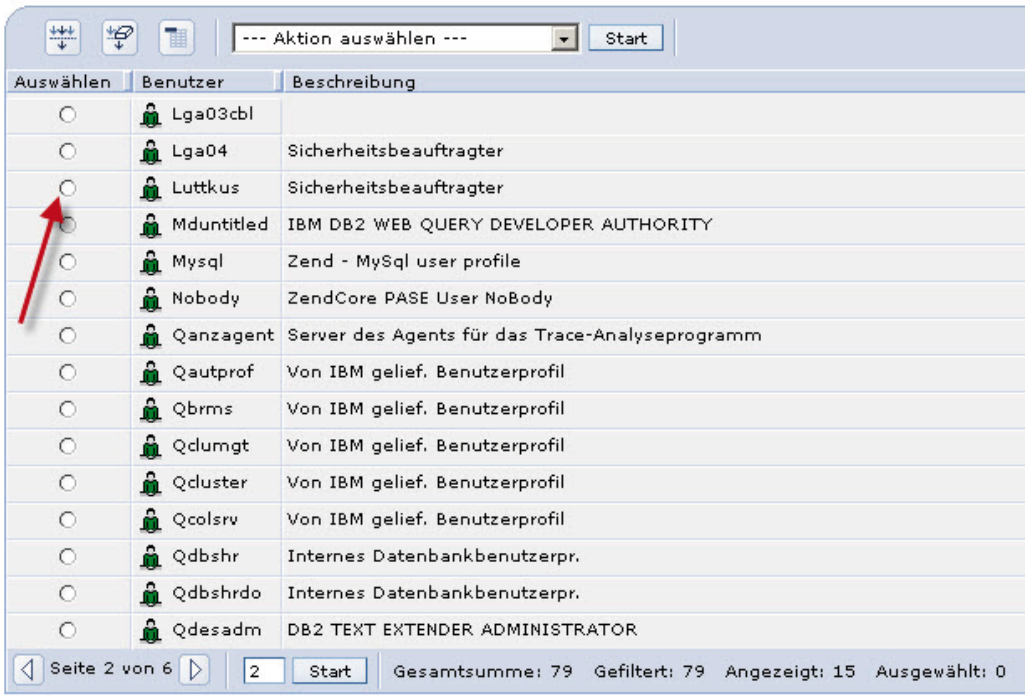
Jetzt öffnet sich der folgende Dialog, der starke Ähnlichkeiten mit dem Bedienerführungsdialog des CL-Befehls WRKUSRJOB aufweist.



Anzeigeoptionen für Jobs

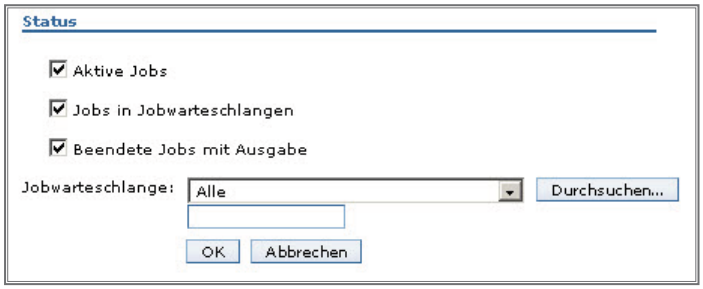
Sie sehen in der Parametergruppe Job den Einzelparameter Name, in den der Standardwert Alle eingetragen ist. Dies bedeutet, dass als Standard alle Jobs aller Benutzer der iSeries angezeigt werden. Diesen Wert werde ich ändern, indem ich auf den Knopf Durchsuchen klicke.

Jetzt öffnet sich der Dialog Benutzer.



Benutzer auswählen

Hier werden alle für Sie berechtigten Benutzer angezeigt. Ich wähle den Benutzer Luttkus aus.



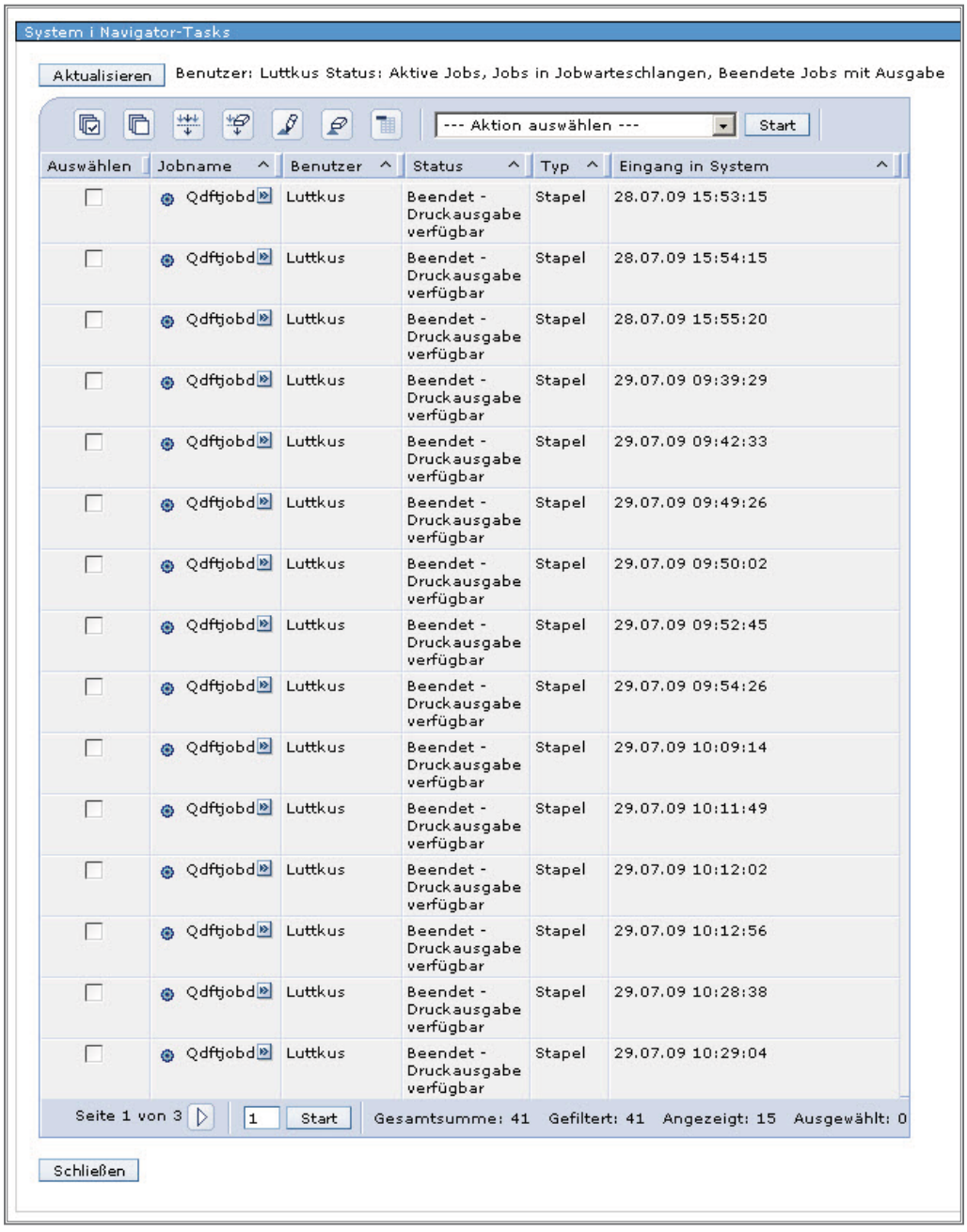
Anzeigeoptionen: mehr Parameter

In der Parametergruppe Status aktiviere ich folgende Parameter:

- Aktive Jobs anzeigen – Ja
- Jobs in Jobwarteschlange anzeigen – Ja
- Beendete Jobs mit Druckausgabe anzeigen – Ja

Danach kehre ich in die Anzeige zurück.

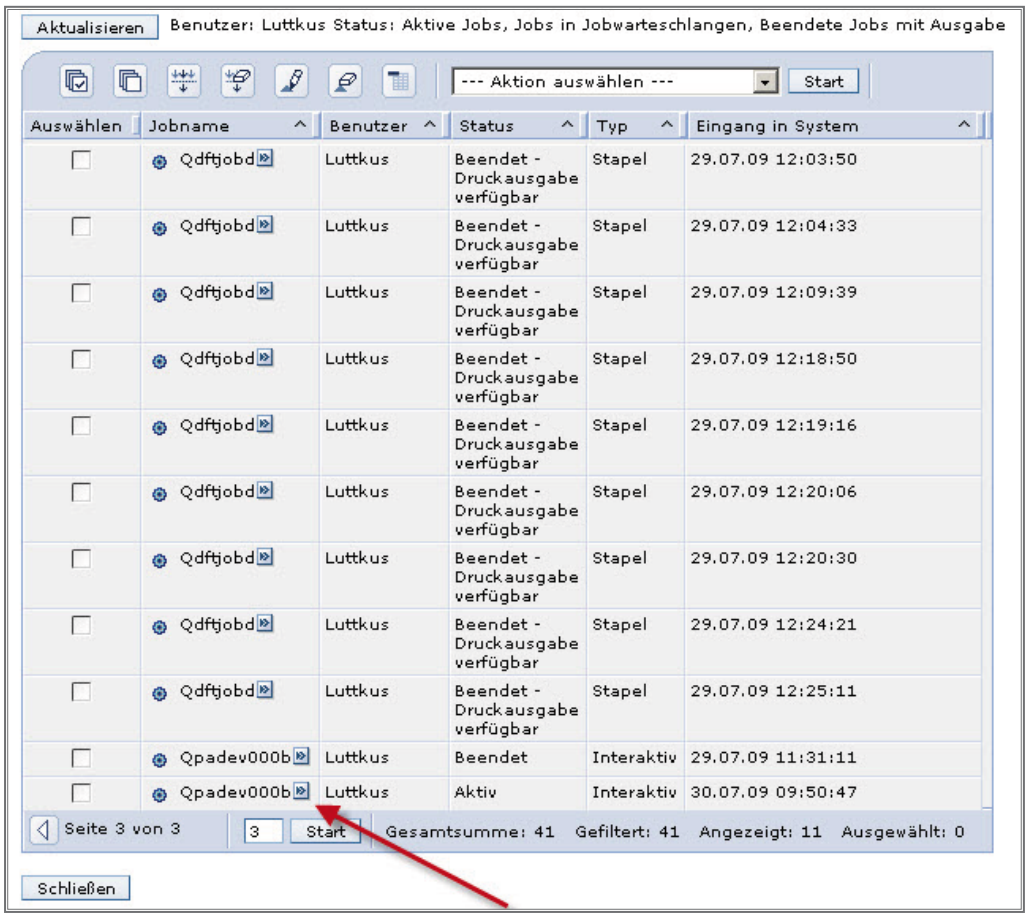
7.7.4.1.5
Seite 4



Jobs des Benutzers Luttkus

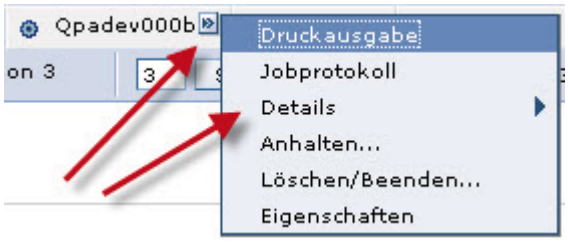
In der Anzeige erkennen Sie jetzt die Jobs des Benutzers Luttkus im System.

Sie können erkennen, dass alle angezeigten Jobs zur Druckausgabe anstehen (Status). Ich blättere weiter und finde einen interaktiven Job.



Interaktiver Job

Das ist interessant. Wir sehen jetzt nach, ob dieser Job ein Jobprotokoll zur Ausgabe anstehen hat. Dazu lasse ich mir das Kontextmenü für diesen Job anzeigen.



Kontextmenü für einen einzelnen Job

7.7.4.1.5
Seite 6

Folgende zusätzliche Funktionen sind je nach Jobstatus möglich:

Druckausgabe:
Druckausgaben für einen Job anzeigen

Jobprotokoll:
Jobprotokoll für einen Job anzeigen

Anhalten:
Laufenden Job anhalten

Freigeben:
Angehaltenen Job wieder freigeben

Löschen/Beenden:
Job sofort oder kontrolliert beenden

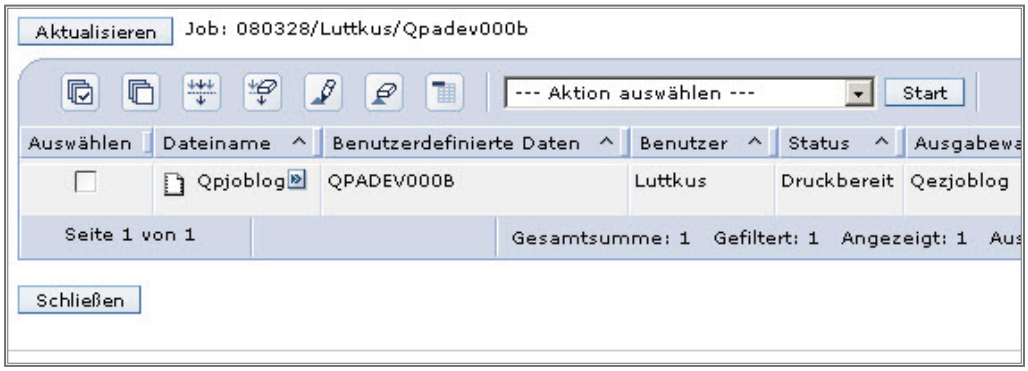
Eigenschaften:
Eigenschaften eines Jobs anzeigen

Details

Jobdetails

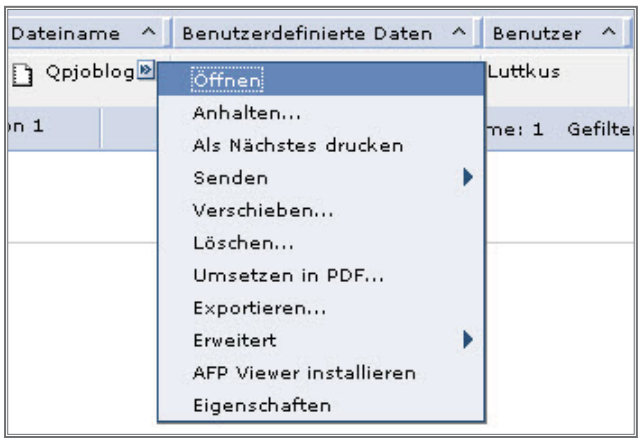
Unter diesem Punkt finden Sie die oben gezeigten Auswahlmöglichkeiten, die insgesamt für sich sprechen.

Ich wähle Druckausgabe und Sie sehen das zur Druckausgabe anstehende Joblog.



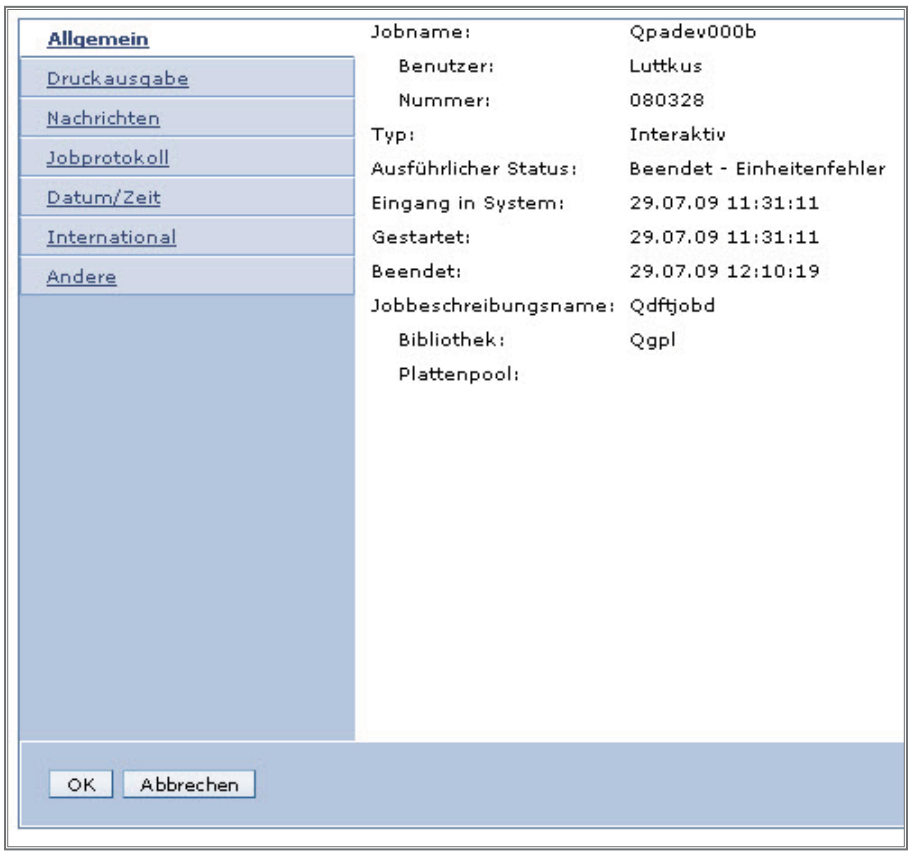
Druckausgabe für Job

Ein Klick auf das Kontextmenüsymbol bringt das Ihnen aus dem vorherigen Abschnitt bekannte Kontextmenü zur Anzeige.



Druckausgabe Kontextmenü

Hier sehen Sie zusätzlich die Eigenschaften dieses Jobs.



Eigenschaften eines Jobs

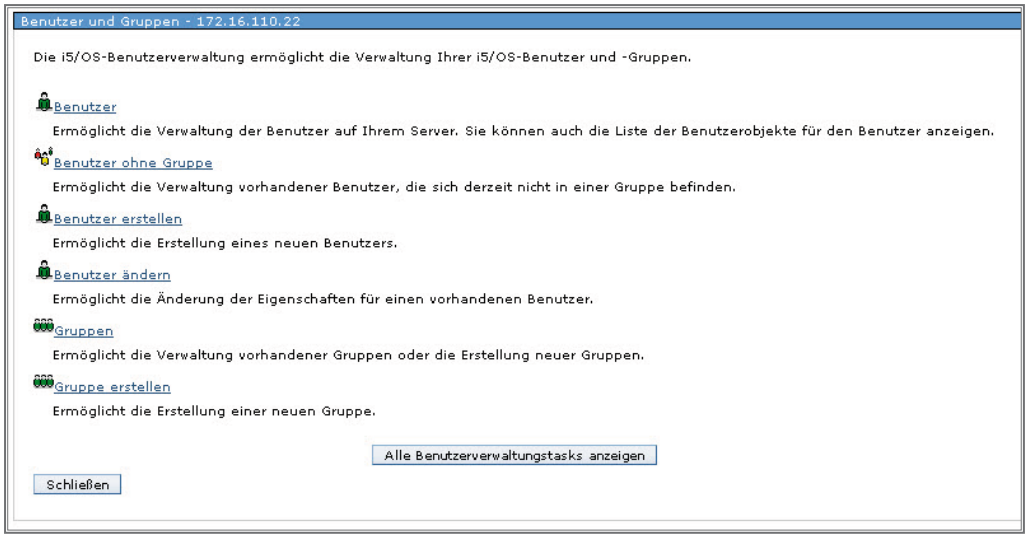
Aus diesem Dialog entnehmen Sie die genauen Kenndaten des beobachteten Jobs.

7.7.4.1.5

Seite 8

Benutzer und Gruppen

Mit der Komponente Benutzer und Gruppen können Sie Listen der Benutzer und Gruppen Ihres Systems anzeigen. Ausgehend von dieser Liste können Sie Benutzer und Gruppen verwalten.



Benutzerverwaltungsaufgaben

Sie können folgende Funktionen verwenden:

- Benutzer: Mit allen Benutzern arbeiten. Dabei bleibt unberücksichtigt, ob ein Benutzer Mitglied einer Benutzergruppe ist oder nicht.
- Benutzer ohne Gruppe: Mit den Benutzern arbeiten, die keiner Benutzergruppe angehören.
- Benutzer erstellen
- Benutzer ändern
- Gruppen: Mit Gruppenprofilen arbeiten: Diese Verarbeitungsvariante besteht mittels CL-Befehlen nicht.
- Benutzer erstellen

7.7.4.2
Seite 2

Das Leistungsspektrum der Komponente Benutzer und Gruppen umfasst unter anderem folgende Funktionen:

– Benutzer erstellen

– Benutzer zu Gruppen hinzufügen

– Benutzer auf andere Systeme IBM iSeries kopieren

– Benutzer löschen

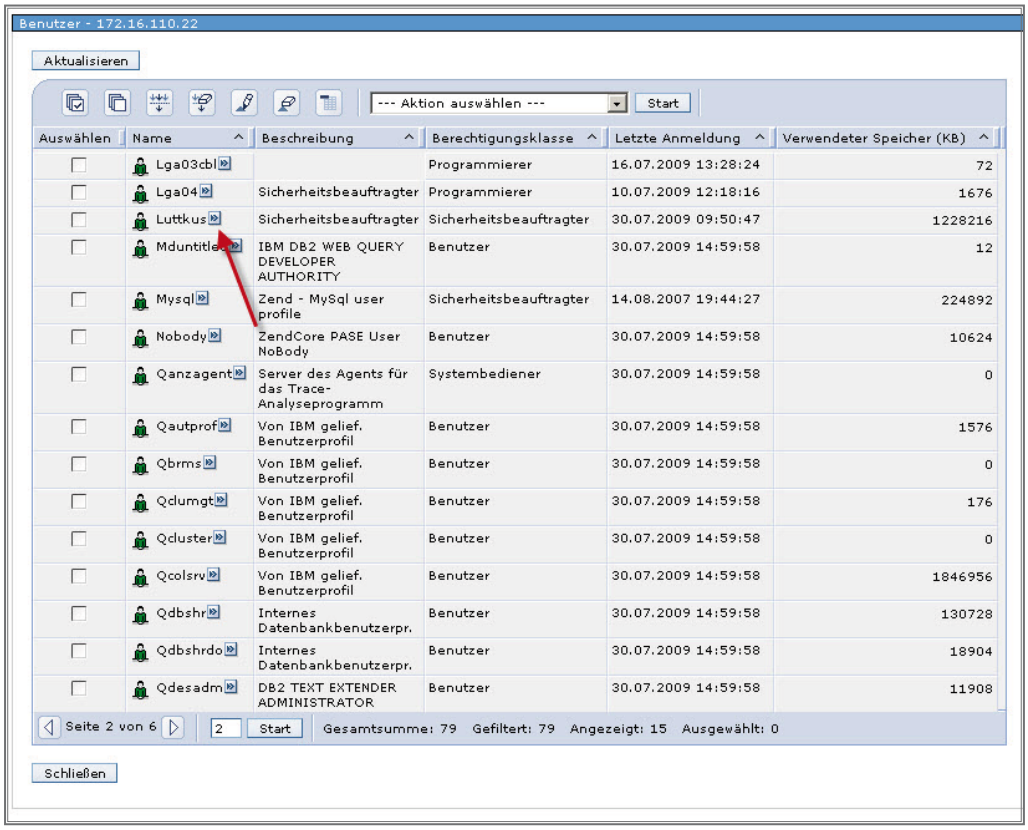
– Gruppe erstellen

– Drag & Drop für Benutzer und Gruppen ausführen

(Sie können beispielsweise einen Benutzer auf „Benutzer ohne Gruppe“ ziehen, um ihn aus allen Gruppen zu entfernen. Sie können einen Benutzer aber auch auf einen Gruppennamen ziehen, um ihn der Gruppe hinzuzufügen, einen Benutzer einem anderen System hinzuzufügen oder den Benutzer in die Gruppe zu verschieben.)

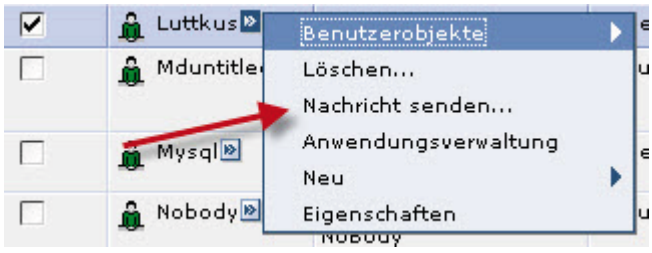
Mit allen Benutzern arbeiten 1

Klicken Sie auf den Link Benutzer. So wird eine Liste aller Benutzer des Systems angezeigt.



Benutzer anzeigen

Ein Klick auf das Kontextmenü-Symbol ermöglicht Ihnen folgende Verarbeitungen vorzunehmen.



Benutzer bearbeiten

- Benutzerobjekte: Mit Hilfe dieses Punktes können Sie mit den Objekten, die einem Benutzer gehören, arbeiten. Mehr dazu in Abschnitt 3.4.7.8.
- Löschen: Löschen Sie einen Benutzer aus dem System.
- Nachricht senden: Senden Sie einem Benutzer eine Nachricht: Diese Funktion kennen Sie bereits aus dem Kapitel Basisoperationen.

7.7.4.2.1
Seite 2

– Anwendungsverwaltung: Beschränken Sie Netzwerk- und Anwendungsfunktionen für einen Benutzer.
– Neu: Kopieren Sie einen Benutzer! Mehr dazu in Kapitel 3.4.7.5.
– Eigenschaften: Lassen Sie sich die Konfiguration für ein Benutzerprofil anzeigen, bzw. ändern Sie ein Benutzerprofil: Mehr dazu in Abschnitt 3.4.7.4.

Mit allen Benutzern arbeiten 2

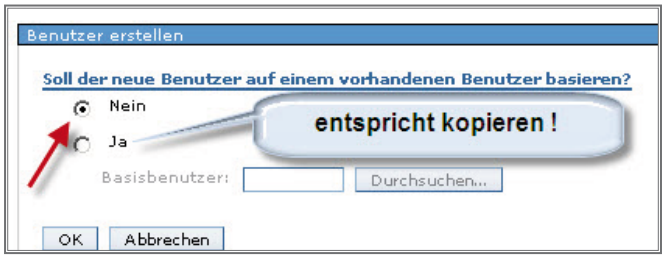
Dieser Punkt ist funktionell identisch mit dem Vorherigen. Der einzige Unterschied besteht darin, dass hier nicht mit allen Benutzern des Systems gearbeitet werden kann, sondern nur mit solchen, die nicht Mitglieder einer Benutzergruppe sind.

7.7.4.2.2

Seite 2

Einen neuen Benutzer anlegen

Um ein neues Benutzerprofil auf Ihrem System anzulegen, folgen Sie dem Link Benutzer erstellen.



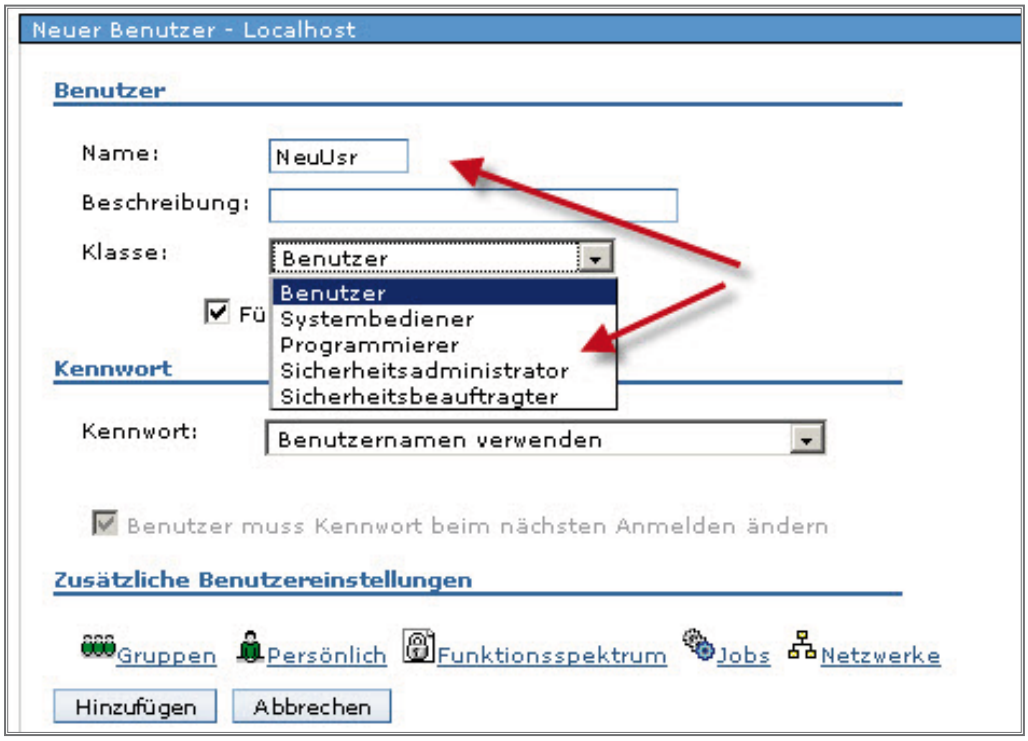
Neuen Benutzer anlegen oder Benutzer kopieren?

Jetzt öffnet sich eine Seite mittels derer Sie einen komplett neuen Benutzer anlegen, wenn Sie die Frage „Soll der neue Benutzer auf einem vorhandenen Benutzer basieren?“ mit Nein beantworten.

Entscheiden Sie sich die Frage mit Ja zu beantworten, dann kopieren Sie einen bestehenden Benutzer.

Ich wähle hier Nein.

Danach öffnet sich ein Dialog, der sich oberflächlich betrachtet mit dem CL-Befehl CRTUSRPRF (Benutzerprofil erstellen) vergleichen lässt. Jedoch hat der Dialog Neuer Benutzer mehr zu bieten als der eben genannte OS/400-Befehl. Dazu gleich mehr. Beginnen wir mit den ersten Feldern dieser Seite.



Startdialog Neuer Benutzer

7.7.4.2.3
Seite 2

In diesem Dialog sind folgende Parameter zu versorgen, die sich von selbst verstehen.

- Benutzername
- Beschreibung
- Kennwort: Über ein Pulldown-Menü lassen sich die Sonderwerte Benutzernamen verwenden (Kennwort ist identisch mit dem Namen des Benutzerprofils) und Kein Kennwort (Es wird ein internes Benutzerprofil erzeugt, das nicht zur Anmeldung am System dient) auswählen. Ansonsten geben Sie hier das gewünschte Kennwort ein.
- Benutzer muss Kennwort bei der nächsten Anmeldung ändern
- Für Verarbeitung aktivieren
- Klasse: Dem Benutzer funktionale Rechte zuordnen

Sie sehen oben, dass ich einen Benutzer NEUUSR anlege, dessen Kennwort ebenfalls NEUUSR ist.

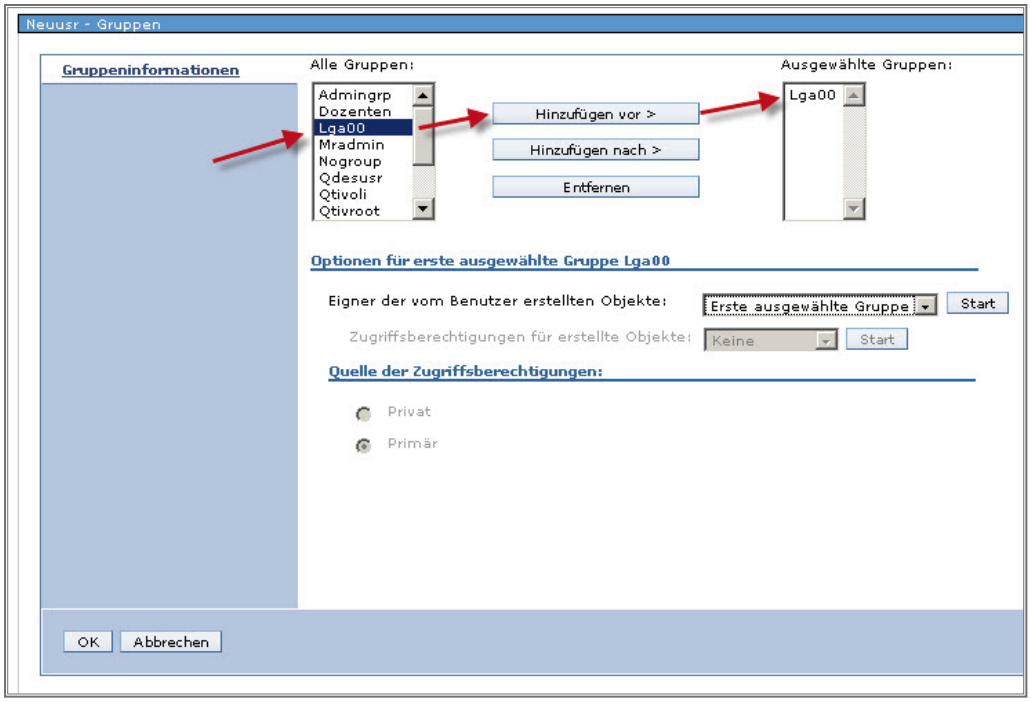
Nach der Versorgung dieser grundsätzlichen Parameter wird die Definition des Benutzers fortgesetzt, indem Sie optional einen der am unteren Dialograndbefindlichen Links klicken.

- Gruppen
- Persönlich
- Funktionsspektrum
- Jobs
- Netzwerke

Gruppen

7.7.4.2.3.1
Seite 1

Jetzt folge ich dem Link Gruppen und der folgende Dialog öffnet sich.



Gruppe zuordnen

Auf der rechten Seite dieses Dialogs sehen Sie alle auf einem System angelegten Gruppenprofile. Um den Benutzer NEUUSR in die Gruppe LGA00 aufzunehmen, markiere ich auf der linken Seite die Gruppe LGA00, drücke dann Hinzufügen vor und die Gruppe LGA00 erscheint daraufhin auf der rechten Seite unter Ausgewählte Gruppen.

Wollen Sie diesen Verarbeitungsschritt wieder rückgängig machen, so markieren Sie rechts unter Ausgewählte Gruppen das betreffende Gruppenprofil und drücken dann entfernen.

Mit Hilfe des Parameters Zugriffsberechtigungen für erstellte Objekte legen Sie fest, ob das angewählte Gruppenprofil die Eignerschaft von Objekten übernehmen soll, die von Mitgliedern einer Gruppe erzeugt werden.

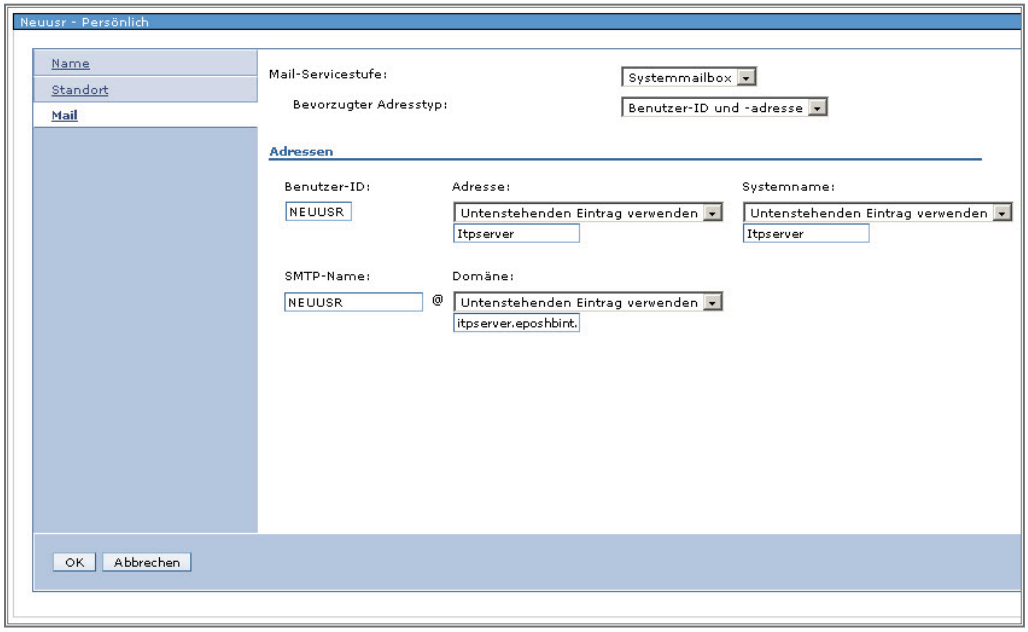
7.7.4.2.3.1

Seite 2

Persönlich

Nachdem der Dialog Gruppen abgeschlossen wurde, folge ich jetzt dem Link Persönlich. Allein das Drücken dieses Knopfes bewirkt, dass auf dem System ein Verzeichniseintrag für den Benutzer NEUUSR angelegt wird. Dieses entspricht der Ausführung des Befehls ADDDIRE (Verzeichniseintrag einem Benutzer hinzufügen).

Im Dialog Persönliche Daten wechsle ich in das Register Mail.



Mail-Adresse erzeugen

Im oben gezeigten Dialog sind die Parameter Mail-Servicestufe, Bevorzugte Adresse, SMTP-Name und Domäne so eingestellt, dass für den Benutzer TESTUSER eine E-Mail-Adresse im Systemadressbuch des Systems angelegt wird.

Die Parameter Benutzer-ID, Adresse und Systemname sind so eingestellt, dass der Benutzer TESTUSER eine korrekte SNA-Adresse im Verteilungsverzeichnis des Systems erhält. Wechseln Sie hier in Systemmailbox, so erhält der Benutzer NEUUSR eine E-Mail-Adresse zugeordnet.

7.7.4.2.3.2

Seite 2

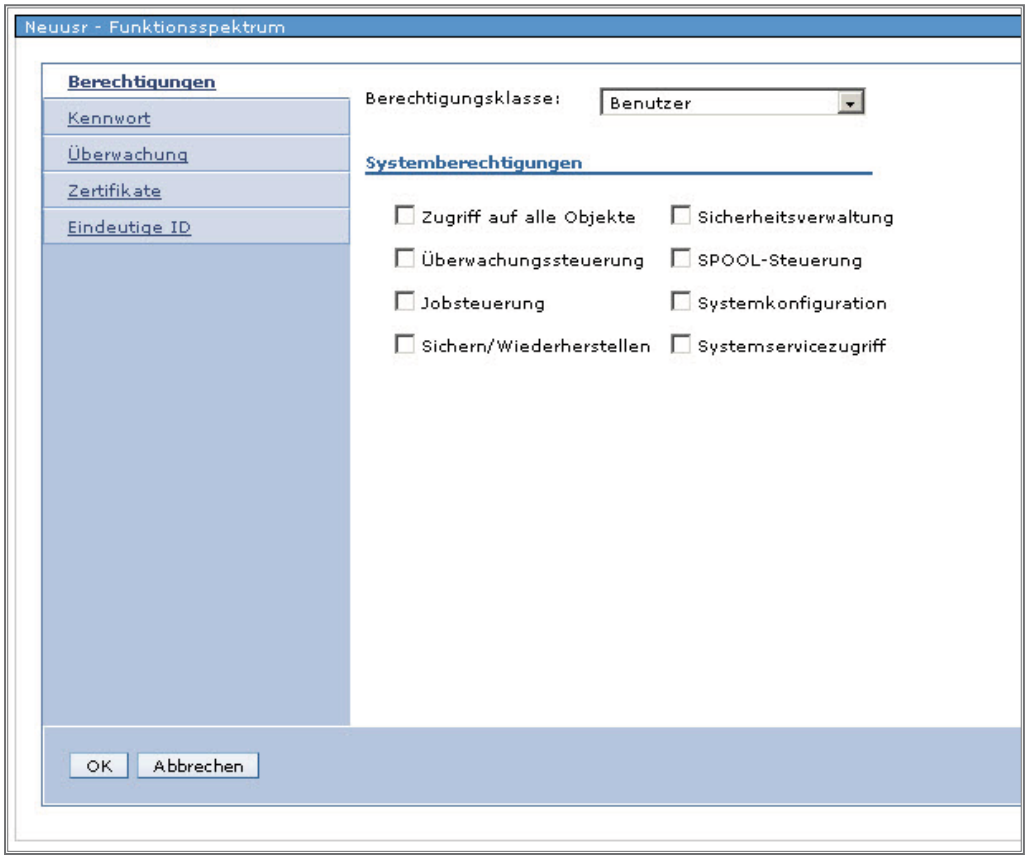
Funktionsspektrum

Danach wechsle ich in den Link Funktionsspektrum in den nachfolgenden Dialog. Hier wird jetzt die funktionale Ausstattung des Benutzers beschrieben.

Dieser Dialog zeichnet sich durch seine aufwendige Registerstruktur aus:

- Berechtigungen
- Überwachungen
- Zertifikate
- Eindeutige ID

Im Register Berechtigungen werden die funktionalen Rechte eines Benutzers beschrieben.



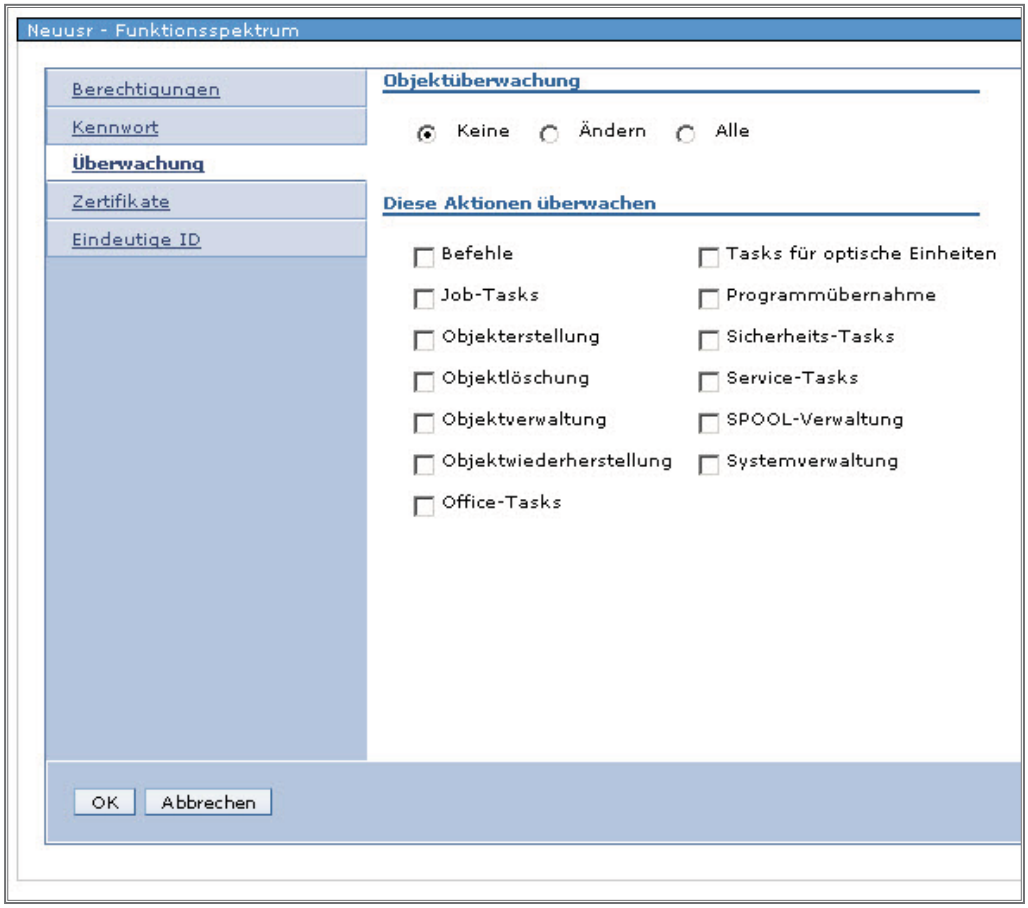
Funktionsspektrum – Berechtigungen

Dieses geschieht durch die Vergabe der Berechtigungsklasse und gegebenenfalls durch die Vergabe von Systemberechtigungen, die im i/OS als Sonderrechte bezeichnet werden.

7.7.4.2.3.3

Seite 2

Das Register Überwachung dient der Protokollierung der Aktivitäten eines Benutzers.

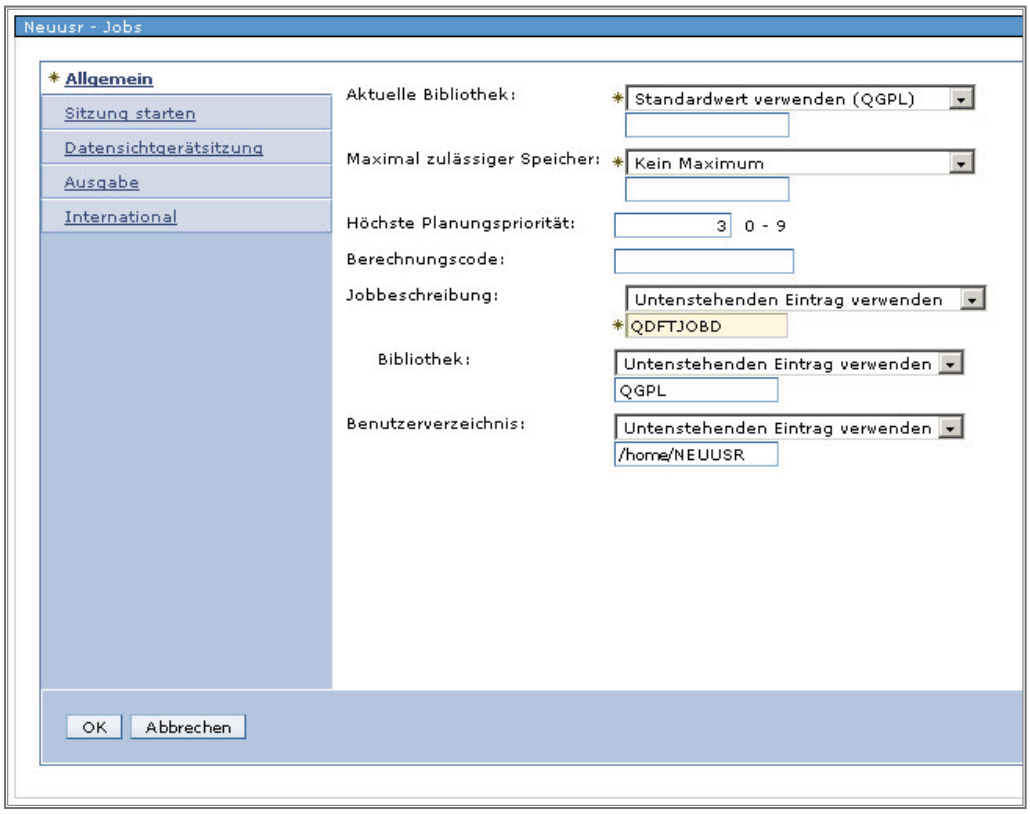


Funktionsspektrum Überwachung

Jobs

Der Dialog Jobs beinhaltet die typischen Parameter, die Sie aus dem Befehl CRTUSRPRF kennen, und bedarf, wie ich meine, keiner näheren Erläuterung.

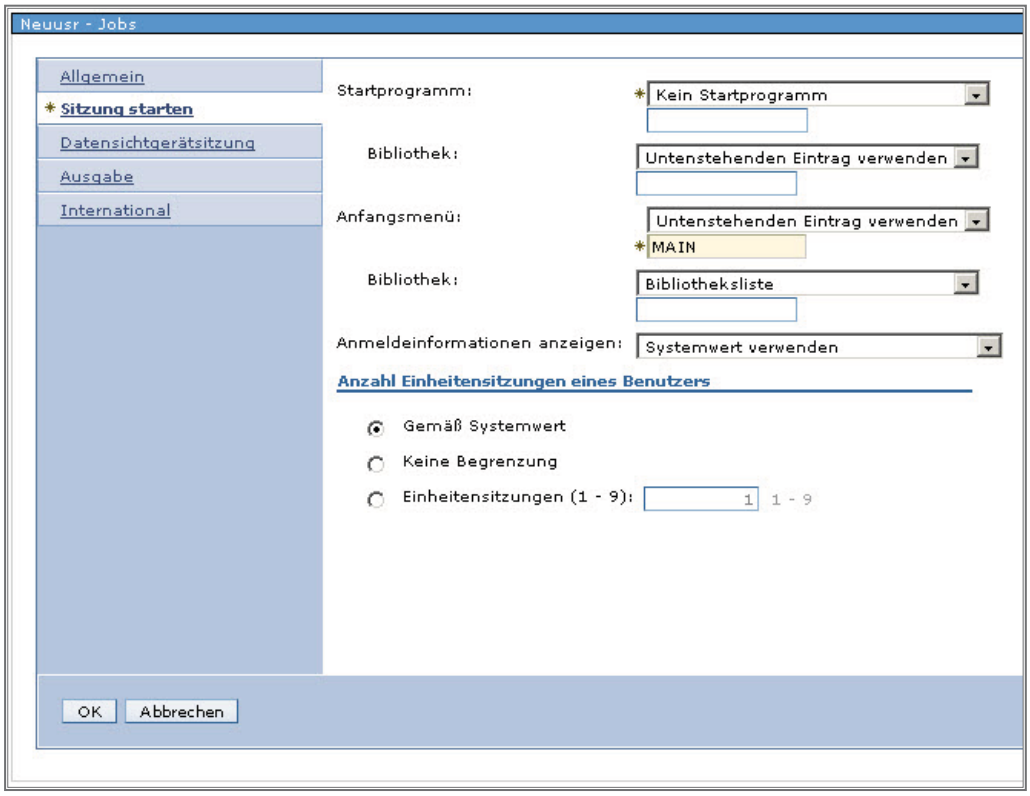
Zur Übersicht zeige ich Ihnen im Folgenden die entsprechenden Register.



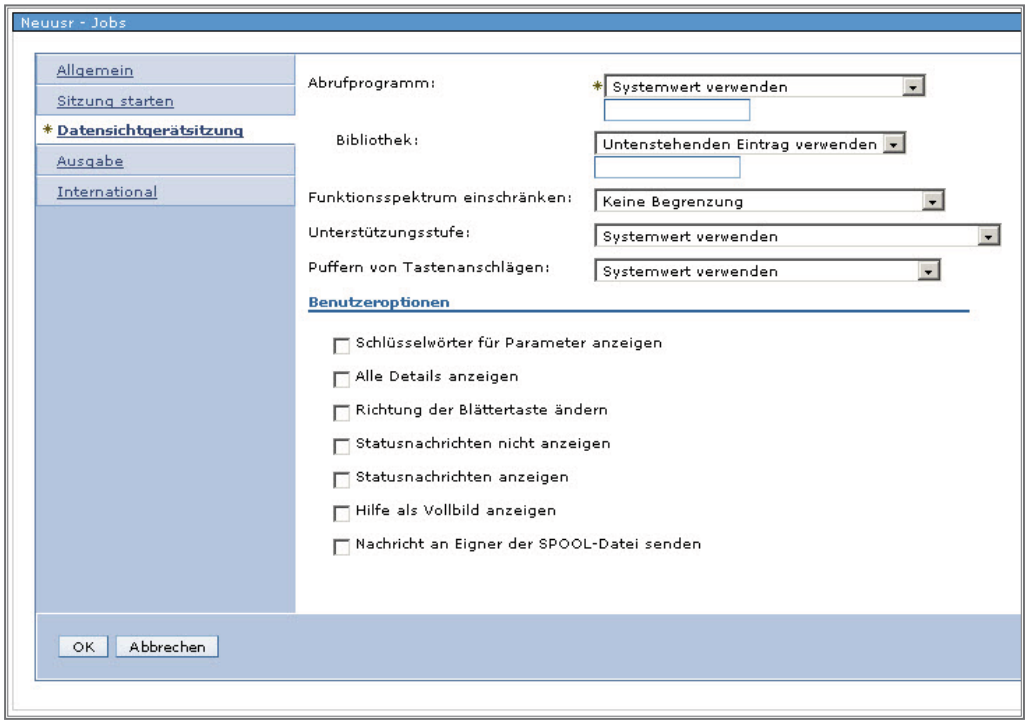
Jobs Allgemein

7.7.4.2.3.4

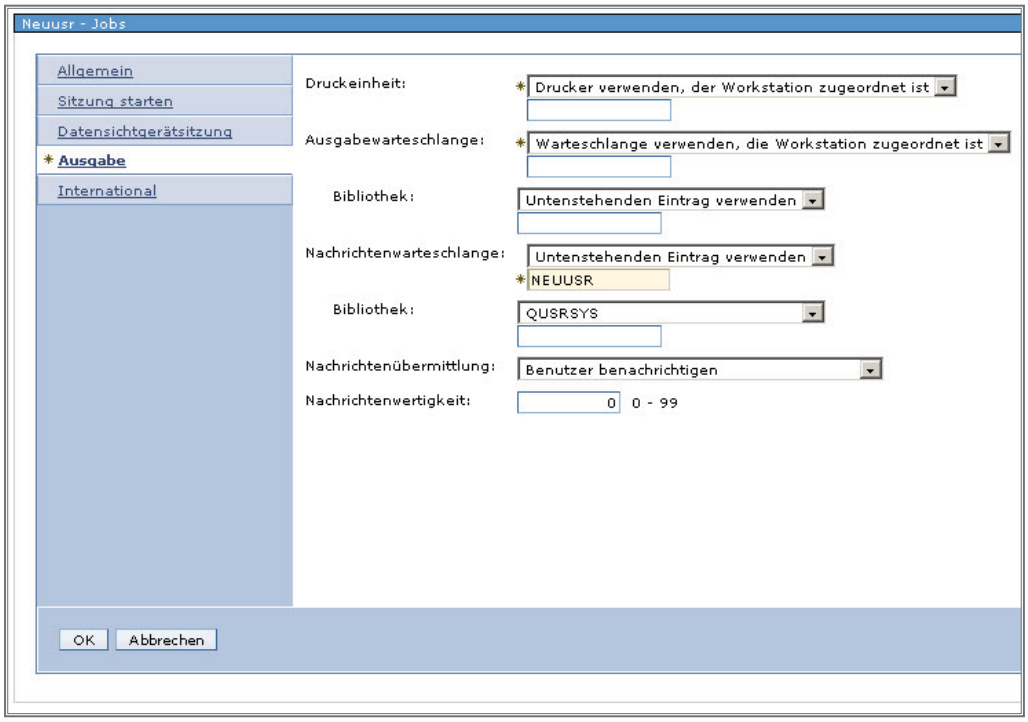
Seite 2



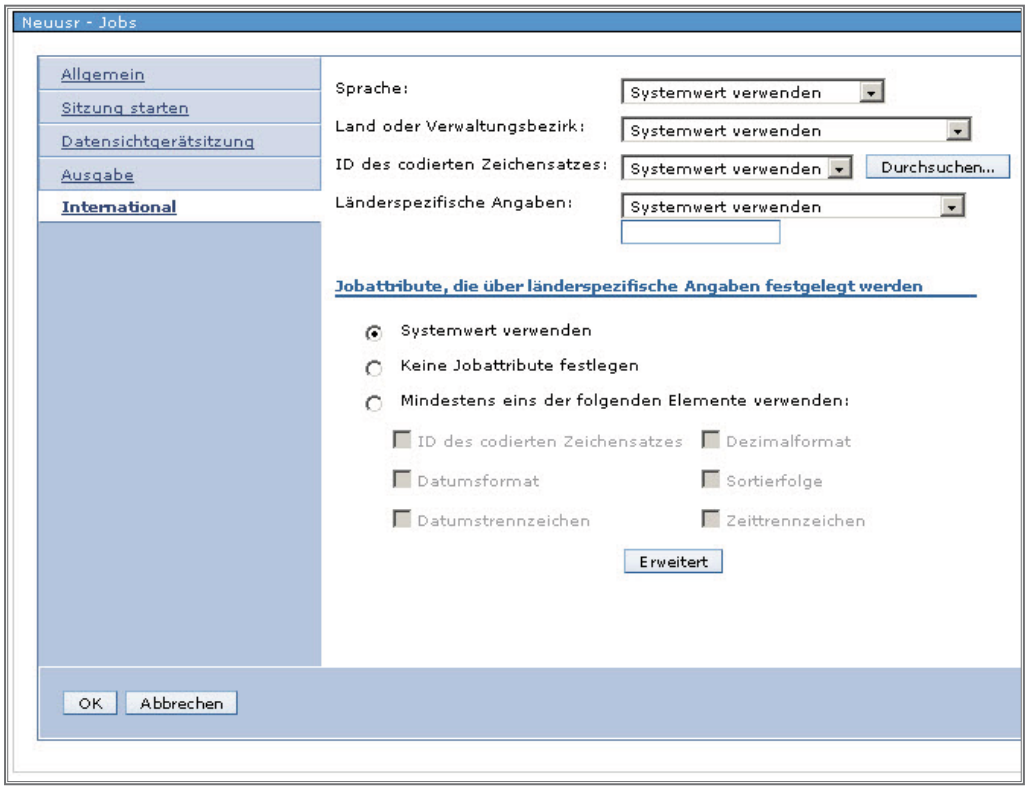
Jobs Sitzung starten



Jobs Datensichtgerätesitzung



Jobs Ausgabe



Jobs International

7.7.4.2.3.4

Seite 4

Wenn Sie einmal die Parameter aller Register der Funktionsjobs mit der Anzeige des Befehls CRTUSRPRF vergleichen, so werden Sie unschwer die Korrespondenzen erkennen.

Einen bestehenden Benutzer ändern

Um einen bestehenden Benutzer zu ändern, folgen Sie einfach dem Link Eigenschaften des zu ändernden Benutzers. Der dann erscheinende Dialog ist identisch mit dem bei der Neuanlage von Benutzern, so dass dieser hier nicht erneut angezeigt werden muss.

7.7.4.2.4

Seite 2

Einen Benutzer kopieren

Ein bestehendes Benutzerprofil zu kopieren, erreichen Sie dadurch, dass Sie das Kontextmenü anfordern und dann die Auswahl Neu basierend auf… wählen.

Anmerkung:

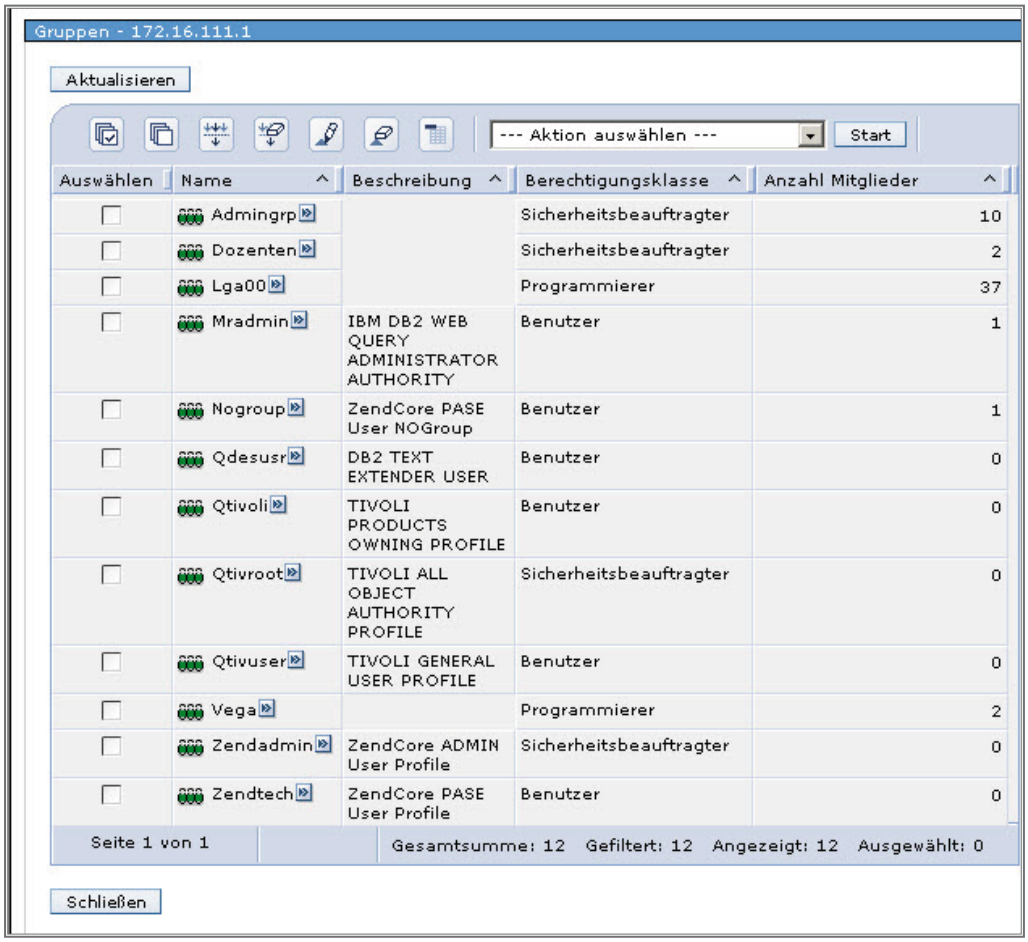
Die Kopie ist nur eine Teilkopie. Es werden so viele Felder wie möglich kopiert. Sie müssen den Benutzernamen angeben und, sofern anwendbar, andere Felder.

7.7.4.2.5

Seite 2

Eine neue Benutzergruppe anlegen

Das Arbeiten mit Benutzergruppen gestaltet sich mit Hilfe des iSeries-Navigators besonders einfach. Im Gegensatz zum i/OS, wo es den Objekttypus des Gruppenprofils nicht gibt, ist hier sehr wohl eine separate Bearbeitung von Benutzerprofilen, die den Status einer Gruppe haben, möglich.



Mit Gruppen arbeiten

Folgen Sie einfach dem Benutzer und Gruppen ➜ Gruppen und Sie sehen die bereits erstellten Gruppenprofile Ihres Systems.

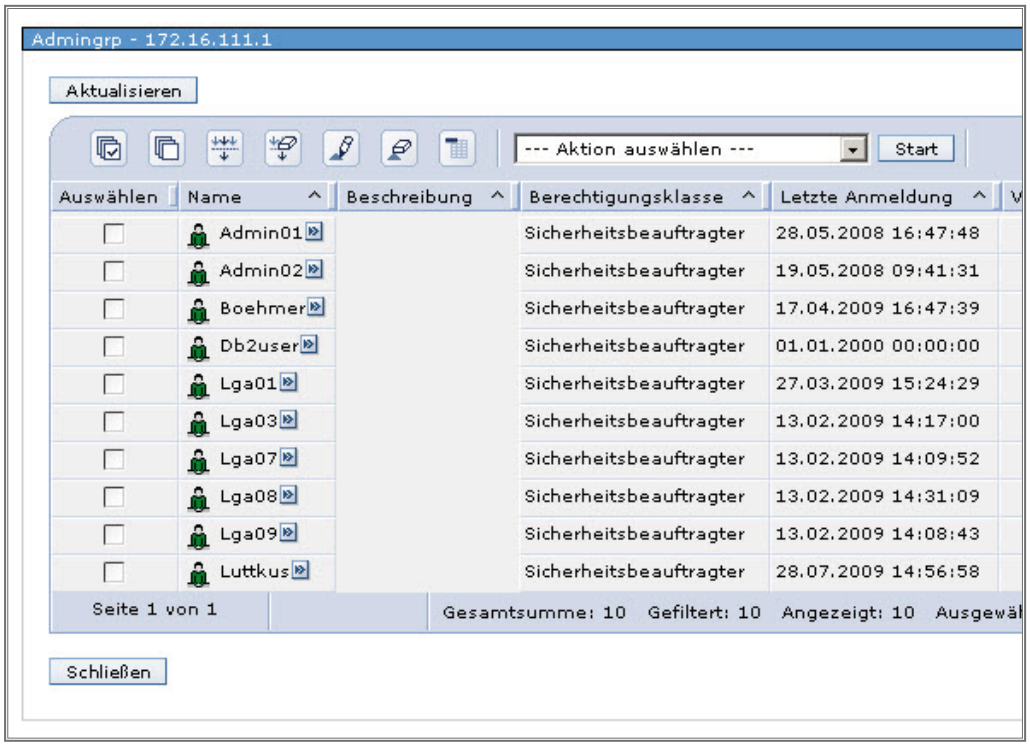
Folgen Sie im Kontextmenü der Auswahl öffnen und Sie sehen in der Detailanzeige die Gruppenmitglieder.

Die Gruppenmitglieder

7.7.4.2.6
Seite 2

Mit Hilfe des angeforderten Kontextmenüs können Sie sodann das Profil eines Gruppenmitglieds ändern.

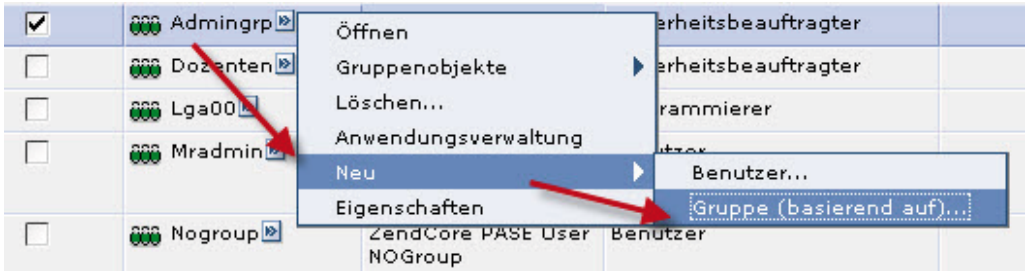
Um eine neue Gruppe anzulegen, öffnen Sie das Kontextmenü für Gruppen in der Gruppenprofilliste und wählen Neue Gruppe (basierend auf)... .



Neue Gruppe erstellen

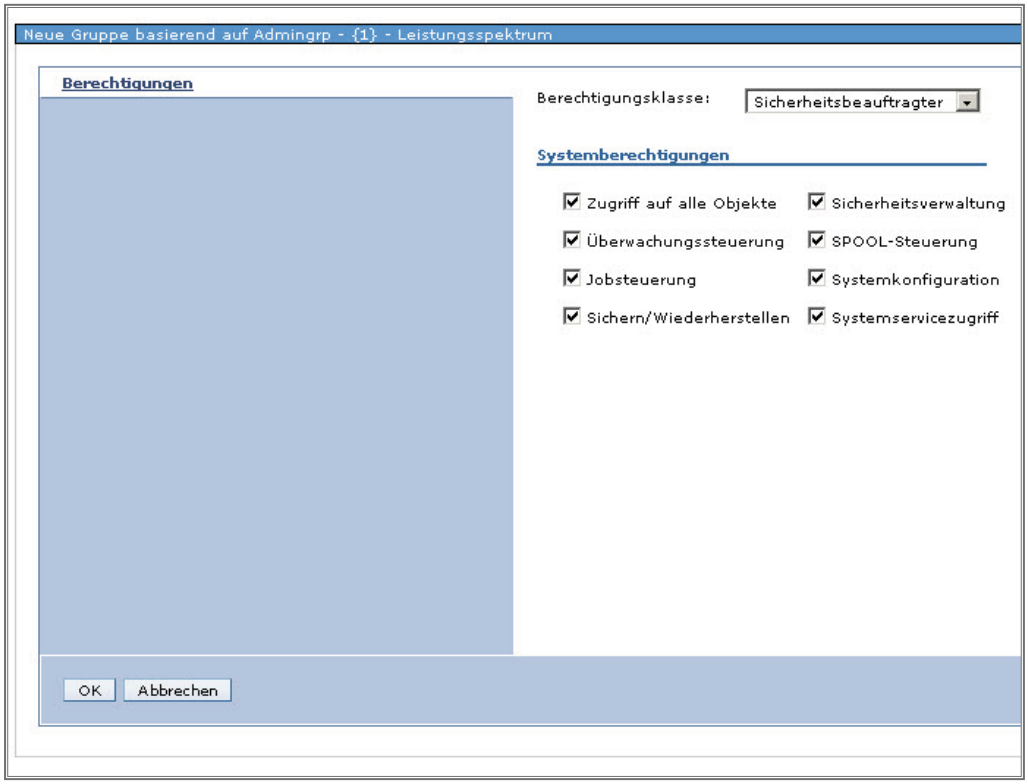
Dieses führt Sie in den zentralen Dialog, der es Ihnen ermöglicht, eine neue Benutzergruppe zu erstellen, dieser Gruppe Benutzer und der Gruppe funktionale Rechte zuzuordnen.

Geben Sie der neuen Gruppe einen entsprechenden Namen. In der Liste Alle Benutzer finden Sie die Benutzer, die Sie der neuen Gruppe zuordnen wollen. Markieren Sie den gewünschten Benutzer und klicken Sie dann auf hinzufügen. Damit ist der Benutzer in die neue Gruppe übernommen worden.



Dialog Neue Gruppe

Klicken Sie auf Funktionsspektrum, so gelangen Sie in einen Dialog, den Sie bereits aus der Neuanlage einzelner Benutzer kennen.



Neue Gruppe Funktionsspektrum

Auch hier finden Sie die Zuordnung funktionaler Rechte in Form von Berechtigungsklassen und Systemberechtigungen.

7.7.4.2.6

Seite 4

Benutzergruppen ändern

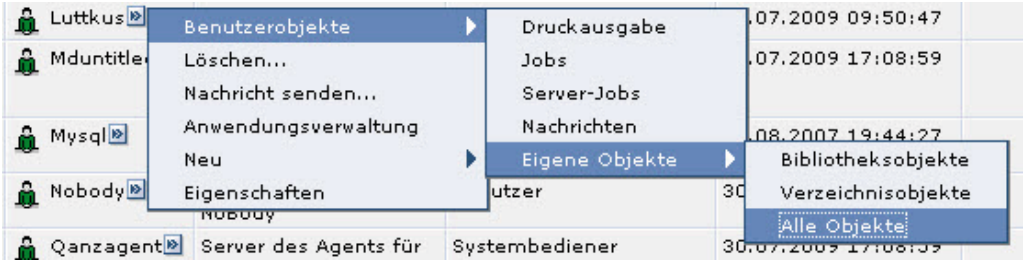
Um eine bestehende Benutzergruppe zu ändern, lassen Sie sich einfach das Kontextmenü für die entsprechende Gruppe anzeigen und wählen die Option Eigenschaften aus. Der dann erscheinende Dialog ist identisch mit dem bei der Neuanlage von Gruppen, so dass er hier nicht erneut angezeigt werden muss.

7.7.4.2.7

Seite 2

Mit Benutzerobjekten arbeiten

Auch die Arbeit mit Objekten, die einem Benutzer gehören, gestaltet sich mit Hilfe des Navigators relativ einfach. Klicken Sie einfach im Kontextmenü für einen Benutzer oder eine Benutzergruppe auf die Auswahl Benutzerobjekte.



Mit Benutzerobjekten arbeiten

In dem gezeigten Submenü folgen Sie den Links Eigene Objekte ➜ Alle Objekte.

Aus diesem Menü heraus können Sie mit folgenden Objekten eines Benutzers arbeiten:

– Mit Bibliotheksobjekten

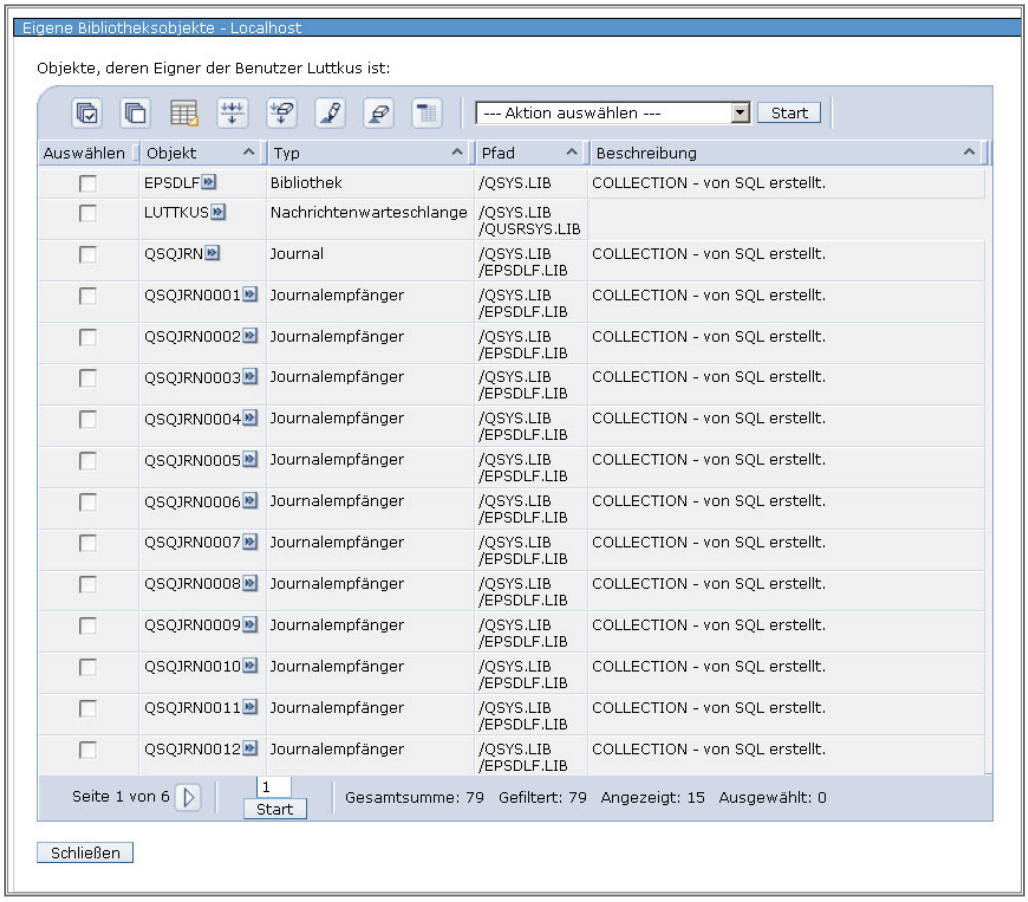
– Mit Verzeichnisobjekten

– Mit allen Objekten

7.7.4.2.8

Seite 2

Sofort erscheint folgender Dialog:



Anzeige Eigene Objekte

Konfiguration und Service

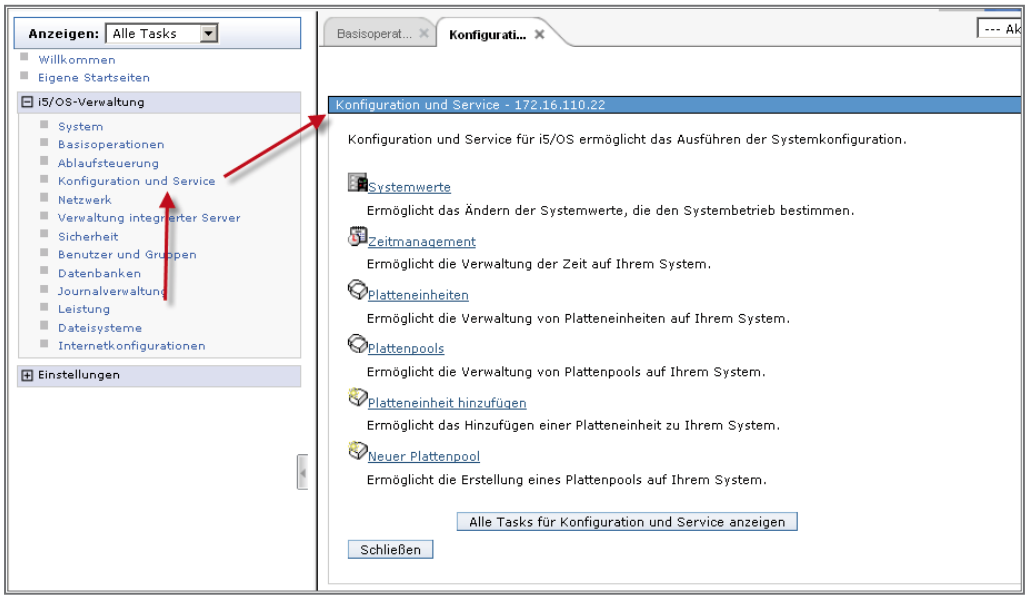
„Konfiguration und Service“ stellt eine Vielzahl von Funktionen bereit, um mit Hardware zu arbeiten.

Sie können

– mit Systemwerten,

– mit Zeiteinstellungsfunktionen

– und mit den Speicherplatten des Systems arbeiten.



Konfiguration und Service

Folgende Funktionsbereiche sind realisiert. Zum leichteren Verständnis benenne ich in der Folge die CL-Befehle, die diesen Funktionsgruppen am ehesten nahekommen.

– Systemwerte anzeigen und ändern
  WRKSYSVAL – Mit Systemwerten arbeiten

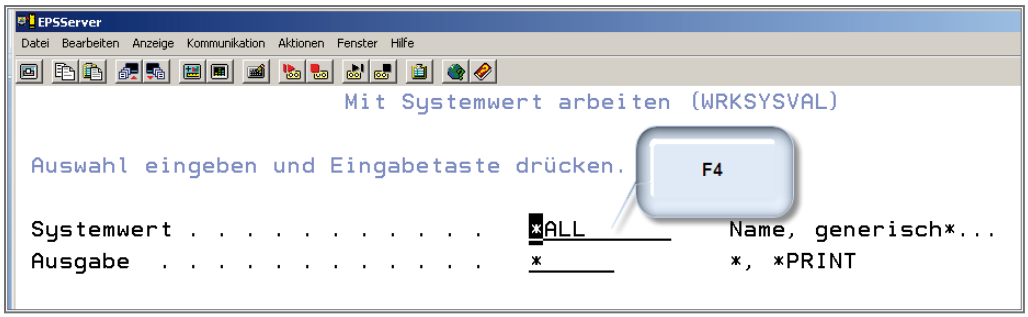
– Mit System Service Tools (SST) arbeiten
  STRSST

7.7.4.3
Seite 2

Systemwerte

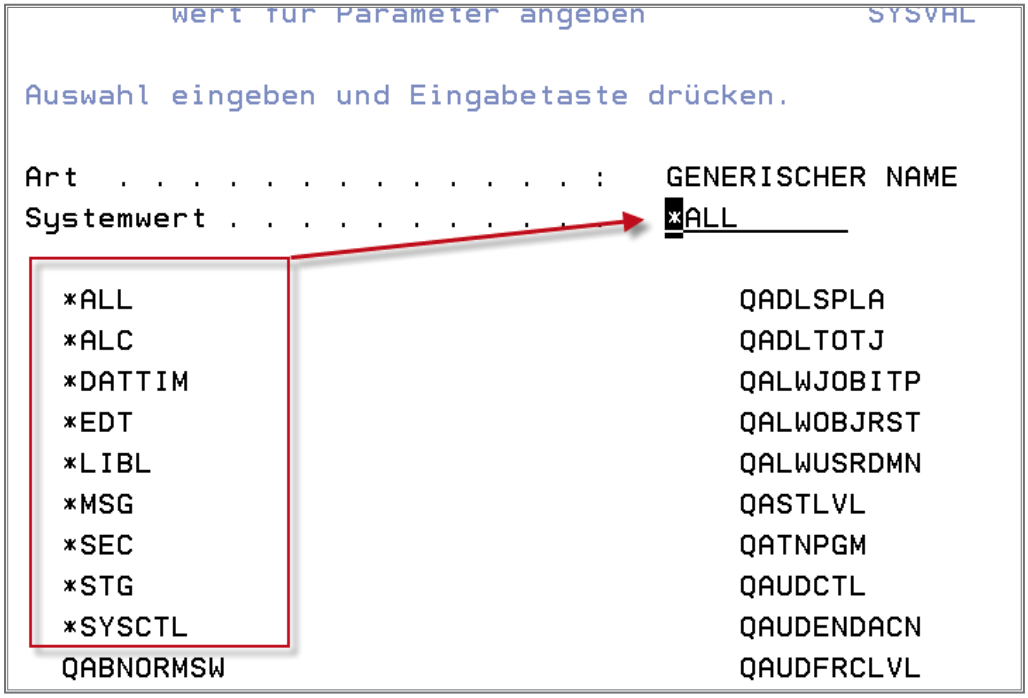
Systemwerte sind Informationselemente, die die Betriebsumgebung im gesamten System beeinflussen. Ein Systemwert enthält Steuerungsinformationen für den Betrieb bestimmter Komponenten des Systems. Ein Benutzer kann die Systemwerte ändern, um z. B. die Arbeitsumgebungen für Jobs zu definieren. Das Systemdatum und die Bibliotheksliste sind Beispiele für im System global hinterlegte Systemwerte.

Im Navigator werden Systemwerte in Funktionsgruppen zusammengefasst. Hierbei geht der Navigator weit über das Gruppenraster hinaus, das uns etwa aus dem Befehl WRKSYSVAL bekannt ist:



Systemwertgruppen anfordern

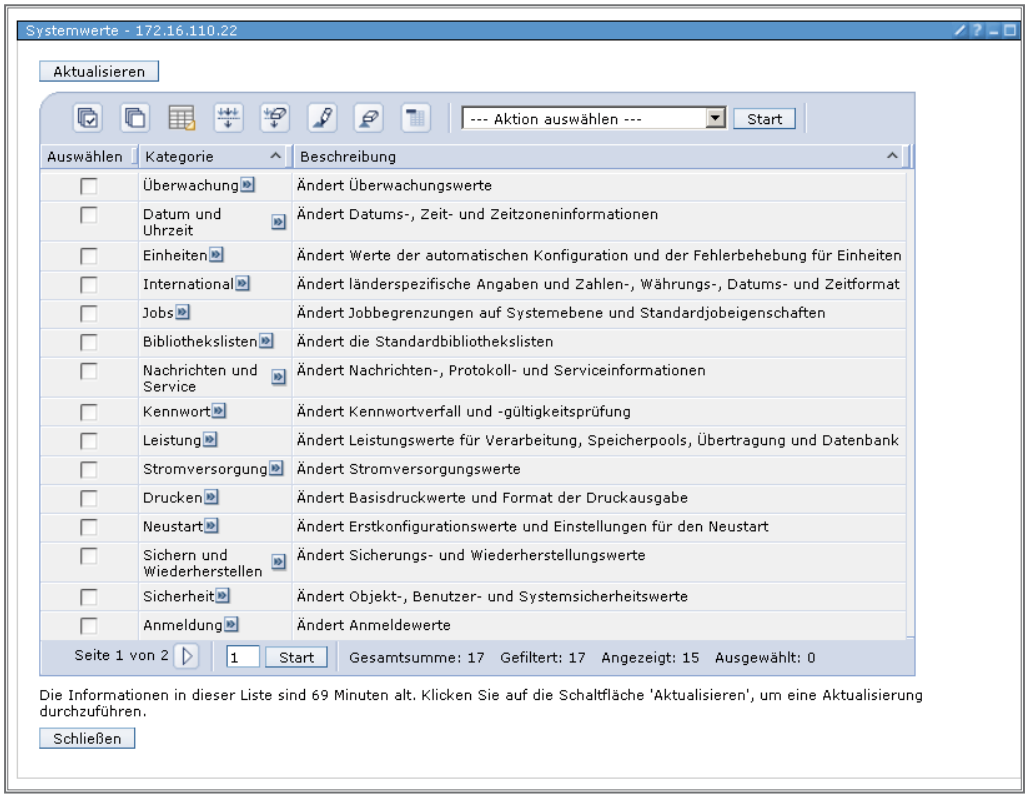
Fordern Sie in der 5250-Umgebung für den Befehl WRKSYSVAL die Systemwertliste an, so erhalten Sie folgende Ausgabe:



Systemwertgruppen

7.7.4.3.1
Seite 2

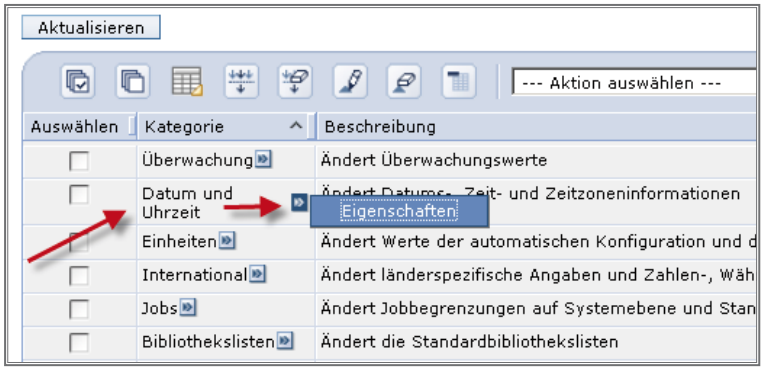
Vergleichen Sie einfach diese Liste mit dem unten gezeigten Dialog.



Systemwertgruppen im Navigator

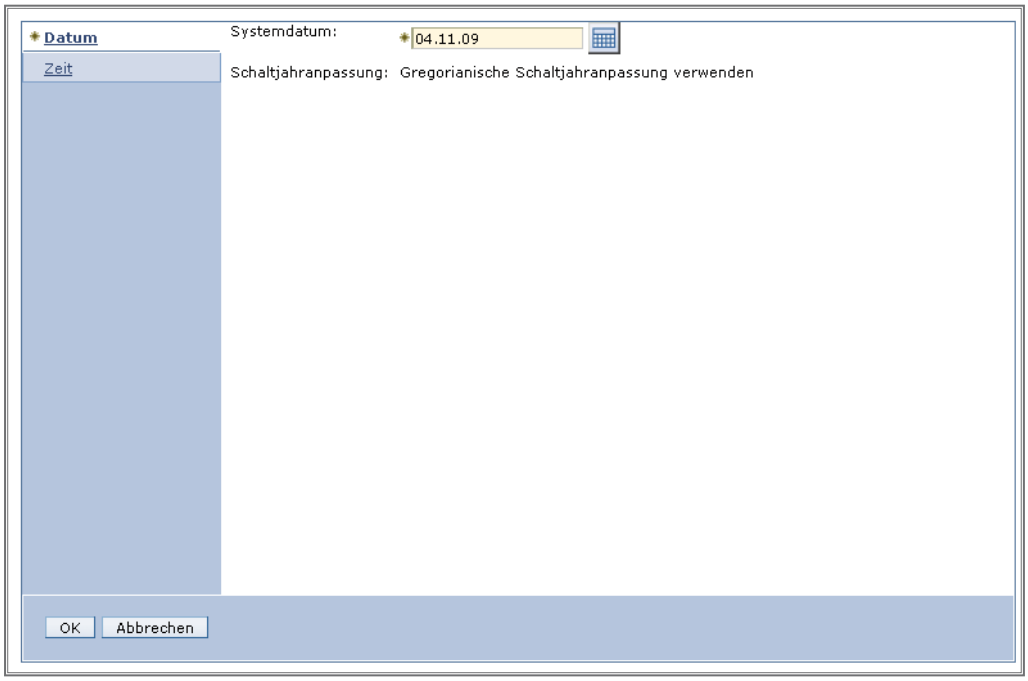
Es ist sicherlich leicht zu erkennen, dass hier eine differenzierte Gruppierung der Systemwertfunktionen vorgenommen wurde.

Ich werde jetzt nicht alle oben gezeigten Gruppen mit Ihnen zusammen durchgehen, sondern nur stellvertretend einige Gruppen kursorisch darstellen.



Datum und Uhrzeit

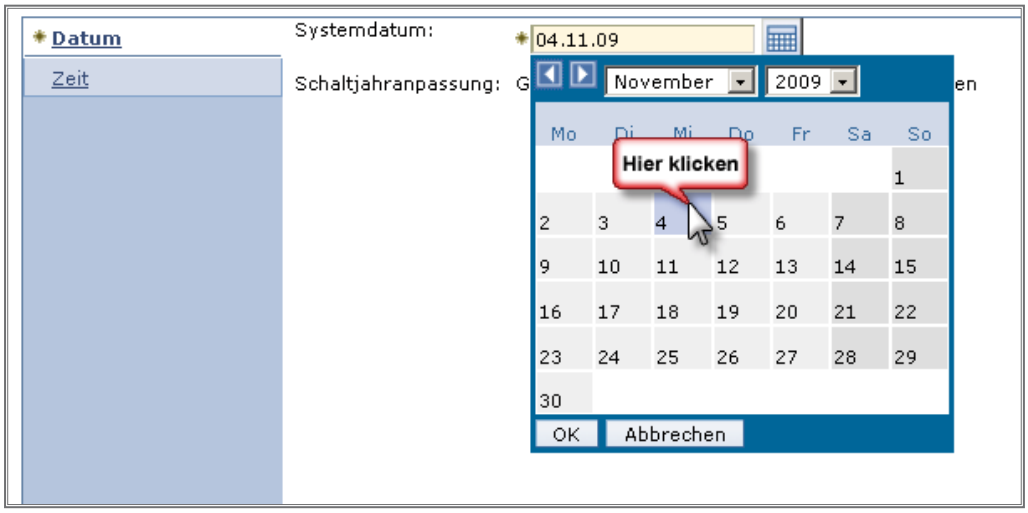
Ich öffne zunächst die Systemwertfunktion „Datum und Uhrzeit“.



Datums- und Zeitwerte: QDATE

Hier sehen Sie die Navigator-Darstellung des Systemwerts QDATE.

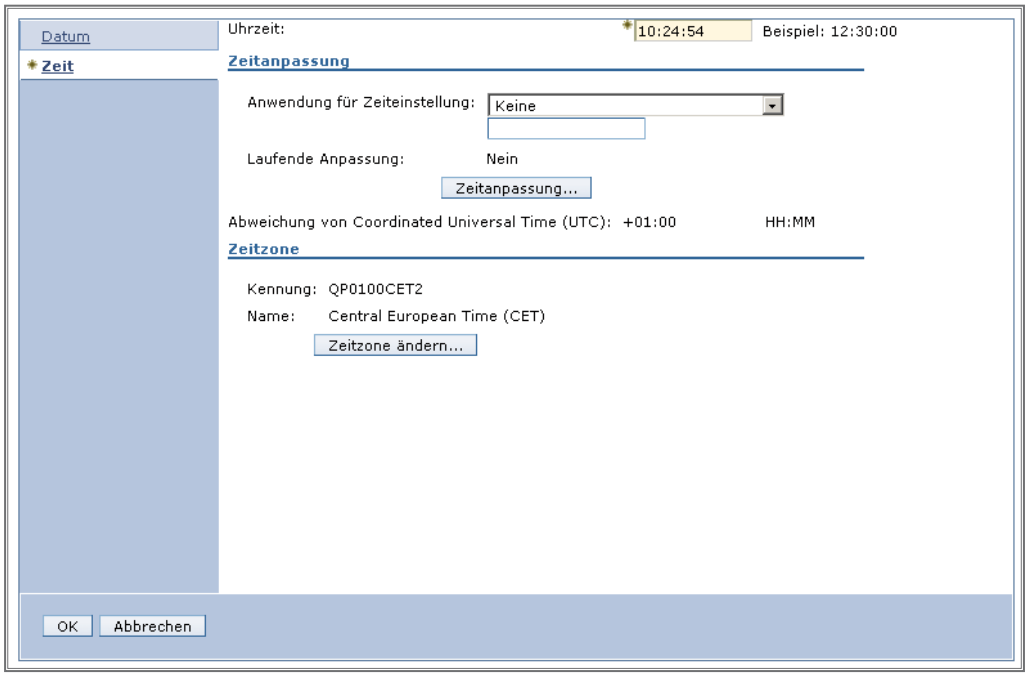
Sie müssen das Systemdatum anpassen.



Systemdatum ändern

So fordern Sie den Kalender an und wählen das entsprechende Datum aus.

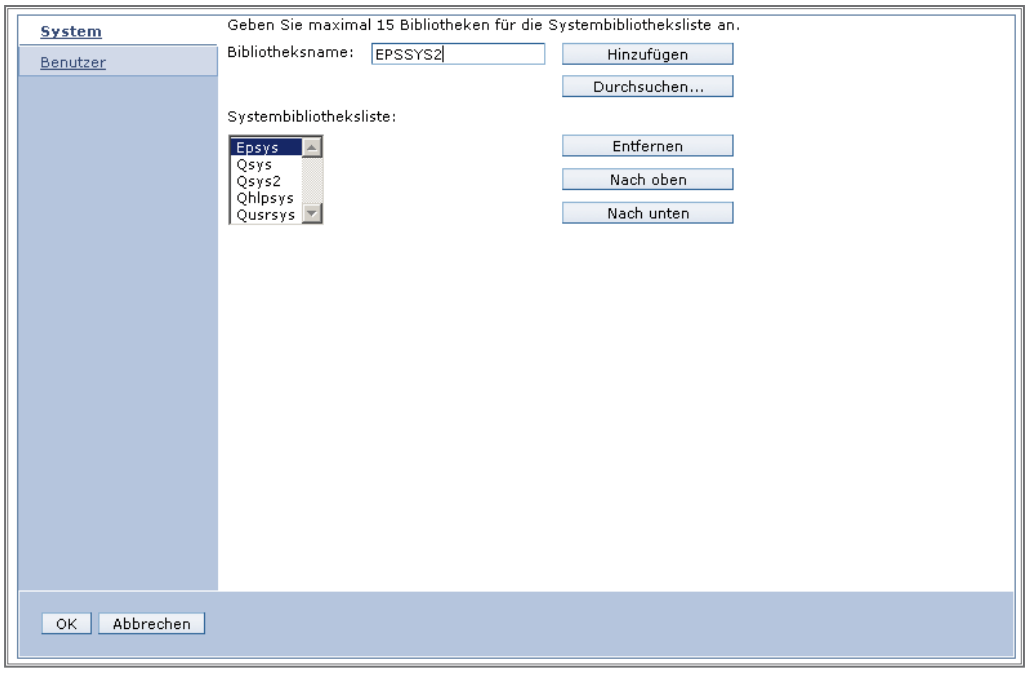
7.7.4.3.1
Seite 4



Datums- und Zeitwerte: QTIMADJ und QTIMZON

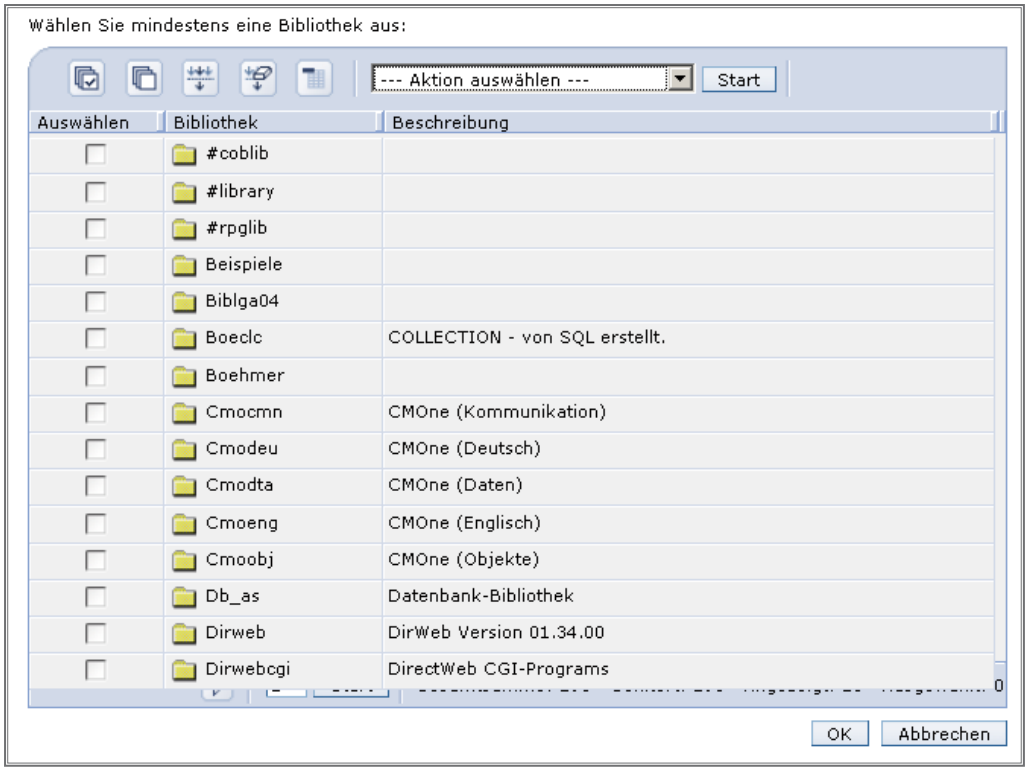
Schauen Sie sich oben den Dialog für die Veränderung der Systemwerte QTIMAD (Zeitanpassung) und QTIMZON (Zeitzone) an. In letzterem geben Sie die zeitliche Verschiebung zur Greenwich-Standardzeit an und aktivieren die Sommer-/Winterzeitanpassung.

Jetzt bearbeiten wir die Systembibliothekssuchliste.



Systemwert QSYSLIBL

Hinter dem gezeigten Dialog steht die Bearbeitung des Systemwerts QSYSLIBL. Soll der Systemteil der Bibliothekssuchliste geändert werden, so geben Sie den Namen der gewünschten Bibliothek in den Parameter „Bibliotheksname“ ein und drücken dann „Hinzufügen“. Ist Ihnen der Name der Bibliothek jedoch nur teilweise oder auch gar nicht bekannt, so gehen Sie in „Durchsuchen...“ und folgender Dialog öffnet sich.



Bibliothek suchen

Wählen Sie aus der gezeigten Liste „Ihre“ Bibliothek aus. Es werden hier nur die Bibliotheken des Systems angezeigt, für die Sie ein Leserecht besitzen.

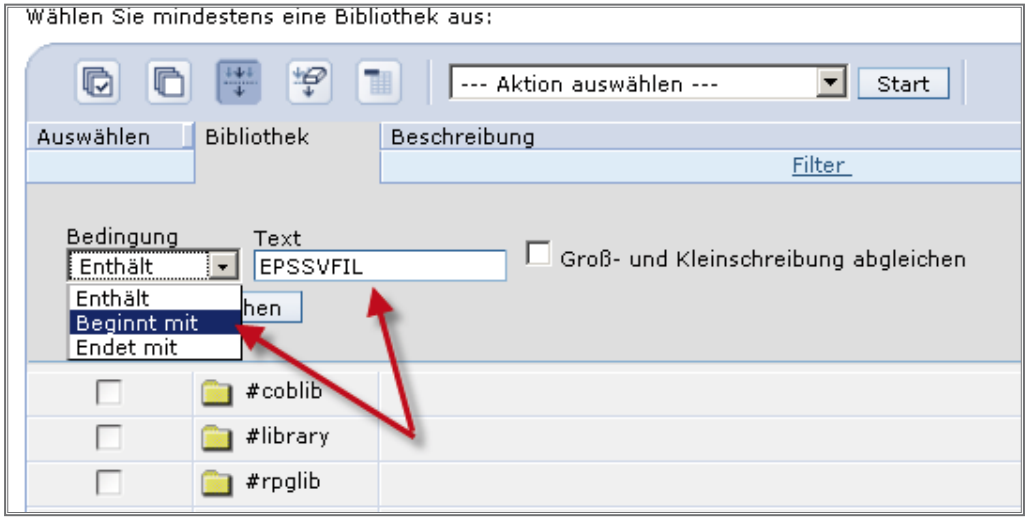
Nachdem Sie die betreffende Bibliothek markiert haben, beenden Sie den Dialog mit „OK“.

Mit Hilfe der Filterfunktion lassen sich auch Bibliotheken finden, von denen nur ein Teil des Namens bekannt ist.

7.7.4.3.1
Seite 6

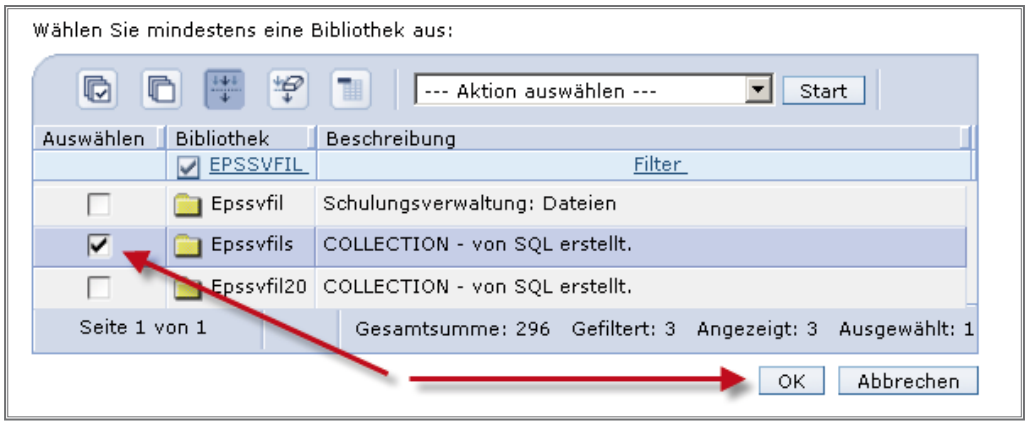
Bibliothek finden

Nach einem Klick auf das Filtersymbol öffnet sich die Filterzeile. Ein Klick auf „Filter“ öffnet den Filterdialog.



Filterwert und Bedingung

Ein Klick auf „OK“ bewirkt die Anzeige der gesuchten Bibliotheken.



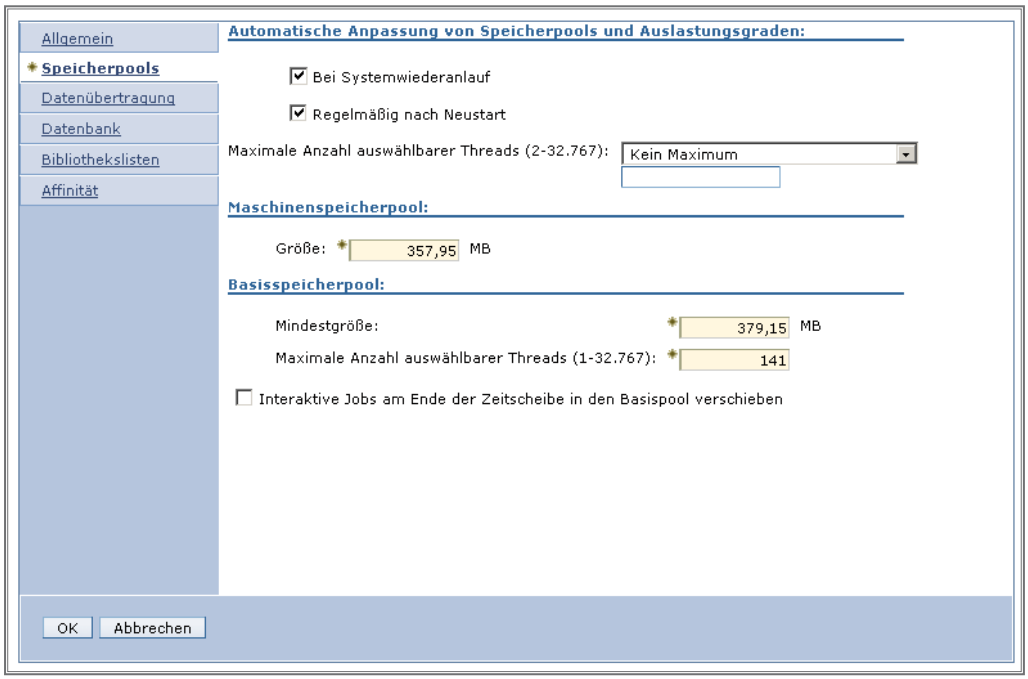
Bibliothek auswählen

Markieren Sie die Bibliothek und beenden Sie den Dialog mit „OK“.

Danach finden Sie die ausgewählte Bibliothek im Feld „Systembibliotheksliste“ angezeigt.

Um mit performancerelevanten Systemwerten zu arbeiten, öffnen Sie die Funktion „Systemwertegruppe Leistung“.

Im unten gezeigten Beispieldialog habe ich das Register „Speicherpools“ geöffnet.



Leistungssystemwerte

Hier können u. a. folgende Systemwerte gesetzt bzw. verändert werden:

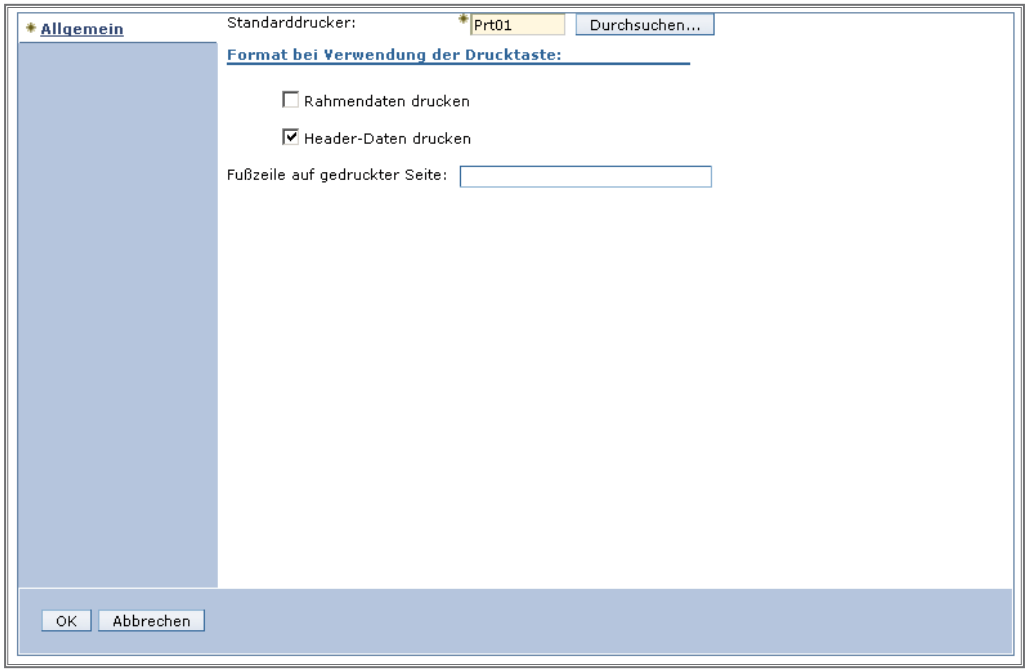
– QMAXACTLVL – maximal auswählbare Threads

– QMCHPOO – Größe des Maschinenspeicherpools

– QBASPOOL – Basisspeicherpool

7.7.4.3.1
Seite 8

Die globale Einstellung für einen Wechsel des Systemdruckers der iSeries finden Sie in der Funktion „Drucken“.



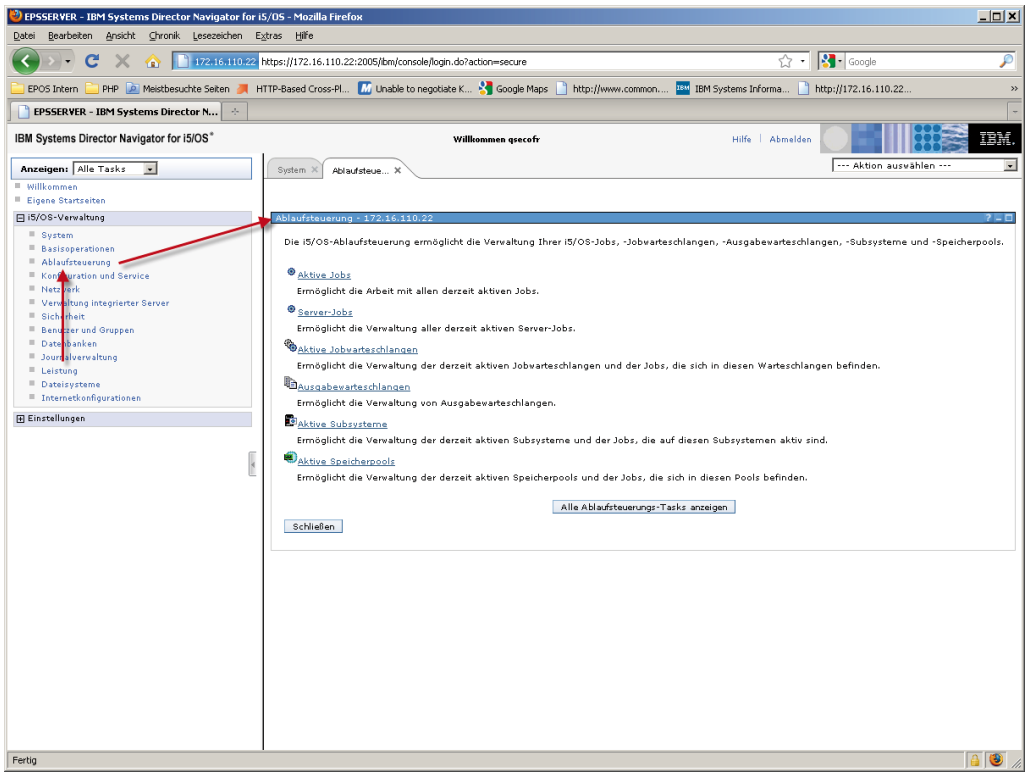
Drucksystemwerte

Dieser Dialog entspricht dem Systemwert QPRTDEV.

Ablaufsteuerung

Der Bereich der Ablaufsteuerung ist im Rahmen des Navigator verantwortlich für die Überwachung und Steuerung alle Prozess- und Jobvorgänge im i/OS. Der Begriff Ablaufsteuerung ist im Deutschen für diesen Sachzusammenhang ein missverständlicher Ausdruck. Im amerikanischen Original des iSeries Navigator wird diese Funktion mit dem Begriff „Work Management" dargestellt. Dieser Begriff ist für mich deutlich aussagekräftiger als die deutsche Übersetzung, wird hier doch genau beschrieben, was der Anwender zu erwarten hat. Dieser Funktionsbereich beschäftigt sich nämlich mit der Steuerung von Jobs im System. Funktionen wie WRKACTJOB (Mit aktiven Jobs arbeiten), WRKUSRJOB (Mit Benutzerjobs arbeiten) etc. dienen diesem Bereich weitgehend als Vorlage.

Welche Funktionsbereiche sind hier zu finden? Ich werde sie einfach durch CL-Befehle erläutern, obwohl Sie sich darüber im Klaren sein sollten, dass dieser Bereich teilweise über die Funktionalitäten der CL-Befehle hinausgeht.



Funktionen der Ablaufsteuerung

7.7.4.4
Seite 2

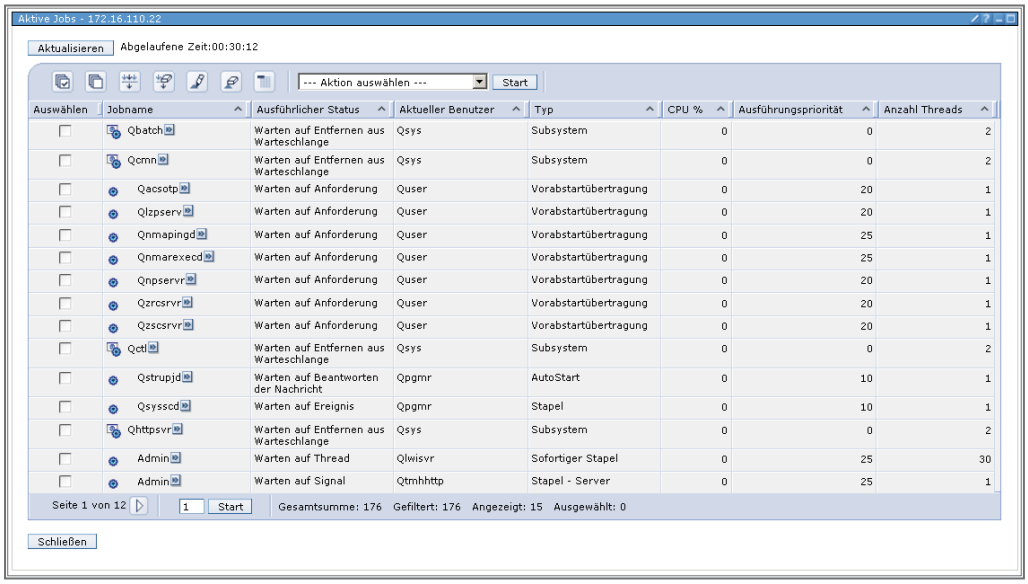
Folgende Funktionen finden wir in der Ablaufsteuerung:

- Aktive Jobs – Mit aktiven Jobs arbeiten (WRKACTJOB)
- Server-Jobs – Mit aktiven Server-Jobs arbeiten
  Dies ist ein WRKACTJOB nur für die Server-Jobs der iSeries.
- Aktive Jobwarteschlangen – Mit Jobwarteschlangen arbeiten (WRKJOBQ)
- Ausgabewarteschlangen – Mit Ausgabewarteschlangen arbeiten (WRKOUTQ)
- Aktive Subsysteme – Mit Subsystemen arbeiten (WRKSBS)
- Aktive Speicherpools – Mit Speicherpools arbeiten
  Eine Mischung aus WRKSYSSTS (Mit dem Systemstatus arbeiten) und WRKSHRPOOL (Mit gemeinsamen Speicherbereichen arbeiten).

Mit aktiven Jobs arbeiten

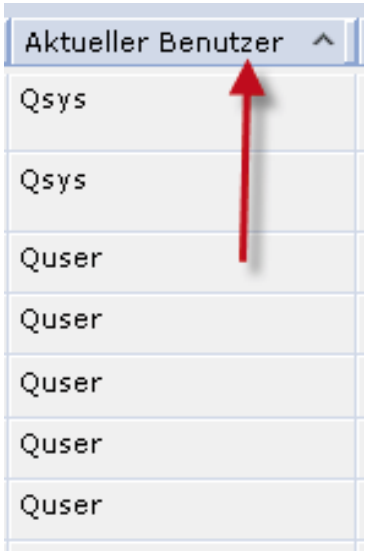
Für gestandene System-Administratoren birgt dieser Verarbeitungspunkt keine Geheimnisse. Wer sich mit der Funktionsweise des CL-Befehls WRKACTJOB (Mit aktiven Jobs arbeiten) auskennt, dem bietet sich in dieser Funktion ein vertrautes Umfeld. Trotzdem werde ich diesen Bereich zusammen mit Ihnen durchgehen.

Nach einem Klick auf den Eintrag „Aktive Jobs“ baut sich im Detailfenster auf der rechten Seite des Navigators das vertraute Bild der Anzeige WRKACTJOB auf.



Mit aktiven Jobs arbeiten

Linksbündig dargestellte Einträge repräsentieren die Subsysteme – präzise ausgedrückt: Subsystem-Monitor-Jobs. Eingerückt sehen Sie die unter der Kontrolle eines Subsystems laufenden Jobs.

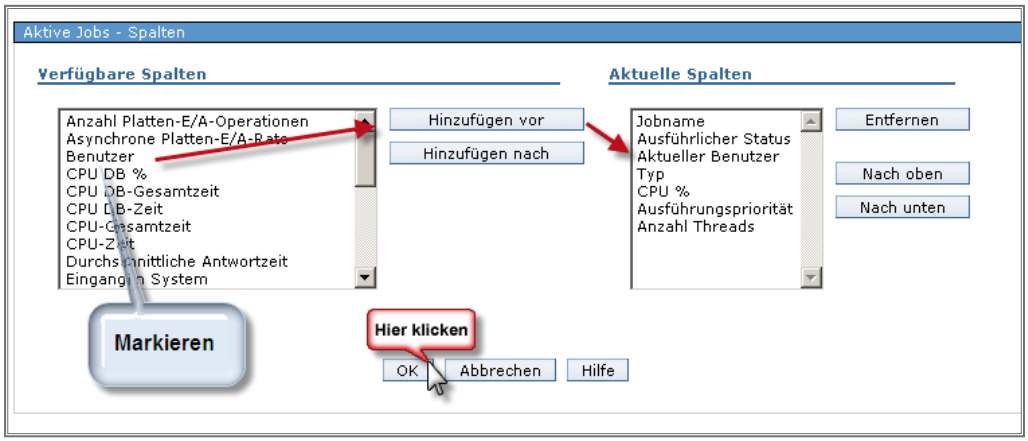


Sortierung ändern

7.7.4.4.1
Seite 2

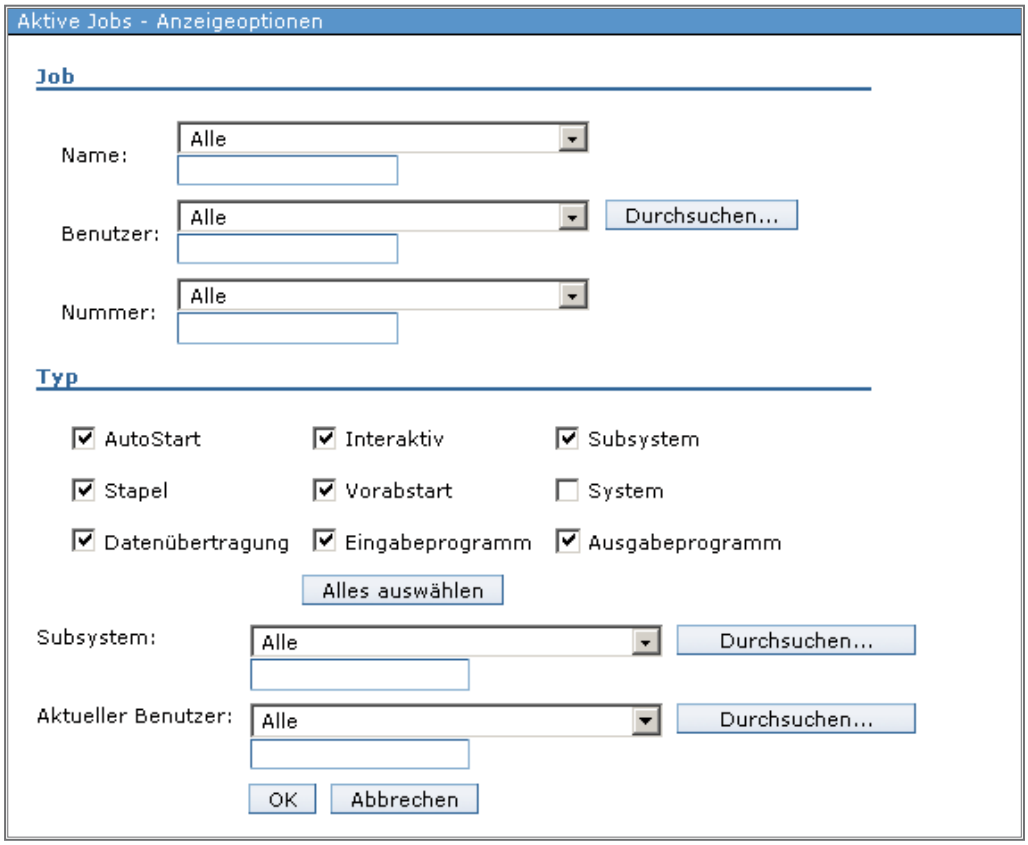
Die Sortierfolge lässt sich durch einen Klick in die Überschrift einer Tabellenspalte aufsteigend bzw. absteigend sortieren.

Die Spaltenanzeige verändern Sie über das Menü „Aktion auswählen ➜ Spalten“.



Spalten auswählen

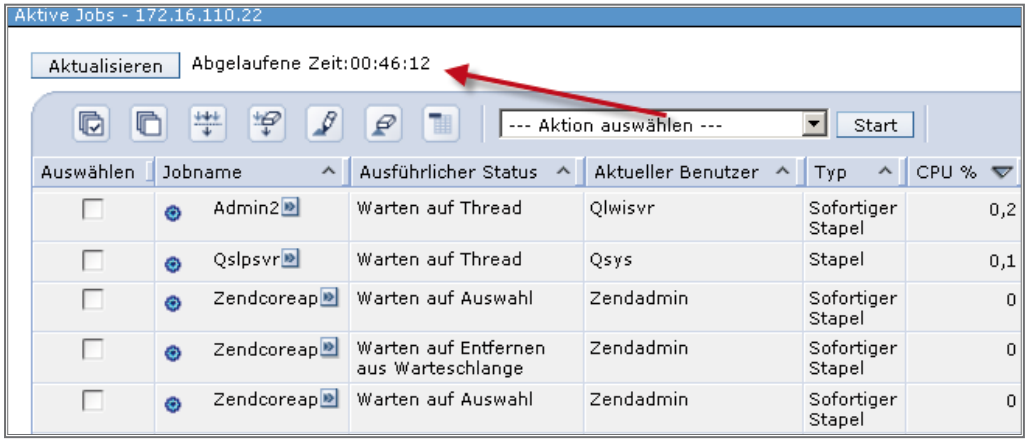
Die Anzeige wird durch „Aktion auswählen ➜ Anzeigeoptionen“ betreffs ihres Gehalts eingeschränkt.



Angezeigte Werte ändern

Ich verändere hier die Anzeigeoptionen nicht, um es bei der ursprünglichen Darstellung zu belassen.

Einen weiteren interessanten Punkt – Statistik zurücksetzen – finden Sie im Aktionsmenü „Aktion auswählen ➜ Statistik zurücksetzen“. Die Anzeige „Aktive Jobs“ zeigt Performance- und Durchsatzdaten bezogen auf die sogenannte „Abgelaufene Zeit“ an. Das ist im Befehl WRKACTJOB nicht anders. Die abgelaufene Zeit beginnt, wenn das erste Mal innerhalb einer Verbindung die Funktion „Aktive Jobs“ aufgerufen wird. Sehen Sie sich in der folgenden Abbildung einmal die Detailkopfzeile an. Dort finden Sie den Eintrag „Abgelaufene Zeit“. Alle angezeigten Werte sind relativ zu diesem Eintrag zu verstehen.

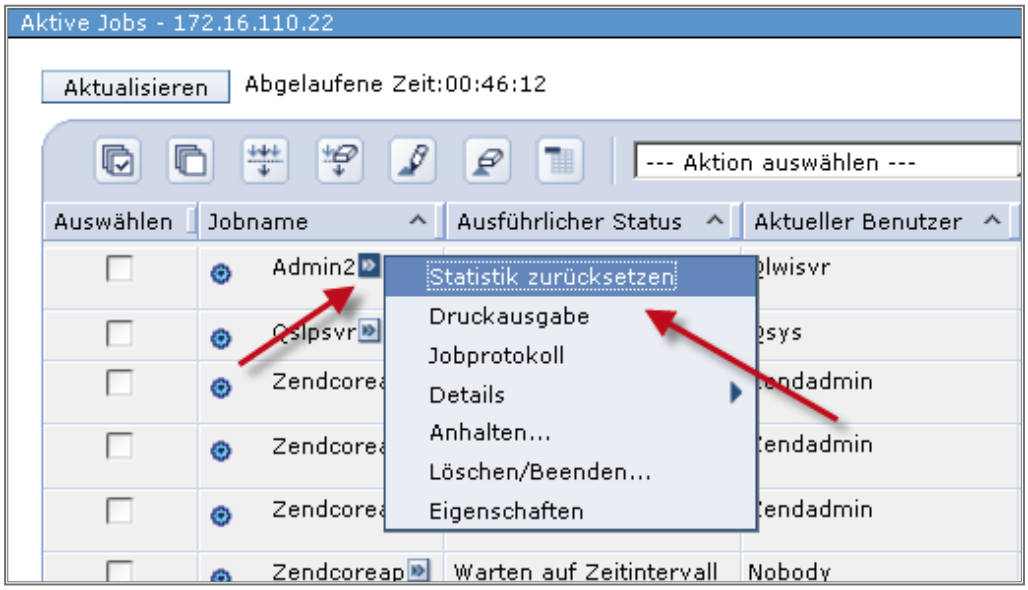


Abgelaufene Zeit

Ein Job, für den 10% CPU-Leistung angezeigt werden, beansprucht in diesem Moment nicht 10% der CPU, sonder vielmehr hat er in den letzten 59 Sekunden 10% der CPU-Leistung beansprucht. Da sich die Anzeige nicht selbständig aktualisiert, kann die aktuelle Situation vom angezeigten Wert erheblich abweichen. Wollen Sie die Anzeige auf den aktuellen Stand bringen, so klicken Sie doppelt auf den Punkt „Statistik zurücksetzen“. Im Dialog Aktive Jobs wird insgesamt – also für alle Jobs – die abgelaufene Zeit auf 00:00:00 zurückgesetzt.

Auch auf der Jobebene lässt sich ein Kontextmenü anzeigen. Markieren Sie einfach wahlfrei irgendeinen Job in der Detailanzeige von „Aktive Jobs“ und drücken Sie dann die rechte Maustaste. Danach sehen Sie folgendes Kontextmenü.

7.7.4.4.1
Seite 4



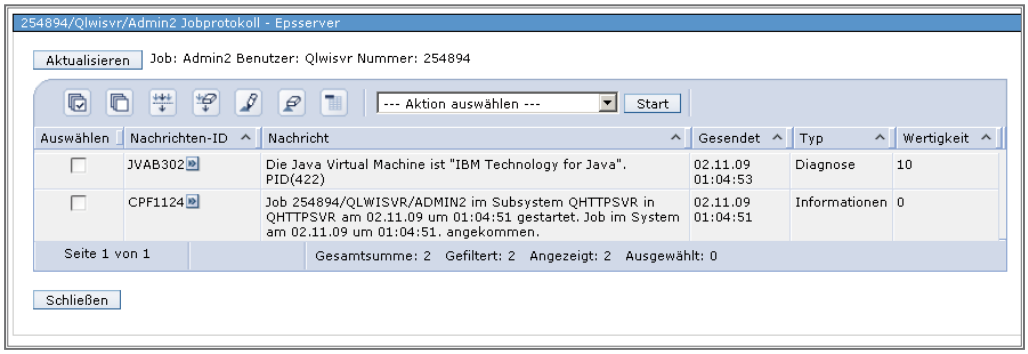
Kontextmenü auf Jobebene

Folgende Verarbeitungsoptionen werden – je nach Jobstatus – angeboten:

– Statistik zurücksetzen:
Diese Funktion ist mit der soeben dargestellten identisch. Die „Abgelaufene Zeit“ wird auf Null gesetzt – hier aber nicht für alle Jobs der Anzeige, sondern nur für die markierten.

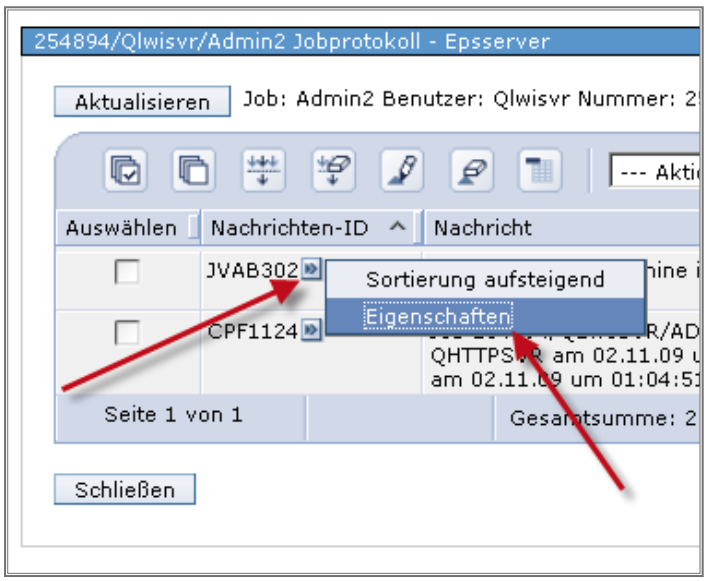
– Druckausgabe:
Dieser Punkt spricht für sich selbst. Erzeugt ein Job eine Druckausgabe, so kann sie mit Hilfe dieses Punktes eingesehen werden. Auch hier wird für die Anzeige der Ausgabedatei der AFP Workbench Viewer genutzt. Die Möglichkeiten kennen Sie bereits.

– Jobprotokoll:
Wählen Sie den Menüeintrag „Jobprotokoll“, so werden die jeweiligen Jobprotokolleinträge für den markierten Job angezeigt.



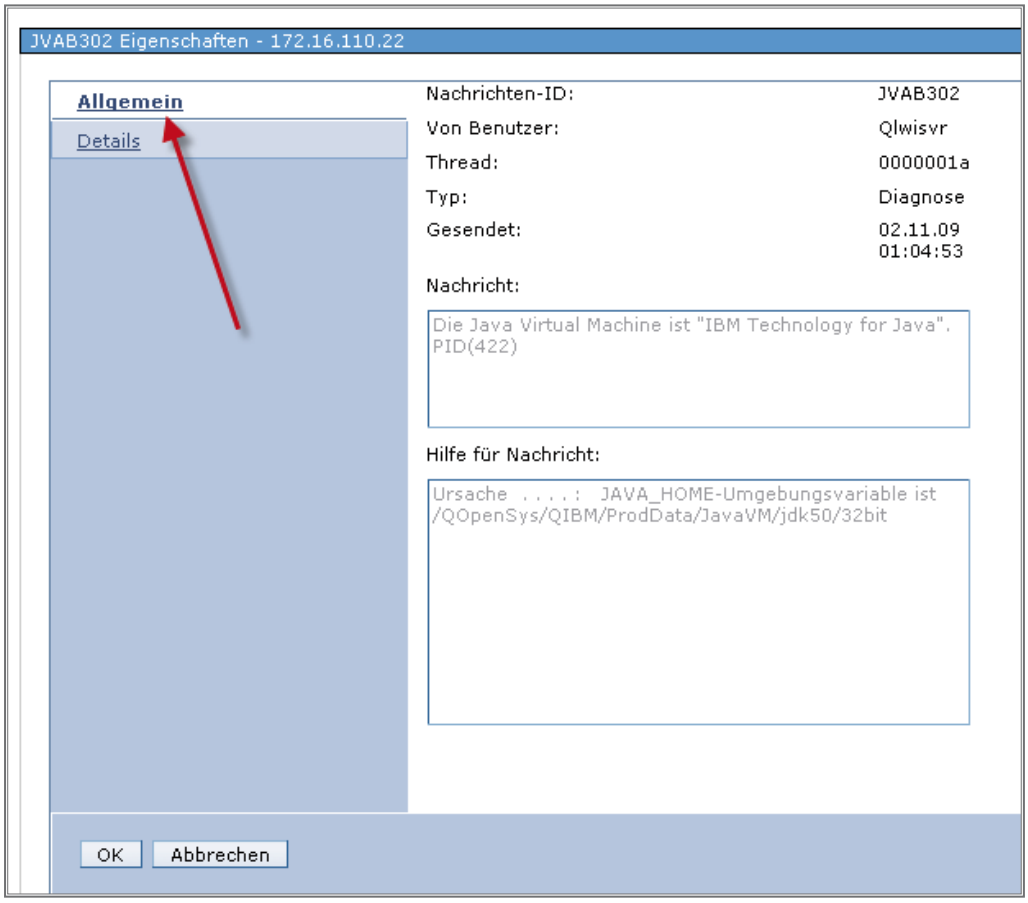
Jobprotokoll anzeigen

Ein Klick auf die jeweilige Nachricht zeigt das Kontextmenü an.



Nachrichten – Kontextmenü

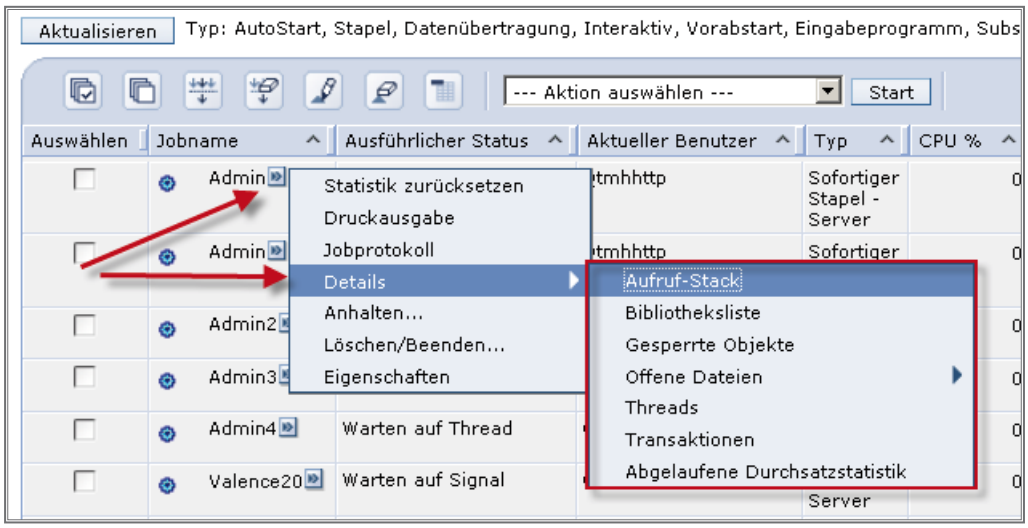
Mit Hilfe der Auswahl „Eigenschaften“ werden die Systemattribute der angewählten Nachricht angezeigt.



Nachrichten – Eigenschaften

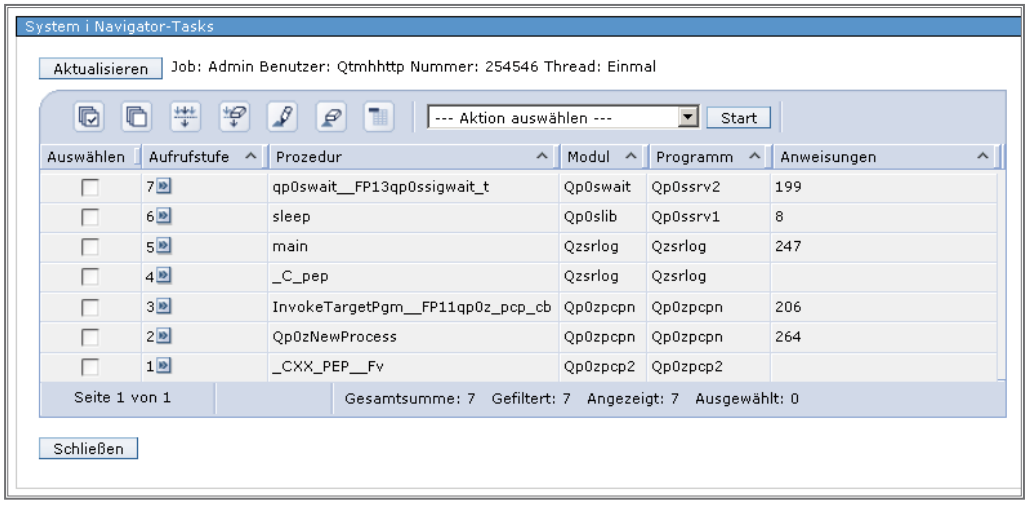
7.7.4.4.1
Seite 6

Im Kontextmenü „Job“ finden Sie die Auswahl „Details“. Ein Doppelklick auf diesen Eintrag öffnet ein Submenü mit den folgenden Punkten:



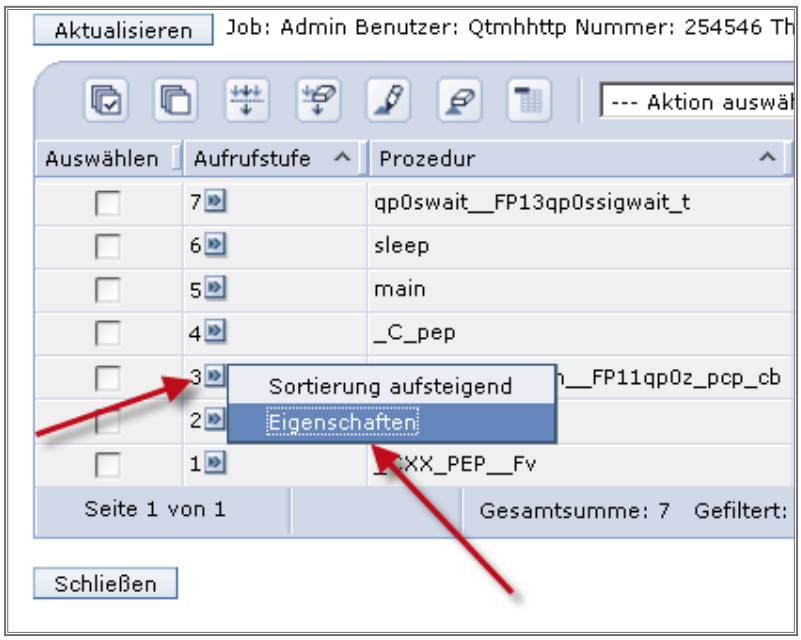
Submenü „Details“

Klicken Sie auf die Auswahl „Aufruf-Stack“, dann sehen Sie den Programmaufrufstapel für den jeweiligen Job. Wenn ein Job in eine Fehlersituation gerät, dann können Sie den Sachverhalt aus dieser Anzeige sehr gut entnehmen.



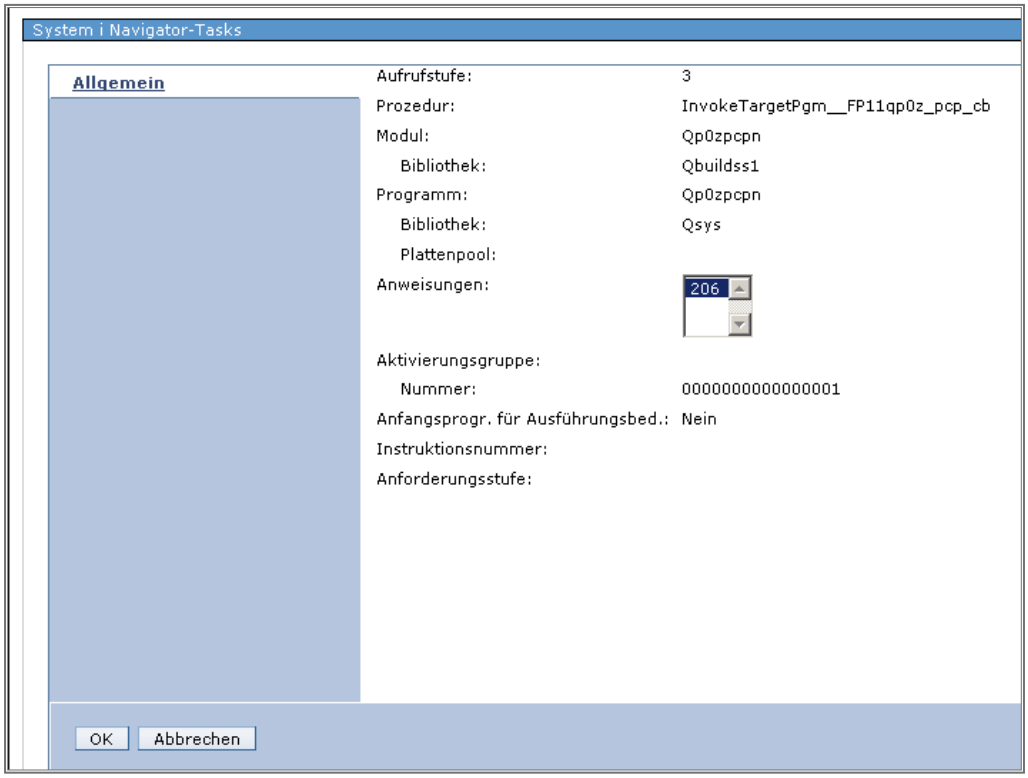
Programmstapel

Interessiert Sie im Programmstapel der Zustand eines bestimmten Programms, so fordern Sie das Kontextmenü für das Programm an.



Programmeigenschaften anfordern

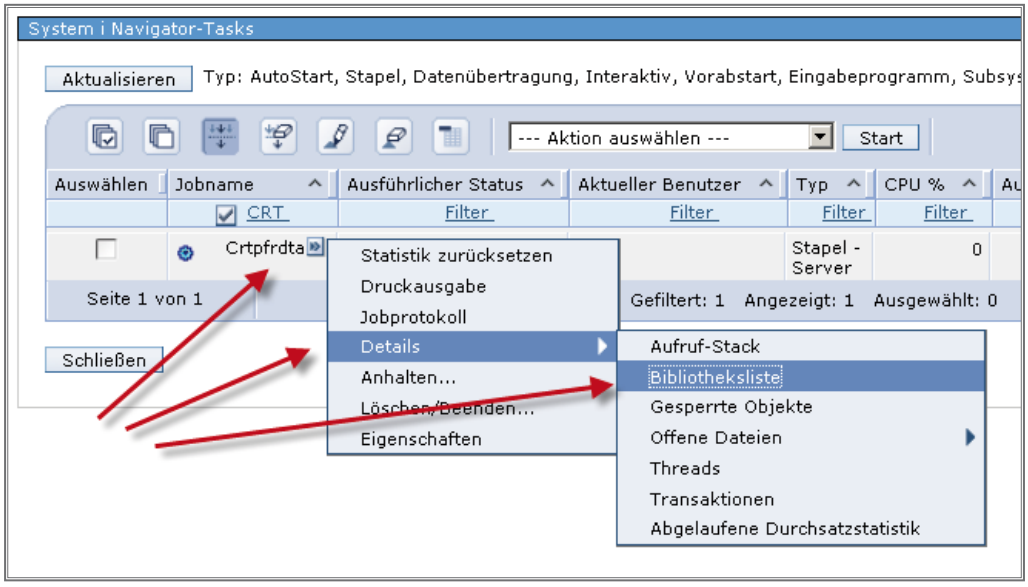
Dann wählen Sie die Eigenschaften des Programms im Aufrufstapel aus.



Programmeigenschaften

7.7.4.4.1
Seite 8

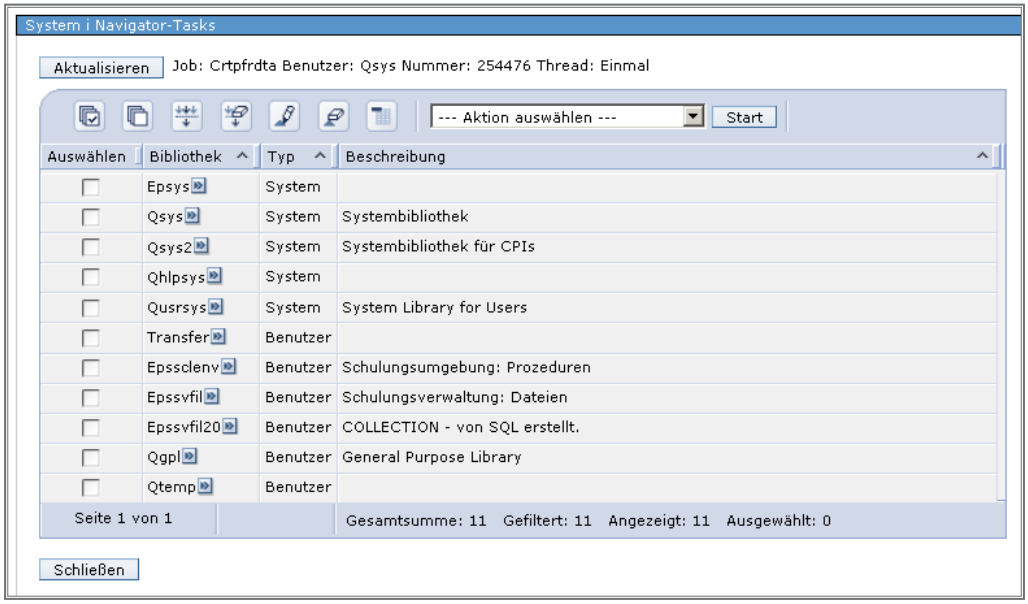
Weitere Anzeigen sind aus dem Submenü „Details“ heraus abrufbar. Ich werde jetzt auf den Eintrag „Bibliotheksliste“ klicken.



Bibliotheksliste

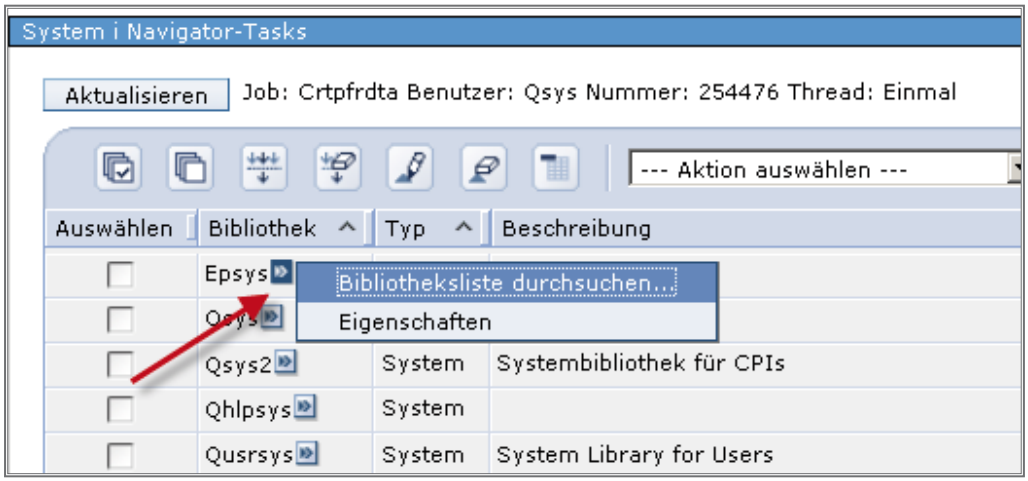
Folgende Anzeige öffnet sich.

Sie sehen in der Anzeige die Bibliotheksliste für den Job CRTPFRDTA.



Bibliotheksliste für Job

Auch hier existiert wieder ein Kontextmenü.

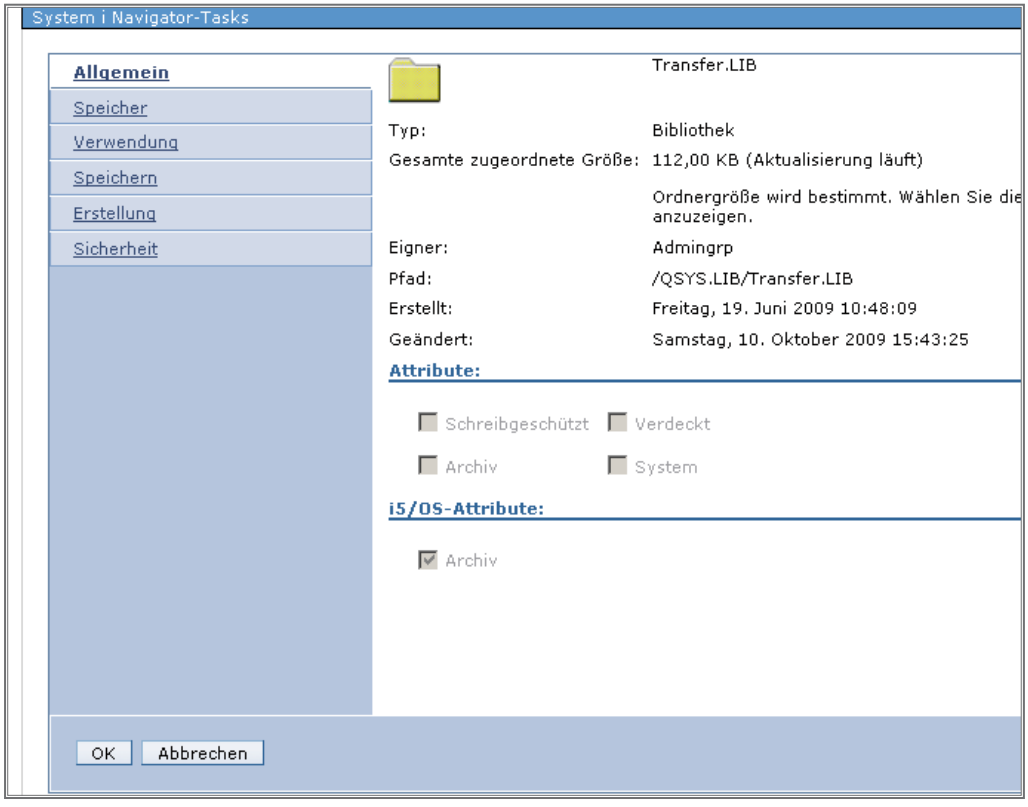


Kontextmenü für Bibliotheken in der Bibliotheksliste

Dieses Menü umfasst folgende Auswahlmöglichkeiten:

– Bibliotheksliste durchsuchen…:
Mit Hilfe dieser Auswahl durchsuchen Sie die Bibliothek gezielt nach ganz bestimmten Objekten.

– Eigenschaften:
Falls Sie die Eigenschaften einer Bibliothek interessieren, sollten Sie in diesen Punkt gehen.

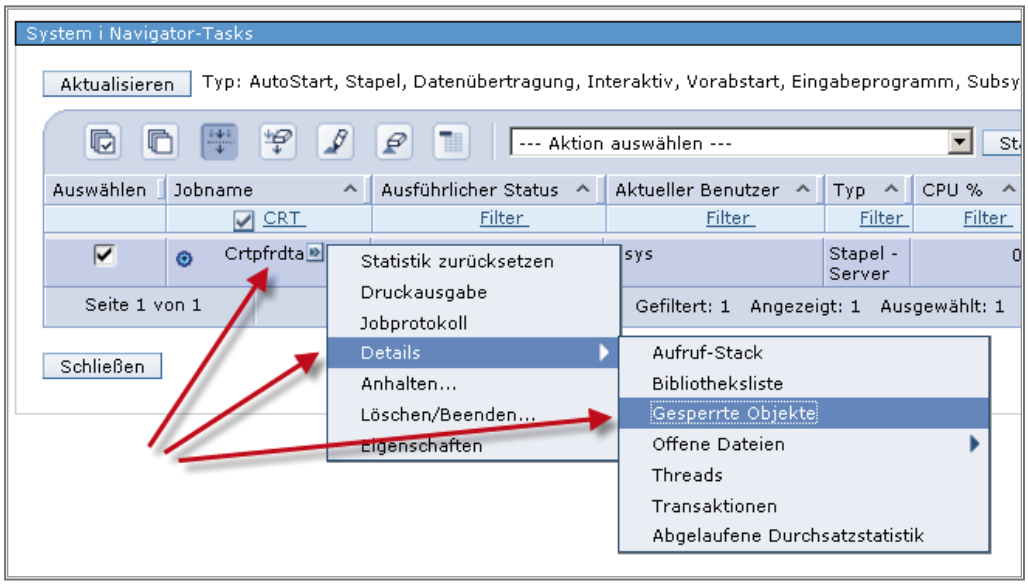


Bibliothekseigenschaften

7.7.4.4.1
Seite 10

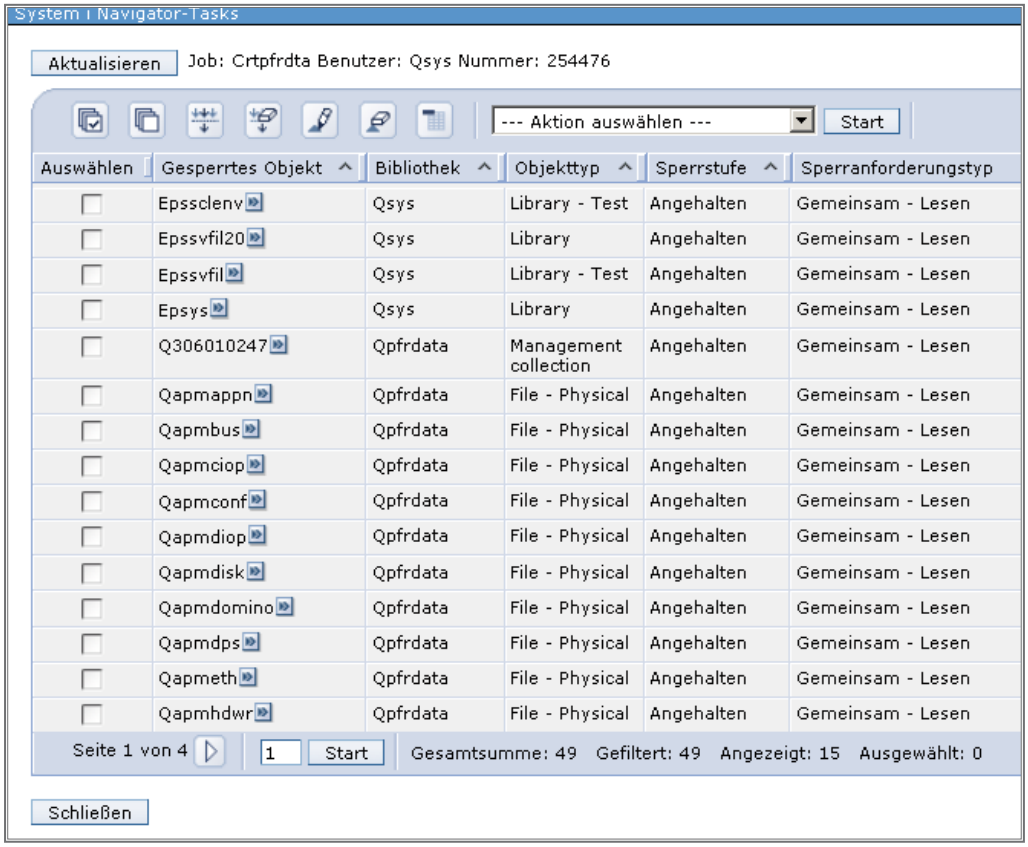
Hier werden folgende Merkmale dargestellt: Größe einer Bibliothek, Erstellungsdatum und Zeit, Datum der letzten Änderung, Anzahl der gespeicherten Objekte, Datum der letzten Sicherung und Wiederherstellung, Eigner und vieles mehr.

Kehren wir zum Submenü „Details“ zurück.



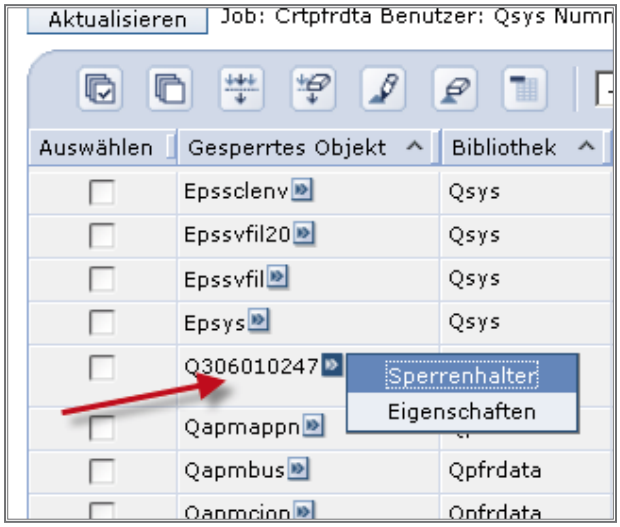
Job-Details

Eine weitere interessante Menüauswahl finden Sie in „Gesperrte Objekte“. In der Folge sehen Sie die Anzeige dieses Menüpunkts:



Objekte, die eine Sperre halten

Sie sehen oben die vom Job CRTPFRDTA gesperrten Objekte. Auch an dieser Stelle existiert ein Kontextmenü.

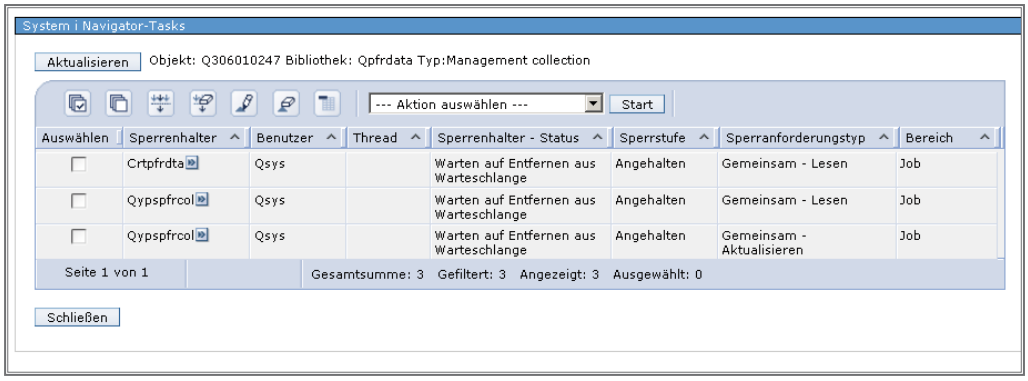


Kontextmenü „Gesperrte Objekte“

7.7.4.4.1
Seite 12

Dieses Menü umfasst zwei Auswahlmöglichkeiten:

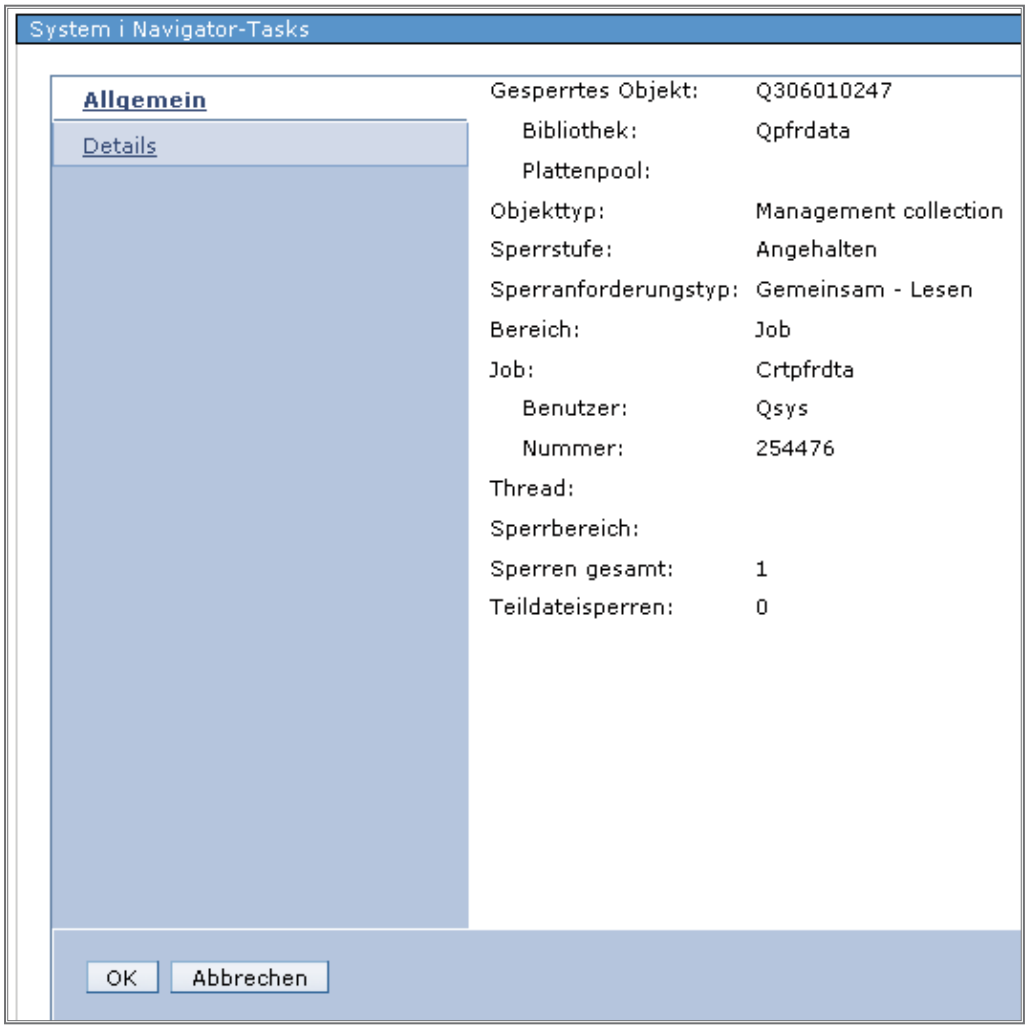
– Sperrenhalter:
Hier sehen Sie, welche Jobs – inklusive dem gerade zu bearbeitenden – eine Sperre auf dieses Objekt halten.



Weitere Jobs, die Sperren halten

In dieser Anzeige erkennen Sie, dass der Job CRTPFRDTA sowie einige andere eine Sperre auf das gekennzeichnete Objekt halten.

– Eigenschaften:
Den Eigenschaften des gesperrten Objekts entnehmen Sie dann Detaildaten.

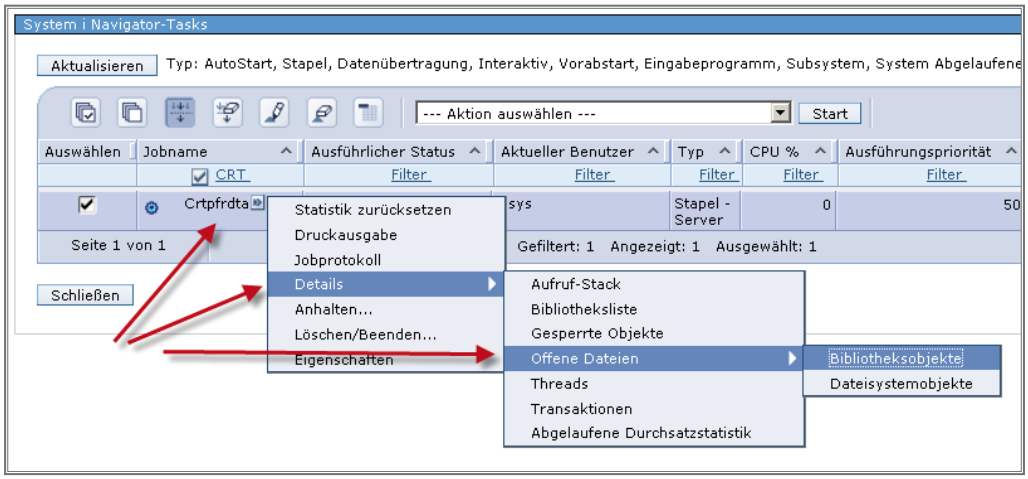


Sperren von Eigenschaften

Folgende Werte werden angezeigt: Sperrstufe, Sperrabforderungstyp, Bereich der Sperre, die Jobkenndaten des sperrenden Jobs, Objektsperren und Teildateisperren.

Wir kehren einmal mehr zum Submenü „Details“ zurück und gehen in die Auswahl „Offene Dateien“ hinein.

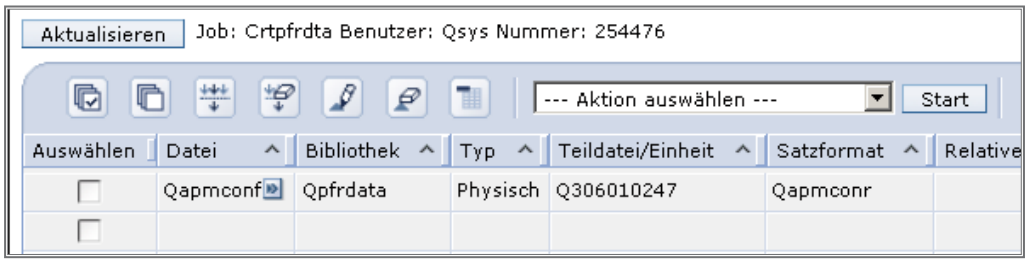
7.7.4.4.1
Seite 14



Menü „Offene Dateien“

In diesem Punkt wird unterschieden zwischen offenen Dateien der Bibliotheksumgebung und des Integrated File System (Dateisystemobjekte).

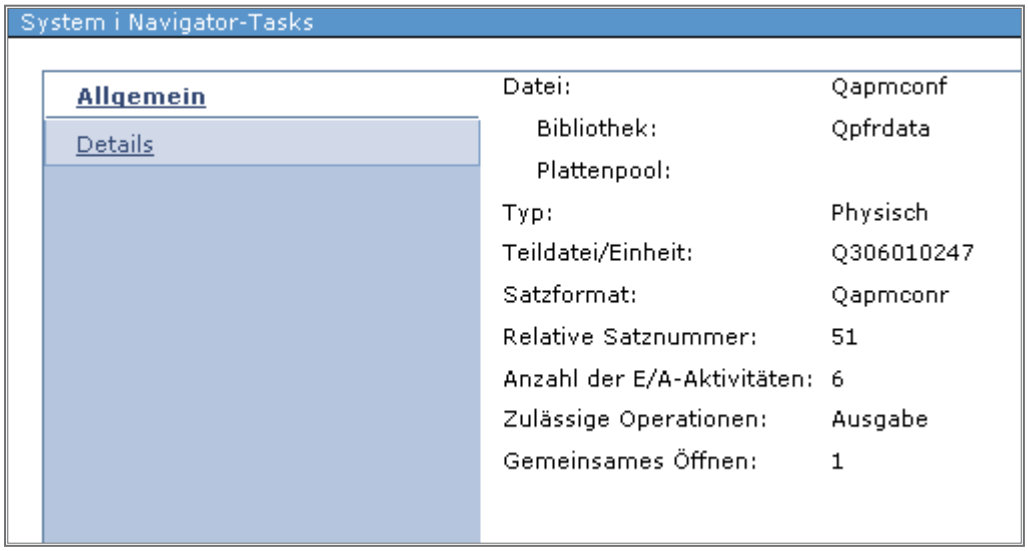
In der folgenden Anzeige sehen Sie, welche Systemdateien der Job CRTPFRDTA geöffnet hat.



Offenen Dateien

In dem oben dargestellten Dialog fordere ich das Kontextmenü für die offene Datei QAPMCONF an. Es wird der Eigenschaften-Dialog für das angesprochene Objekt angezeigt.

Hier finden Sie einerseits Beschreibungsdaten für die Datei, so z. B. ihren Namen, die Bibliothek, in der sie gespeichert wurde, den Dateityp, den Namen der Teildatei und den Namen des Satzformats.

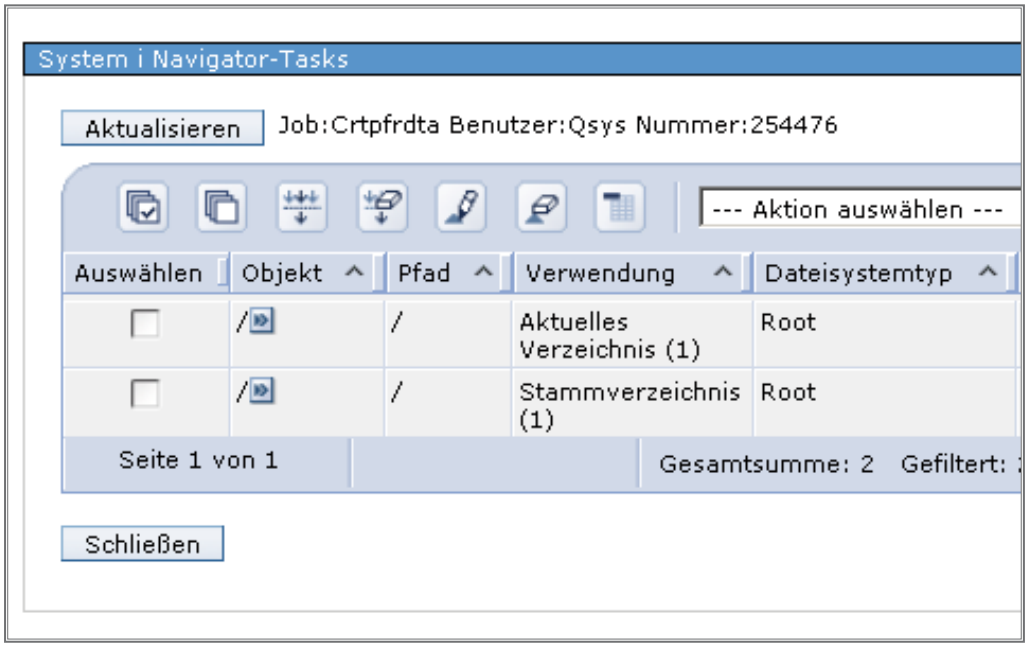


Eigenschaften des Objekts

Andererseits sehen Sie hier die relative Satznummer des gerade verarbeiteten Satzes, die Anzahl der Eingabe- und Ausgabeaktivitäten sowie welche Ausgabeoperationen für diese Datei zulässig sind.

Für Programmierer ist das ein fast unentbehrlicher Dialog, um Konflikte bei Dateizugriffen zu klären.

Folgen wir im Punkt „Offene Dateien“ dem Pfad „Dateisystemobjekte“, so sehen wir, auf welche Objekte des Integrated File System (IFS) der Job CRTPFRDTA zugreift.

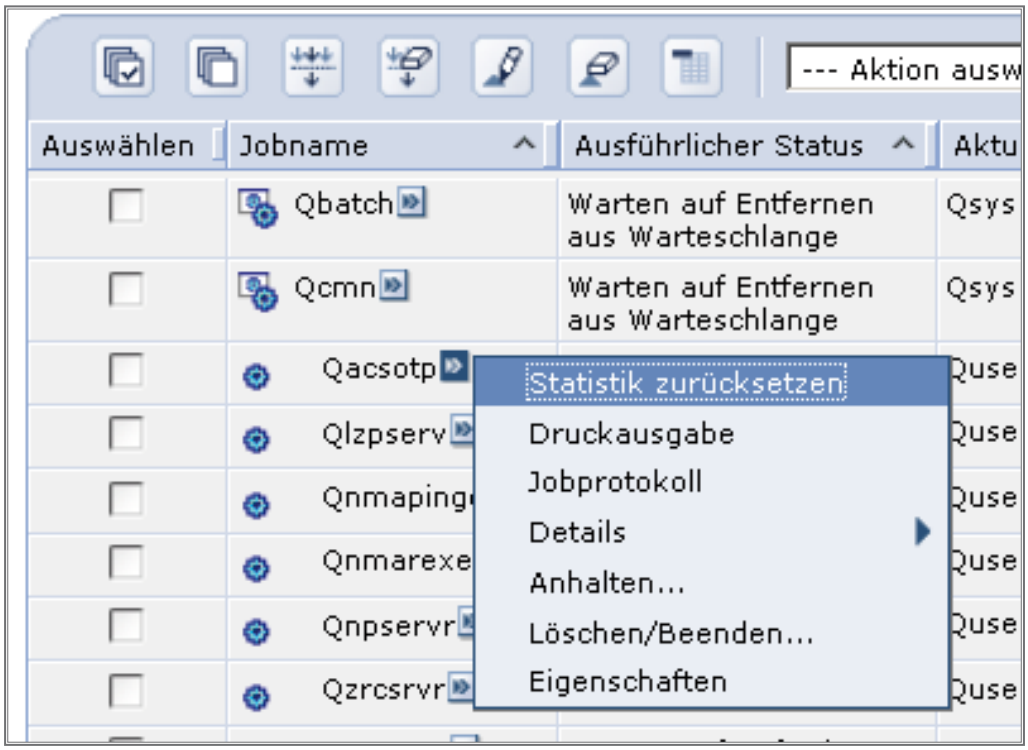


IFS-Objekte im Zugriff

7.7.4.4.1
Seite 16

Nach unserer Reise durch die Punkte und Auswahlmöglichkeiten des Submenüs „Details“ kehren wir nun auf die Jobebene zurück, um mit dem Kontextmenü für Jobs weiterzumachen.

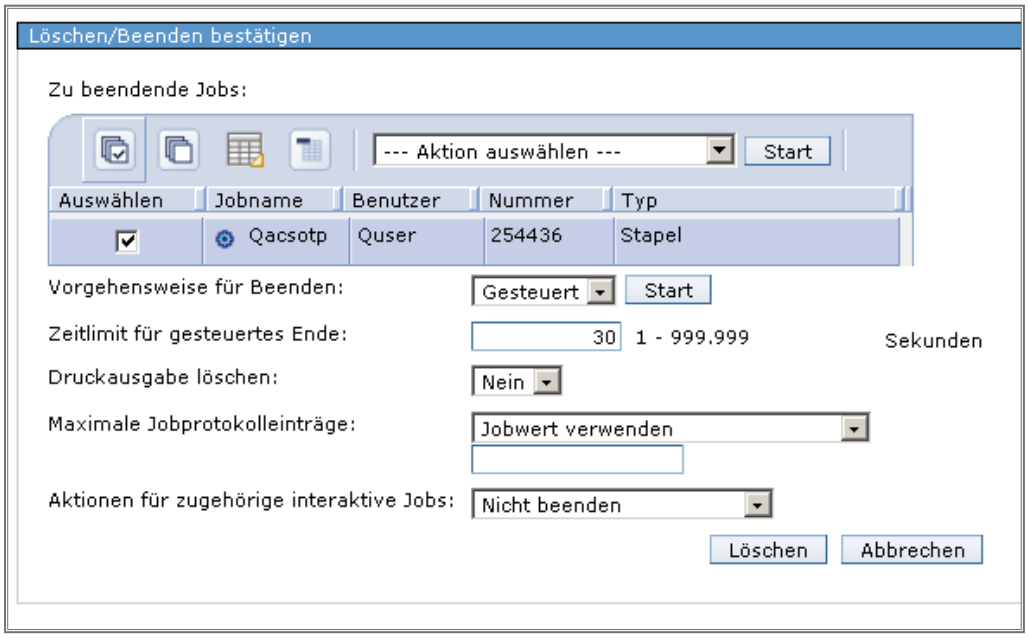
Zur Erinnerung zeige ich Ihnen in der folgenden Abbildung noch einmal das Kontextmenü.



Kontextmenü auf Jobebene

– Anhalten:
  Einen laufenden Job anhalten.

– Löschen/Beenden:
  Wählen Sie diesen Punkt an, so können Sie einen laufenden Job beenden.



Job beenden

Mit dem Parameter „Vorgehensweise für Beenden“ wird die Beendigungsart eingestellt:

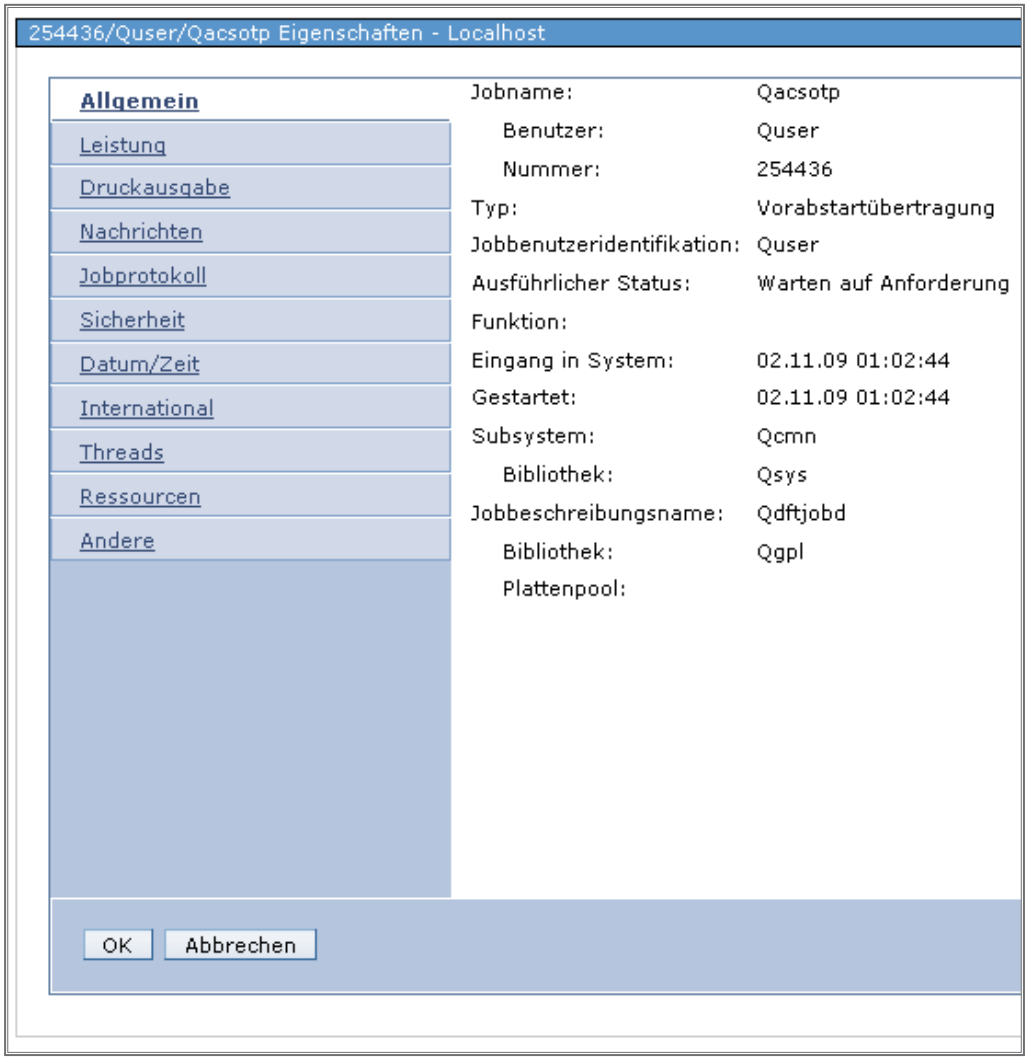
Sofort: Der Job wird augenblicklich beendet. Er gilt als abnormal beendet und erzeugt ein Jobprotokoll. Achtung! Daten können verloren gehen!

Gesteuert: Der Job schließt alle Dateien und beendet kontrolliert nach Wegschreiben aller Pufferdaten seine Arbeit. Daten gehen nicht verloren.

– Eigenschaften:
  Diese Auswahl zeigt Ihnen die Kenndaten eines Jobs an. Der Eigenschaften-Dialog ist vergleichbar mir der Anzeige des Befehls DSPJOB (Jobeigenschaften anzeigen).

7.7.4.4.1
Seite 18

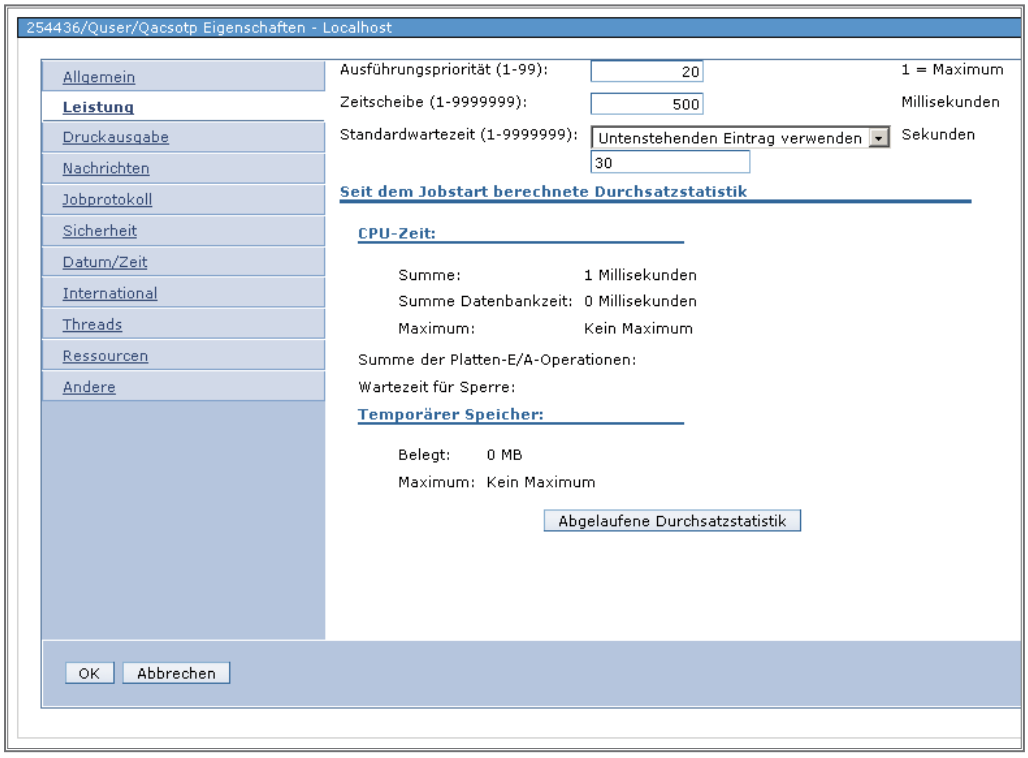
Ich stelle im Folgenden nur einige Register dieser Anzeige dar. Im Register „Allgemein“ sehen Sie die allgemeinen Kenndaten eines Jobs.



Allgemeine Kenndaten

Betrachten Sie oben den Dialog im Register „Allgemein“. Sie sehen hier Jobidentifikationsdaten wie Jobname, Eigner und Jobnummer, Jobtyp, aktueller Benutzer des Jobs, derzeitiger Status, das vom Job aufgerufene Programm, den Zeitpunkt des Eintritts in das System und den Zeitpunkt des Jobstarts. Darüber hinaus können Sie das diesen Job steuernde Subsystem und die vom Job verwendete Jobbeschreibung erkennen.

Im Register „Leistung“ werden performancerelevante Daten und Durchsatzstatistikdaten angezeigt. Es können in diesem Dialog aber auch Parameterwerte geändert bzw. angepasst werden.



Leistungsdaten

Parameterwerte, die geändert werden können:

– Ausführungspriorität

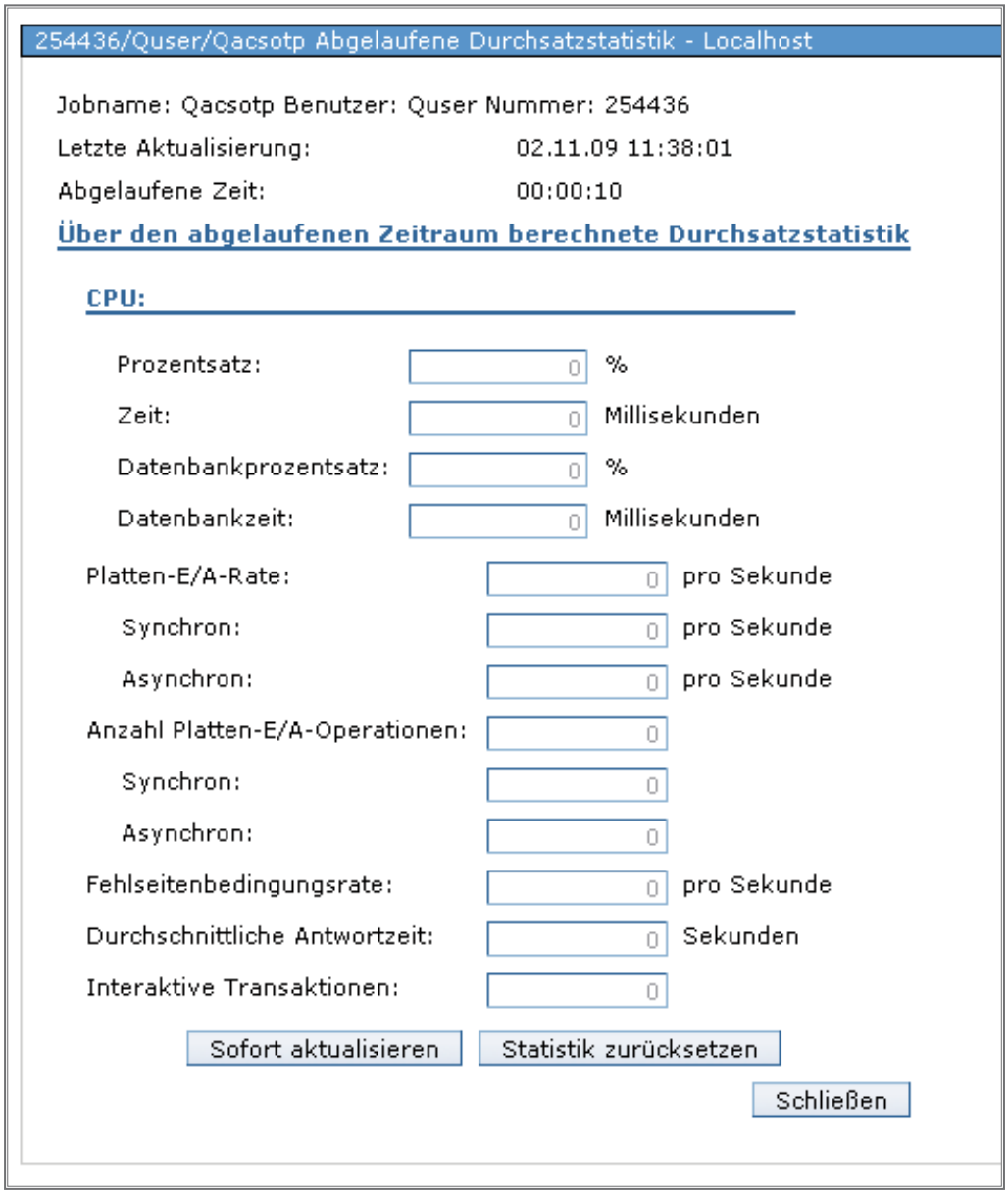
– Zeitscheibe

– Standardwartezeit

Diese Werte überschreiben die dem Job durch seine Jobklasse zugeordneten Werte.

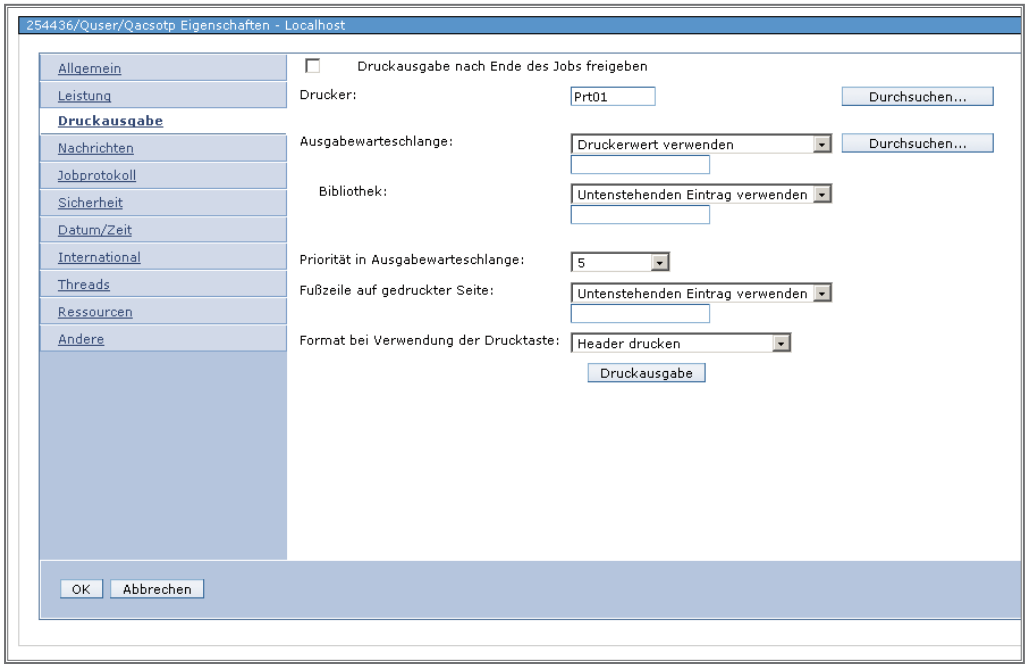
7.7.4.4.1
Seite 20

Im unteren Teil des Dialogs sehen Sie dann die aktuelle Durchsatzstatistik dieses Jobs. Historische Durchsatzstatistikdaten erhalten Sie, indem Sie auf den Knopf „Abgelaufene Durchsatzstatistik“ drücken.



Durchsatzstatistik

Im Register „Druckausgabe“ finden Sie relevante Parameter zur Drucker- und Ausgabewarteschlangenzuordnung.



Druckausgabe

Folgende wichtige Parameter finden Sie hier:

Drucker:
Ordnen Sie dem Job temporär einen neuen Drucker zu. Mit Hilfe des Dialogs „Durchsuchen…“ sind Sie in der Lage, aus der Systemumgebung der iSeries interaktiv einen Drucker auszusuchen.

Ausgabewarteschlange:
Ordnen Sie dem Job temporär eine neue Ausgabewarteschlange zu. Auch hier können Sie interaktiv das System nach einer geeigneten Warteschlange durchsuchen.

Priorität in Ausgabewarteschlange:
Durch Reduzierung dieses Werts erhält Ihr Job eine höhere Priorität in der Ausgabewarteschlange und wird somit früher gedruckt.

7.7.4.4.1

Seite 22

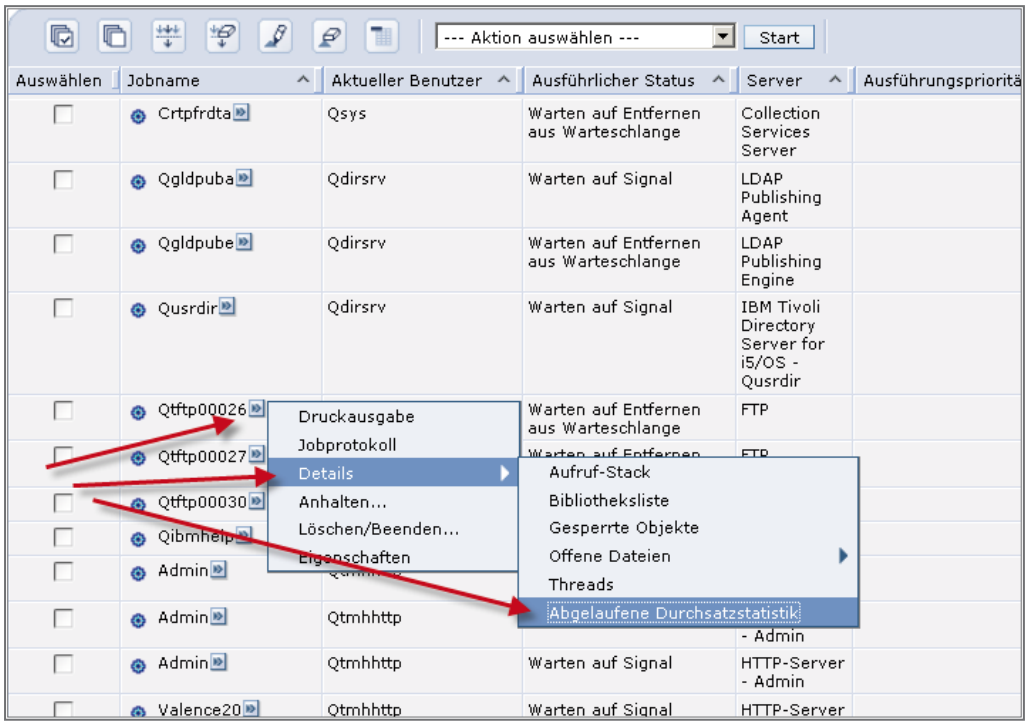
Mit Server-Jobs arbeiten

Die Funktion „Mit Server-Jobs arbeiten“ hat auf der Befehlsebene des i/OS keine Entsprechung. Es handelt sich bei dieser Funktion auch nicht um eine eigenständige und neue Funktion, vielmehr ist sie eine Untermenge der Funktion „Mit aktiven Jobs arbeiten“. Da die Power i-Systeme mittlerweile als Infrastruktur-Server zu betrachten sind, sind Funktionen – u. a. des Netzwerks und der Datenbank – als Server-Jobs im Betriebssystem implementiert.

Aufgrund der mit den letzten Versionen des Betriebssystems rasant angestiegenen Zahl von Serverdiensten sowie den ungenügenden Darstellungsmöglichkeiten von Server-Jobs im traditionellen Bildschirm-Interface wurde die hier zu besprechende Funktion notwendig.

Die Dialoge der Funktion „Server-Jobs“ sind identisch mit denen der Funktion „Aktive Jobs“. Deshalb werde ich nur auf wenige zusätzliche Punkte im Funktionsbereich „Server-Jobs“ eingehen.

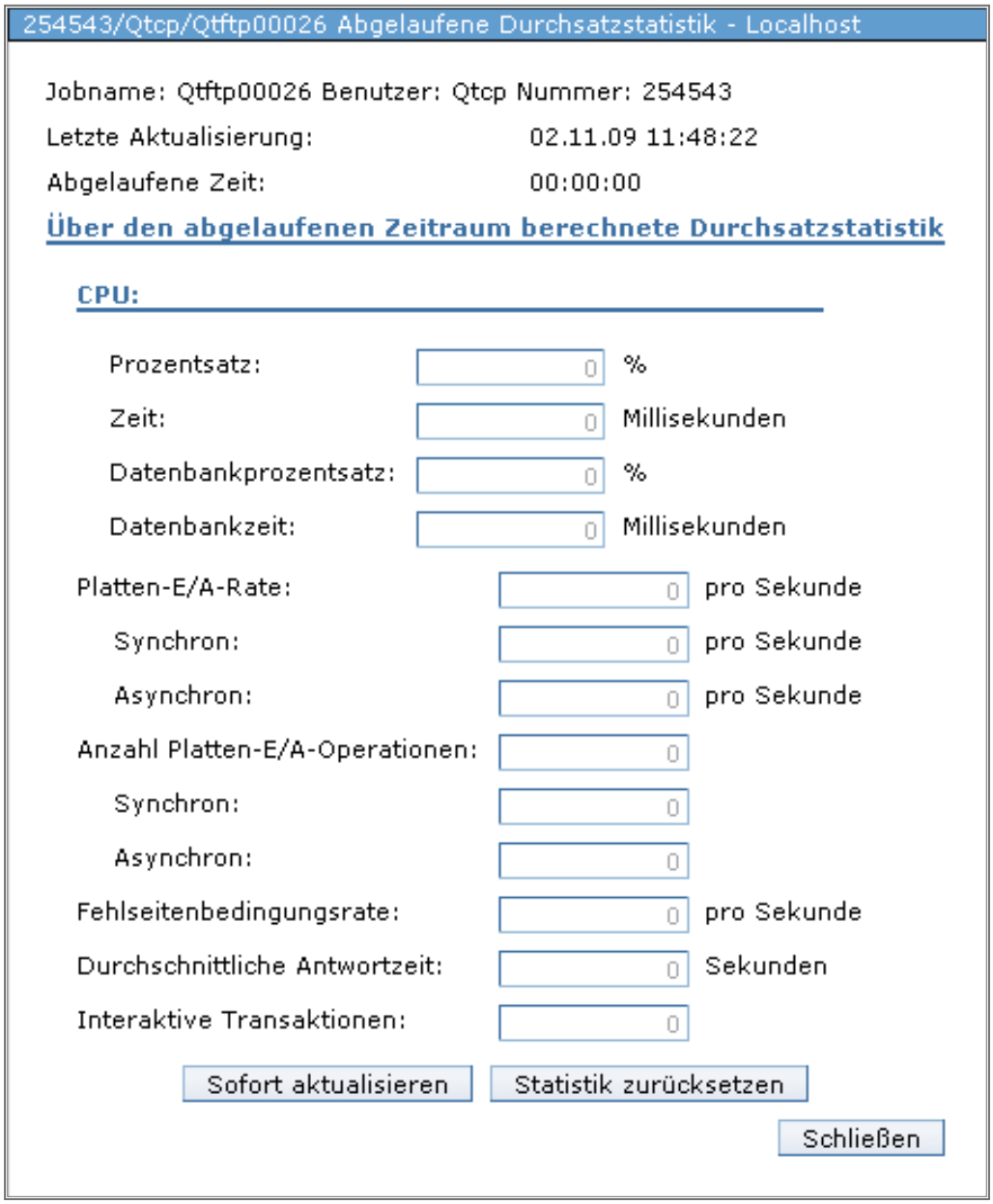
Im Submenü „Details“ möchte ich nur kurz auf die Auswahl „Abgelaufene Durchsatzstatistik“ eingehen.



Mit Server-Jobs arbeiten

Diese Funktion zeigt historische Durchsatzdaten für den in der Jobliste markierten Job an, die sich in der Anzeige regelmäßig oder auf Anforderung aktualisieren lassen.

7.7.4.4.2
Seite 2



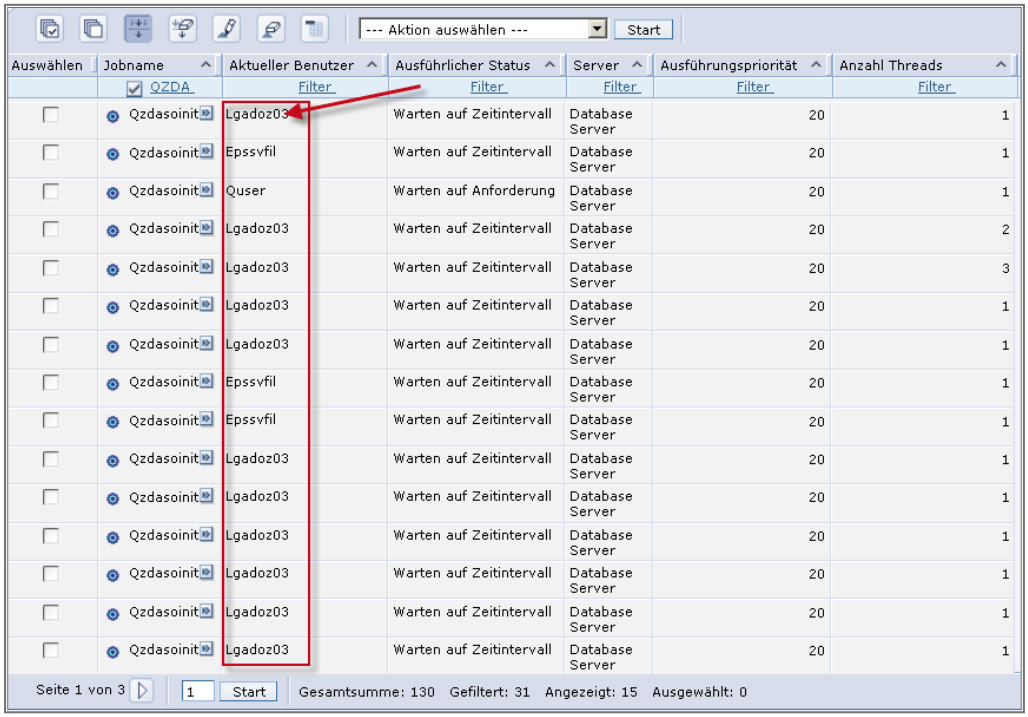
Abgelaufene Durchsatzstatistik

Folgende Werte lassen sich darstellen:
- CPU-Durchsatzdaten
- Plattenzugriffsraten pro Sekunde
- Durchgeführte Platten-I/O
- Paging-Raten
- Durchschnittliche Antwortzeiten
- Interaktive Transaktionen

Die meisten Clientprodukte der IBM und auch anderer Hersteller verwenden sogenannte Daemon-Jobs (Server-Jobs), um die Kommunikation zwischen Client und Server herzustellen. Aus dem 5250-Umfeld heraus ist die Administration dieser Daemons jedoch relativ schwierig, da in CL-Befehlen zur Jobüberwachung nicht alle Kennwerte für diese Jobs direkt ausgegeben werden.

Hier ein kleines Beispiel: Daemon-Jobs werden beim Start eines Subsystems als vorgestartete Jobs automatisch gestartet. Beim Start wird diesen Jobs ein Standardbenutzer wie z. B. QUSER als Eigentümer zugeordnet. Kommt jetzt ein Client, so muss sich der Anwender des Clients am Server anmelden. D. h., dem Daemon wird ein zweiter Benutzer zugeordnet – der Anwender, der den Daemon gerade benutzt – der sogenannte aktuelle Benutzer.

Gibt es Probleme mit einem Daemon, so müssen wir in Erfahrung bringen, wer der aktuelle Benutzer eines Daemon-Jobs ist.



Aktuelle Benutzer

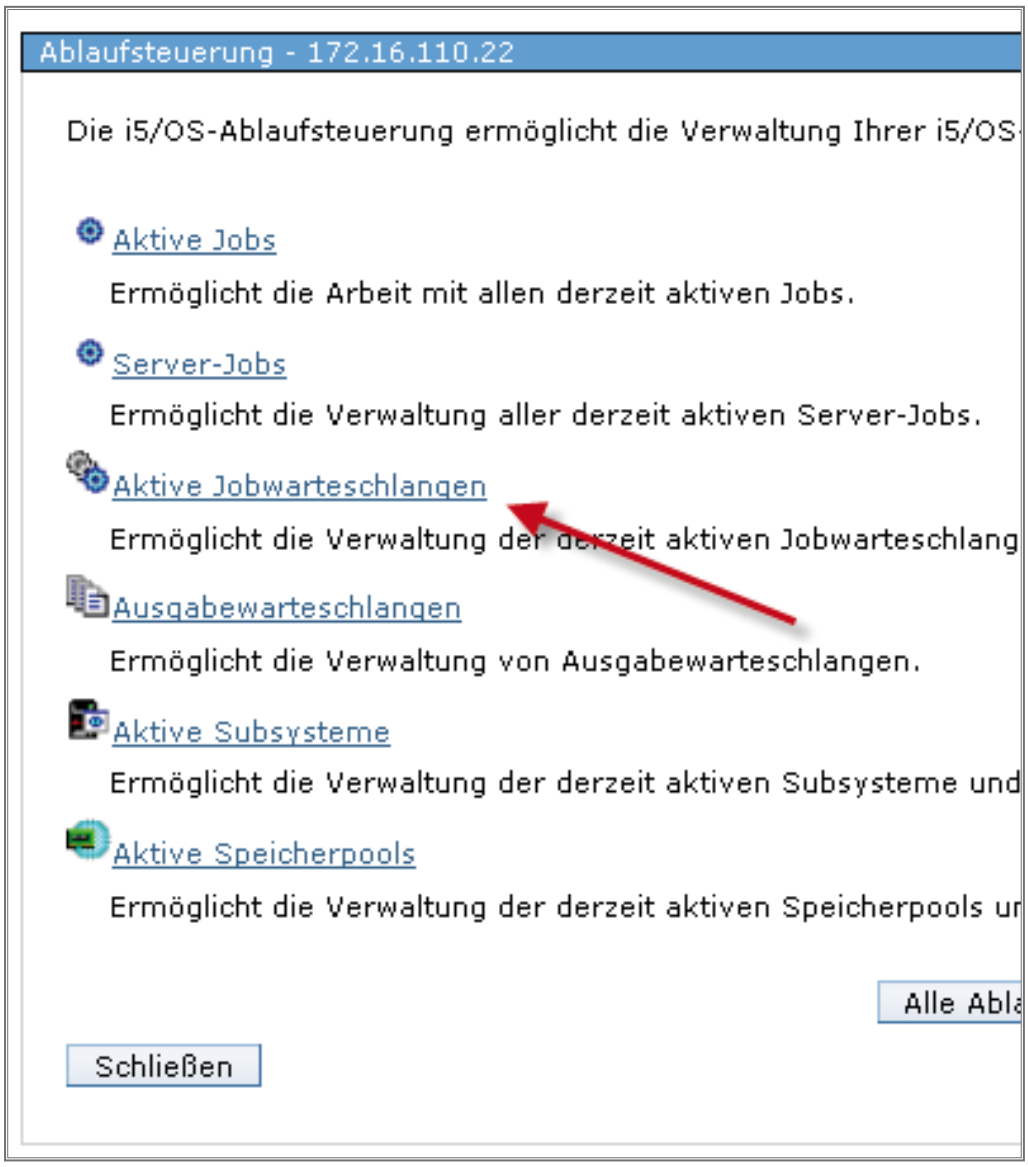
Gehen Sie in der Ablaufsteuerung in die Funktion „Mit Server-Jobs arbeiten“, so wird der aktuelle Benutzer angezeigt. Ein unschätzbarer Vorteil bei der Problemanalyse von Client/Server-Problemen.

7.7.4.4.2

Seite 4

Mit Jobwarteschlangen arbeiten

Zum Bereich der Ablaufsteuerung im iSeries Navigator gehört auch die Arbeit mit Jobwarteschlangen. In dieser Umgebung wird folgende Option angeboten.

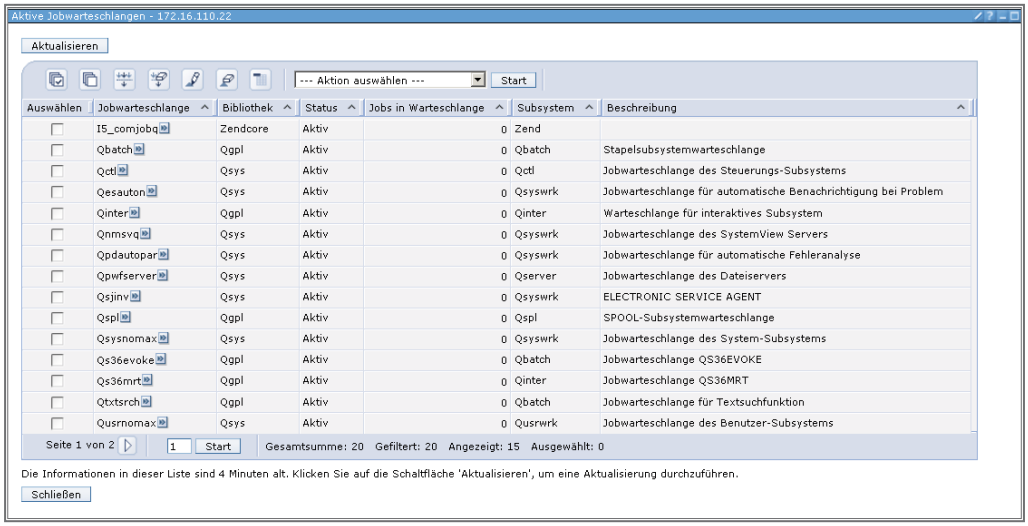


Mit aktiven Jobwarteschlangen arbeiten

Unter aktiven Jobwarteschlangen sind solche zu verstehen, die eine Verbindung zu einem gestarteten Subsystem unterhalten.

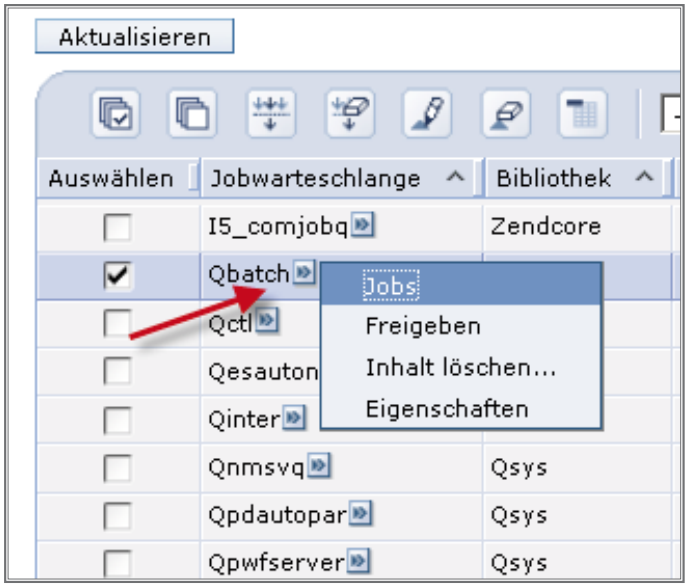
7.7.4.4.3
Seite 2

Im folgenden Dialog habe ich die Funktion „Aktive Jobwarteschlangen“ aufgerufen.



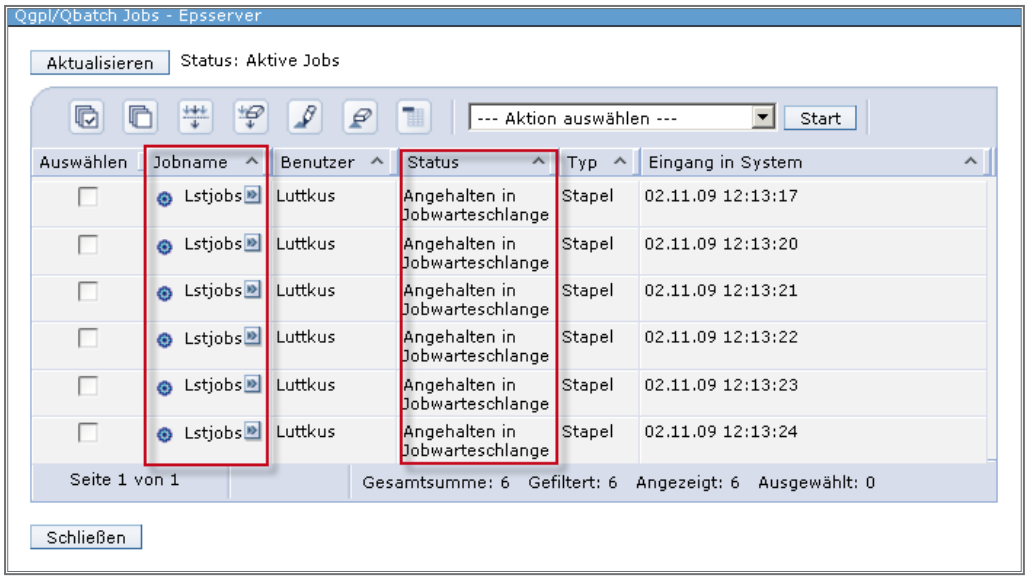
Aktive Jobwarteschlangen

Ich fordere für die Jobwarteschlange QBATCH das Kontextmenü an.



Jobwarteschlange – Kontext

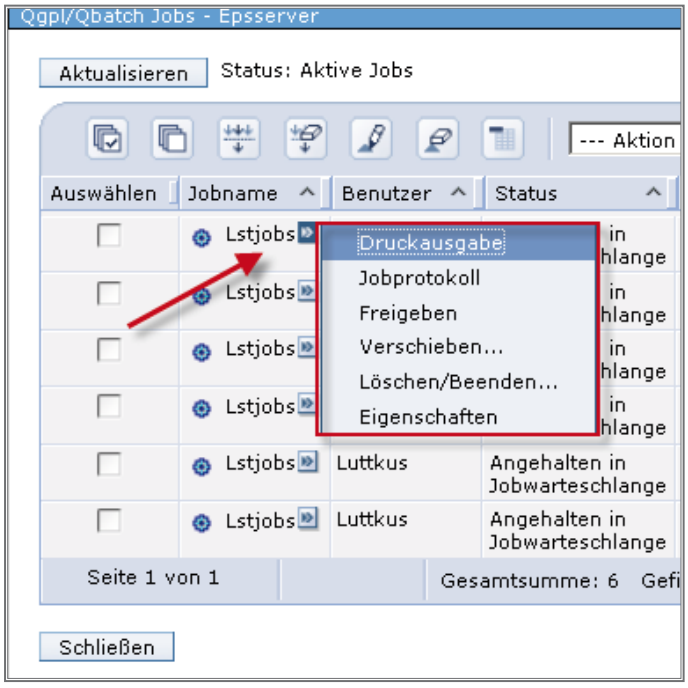
Mit Hilfe des Eintrags „Jobs“ lasse ich die in dieser Warteschlange auf Verarbeitung wartenden Jobs anzeigen.



Wartende Jobs

In dem gezeigten Dialog sehen Sie, dass Jobs mit dem Namen LSTJOB auf Verarbeitung warten. Diese Jobs haben derzeit den Status „Angehalten“. Für einen dieser Jobs werden wir jetzt die Verarbeitungsmöglichkeiten durchgehen, die zur Verfügung stehen.

Dazu lasse ich mir das Kontextmenü anzeigen. Folgende Auswahlmöglichkeiten bestehen:

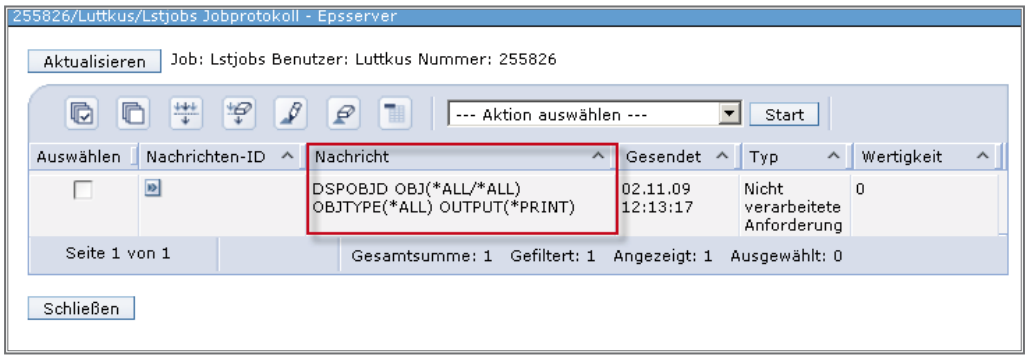


Kontextmenü Jobs

7.7.4.4.3
Seite 4

Druckausgabe:
Da der Job derzeit in der Jobwarteschlange auf Freigabe wartet, konnte er noch keine Ausgabe erstellen. Folglich ist hier derzeit keine sinnvolle Anzeige zu erwarten.

Jobprotokoll:
In der Auswahl „Jobprotokoll“ sehen wir den Befehl, der zur Verarbeitung innerhalb des Jobs ansteht.

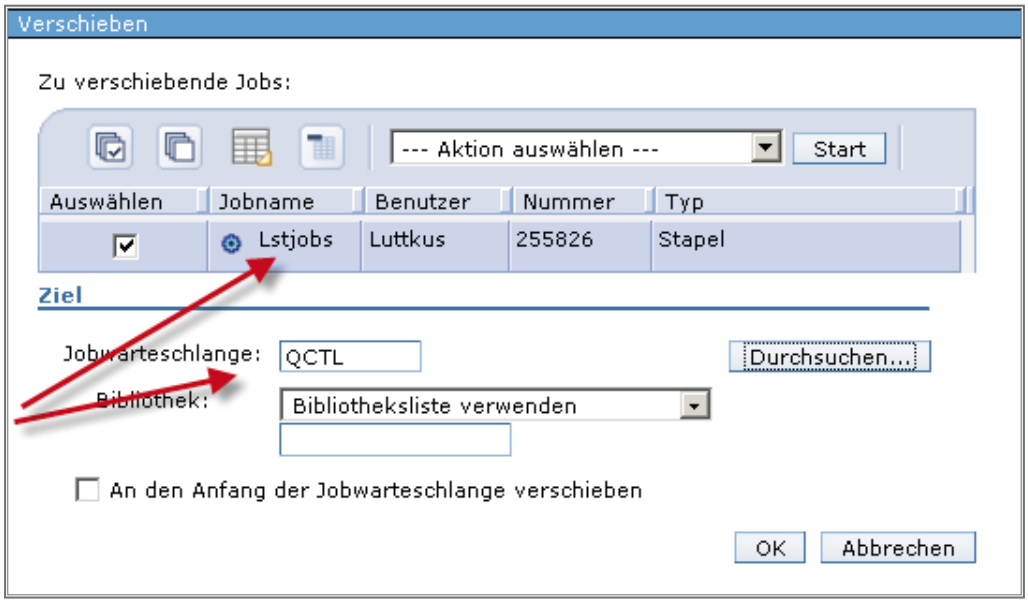


Zur Verarbeitung anstehender Befehl

Anhalten:
Da der Job angehalten wurde, steht diese Auswahl derzeit nicht zur Verfügung.

Freigeben:
Geben Sie den Job mit dieser Auswahl frei. Ist der Job freigegeben, wird die Auswahl „Anhalten“ aktiviert und „Freigeben“ deaktiviert.

Verschieben:
Mit dieser Auswahl verschieben Sie den Job in eine andere Warteschlange bzw. an den Anfang der aktuellen Jobwarteschlange.



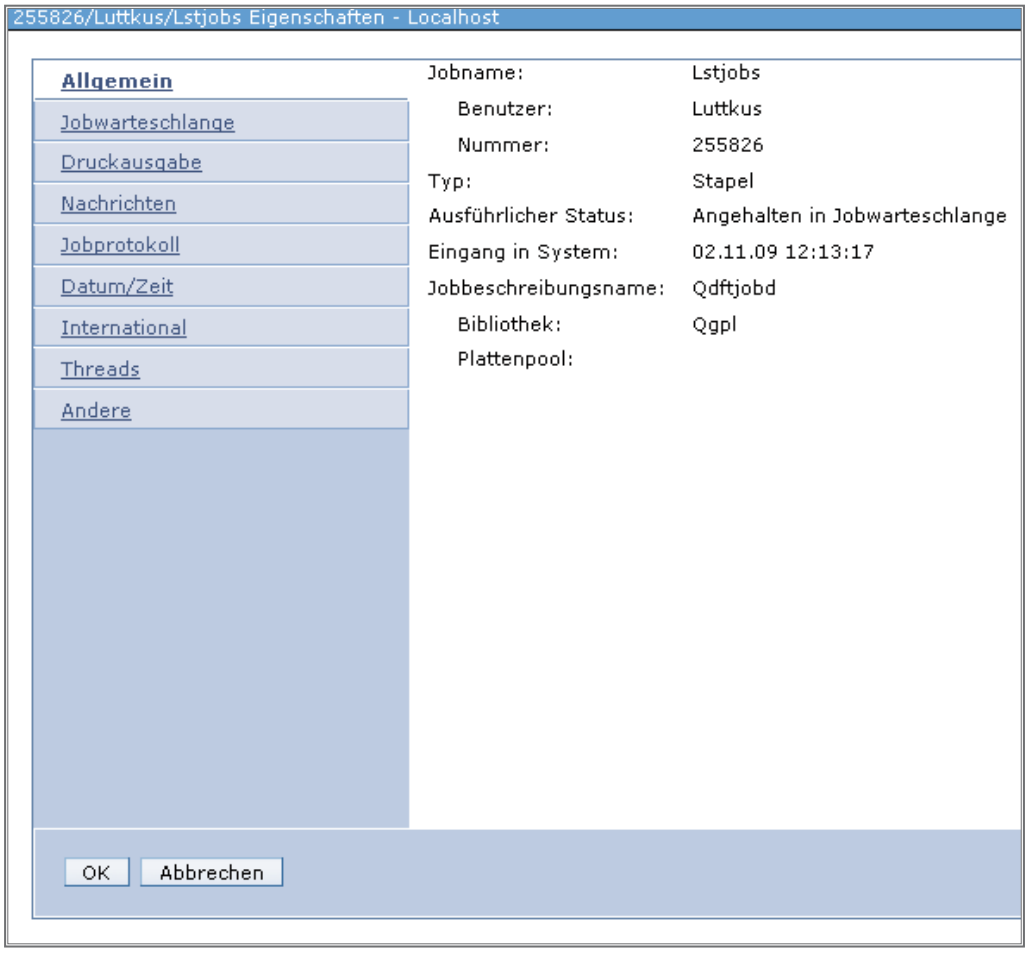
Job verschieben

Im Dialog „Verschieben“ soll der Job „Lstjobs“ in die Jobwarteschlange QCTL verschoben werden. Mit Hilfe des Dialogs „Durchsuchen…“ können Jobwartschlangen interaktiv gesucht werden.

Löschen/Beenden:
Mit dieser Auswahl entfernen Sie einen Job aus der Jobwarteschlange.

Eigenschaften:
Sollen die Eigenschaften eines Jobs, der auf Verarbeitung wartet, bewertet und ggf. verändert werden, so gehen Sie in den Dialog „Eigenschaften“.

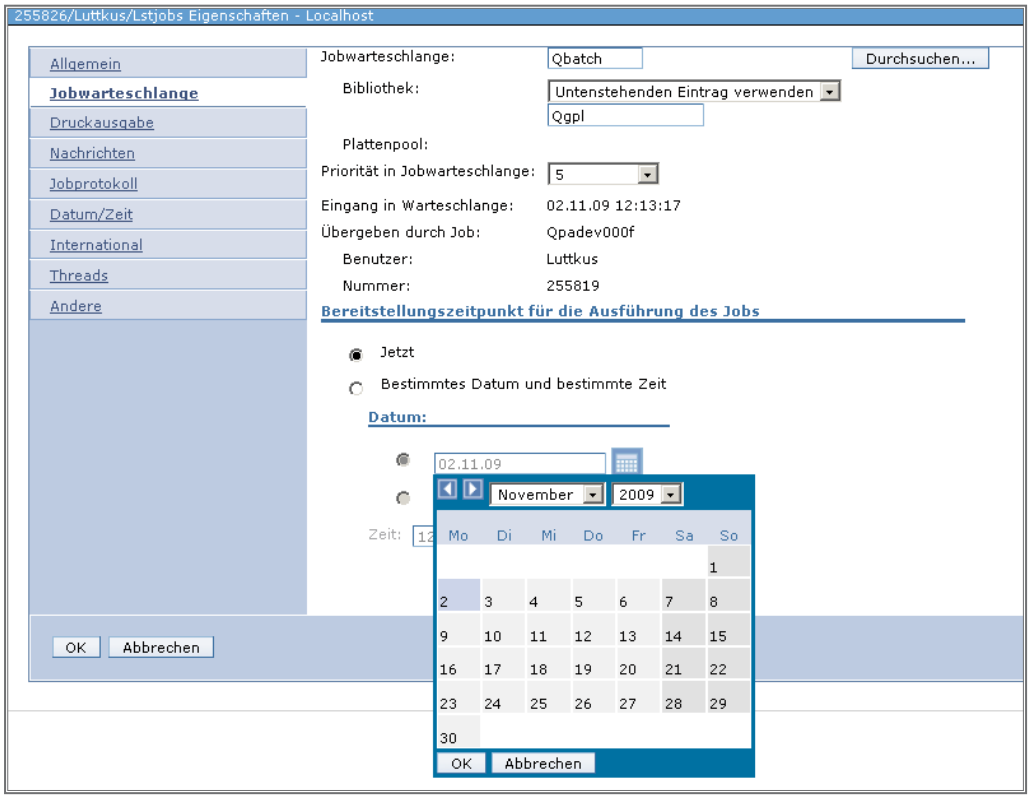
7.7.4.4.3
Seite 6



Register „Allgemein“ – Jobeigenschaften

Im Register „Allgemein“ finden Sie allgemeine Jobkenndaten:
- Jobname, Benutzer und Jobnummer
- Jobtyp
- Jobstatus
- Eingang ins System
- Zugeordnete Jobbeschreibung

Das Register „Jobwarteschlange“ stellt Ihnen Steuerungsmöglichkeiten für den in der Jobwarteschlange wartenden Job zur Verfügung.



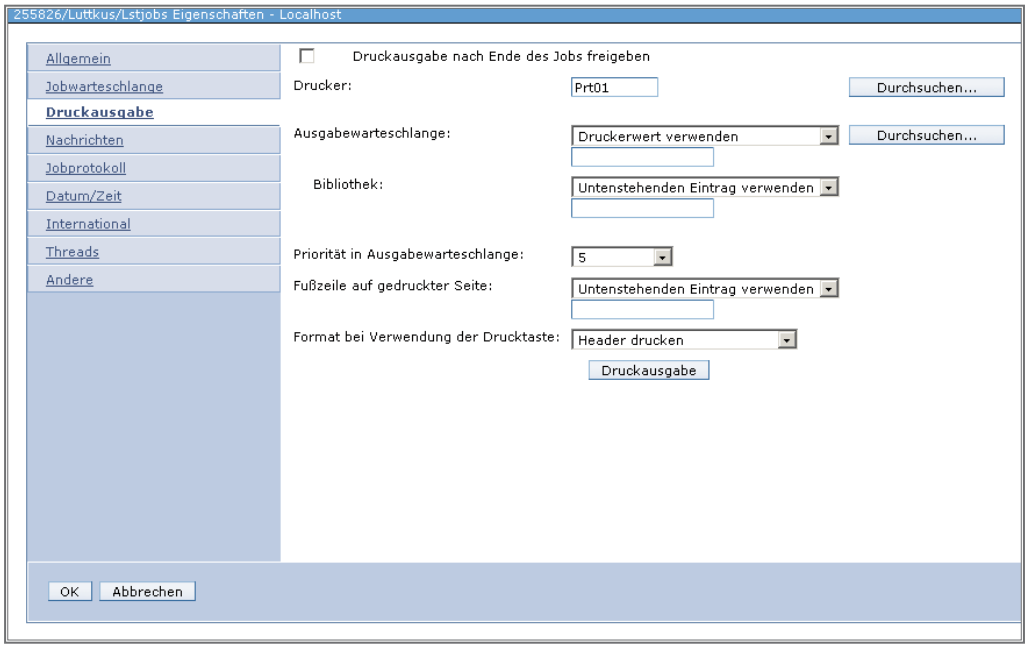
Register „Jobwarteschlange" – Jobeigenschaften

Folgende Parameter sind dabei wichtig:

- Jobwarteschlange/Bibliothek: Mit Hilfe dieser Parameter ordnen Sie den Job einer neuen Jobwarteschlange zu.
- Priorität: Durch Vergabe eines niedrigeren Werts wird der Job früher in das System zur Verarbeitung übergeben.
- Parametergruppe Bereitstellungszeitpunkt: Geben Sie hier ein Datum und eine Uhrzeit ein, um anzuzeigen, wann der Job zur Verarbeitung an das System übergeben werden soll.

Im Register „Druckausgabe" finden Sie relevante Parameter zur Drucker- und Ausgabewarteschlangenzuordnung.

7.7.4.4.3
Seite 8



Register „Druckausgabe" – Jobeigenschaften

Folgende wichtige Parameter finden Sie hier:

Drucker:
Ordnen Sie dem Job temporär einen neuen Drucker zu. Mit Hilfe des Dialogs „Durchsuchen…" sind Sie in der Lage, interaktiv aus der Systemumgebung der iSeries einen Drucker auszusuchen.

Ausgabewarteschlange:
Ordnen Sie dem Job temporär eine neue Ausgabewarteschlange zu. Auch hier können Sie interaktiv das System nach einer geeigneten Warteschlange durchsuchen.

Priorität in Ausgabewarteschlange:
Durch Reduzierung dieses Werts erhält Ihr Job eine höhere Priorität in der Ausgabewarteschlange und wird somit früher gedruckt.

Neben der Arbeit mit Jobs in einer Jobwarteschlange kann auch mit der Jobwarteschlange selbst gearbeitet werden. Diese Arbeit initiieren Sie, indem Sie die gewünschte Jobwarteschlange markieren und sich dann das Kontextmenü anzeigen lassen.

Mit Jobwarteschlangen arbeiten

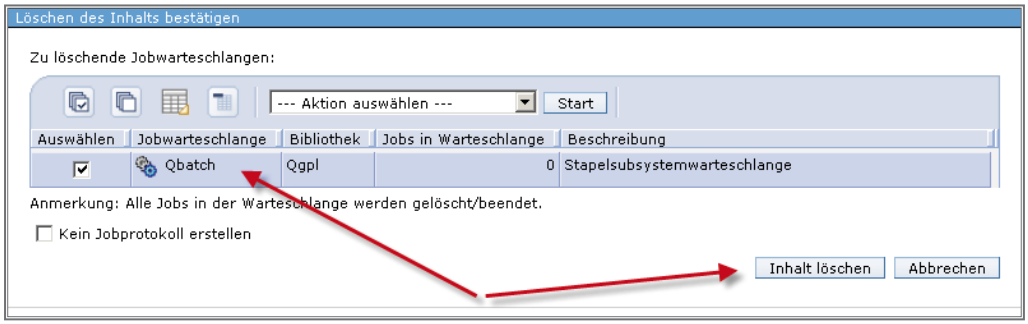
Hier sehen Sie das Kontextmenü für Jobwarteschlangen:

Folgende Auswahlmöglichkeiten bestehen:

Anhalten:
Die markierte Jobwarteschlange anhalten. Wird nur zur Verfügung gestellt, wenn die Jobwarteschlange freigegeben ist.

Freigeben:
Die markierte Jobwarteschlange freigeben. Wird nur zur Verfügung gestellt, wenn die Jobwarteschlange angehalten ist.

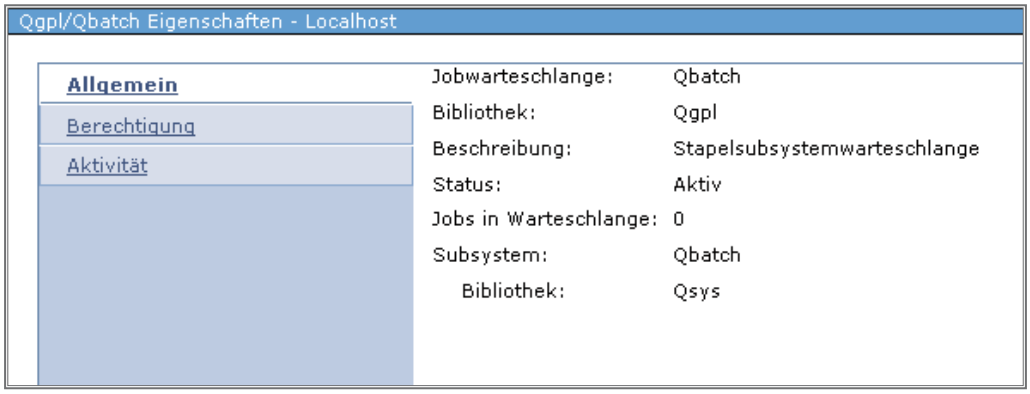
Inhalt löschen:
Sollen alle Jobs innerhalb einer Jobwarteschlange gelöscht werden, so gehen Sie in diesen Punkt.



Wartende Jobs löschen

7.7.4.4.3
Seite 10

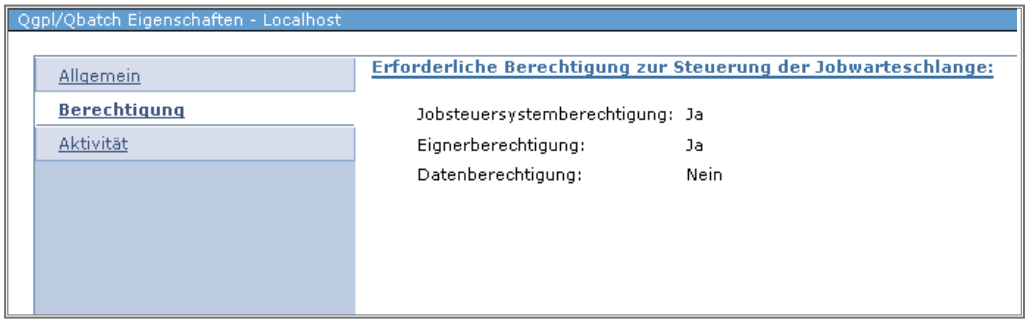
Eigenschaften:



Allgemeine Eigenschaften einer Jobwarteschlange

Wichtiger Parameter im Register „Allgemein“:

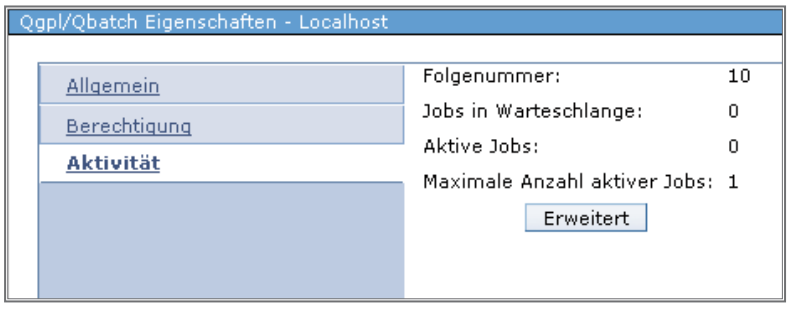
Status:
· aktiv – Jobwarteschlange ist einem Subsystem zugeordnet.



Berechtigungseigenschaften einer Jobwarteschlange

In diesem Register lassen Sie sich anzeigen, wer auf welche Weise mit dieser Jobwarteschlange und mit den darin enthaltenen Jobs arbeiten darf.

Im Register „Aktivität“ finden Sie u. a.:

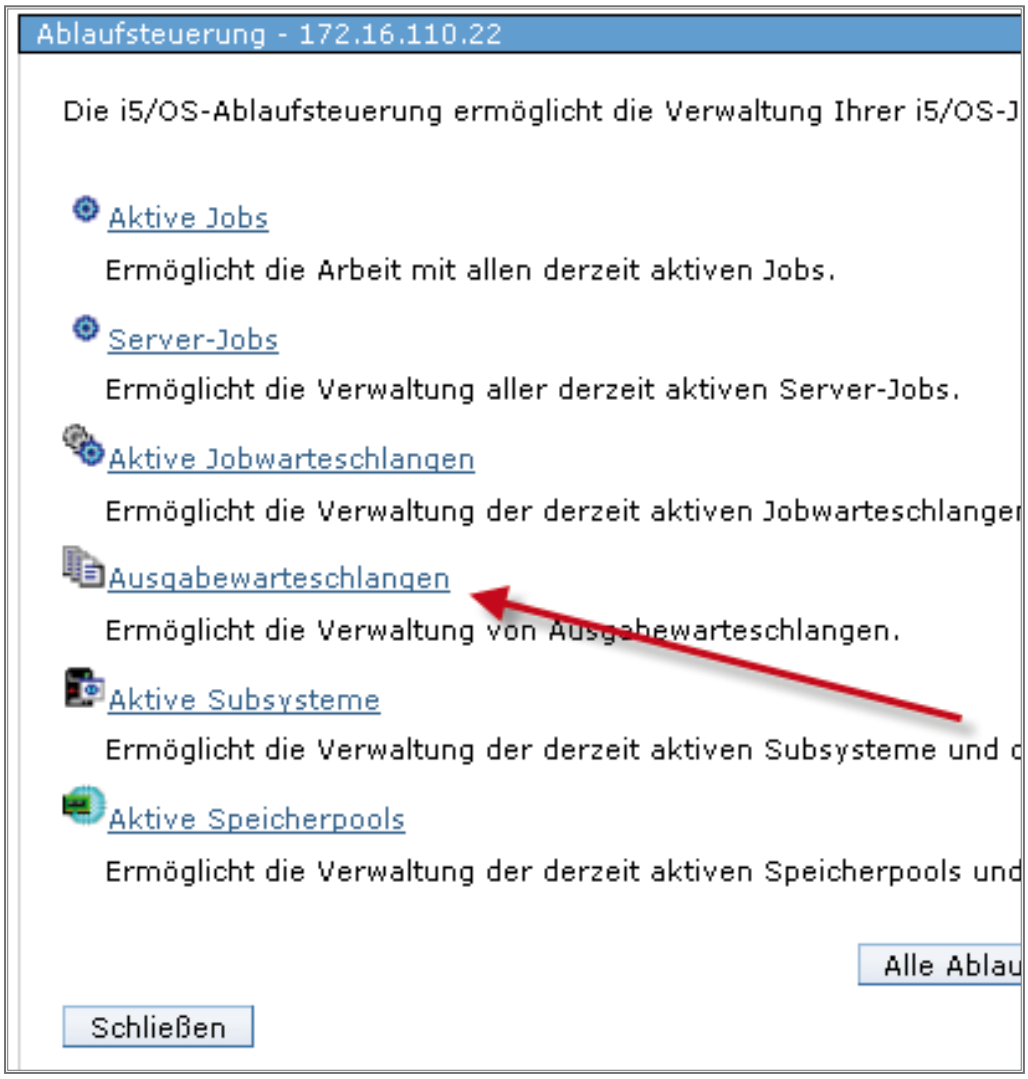


Aktivitätseigenschaften

Die Anzahl der Jobs, die eine Jobwarteschlange gleichzeitig an das angeschlossene Subsystem übergeben kann.

Mit Ausgabewarteschlangen arbeiten

Diese Funktion ermöglicht es Ihnen, direkt mit einer Ausgabewarteschlange zu arbeiten.

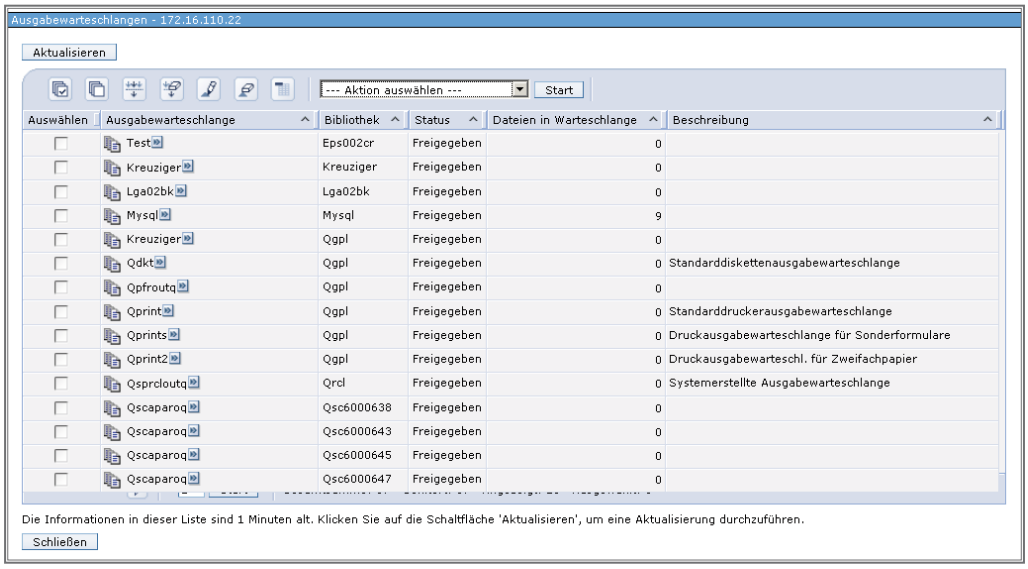


Mit Ausgabewarteschlangen arbeiten

Folge ich dem gezeigten Link, so wird eine Liste aller Ausgabewarteschlangen angezeigt, für die ich ein Zugriffsrecht habe.

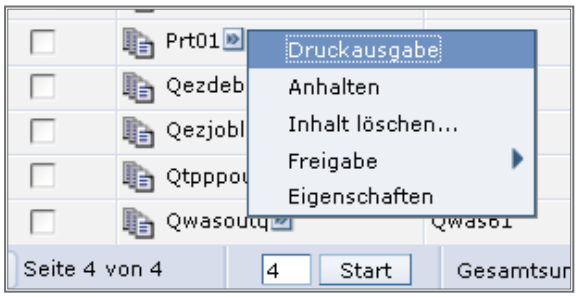
Sie stellt quasi eine direkte Implementierung des CL-Befehls WRKOUTQ (Mit Ausgabewarteschlangen arbeiten) dar, inklusive der entsprechenden Subbefehle.

7.7.4.4.4
Seite 2



Anzeige der Ausgabewarteschlangen

Um mit einer Warteschlange zu arbeiten, fordere ich das Kontextmenü an:



Kontextmenü für Ausgabewarteschlangen

Druckausgabe:
Ich gehe hier nicht weiter auf diese Funktionen ein, da Sie im Bereich der Basisfunktionen bereits beschrieben wurden.

Anhalten/Freigeben:
Je nach aktuellem Status der Ausgabewarteschlange ermöglicht diese Auswahl das Anhalten bzw. Freigeben einer Warteschlange.

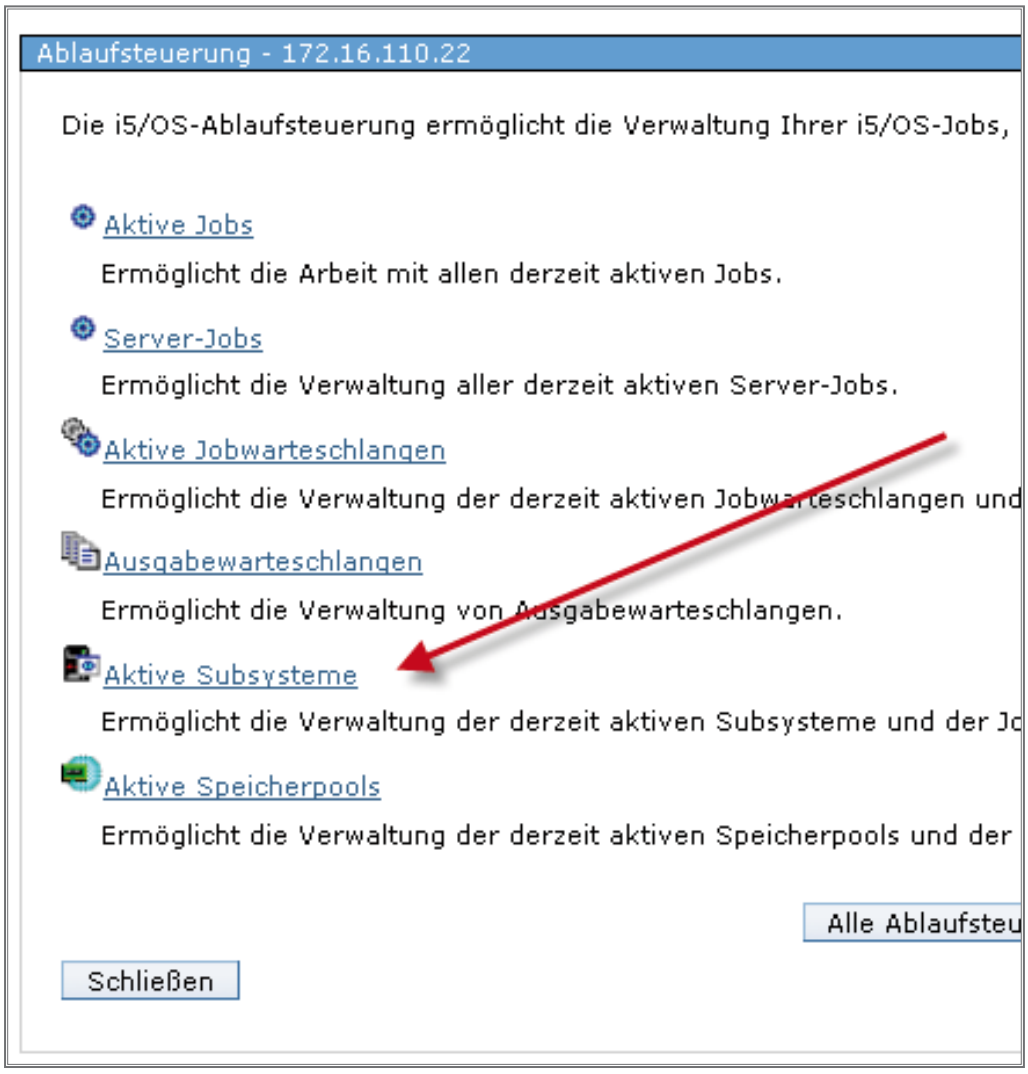
Inhalt löschen:
Der Inhalt einer Ausgabewarteschlange wird gelöscht.

Freigabe:
Eine Ausgabewarteschlange wird als Drucker im Windows-Netzwerk angesehen. Eine freigegebene Ausgabewarteschlange kann hier bearbeitet werden.

Eigenschaften:
Hier sehen Sie den aktuelle Status der Warteschlange (Freigegeben/Angehalten) und erfahren, welchem Druckausgabeprogramm diese Warteschlange zugeordnet ist.

Mit Subsystemen arbeiten

Die Funktion „Mit Subsystemen arbeiten“ ist im weiteren Sinne auch eine Darstellung von aktiven Jobs im System. Anders als bei einer normalen WRKAKTJOB-Anzeige ist hier der Fokus der Anzeige jedoch auf das Subsystem gesetzt worden.



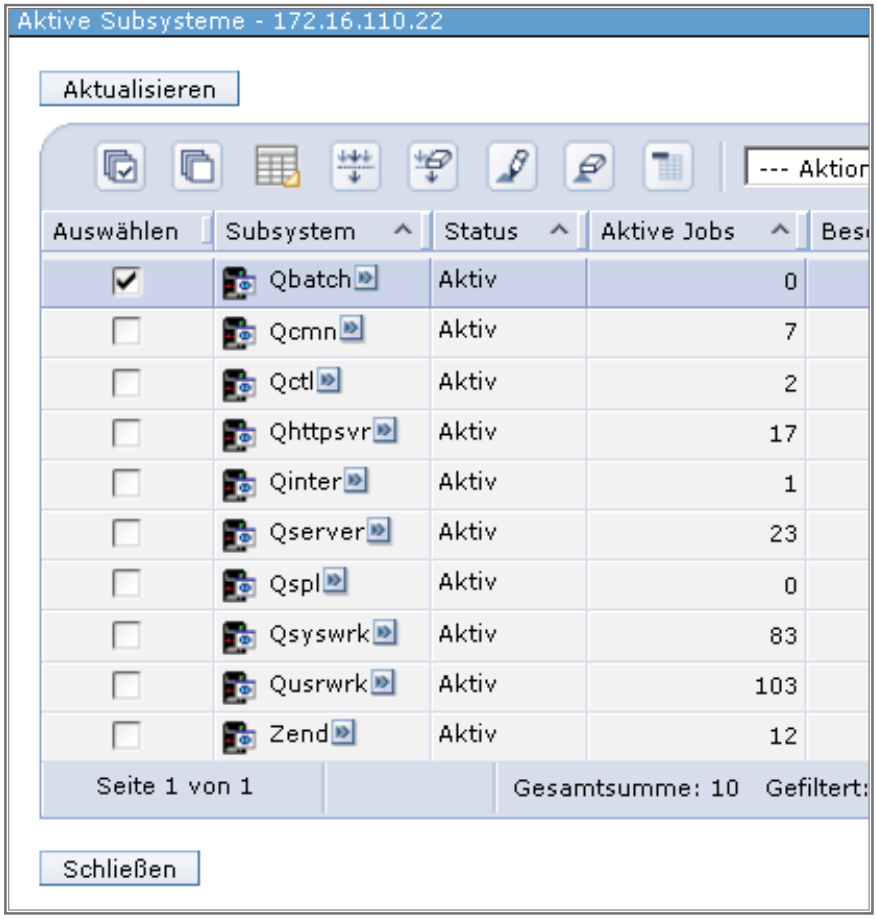
Funktion „Aktive Subsysteme“ aufrufen

Fragen, die in diesem Bereich beantwortet werden können, lauten:
- Welche Eigenschaften besitzt ein Subsystem?
- Welche Jobs werden durch ein Subsystem gesteuert?
- Welche Jobwarteschlangen bedienen ein Subsystem?

Verglichen mit CL-Befehlen werden hier Funktionen ausgelöst wie DSPSBSD (Subsystembeschreibung anzeigen – nur auszugsweise) und WRKSBSJOB (Mit Subsystemjobs arbeiten). Die zuletzt genannte Funktion ermöglicht dann den Drill-Down auf die Eigenschaften eines Jobs (WRKJOB – Mit Jobs arbeiten).

7.7.4.4.5
Seite 2

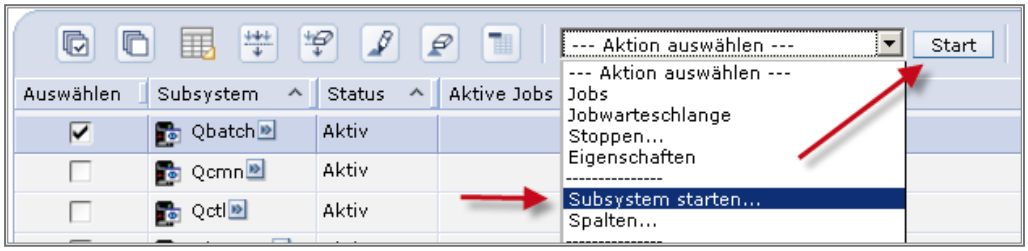
Lassen Sie uns einen kurzen Blick auf den folgenden Dialog im iSeries Navigator werfen.



Mit Subsystemen arbeiten

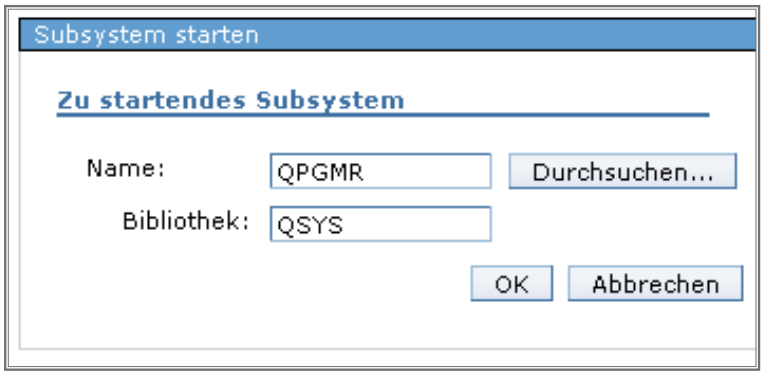
Sie sehen eine Liste der aktiven Subsysteme. An dieser Stelle zeigt sich auch eine erste Einschränkung. Wir sehen hier nicht alle Subsysteme, die im System definiert sind, sondern nur diejenigen, die derzeit im System aktiv sind. Eine Anzeige der nicht aktiven Subsysteme ist nicht möglich.

Um ein derzeit nicht gestartetes Subsystem zu starten, rufe ich aus dem Aktionsmenü die Funktion „Subsystem starten“ auf.



Subsystem starten

Nachdem Sie im Aktionsmenü die Funktion „Subsystem starten…“ gewählt und auf Start geklickt haben, erscheint der unten gezeigte Dialog.



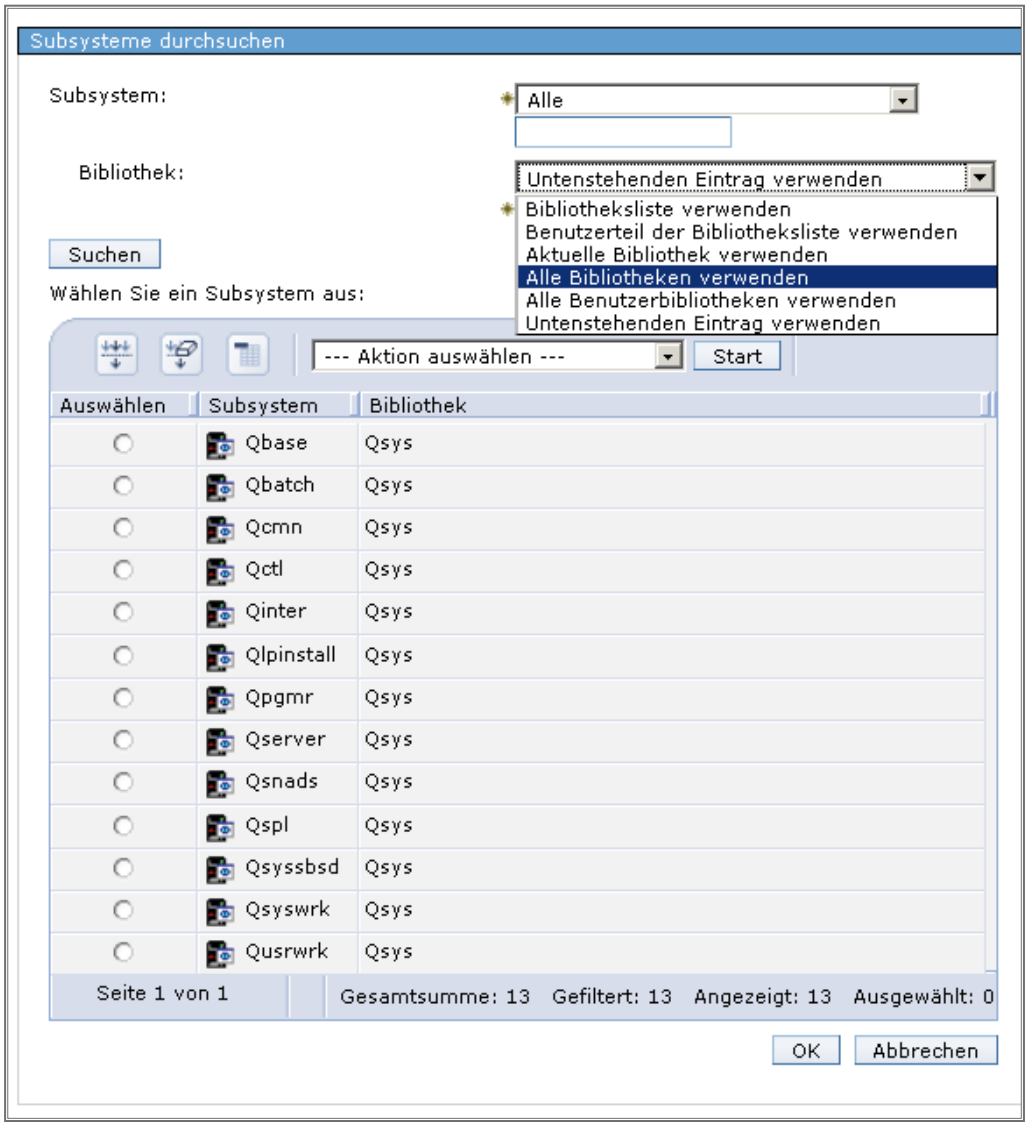
Nicht aktives Subsystem starten

Sind Ihnen der Name und die Bibliothek des zu startenden Subsystems bekannt, so geben Sie beide Werte einfach in die Parameter „Name“ bzw. „Bibliothek“ ein. Drücken Sie dann „OK“ und das betreffende Subsystem wird gestartet. Dies entspricht dem CL-Befehl STRSBS (Subsystem starten).

Sollten Sie jedoch den Namen des zu startenden Subsystems nicht genau kennen, so ermöglicht Ihnen der Dialog „Durchsuchen…“ die interaktive Suche nach dem in Frage kommenden Subsystem.

Ich drücke also auf „Durchsuchen…“. Im unten gezeigten Dialog finden Sie eine mächtige Suchfunktion, die Ihnen – Berechtigungen vorausgesetzt – eine Suche nach Subsystemen im gesamten System ermöglicht.

7.7.4.4.5
Seite 4



Nach einem Subsystem suchen

Die folgenden Parameter sind wichtig:

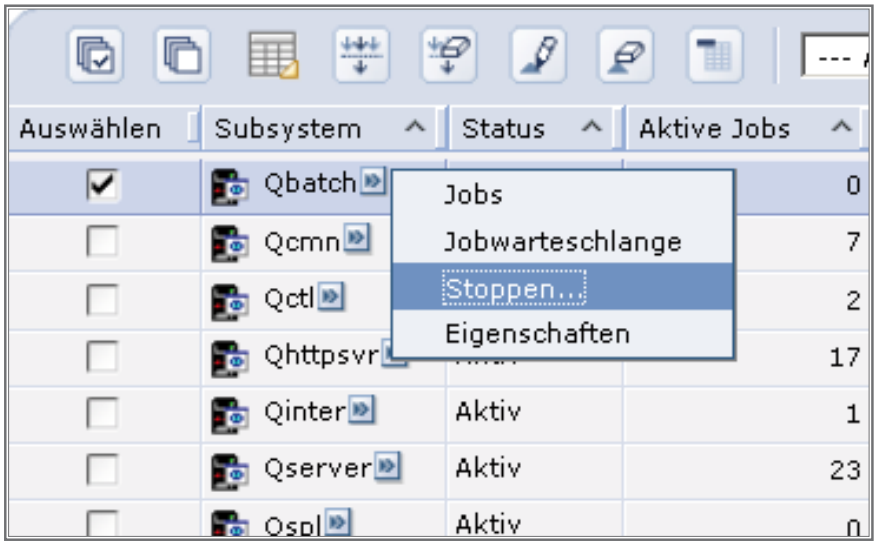
Subsystem:
Im Parameter „Subsystem“ sehen Sie den Standardwert „Alle“. Es ist jedoch auch möglich, Teilnamen einzugeben. Suchen Sie ein spezielles Subsystem, dessen Namen Ihnen bekannt ist, so geben Sie den entsprechenden Namen hier ein. Wichtig ist jetzt Ihr Eintrag in den Parameter „Bibliothek“.

Bibliothek:
In diesem Parameter sehen Sie den Einsetzwert „Bibliotheksliste verwenden“. Bei der Suche nach Subsystemen wird die Bibliothekssuchliste desjenigen herangezogen, der implizit oder explizit die iSeries Navigator-Anmeldung am System iSeries durchgeführt hat.

Ich belasse es bei den Standardwerten und klicke mit der Maus auf „Suchen“. Im Dialog „Wählen Sie ein Subsystem aus“ erscheint eine Liste aller Subsystembeschreibungen, die durch das Heranziehen meiner Bibliotheksliste gefunden wurden.

Ich markiere das Subsystem QSPL und drücke „OK“. Dann wird das ausgewählte Subsystem in den Dialog „Subsystem starten“ übernommen und kann gestartet werden.

Um jetzt mit einem aktiven Subsystem arbeiten zu können, fordere ich das folgende Kontextmenü an.



Kontextmenü „Subsysteme“

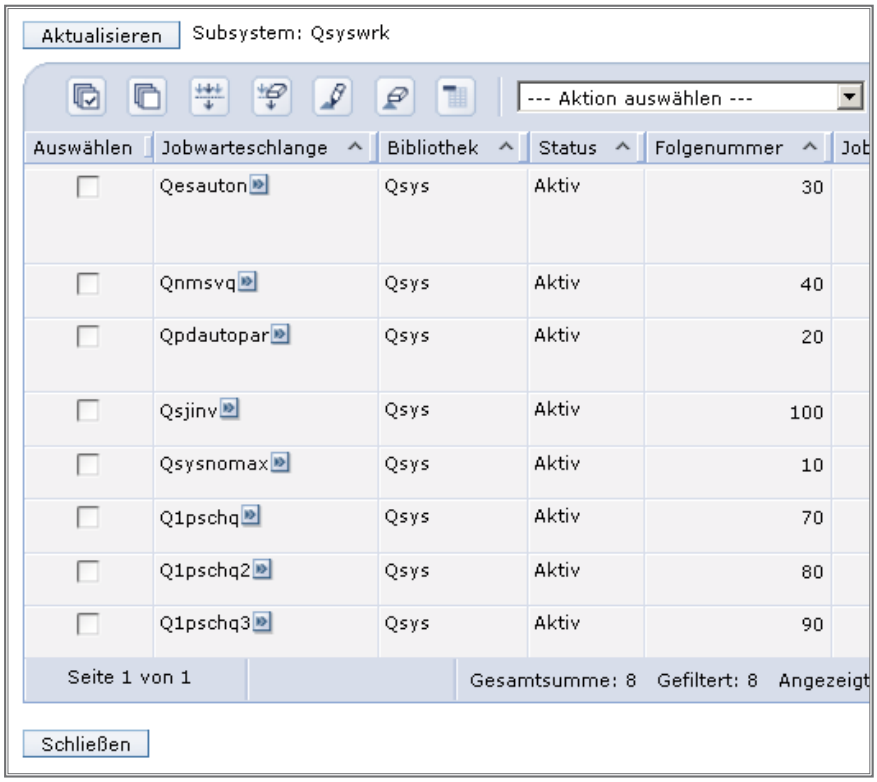
Folgende interessante Auswahlmöglichkeiten bestehen:

Stoppen…:
Beenden Sie das gewählte Subsystem. Subsysteme können sowohl kontrolliert als auch sofort beendet werden.

Eigenschaften:
Lassen Sie sich die Definitionsdaten eines Subsystems anzeigen.

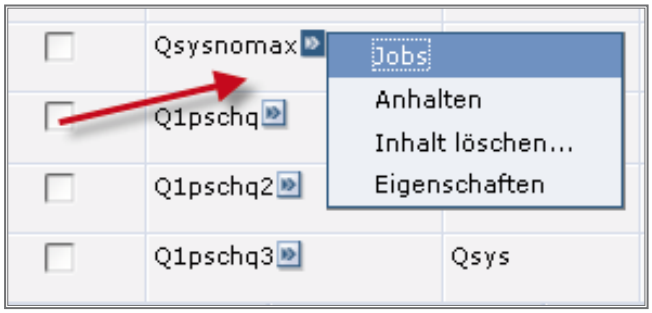
Jobwarteschlange:
Ich gehe jetzt für das Subsystem QSYSWRK in die Kontextmenü-Auswahl „Jobwarteschlange“. Es öffnet sich ein Dialog, der Ihnen eine Liste aller Jobwarteschlangen anzeigt, die für das Subsystem QSYSWRK aktiv sind.

7.7.4.4.5
Seite 6



Anzeige der Jobwarteschlangen für QSYSWRK

Auch in dieser Anzeige gibt es ein Kontextmenü. Ich markiere einfach die Jobwarteschlange QSYSNOMAX und fordere mit der rechten Maustaste folgendes Kontextmenü an.



Kontextmenü für Jobwarteschlange

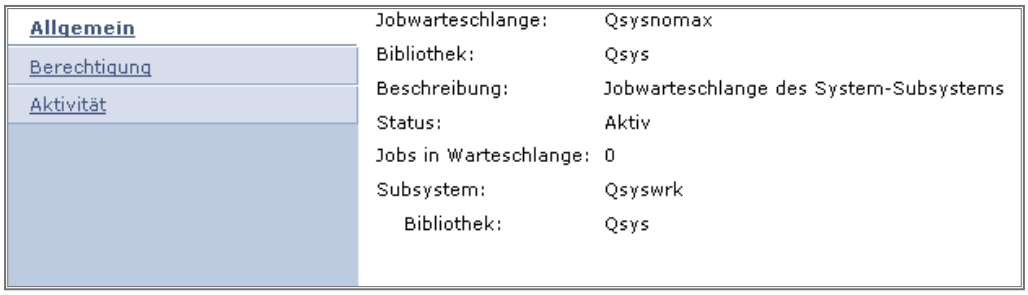
Hier finden wir wiederum die typischen Navigator-Verarbeitungsmöglichkeiten für Jobwarteschlangen, die im Abschnitt „Mit Jobwareschlangen arbeiten“ bereits beschrieben wurden.

Anhalten:
Die markierte Jobwarteschlange anhalten.

Freigeben:
Die markierte Jobwarteschlange freigeben.

Inhalt löschen:
Sollen alle Jobs innerhalb einer Jobwarteschlange gelöscht werden, so gehen Sie in diesen Punkt.

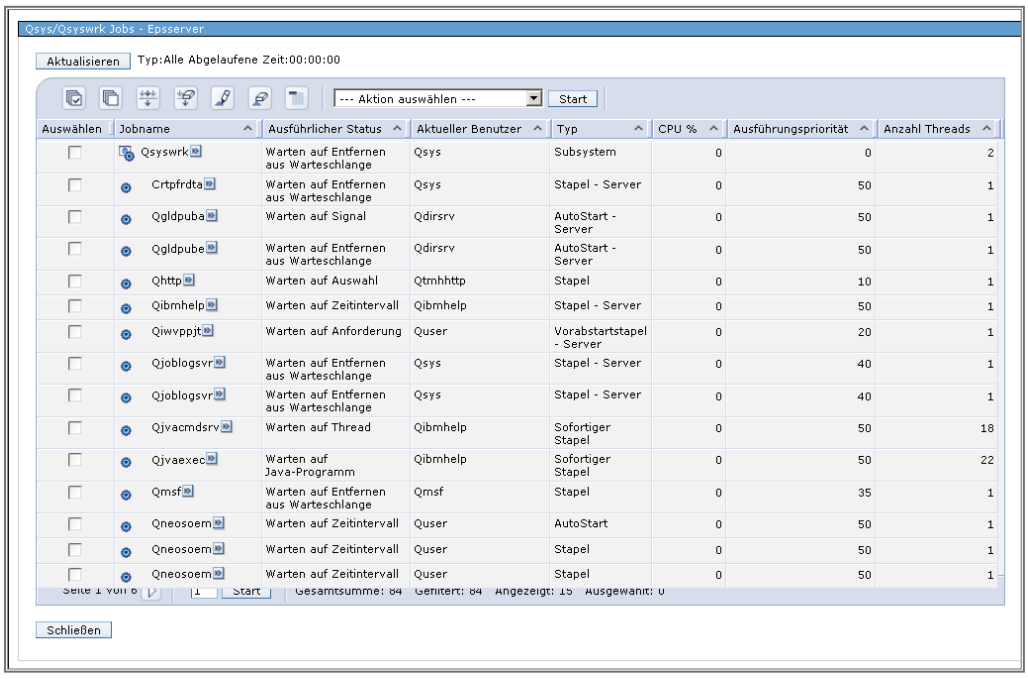
Eigenschaften:
Auch diese Auswahl entspricht weitgehend dem Eigenschaften-Dialog für Jobwarteschlangen.



Eigenschaften einer Jobwarteschlange

Eine Erweiterung besteht darin, dass subsystembezogene Daten den Eigenschaften der Jobwarteschlange hinzugefügt wurden.

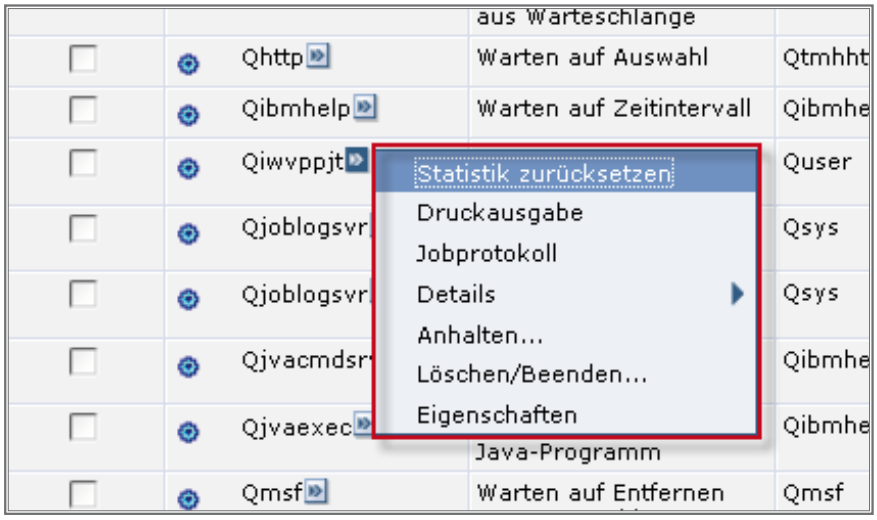
Jobs:
Ich gehe jetzt in die Hauptanzeige der Funktion „Aktive Subsysteme“ zurück und fordere die Auswahl „Jobs“ aus dem Kontextmenü des Subsystems QSYSWRK an.



Mit Subsystemjobs arbeiten

7.7.4.4.5
Seite 8

Im Prinzip entspricht die Detailanzeige der Funktion WRKSBSJOB (Mit Subsystemjobs arbeiten). In der Anzeige sehen Sie die derzeit aktiven Jobs im Subsystem QSYSWRK.

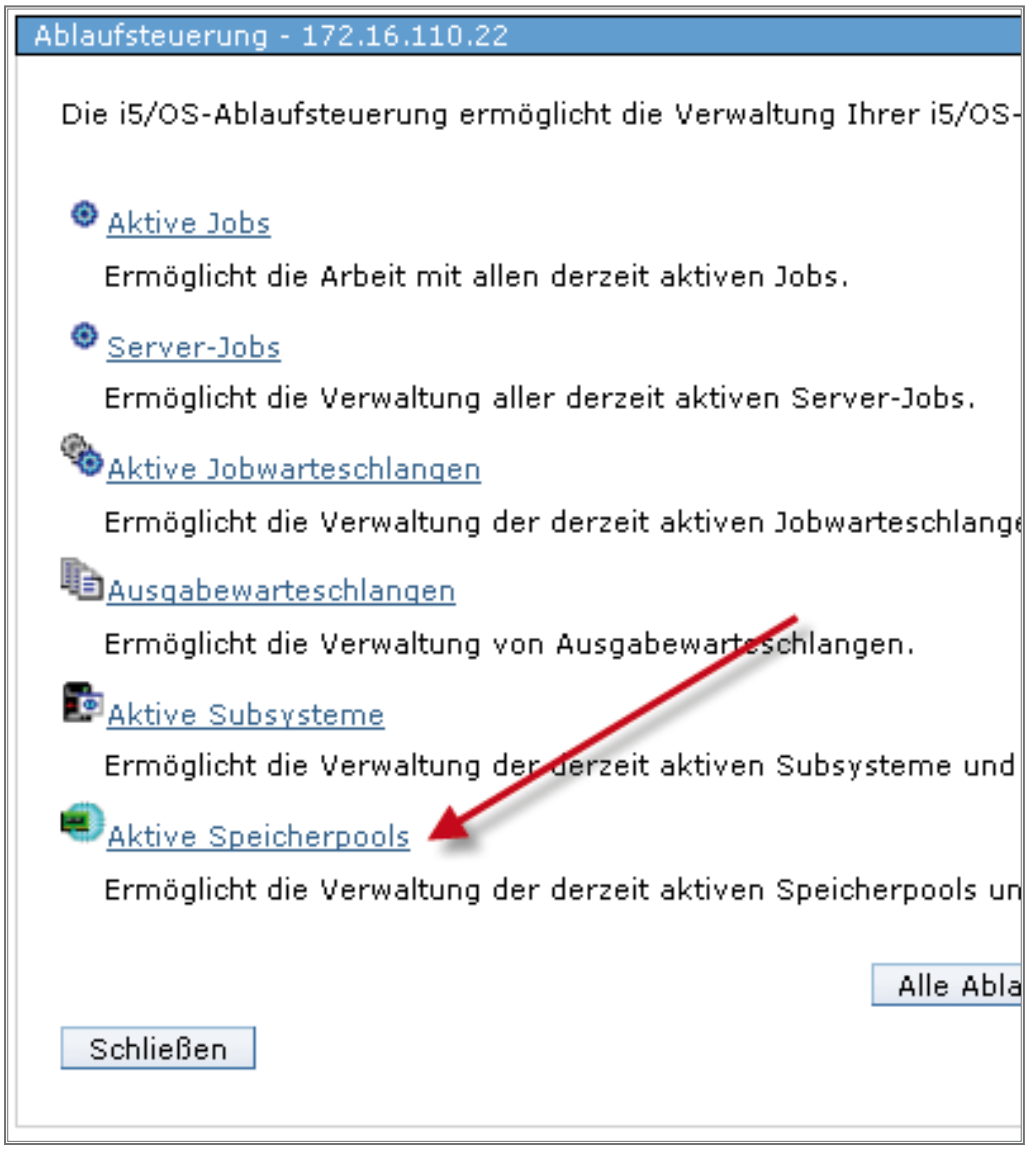


Kontextmenü für Subsystemjobs

In der oben gezeigten Abbildung habe ich für einen Job das Kontextmenü angefordert. Es ist unschwer zu erkennen, dass es sich hierbei um das gleiche Kontextmenü handelt, das Sie bereits aus dem Abschnitt für die Arbeit mit einzelnen Jobs kennen. Sollten Ihnen die Möglichkeiten dieses Menüs nicht geläufig sein, so lesen Sie bitte die entsprechenden Passagen noch einmal durch.

Mit Speicherpools arbeiten

Die Funktion „Mit Speicherpools arbeiten“ ist im weiteren Sinne auch eine Darstellung von aktiven Jobs im System. Gegenüber einer normalen WRKAKTJOB-Anzeige liegt jedoch die Betonung dieser Funktion auf der Darstellung von Jobaktivitäten in Bezug zum betrachteten Speicherpool.



Mit aktiven Speicherpools arbeiten

Fragen, die in diesem Bereich beantwortet werden können, lauten:

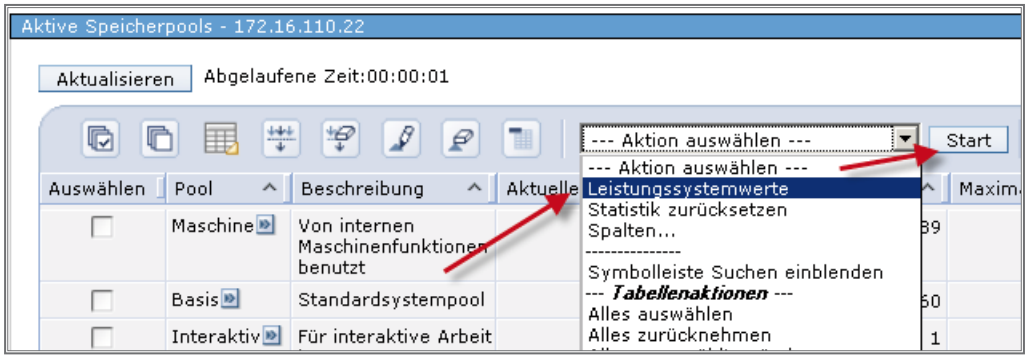
- Welche Eigenschaften besitzt ein Speicherpool?
- Welche Systemwerte steuern Speicherpools?
- Welche Jobs werden in einem Speicherpool ausgeführt?
- Welche Subsysteme bedienen sich von einem Speicherpool?

7.7.4.4.6
Seite 2

CL-Befehle, die zum Vergleich herangezogen werden können, sind:

- WRKSYSSTS – Mit Systemstatus arbeiten (in Auszügen)
- WRKSYSVAL – Mit Systemwerten arbeiten (in Auszügen)
- WRKSBS – Mit Subsystemen arbeiten (in Auszügen)
- WRKJOB – Mit Jobs arbeiten
- WRKJOBQ – Mit Jobwarteschlangen arbeiten
- WRKACTJOB – Mit aktiven Jobs arbeiten
- WRKSHRPOOL – Mit gemeinsamen Speicherpools arbeiten

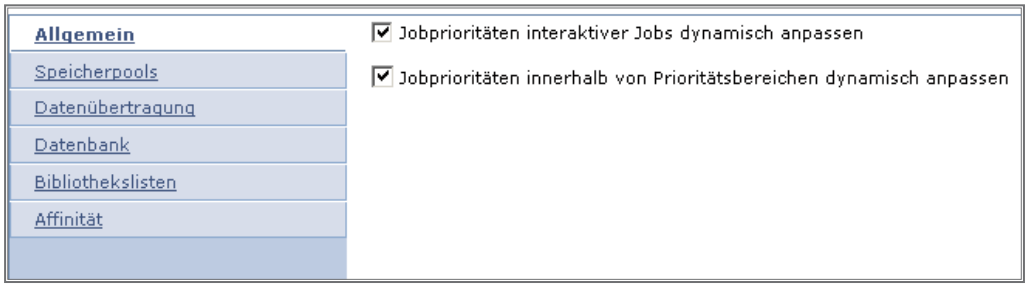
Lassen Sie uns einen kurzen Blick auf den folgenden Dialog im Navigator werfen. Ich fordere das Aktionsmenü für die Funktion „Arbeitsspeicherpools“ an und wähle die Option „Leistungssystemwerte“.



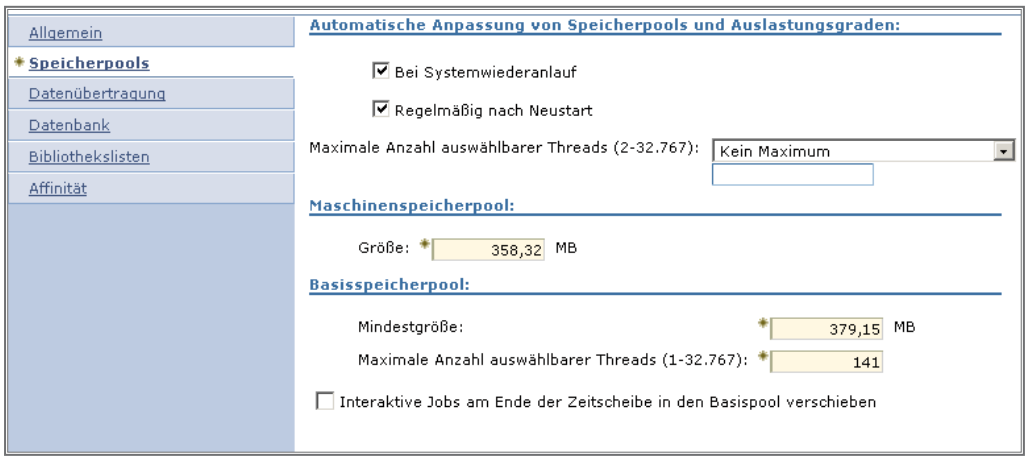
Mit Systemwerten für Pools arbeiten

Nach einem Klick auf „Start“ erscheint der folgende Dialog für die Arbeit mit Systemwerten, die die Arbeit von Speicherpools im System steuern.

Im Folgenden werde ich die Systemwerte nicht einzeln durchgehen, da deren Bezeichnungen sehr gut ins Deutsche übersetzt wurden, so dass die meisten der dargestellten Parameter für sich selbst sprechen.



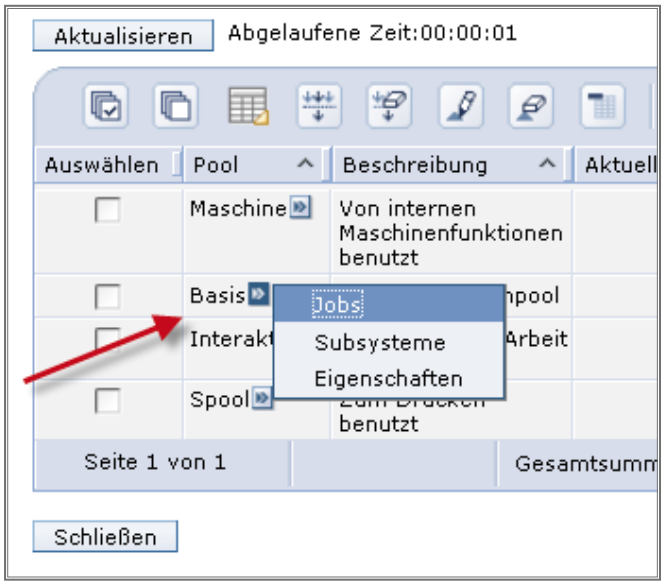
Register „Allgemein“



Register „Speicherpools“

Wichtig ist für Sie, zu erkennen, dass es sich bei diesem Dialog um eine Zusammenfassung aller Systemwerte für die Steuerung von Speicherpools und deren assoziierte Funktionen handelt.

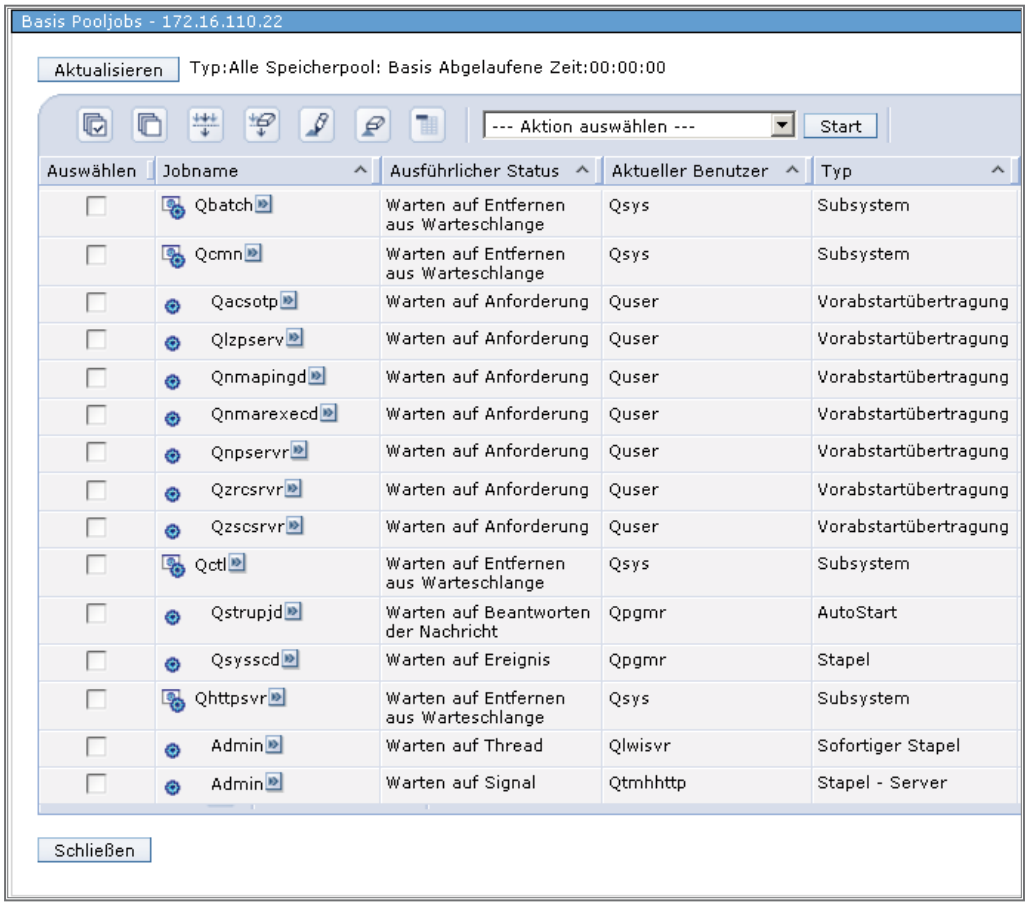
Ich markiere jetzt den Basepool meines Systems. Betrachten Sie das von mir angeforderte Kontextmenü.



Kontextmenü für Basepool

Die Funktion „Jobs“, die Sie hier erreichen, stellt alle aktiven Jobs dar, die derzeit im Basepool meines Systems laufen.

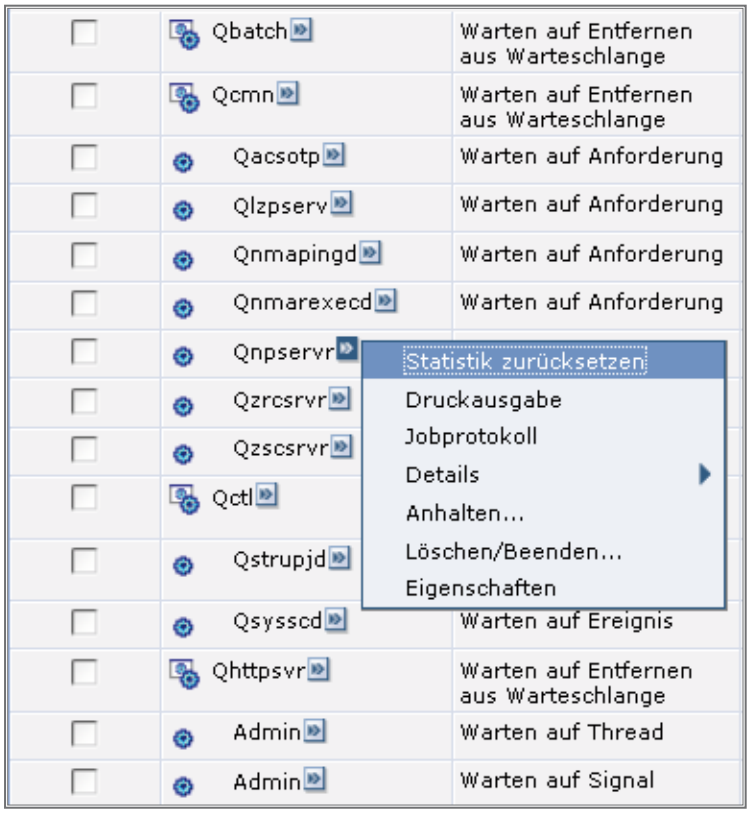
7.7.4.4.6
Seite 4



Auswählen	Jobname	Ausführlicher Status	Aktueller Benutzer	Typ
☐	Qbatch	Warten auf Entfernen aus Warteschlange	Qsys	Subsystem
☐	Qcmn	Warten auf Entfernen aus Warteschlange	Qsys	Subsystem
☐	Qacsotp	Warten auf Anforderung	Quser	Vorabstartübertragung
☐	Qlzpserv	Warten auf Anforderung	Quser	Vorabstartübertragung
☐	Qnmapingd	Warten auf Anforderung	Quser	Vorabstartübertragung
☐	Qnmarexecd	Warten auf Anforderung	Quser	Vorabstartübertragung
☐	Qnpservr	Warten auf Anforderung	Quser	Vorabstartübertragung
☐	Qzrcsrvr	Warten auf Anforderung	Quser	Vorabstartübertragung
☐	Qzscsrvr	Warten auf Anforderung	Quser	Vorabstartübertragung
☐	Qctl	Warten auf Entfernen aus Warteschlange	Qsys	Subsystem
☐	Qstrupjd	Warten auf Beantworten der Nachricht	Qpgmr	AutoStart
☐	Qsysscd	Warten auf Ereignis	Qpgmr	Stapel
☐	Qhttpsvr	Warten auf Entfernen aus Warteschlange	Qsys	Subsystem
☐	Admin	Warten auf Thread	Qlwisvr	Sofortiger Stapel
☐	Admin	Warten auf Signal	Qtmhhttp	Stapel - Server

Aktive Jobs im Pool

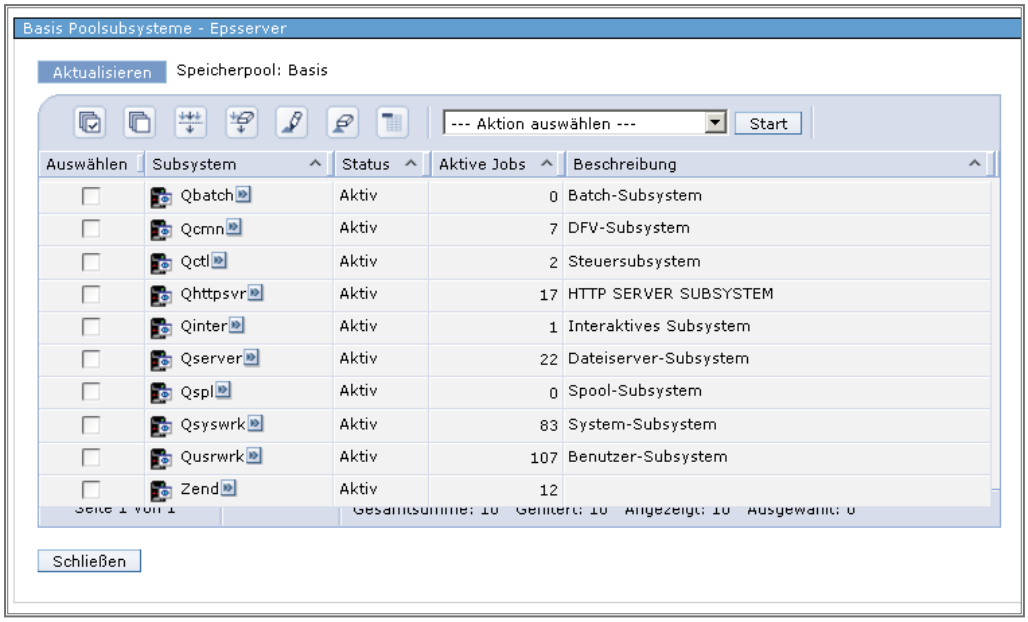
Fordern Sie das Kontextmenü für einen Job an, so gelangen Sie wieder in das Kontextmenü, das Sie mit aktiven Jobs arbeiten lässt.



Mit Jobs arbeiten

Also handelt es sich hier erneut um eine Variante der Funktion „Aktive Jobs“.

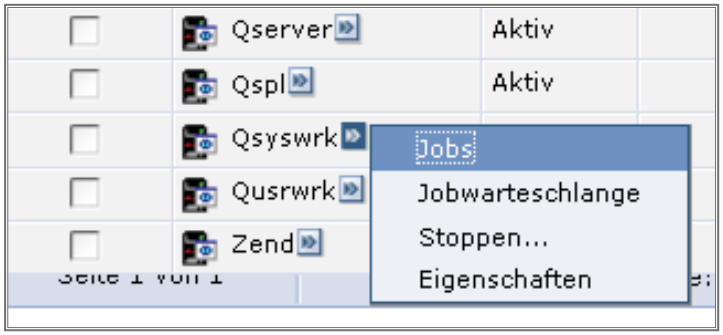
Gehen Sie im Kontextmenü für den Basepool in die Auswahl „Subsysteme“, so erhalten Sie eine Liste aller Subsysteme, die vom Basepool Speicher beziehen.



Subsysteme im Pool

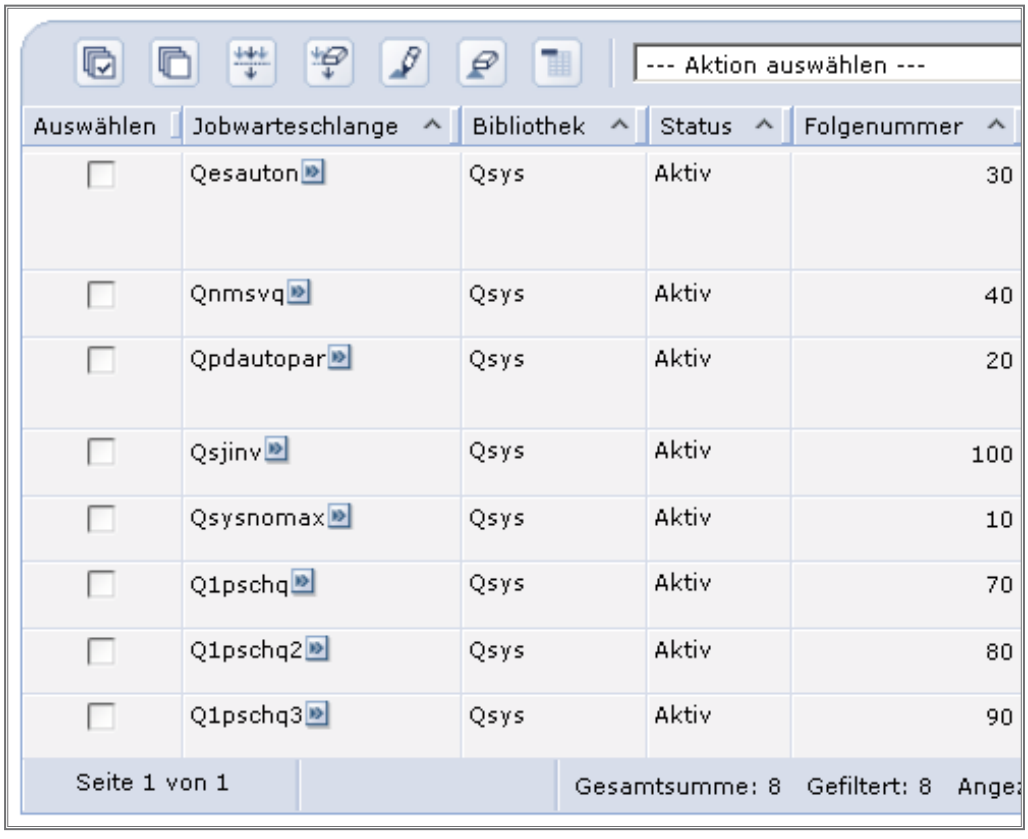
7.7.4.4.6
Seite 6

Öffnen Sie das Kontextmenü eines Subsystems, so können Sie mit den Objekten, die einem Subsystem zugeordnet sind, arbeiten.



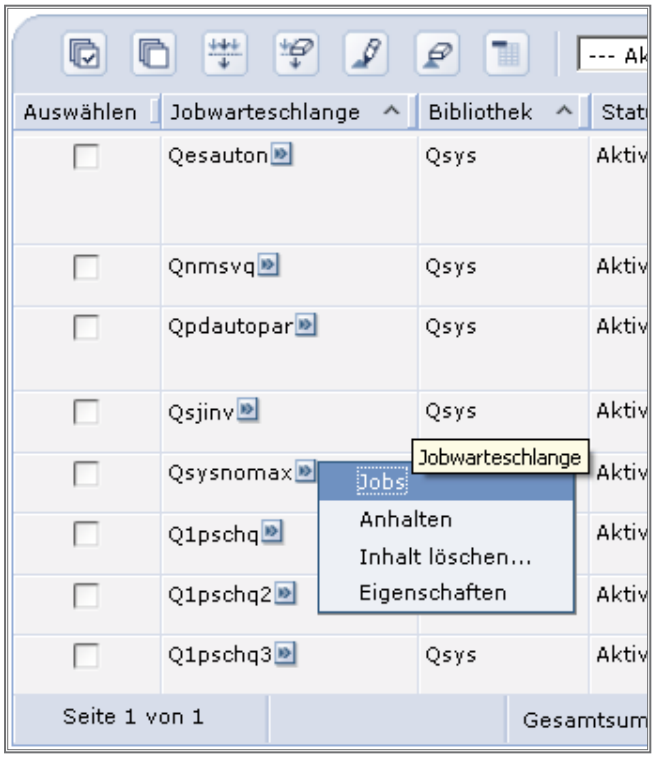
Mit Subsystemen arbeiten

Von hier gelangen Sie u. a. wieder in die Jobwarteschlangen, die an das betreffende Subsystem gebunden wurden.



Jobwarteschlangen für ein Subsystem

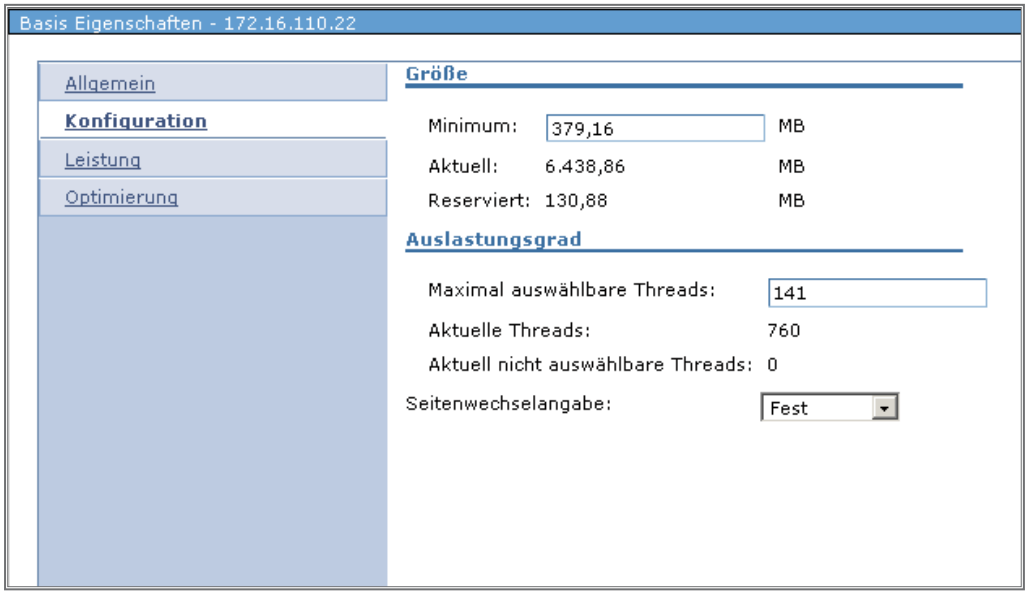
Über diesen Dialog können Sie nun einmal mehr mit den entsprechenden Jobwarteschlangen via Kontextmenü arbeiten.



Mit Jobwarteschlangen arbeiten

Sie sehen also, dass Sie in diesem Bereich eine mächtige Funktion haben, die Sie durch sämtliche Hierarchie-Instanzen der Arbeit mit Jobs im System führt.

Einen anderen Themenkomplex betreten Sie, wenn Sie sich die Eigenschaften eines Speicherpools anzeigen lassen. Stellvertretend für alle Pools im System öffne ich die Eigenschaften des Basepools.



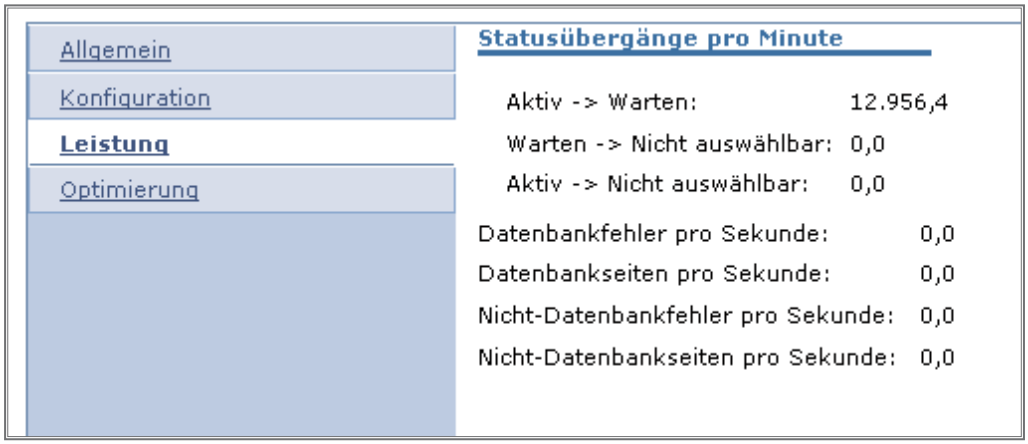
Eigenschaften Basepool

7.7.4.4.6
Seite 8

Im oben gezeigten Dialog sehen Sie sehr schön, dass es sich hier weitgehend um die Anzeige des Befehls WRKSYSSTS für einen Speicherpool – hier Basepool – handelt.

Wir sehen die aktuelle Dimensionierung des Pools, den Aktivitätslevel sowie das Caching-Verhalten (Seitenwechselangabe) für diesen Pool.

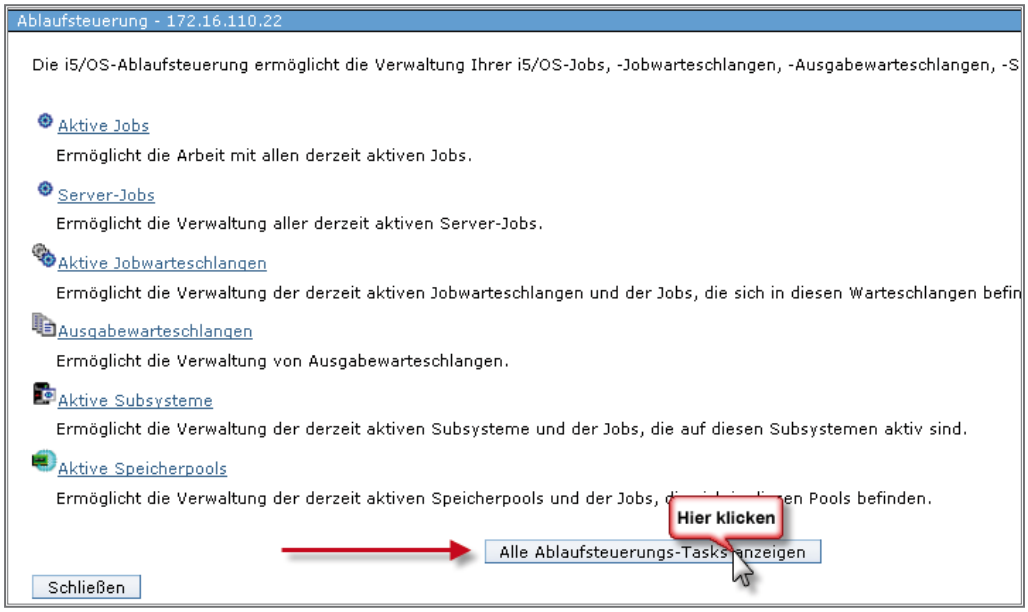
Dem nächsten Register „Leistung“ entnehmen Sie dann Angaben zum Paging-Verhalten im gewählten Speicherpool.



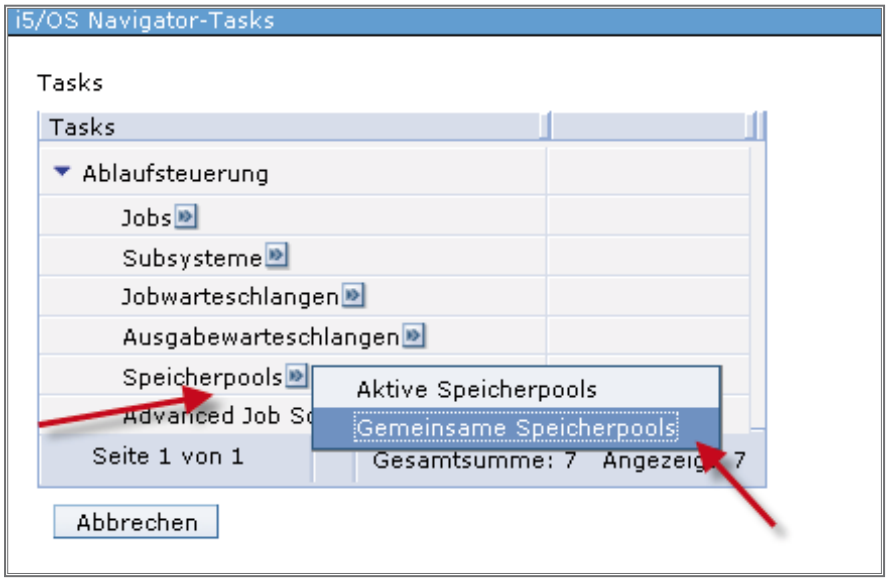
Register „Leistung“

Ausgegliedert von der Funktion „Aktive Pools“ ist die Funktion „Gemeinsame Pools“.

Die Arbeit mit gemeinsamen Pools ist etwas „versteckt“ untergebracht worden. Um diese Funktion zu finden, öffnen Sie das Hauptmenü der Funktion „Ablaufsteuerung“.

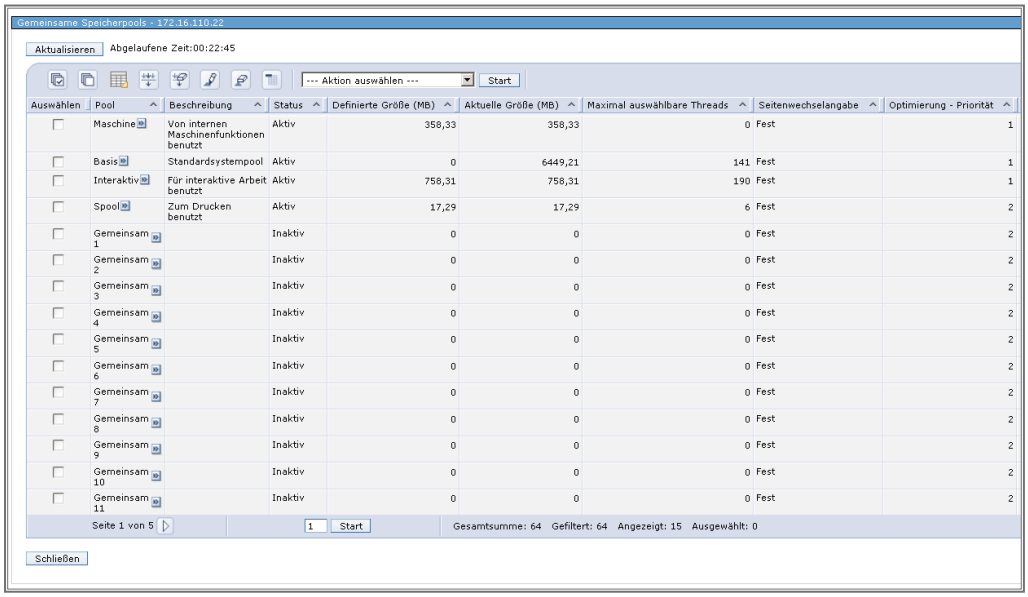


Mit Gemeinsamen Pools arbeiten – 1



Mit Gemeinsamen Pools arbeiten – 2

Hier öffnen Sie das Kontextmenü für Speicherpools und wählen die Option „Gemeinsame Speicherpools“.



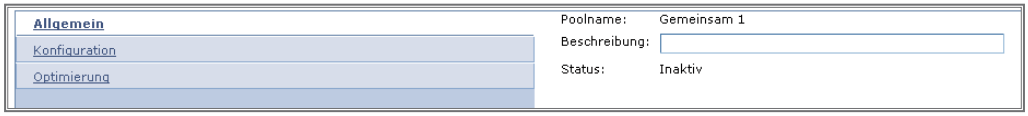
Mit Gemeinsamen Pools arbeiten – 3

Sie erhalten dann eine Liste aller gemeinsamen Speicherpools im System.

Mit Gemeinsamen Pools arbeiten – 4

Mit Hilfe des Kontextmenüs können Sie dann einen konfigurierten Speicherpool aktivieren (hier: „Freigeben“) oder wahlweise neue „Shared Pools“ definieren.

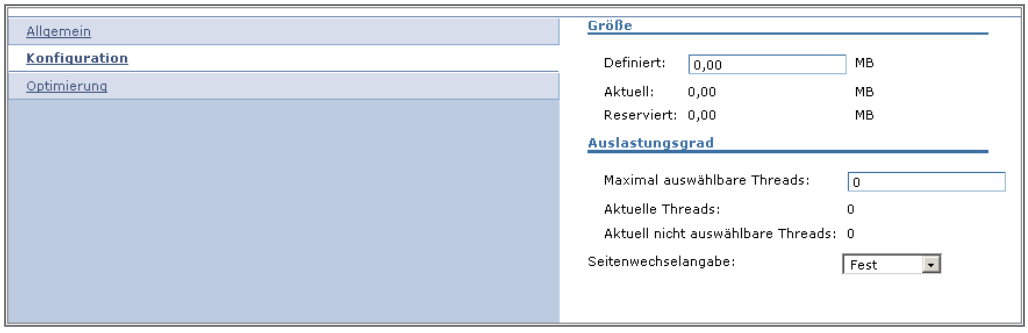
Ich gehe in den Eigenschaften-Dialog, der sich mit Hilfe des Kontextmenüs für den „Gemeinsamen Pool 1“ anzeigen lässt. Das Register „Allgemein“ erklärt sich von selbst.



Register „Allgemein“

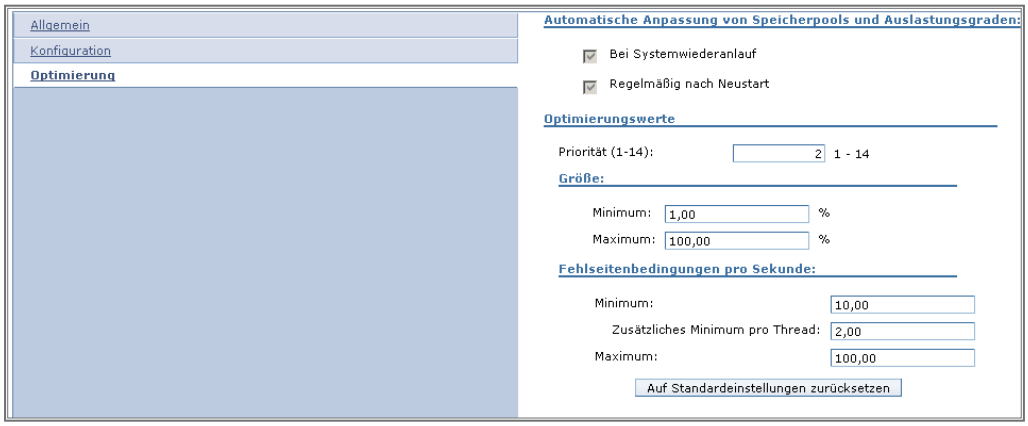
Hinterlegen Sie hier einfach einen beschreibenden Text für den „Gemeinsamen Pool 1“.

Im Register „Konfiguration“ wird die Größe des „Gemeinsamen Pool 1“ sowie der Aktivitätslevel für diesen Pool eingestellt. Darüber hinaus lässt sich auch das Caching-Verhalten für diesen Pool definieren.



Register „Konfiguration“

Im nächsten und letzten Register „Optimierung“ stellen Sie dann ein, inwieweit bei aktivierter Performanceanpassung dieser „Gemeinsame Pool 1“ bei der automatischen Anpassung von Leistungsdaten berücksichtigt werden soll.



Register Optimierung

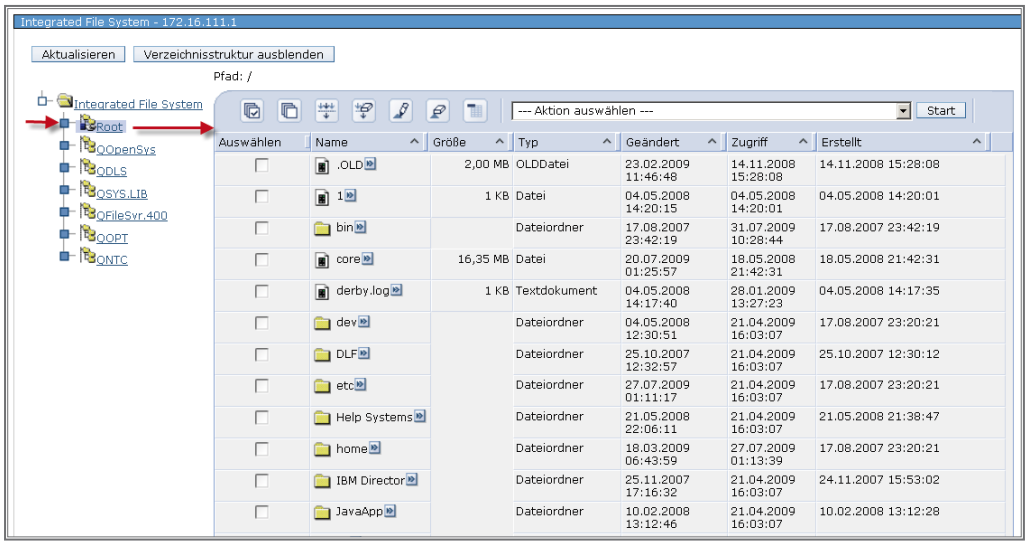
7.7.4.4.6

Seite 12

Dateisysteme

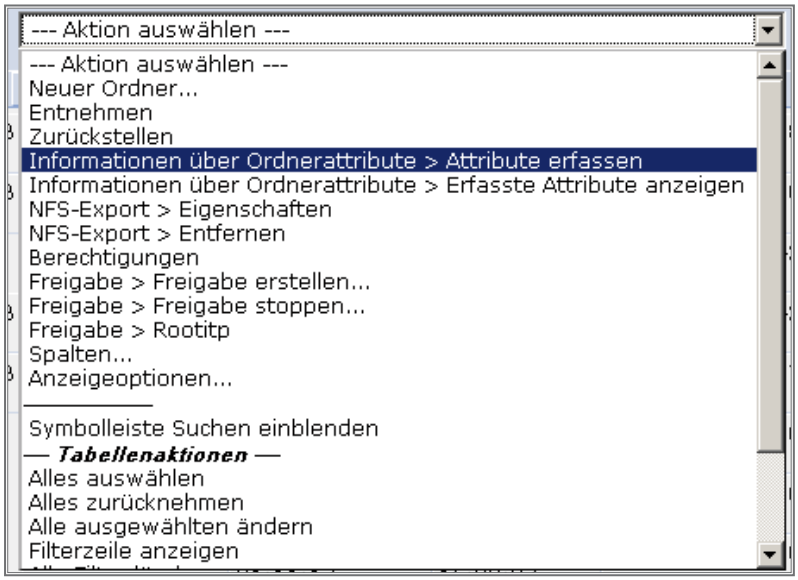
Der Bereich Dateisysteme im Director dient der Verwaltung des Integrated File Systems (IFS) im i/OS. Das Integrated File System ist dem Verzeichnissystem der PC- und UNIX-Welt nicht unähnlich. Durch eine hierarchische Struktur werden Daten logisch angeordnet, so dass – eine vernünftige Struktur vorausgesetzt – auf sie optimal zugegriffen werden kann.

Auch Bibliotheken sind, betrachtet aus der Sicht des Integrated File System, nichts anderes als Verzeichnisse, jedoch mit einer relativ flachen Struktur.



IFS-Struktur

Links sehen Sie eine Liste aller Dateisysteme, für die Sie berechtigt sind. Sie können auf Systemobjekte zugreifen, indem Sie den Verzeichnispfad für das Objekt expandieren.



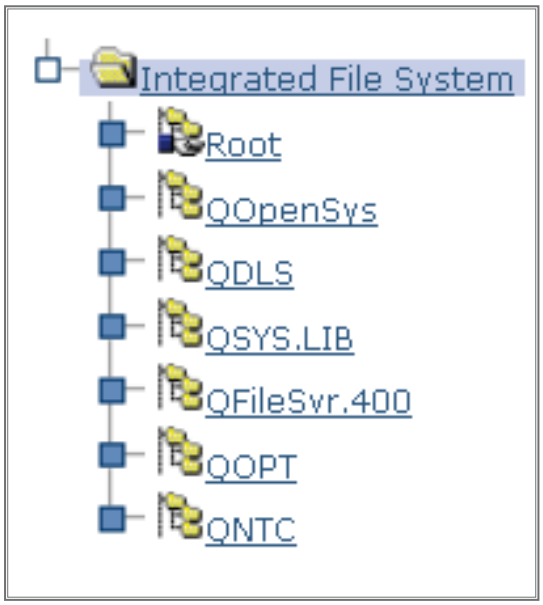
Verarbeitungsoptionen

7.7.4.5
Seite 2

Mit Hilfe der Combo-Box „Aktion auswählen“ haben Sie eine ganze Reihe von Verarbeitungsmöglichkeiten.

Mit dem Integrated File System arbeiten

In der folgenden Abbildung sehen Sie die hierarchische Struktur des Integrated File System.



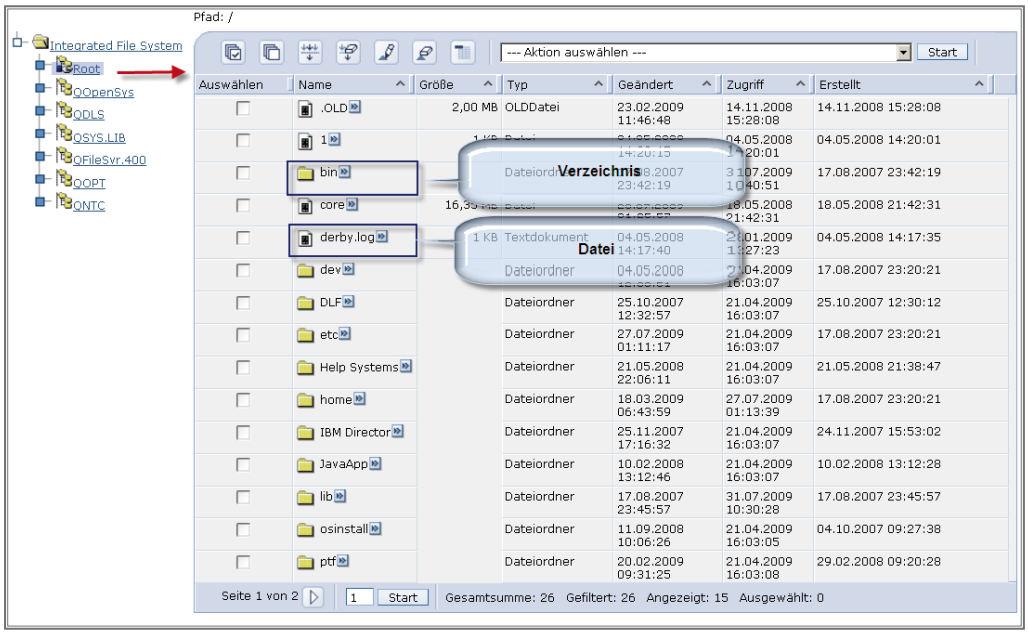
Dateisysteme

7.7.4.5.1

Seite 2

Mit den Dateisystemen arbeiten

Klicke ich auf eines der Dateisysteme, so öffnet sich auf der rechten Seite das angewählte Dateisystem.



Inhalt Dateisysteme

Folgende Dateisysteme werden angezeigt:

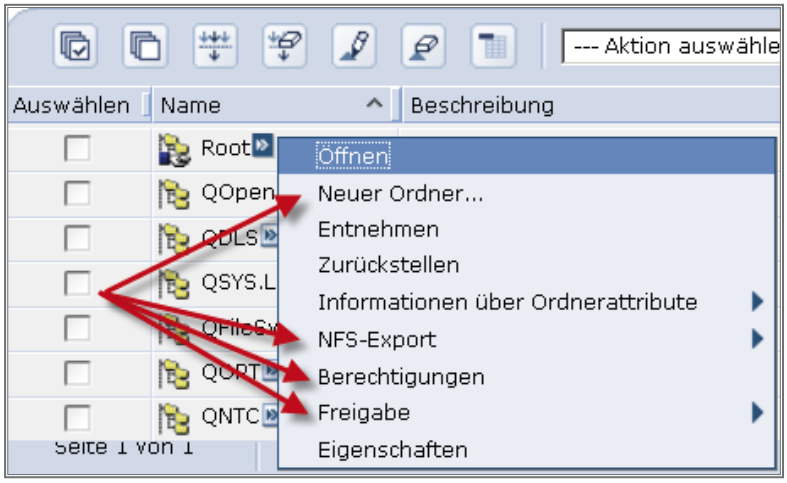
- Root: Root ist das sogenannte Windows- oder OS/2- kompatible Dateisystem. Daten, die hier gespeichert sind, werden im ASCII-Format als sogenannte Streamfiles abgelegt. Namenskonventionen entsprechen denen der Windows-Welt.
- QopenSys: Dieses Dateisystem ist UNIX-kompatibel. Daten, die hier gespeichert sind, werden im ASCII-Format als sogenannte Streamfiles abgelegt. Namenskonventionen entsprechen denen der UNIX-Welt.
- QDLS: QDLS oder das Document Library System beinhaltet die Speicherung von Office Vision-Dokumenten oder von PC-Dateien. Daten, die hier gespeichert sind, werden im EBCDIC-Format als sogenannte Streamfiles abgelegt. Namenskonventionen entsprechen denen der MS-DOS (PC-DOS)-Welt.
- QSYS.LIB: Das Bibliothekssystem des OS/400. Daten, die hier gespeichert sind, werden im EBCDIC-Format als sogenannte Record-Files abgelegt. Namenskonventionen entsprechen denen der OS/400-Welt.
- QfileSvr.400: Dies ist ein sogenanntes Brückenverzeichnis. Erzeugen Sie unterhalb von QfileSvr.400 ein Verzeichnis, das den Host-Namen einer entfernten iSeries trägt, so gelangen Sie in das entsprechende Integrated File System des entfernten Systems.

7.7.4.5.2
Seite 2

– QOPT: Qopt erlaubt den Zugriff auf optische Speichermedien. QOPT erlaubt den Zugriff auf das CD-ROM-Laufwerk des iSeries Servers.
– QNTC: Auch dieses Verzeichnis ist ein Brückenverzeichnis zu anderen Systemen, und zwar zu denen der Microsoft-Windows-Welt. Erzeugen Sie unterhalb von QNTC ein Verzeichnis mit dem Computernamen eines im Netzwerk befindlichen Windows-Systems, so erhalten Sie hierüber einen Zugriff auf die auf diesem System konfigurierten Freigaben. Voraussetzung dafür, dass diese Brückenfunktion ausgeführt werden kann, ist eine lauffähige Konfiguration des iSeries-Netservers.

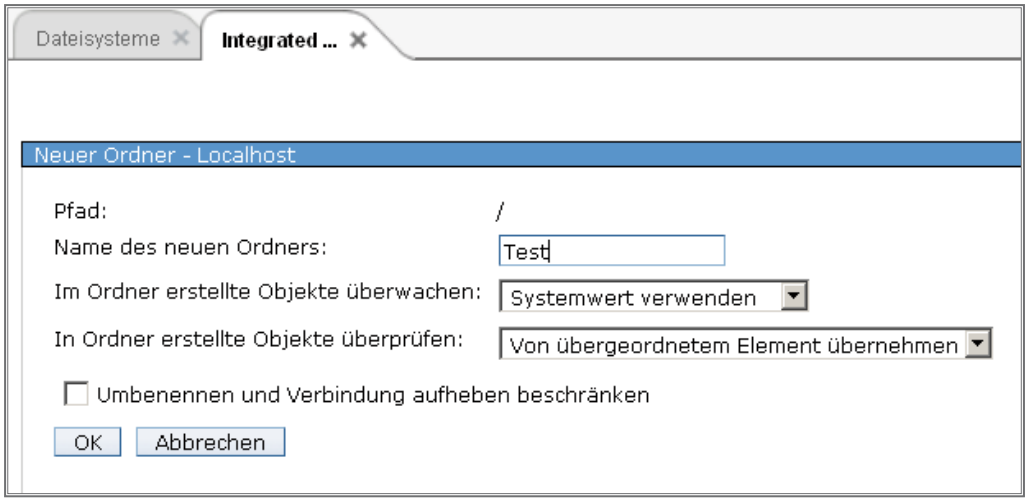
Sie erkennen an der gewählten Symbolik, dass Verzeichnisse und Dateien angezeigt werden.

Ich fordere das Kontextmenü für das Dateisystem „Root“ an und sehe folgende Bearbeitungsmöglichkeiten:



Kontextmenü – Neuer Ordner

Wählen Sie die Funktion „Neuer Ordner“, so öffnet sich folgender Dialog.



Neuer Ordner

In das Eingabefeld tragen Sie den Namen des neuen Ordners ein. In diesem Fall wird der Ordner „Test“ unterhalb des Root-Verzeichnisses erstellt. Beabsichtigen Sie den Ordner „Test“ unterhalb von /Home zu erstellen, so müssen Sie in den oben gezeigten Dialog den kompletten Pfad eintragen: /home/test.

Im Parameter „In Ordner erstellte Objekte überprüfen“ stellen Sie ein, ob die *PUBLIC-Rechte aus dem übergeordneten Verzeichnis an das untergeordnete, neue Verzeichnis vererbt werden sollen.

Mittels des Parameters „Umbenennen…“ können Sie verhindern, dass der neue Ordner umbenannt bzw. gelöscht werden kann.

Berechtigungen: Integrated File System-Berechtigungen vergeben

Freigabe: Arbeiten Sie mit dem Netserver, so kann für das Dateisystem eine Verzeichnisfreigabe erfolgen, die als Netzwerklaufwerk Ihrem Windows-Desktop zugeordnet wird.

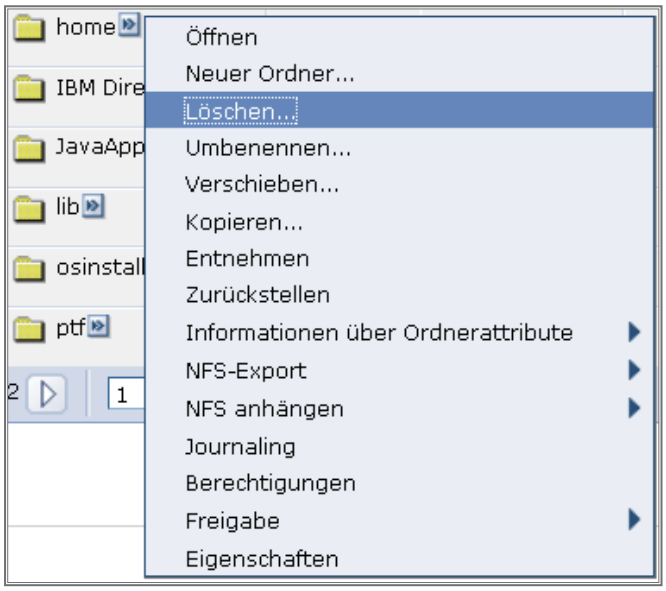
NFS-Export: Arbeiten Sie mit UNIX-Servern, so können Sie das gewählte Dateisystem in die UNIX-Umgebung exportieren.

7.7.4.5.2

Seite 4

Mit den Ordnern arbeiten

Markieren Sie ein Verzeichnis innerhalb eines Dateisystems, so sehen Sie folgendes Kontextmenü.



Kontextmenü für Verzeichnis

Dieses verzeichnisbezogene Kontextmenü enthält die meisten der Verarbeitungsoptionen, die auch für Dateisysteme gelten. Deshalb werde ich jetzt die neu hinzugekommenen Funktionen besprechen.

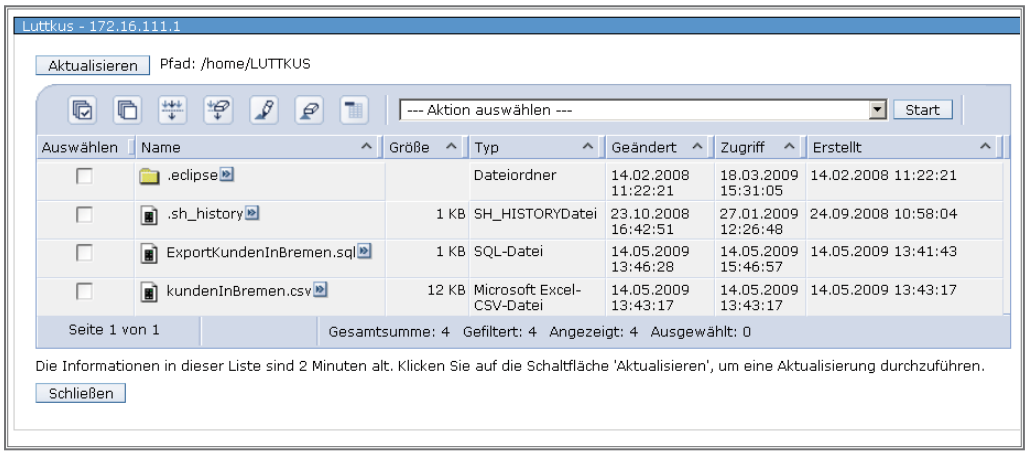
- Löschen: Ein Verzeichnis löschen. Vor dem eigentlichen Löschen erfolgt eine Sicherheitsabfrage.
- Umbenennen: Ein bestehendes Verzeichnis umbenennen.
- Kopieren: Windows Clipboard-Funktion – Verzeichnis in das Windows-Clipboard kopieren. Das Verzeichnis bleibt im System erhalten.
- Journaling: Zugriffe auf das Verzeichnis protokollieren.

7.7.4.5.3

Seite 2

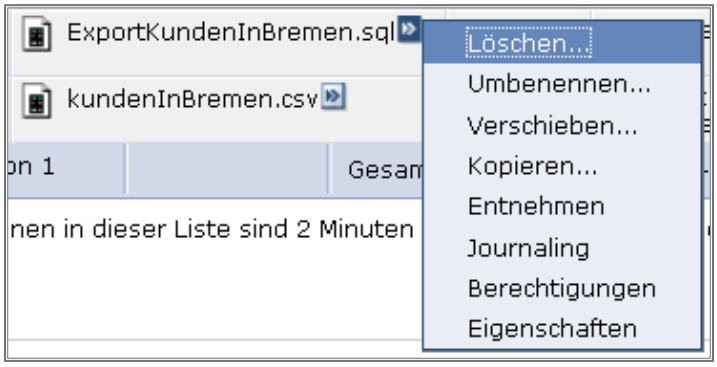
Mit den Objekten in Ordnern arbeiten

Markieren Sie ein bestimmtes Verzeichnis in „Root“ und wählen die Option „Öffnen“ im Kontextmenü, so wird der Inhalt eines Verzeichnisses angezeigt.



Objekte bearbeiten

Im Folgenden sehen Sie das Kontextmenü für ein einzelnes Objekt.



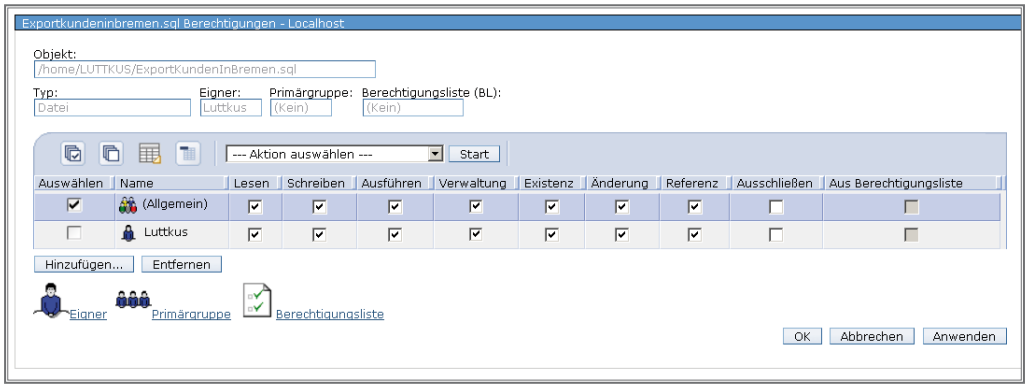
Kontextmenü für einzelnes Objekt

Die meisten dieser Menüeinträge sprechen für sich und bedürfen daher keiner Erklärung.

Um im Integrated File System mit Berechtigungsvergaben zu arbeiten, werden in den Kontextmenüs der diversen Hierarchiestufen entsprechende Auswahlen zur Verfügung gestellt. Auch im Kontextmenü für einzelne Objekte finden Sie den Eintrag „Berechtigung“.

7.7.4.5.4
Seite 2

Ich habe diese Auswahl getroffen und der nachfolgende Dialog öffnet sich.



Berechtigung

Dieser Dialog teilt sich in mehrere Bereiche auf. Der obere Bereich besteht aus der Anzeige folgender Inhalte:

– Objekt

– Typ

– Eigner

– Primärgruppe

– Berechtigungsliste

Der mittlere Bereich dient der Arbeit mit Benutzern und deren Rechten am Objekt. Hier werden Lese-, Schreib und Ausführungsrechte sowie die sogenannten Objektverwaltungsrechte gewährt bzw. verweigert.

Durch „Hinzufügen“ werden neue Benutzer, denen Objektzugriffsrechte gewährt werden sollen, interaktiv aus der Benutzerverwaltung übernommen. „Entfernen“ löscht einen Benutzer aus der Liste der berechtigten Anwender.

Im unteren Bereich finden Sie eine Reihe von Links, mit denen Sie u. a. die Eignerschaft eines Objekts ändern. Auch hier greifen Sie wieder direkt auf die Daten der Benutzerverwaltung zu. Mit dem Link Berechtigungsliste wählen Sie aus den im System gespeicherten Berechtigungslisten diejenige aus, durch die Ihr Objekt geschützt werden soll.

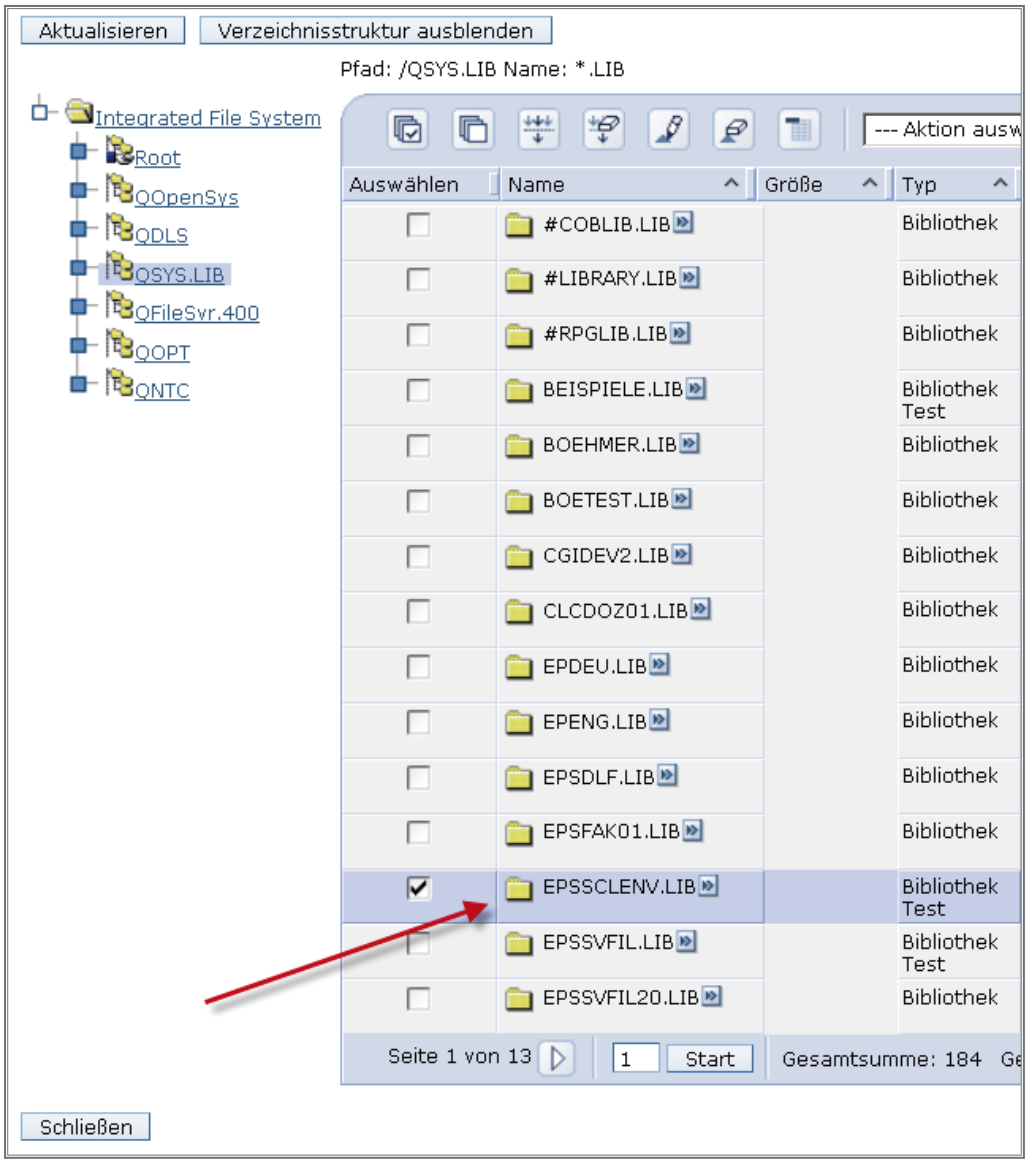
Mit QSYS.LIB arbeiten

Das Dateisystem QSYS.LIB stellt die Bibliotheken des iSeries-Servers dar. Hier gelten die Namenskonventionen der eigentlichen OS/400-Umgebung: 10-stellige Namen, ausschließlich Großschreibung etc.

Die für die OS/400-Umgebung typischen Objektarten werden hier als Namenserweiterungen gesehen.

Die Bibliothek BEISPIELE heißt hier BEISPIELE.LIB, Das Programm CHGALLOWN heißt CHGALLOWN.PGM, der Bildschirm DSP01 heißt DSP01.DEVD, die Ausgabewarteschlange PRT01 heißt PRT01.OUTQ usw.

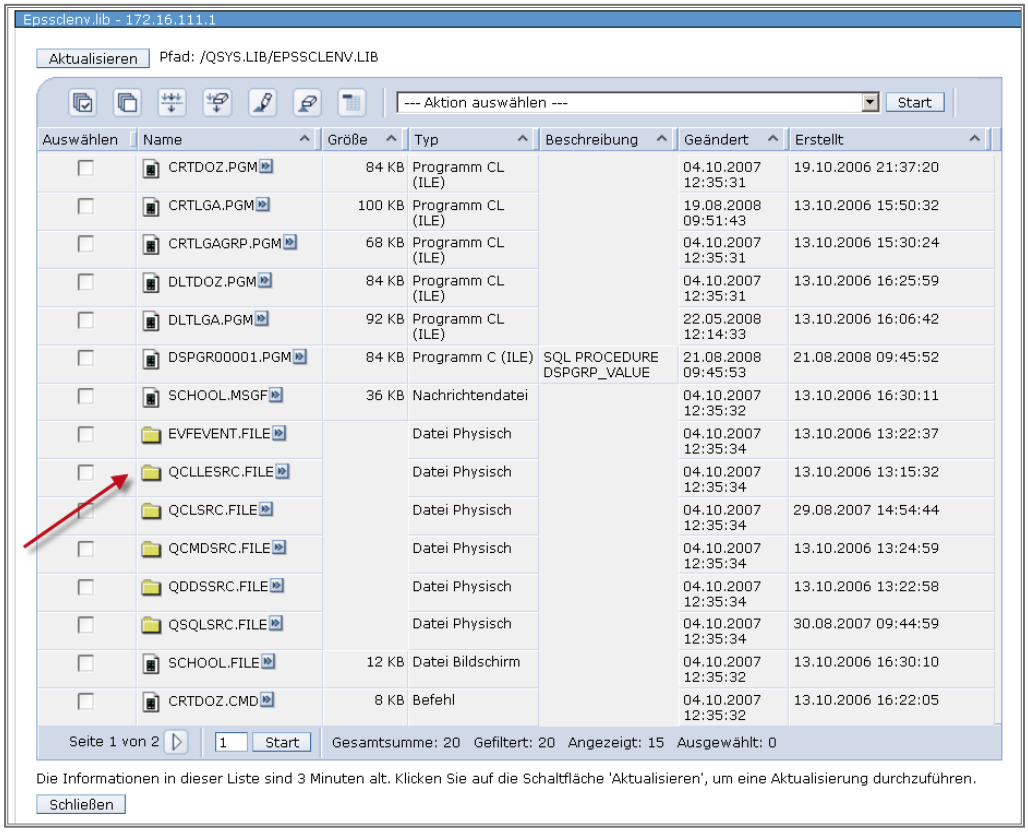
Mit Hilfe der Funktion „Dateisysteme“ werfe ich einen Blick in das Verzeichnis QSYS.LIB (Bibliothek!). Innerhalb des Verzeichnisses QSYS.LIB gibt es ein Verzeichnis EPSSCLENV.LIB (Bibliothek!).



Dateisystem QSYS.LIB

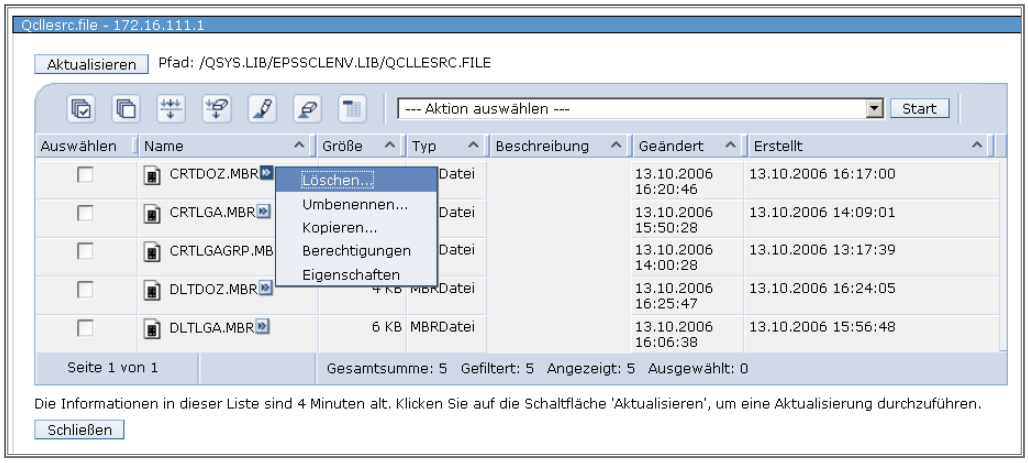
7.7.4.5.5
Seite 2

Im Verzeichnis EPSSCLENV.LIB wird ein Verzeichnis QCLLESRC.FILE (Datei!) verwaltet.



Quellendatei QCLLESRC

Innerhalb des Verzeichnisses QCLSRC.FILE wird ein Objekt CRTDOZ.MBR (Teildatei!) gespeichert. Mit diesem Objekt beabsichtige ich zu arbeiten.



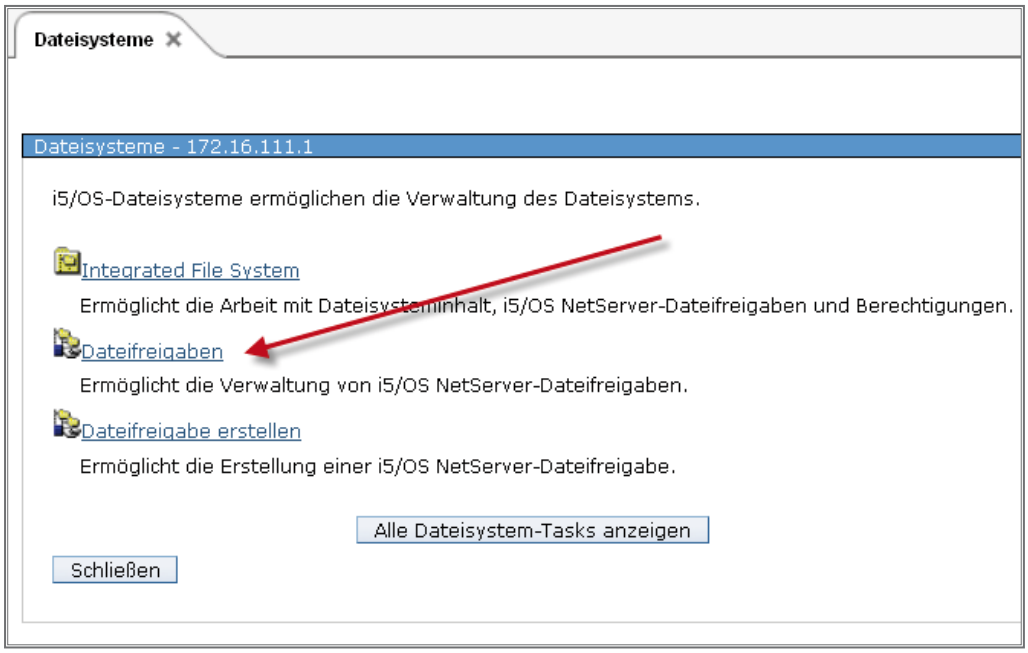
Teildatei SNDDTAQ.MBR

Für die Teildatei fordere ich das Kontextmenü an. Die einzelnen Menüpunkte sind in den vorherigen Abschnitten hinlänglich erklärt worden, so dass ich hier keine weiteren Erklärungen anfüge.

Mit Dateifreigaben arbeiten

Weitere interessante Funktionen in „Dateisysteme“ finden Sie unter „Dateifreigaben“. Hier arbeiten Sie mit den Objekten und Funktionen des iSeries Netservers. Der Netserver ist seit OS/400 V4R2 eine Komponente, die die NETBIOS-Schnittstelle im TCP/IP-Protokollstapel aktiviert und damit das System i Microsoft-Netzwerk-kompatibel macht.

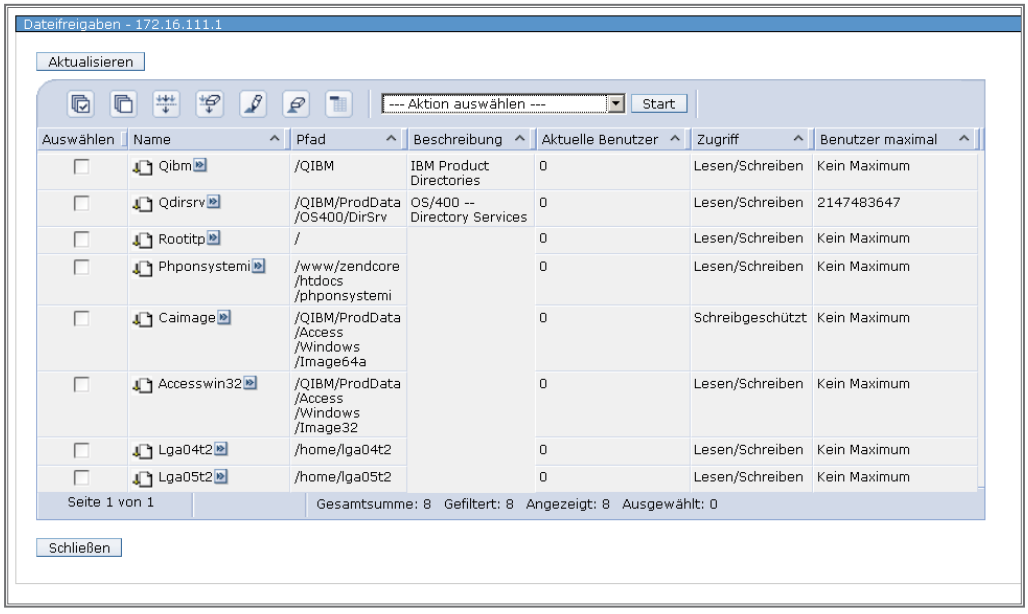
Aus dem Hauptmenü für „Dateisysteme ➜ Dateifreigaben“ heraus öffnen Sie den Netserver Administrator.



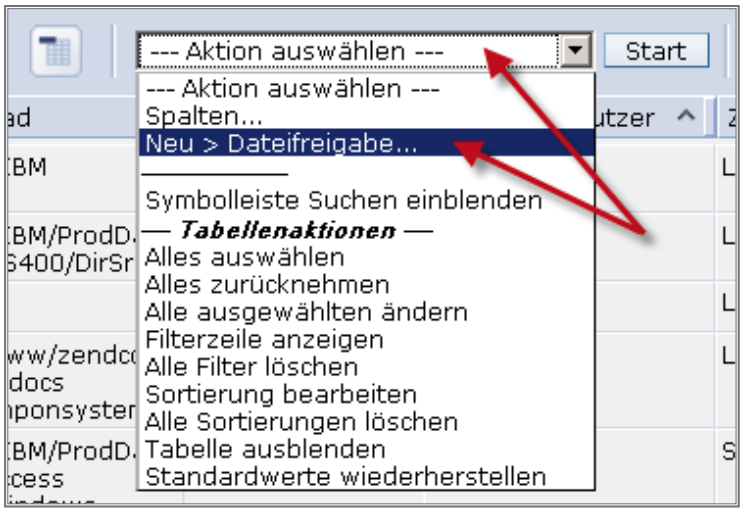
Dateifreigaben

7.7.4.5.6
Seite 2

Folgen Sie dem Link, so erhalten Sie eine Liste aller im Root-Bereich definierten Freigaben.

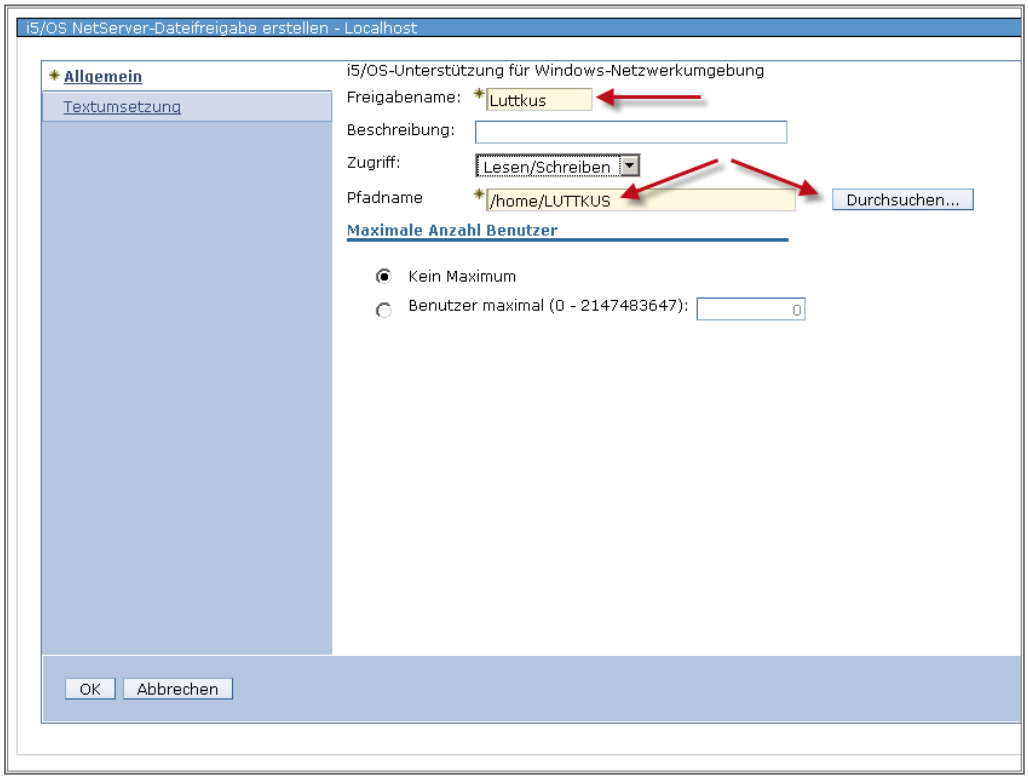


Dateifreigaben verwalten



Neue Dateifreigaben erstellen 1

Um eine neue Freigabe zu erstellen, öffnen Sie die Combo-Box „Aktion auswählen“ und markieren die Option „Neu ➜ Dateifreigabe“ und klicken auf „Start“.



Neue Dateifreigaben erstellen 2

Zugriff: Folgende Zugriffsrechte können vergeben werden:

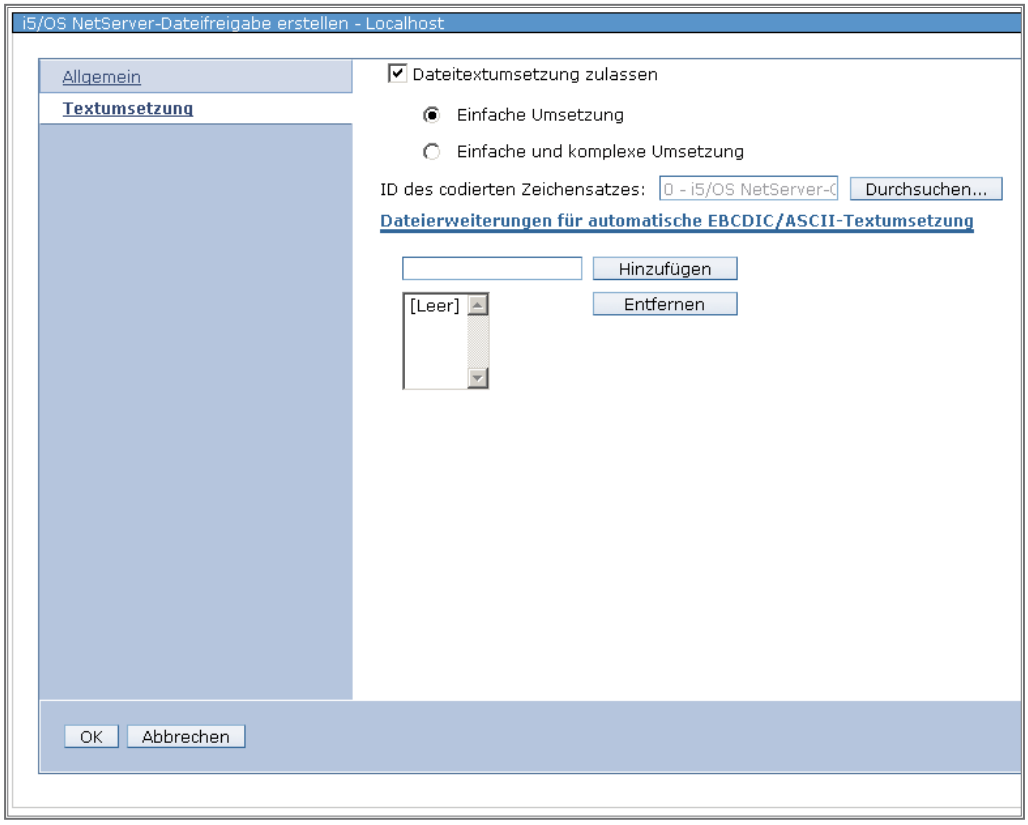
– Lesezugriff: Nur lesender Zugriff über diese Freigabe möglich

– Schreib-/Lesezugriff: Lesender und schreibender Zugriff über diese Freigabe möglich

Im Register „Textumsetzung“ wird die Übersetzung von EBCDIC-Inhalten nach ASCII gesteuert.

7.7.4.5.6
Seite 4

Dabei steht im Vordergrund, welche Objekte automatisch an der Zeichenumsetzung teilhaben.



Textumsetzung

P Parameter: Dateierweiterungen für automatische EBCDIC-ASCII-Textumsetzung

Gibt an, bei welchen Dateierweiterungen der Dateiinhalt automatisch von EBCDIC in ASCII umgesetzt wird. Um eine Dateierweiterung zur Liste hinzuzufügen, geben Sie die Dateierweiterung im Bearbeitungsfeld ein und klicken auf „Hinzufügen“. Um eine Dateierweiterung aus der Liste zu entfernen, wählen Sie die entsprechende Dateierweiterung aus und klicken auf „Entfernen“. Bei der Textumsetzung werden auch bestimmte Werte für Platzhalterzeichen unterstützt. Hierbei handelt es sich um folgende Werte:

* Gibt an, dass der Server alle Dateien umsetzt.

. Gibt an, dass der Server alle Dateien umsetzt, die keine Erweiterung haben.

TXT Gibt an, dass der Server alle Dateien umsetzt, die die Erweiterung .TXT haben. Beispiel: hans.TXT.

.TXT Gibt an, dass der Server alle Dateien umsetzt, die die Erweiterung .TXT haben. Beispiel: CONNY1.TXT.

T?T Gibt an, dass der Server alle Dateien mit einer Erweiterung umsetzt, in der das Platzhalterzeichen (?) durch ein beliebiges Zeichen ersetzt ist. Beispiel: linda.T1T oder hans.TXT.

T*T Gibt an, dass der Server alle Dateien mit einer Erweiterung umsetzt, in der das Platzhalterzeichen (*) durch eine beliebige Anzahl von Zeichen ersetzt ist. Beispiel: hans.T1234T oder hans.TXT oder hans.TEST.

7.7.4.5.6

Seite 6

Der System i Navigator

In einigen vorangegangenen Kapiteln habe ich bereits auf die Administrationsmöglichkeiten hingewiesen, die Ihnen der System i Navigator bietet. In diesem Kapitel, das auf Wunsch vieler Leser entstanden ist, möchte ich System i Navigator-Funktionen beschreiben, die bei der Administration von System i Access- und Systemfunktionen hilfreich sein können.

Bevor der System i Navigator zum Einsatz kommen kann, sind folgende Arbeitsschritte zu erledigen:

- Softwaremedium bereitlegen
- System i vorbereiten
- TCP/IP-Netzwerkkonfiguration der System i überprüfen
- ggf. Hardwarekonfiguration der System i modifizieren
- System i Navigator installieren
- sich mit der Bedienung des System i Navigator vertraut machen

Ich werde Ihnen bei all diesen Schritten helfen, so dass Ihr Einstieg in die Anwendung des System i Navigator ohne Probleme verlaufen sollte.

7.8
Seite 2

Softwaremedium bereitlegen

Der System i Navigator wird im Rahmen des System i Access bzw. des System i Access für Windows ausgeliefert. Um den System i Navigator komplett zu installieren oder Teile nachzuinstallieren, sollten Sie über die System i Access- bzw. System i Access für Windows-Installationsmedien verfügen. Sollten Sie weder über eine Installations-CD noch über einen Zugriff auf eine auf einem File-Server abgelegte Kopie von System i Access verfügen, so lässt sich ggf. die Installation auch direkt aus dem System i-Verzeichnispfad /QIBM/ProdData/Access/Windows/cwblaunch.exe installieren.

Falls Sie über keine Zugriffsmöglichkeiten auf Installationsmedien bzw. Zugriffsrechte auf Installationspfade verfügen, so sollten Sie Ihren Netzwerkadministrator diesbezüglich ansprechen.

Weiterhin ist es wichtig – sofern Sie unter Windows XP/Vista/7 arbeiten –, dass Sie für den Arbeitsplatz, an dem System i Navigator installiert werden soll, über die lokalen Administratorrechte verfügen.

Legen Sie außerdem das aktuelle Service Pack für System i Access bereit. Nach der Installation des System i Navigator bzw. nach der Installation neuer System i Navigator-Teile muss das aktuelle Service Pack erneut eingespielt werden.

Achtung!

Das ist auch dann notwendig, wenn Sie das Service Pack bereits für System i Access installiert haben. Bedenken Sie, ohne angelegtes Service Pack wird Ihr System i Navigator Fehler anzeigen, mit denen Sie sich eigentlich nicht herumzuschlagen bräuchten.

7.8.1
Seite 2

System i vorbereiten

Auch Ihre System i will auf den Einsatz vorbereitet werden. Wichtig ist an dieser Stelle, dass die System i Access- bzw. System i Access für Windows-Schnittstelle fehlerfrei zu betreiben ist.

Stellen Sie sicher, dass durch das Einspielen aktueller CUM-PTFs folgendes Lizenzprogramm aktualisiert wird:

- 5722 – XW1

Darüber hinaus sollten Sie Sorge dafür tragen, dass folgende aktuelle Gruppen-PTFs eingespielt werden:

- DB2 for IBM i

Sollten diese Gruppen-PTFs nicht eingespielt worden sein, so ist der Betrieb der Datenbankfunktionen im System i Navigator fehlerbehaftet.

Des Weiteren sollten die JAVA-Gruppen-PTFs und die HIPER-Gruppen-PTFs eingespielt werden.

- GROUP HIPER
- JAVA

Die aktuellen PTF-Stände erfahren Sie auf der Homepage des IBM Technical Support: http://www-912.ibm.com/s_dir/sline003.nsf/GroupPTFs?OpenView&view=GroupPTFs

Zögern Sie also nicht, die erforderlichen PTFs auf Ihrer System i einzuspielen. Sowohl das ganze System als auch der System i Navigator profitieren insgesamt von dieser Maßnahme. Denken Sie daran: Ohne diese PTFs werden Sie Probleme bekommen, mit denen Sie sich eigentlich nicht beschäftigen müssten.

7.8.2
Seite 2

TCP/IP-Netzwerkkonfiguration der System i überprüfen

Bestandteil des System i Navigator ist eine Funktion, die als Management Central bezeichnet wird. Damit diese Funktion vom System i Navigator aus wahrgenommen werden kann, muss im TCP/IP-Netzwerk des System i der sogenannte Management Central Server gestartet worden sein. Ich habe es in vielen Fällen erlebt, dass dieser Server, nachdem er manuell gestartet wurde, sofort seinen Dienst beendet hat. Nach Durchsicht der Joblogs konnte dann in Erfahrung gebracht werden, dass TCP/IP-Adressierungsprobleme zum Absturz dieses Server-Dienstes führten.

Damit Sie nicht Gleiches erleben, sollten folgende Arbeitsschritte realisiert werden:
Tragen Sie den Hostnamen der System i, auf dem der Management Central Server läuft, in die lokale Hosttabelle der System i ein. Alternativ ist es auch möglich, hierfür den DNS – Domain Name Service – im Netzwerk zu nutzen. Schreiben Sie sich diesen Namen genau auf. Falls Sie über ein gutes Gedächtnis verfügen, können Sie sich diesen Namen auch merken ...

Danach tragen Sie den in der Hosttabelle verwendeten Namen in die TCP/IP-Domäneninformationen Ihrer System i ein.

Die Hosttabelleneinträge finden Sie im Menü CFGTCP in der Auswahl 10, die Domäneninformationen ebenfalls im Menü CFGTCP, jedoch unter der Auswahl 12.

Achtung!

Arbeiten Sie anstatt mit einer Hosttabelle mit einem DNS, so muss der Hostname Ihrer System i dort eingetragen werden, sofern er nicht bereits dort existiert, und der DNS muss in die Domäneninformationen des System i (Menü CFGTCP, Auswahl 12) eingetragen werden.

7.8.3
Seite 2

Gegebenenfalls Hardwarekonfiguration der System i modifizieren

Da der System i Navigator zusammen mit den neuen System i-Servern direkt mit der Hardware arbeiten kann, ist es ggf. notwendig, die Hardwarekonfiguration Ihrer System i zu ändern.

Welche Arbeiten sind notwendig? Der System i Navigator wird später über einen Netzwerkadapter der System i auf Hardwarekomponenten zugreifen. Damit das möglich ist, muss einer der Netzwerkkarten eine sogenannte Service-Netzwerkschnittstelle zugeordnet werden. Dies ist eine besondere IP-Adresse, die nur für den Zugriff auf die System i-Hardware eingerichtet wird. Werksseitig ist diese Konfiguration nicht vorhanden.

Die Einrichtung dieser Service-Netzwerkschnittstelle erfolgt aus dem System i DST (Dedicated Service Tools) heraus. DST-Arbeiten werden über das Systemterminal (DSP01) erledigt.

Um in die DST-Umgebung zu gelangen, müssen Sie das QSECOFR-Profil des DST kennen. Jedes System i-System wird u. a. mit dem Service-Benutzerprofil QSECOFR, Kennwort QSECOFR, ausgeliefert.

Achtung!

Bei dem Service-Benutzerprofil QSECOFR handelt es sich nicht um das Benutzerprofil QSECOFR, mit dem wir uns an jedem Bildschirm der System i zum Erledigen von Systempflegearbeiten anmelden können. Im DST existiert eine eigene „Familie“ von Benutzern! Haben Sie das Kennwort des Benutzers QSECOFR geändert, so ist damit das Kennwort des Service-Benutzers QSECOFR im DST noch lange nicht geändert.

Nochmals Achtung!

Kennwörter im DST haben eine Länge von 128 Byte und unterscheiden Groß- und Kleinschreibung. Darüber hinaus sind Sonderzeichen möglich!

Wie gelangen Sie in die DST-Umgebung? Ich gehe davon aus, dass Ihr System i-System bereits läuft.

- Schalten Sie DSP01 (Systemterminal) ein.
- Stellen Sie die Startart der System i an der Systemeinheit auf manuell um.
- Stellen Sie danach an der Systemeinheit den Code 21 ein.
- Jetzt erscheint die DST-Anmeldung.

7.8.4
Seite 2

Geben Sie nun den Benutzernamen QSECOFR an sowie das entsprechende Kennwort. Denken Sie daran, dass das Standardkennwort groß geschrieben wird!

Tipp: Sollte der DST-Benutzer QSECOFR aufgrund von fehlerhaften Kennworteingaben deaktiviert worden sein, so ist er relativ leicht wieder zu reanimieren. Melden Sie sich an DSP01 der System i als der Benutzer QSECOFR an. Mittels des IBM i-Befehls CHGDSTPWD (Change DST Password) setzen Sie dann dem DST-Benutzer QSECOFR sein Kennwort auf den Auslieferungszustand des Systems zurück. Dazu ist die Kenntnis des alten Kennworts nicht notwendig. Danach melden Sie sich in den SST (System Service Tools) mit STRSST an.
Name: QSECOFR
Kennwort: QSECOFR

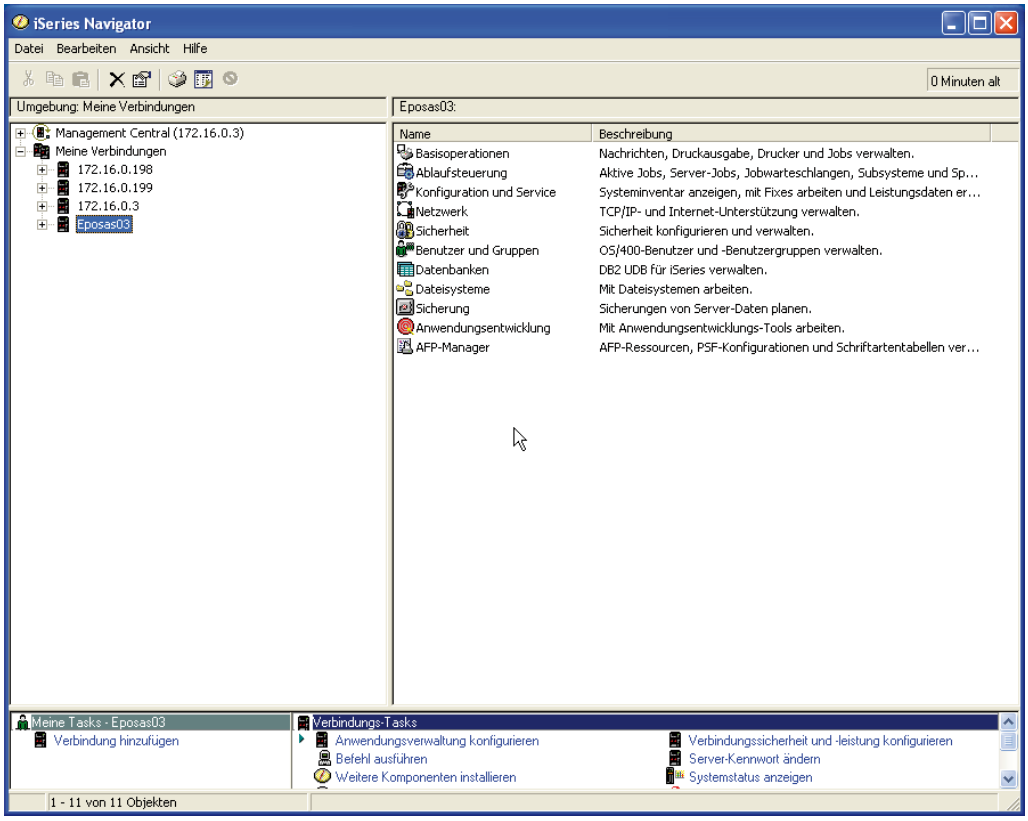
Haben Sie die DST-Anmeldung überstanden, so gehen Sie in den Menüpunkt „DST-Environment“ und ordnen dort einer Netzwerkschnittstelle Ihrer System i eine TCP/IP-Adresse zu.

Ist kein Zugriff auf die System i-Hardware aus dem System i Navigator heraus erwünscht, so ist dieser Abschnitt für Sie nicht maßgeblich.

Sich mit der Bedienung des System i Navigator vertraut machen

Was bei jeder anderen Software gilt, gilt auch für den System i Navigator. Bevor die eigentliche Arbeit beginnt, müssen Sie sich zuerst einmal mit den Prinzipien der Bedienung vertraut machen. Positiv wirkt sich aus, dass die IBM-Programmierer des System i Navigator sich im Großen und Ganzen an die Prinzipien der Windows-Schnittstelle gehalten haben, so dass die Einarbeitung in den System i Navigator relativ leicht fällt – sofern Sie mit der Bedienung von Windows-Programmen vertraut sind.

Hier sehen Sie den System i Navigator-Hauptdialog.

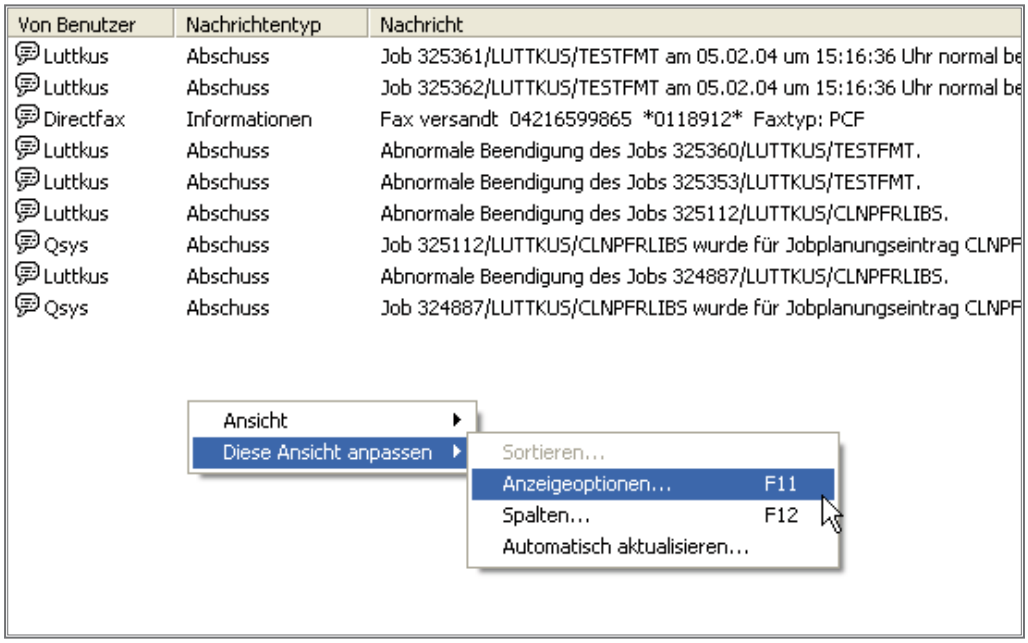


System i Navigator-Menü

Wie in Windows-Programmen üblich, finden Sie unterhalb der Titelleiste ein Menü. Wichtig ist, dass dieses Menü kontextsensitiv ist. Das heißt, in Abhängigkeit von Ihrer gerade ausgeführten Tätigkeit werden in den sogenannten Submenüs von „Datei“, „Bearbeiten“, „Ansicht“ verschiedene Auswahlen zu sehen sein.

7.8.5
Seite 2

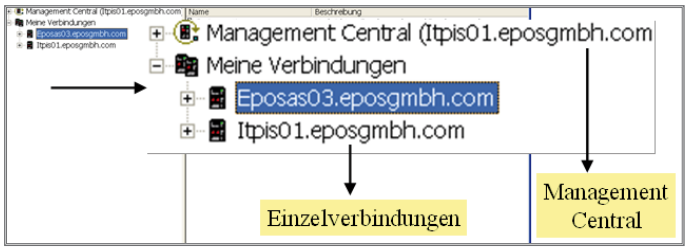
Beabsichtigen Sie die Anzeige der darzustellenden Daten zu beeinflussen, so klicken Sie einfach mit der rechten Maustaste in den Hintergrund des rechten Bereichs des System i Navigator und folgende Kontextmenüs öffnen sich:



Ansichten anpassen

Hier werden je nach Kontext die Funktionen „Anzeigeoptionen“, „Spalten“, „Sortieren“ und „Automatisch aktualisieren“ aktiviert bzw. deaktiviert dargestellt. Darüber hinaus können Sie diese Optionen auch aus dem Menü „Ansicht“ erreichen.

Auf der linken Seite finden Sie die Funktionen, die mit Hilfe des System i Navigator ausgeführt werden.



Funktionsbaum

Diese Funktionen sind in Form einer Verzeichnisstruktur hierarchisch angeordnet. Hierbei unterscheiden wir zwei grundsätzliche Funktionsbereiche: Einzelverbindungen und Management Central.

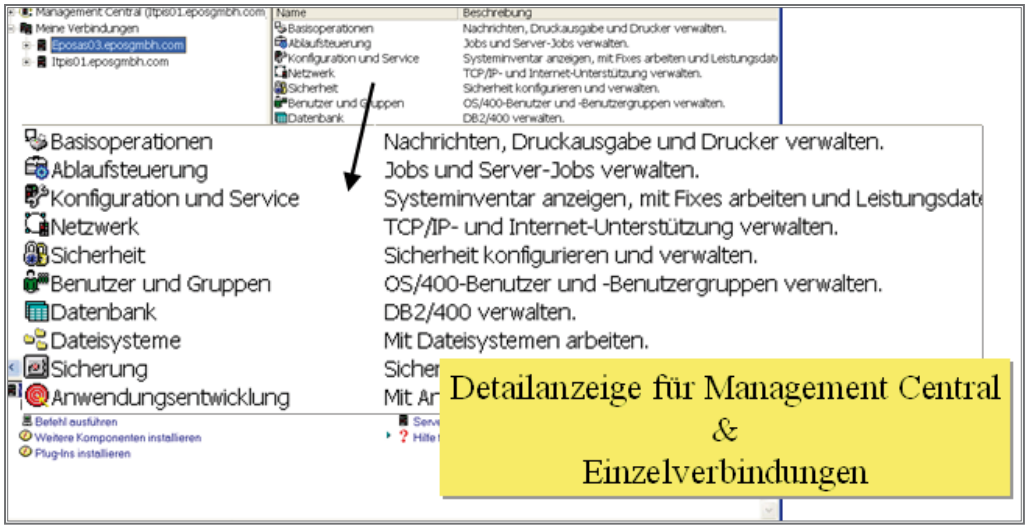
Einzelverbindungen: Für jedes System i-System bzw. jede Partition in Ihrem Netzwerk sollte eine Einzelverbindung definiert werden, so dass Sie mit dem jeweiligen System arbeiten können. Soll eines Ihrer Systeme nicht mit System i Navigator bearbeitet werden, so muss natürlich auch keine Einzelverbindung zu diesem System aufgebaut werden. Wie der Begriff verrät, ermöglicht ein Eintrag in den Bereich „Einzelverbindungen" administrative Arbeiten an einem einzelnen System.

Management Central: Im Management Central werden System i-Systeme, zu denen Sie Einzelverbindungen unterhalten, als administratives Netzwerk angesehen, so dass Sie hier quasi gleichzeitig mit mehreren Systemen und/oder Partitionen arbeiten können.

Sie werden jetzt möglicherweise denken, dass für Sie das Management Central nicht in Frage kommt, weil Sie nur über ein System i-System verfügen. Damit ließe sich dann auch der zusätzliche Netzwerkkonfigurationsaufwand vernachlässigen. So einfach ist es jedoch nicht! Auch Einzelverbindungen greifen auf Funktionen des Management Central zu, so dass die damit einhergehenden Konfigurationsaufgaben in jedem Fall realisiert werden müssen.

Fazit: Die im System i Navigator dargestellte Zweiteilung von Funktionen in die des Management Central und in die der Einzelverbindungen existiert so eigentlich nicht. Im Grunde kann man hier nur von einer groben Funktionsgruppierung sprechen.

Auf der rechten Seite des System i Navigator sehen Sie den Bereich der Detailanzeigen.

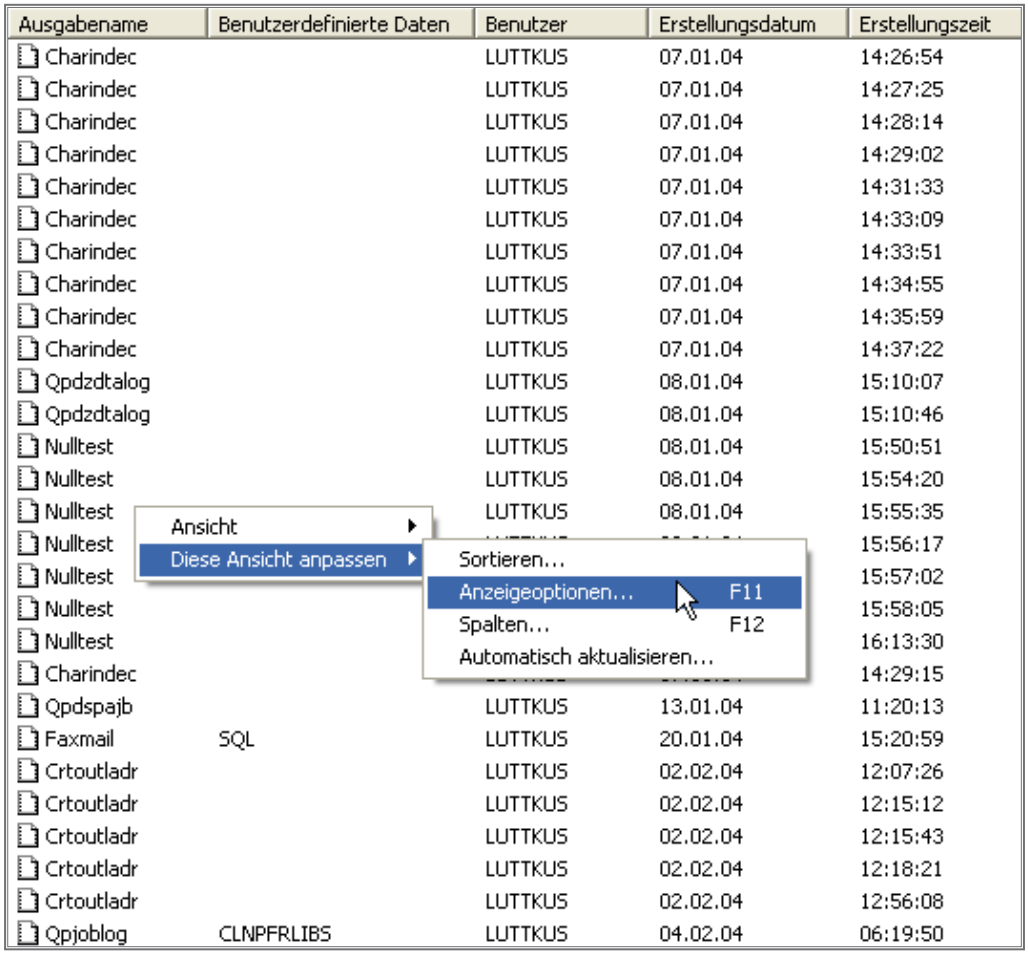


Detailanzeigen

Detailanzeigen verändern sich, je nachdem welche Funktion Sie im Funktionsbaum des System i Navigator angewählt haben.

7.8.5
Seite 4

Um die Aussagefähigkeit von Detailanzeigen steuern zu können, steht uns das Kontextmenü „Diese Ansicht anpassen…“ zur Verfügung.

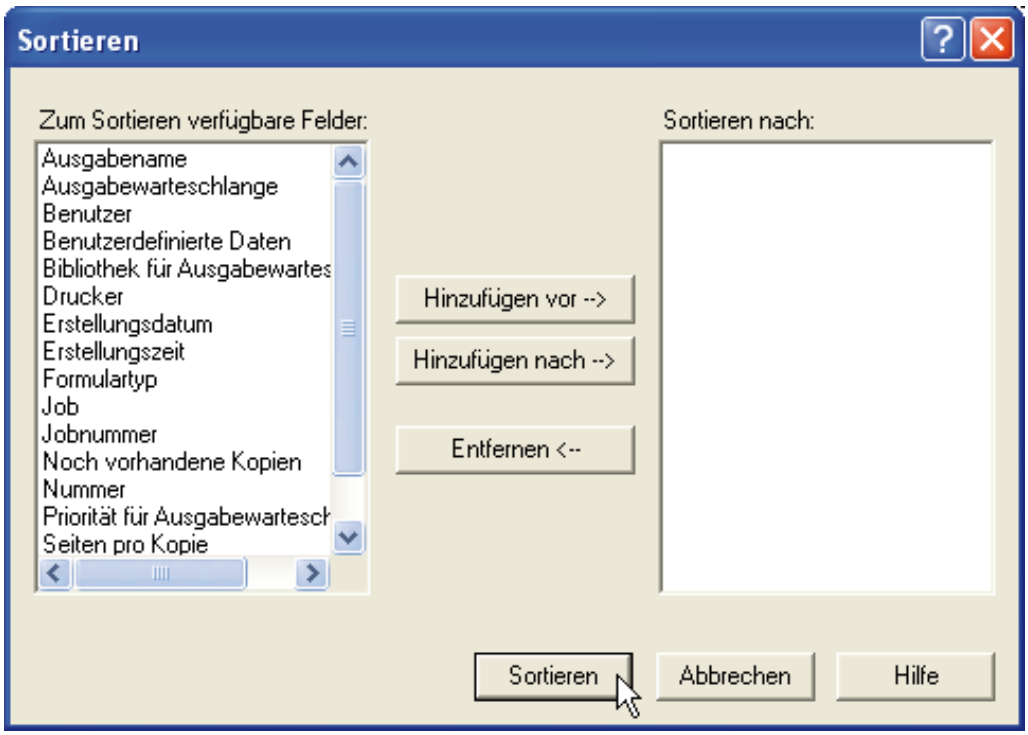


Menü-Optionen

Um die Detailanzeige zu modifizieren, gibt es drei wichtige Auswahlpunkte:

- Sortieren...
- Anzeigeoptionen...
- Spalten...

Die Funktion des Dialogs „Sortieren“ erklärt sich selbst.



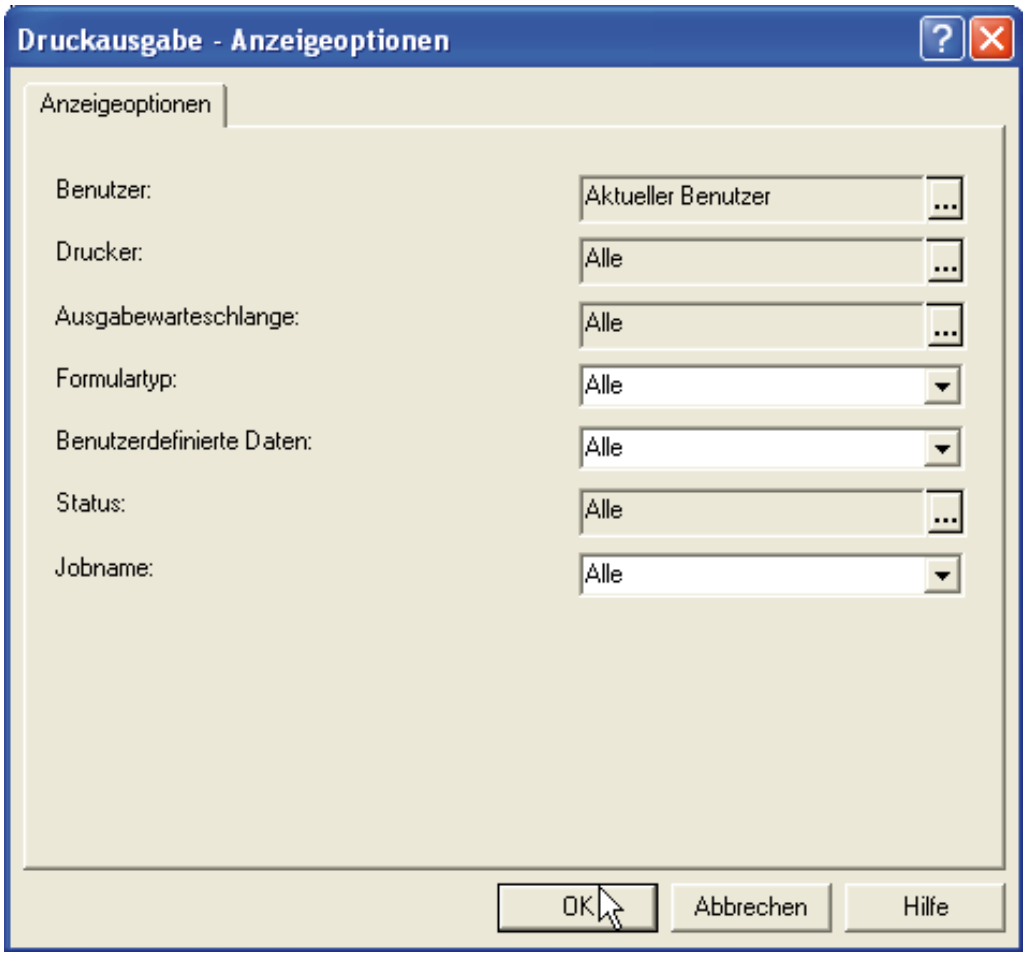
Menü-Optionen: „Sortieren“

Dieser Dialog strukturiert sich wie folgt: Auf der linken Seite finden Sie die für eine System i Navigator-Funktion möglichen Sortierkriterien – zum Sortieren verfügbare Felder. Mithilfe der Hinzufügen-Knöpfe werden gewünschte Sortierbegriffe auf die rechte Seite des Dialogs – „Sortieren nach“ – transferiert.

Komplexe Sortierbegriffe erstellen Sie, indem Sie mehrere Sortierbegriffe auswählen. Logische Reihenfolgen bilden Sie, indem Sie je nach gewünschter Hierarchie Sortierbegriffe durch Drücken von „Hinzufügen vor“ und „Hinzufügen nach“ auf die rechte Seite des Dialogs bringen.

Einzelne Sortierbegriffe entfernen Sie aus der rechten Dialogseite, indem Sie den zu entfernenden Begriff markieren und dann auf den „Entfernen“-Knopf drücken.

Die Funktion des Dialogs „Anzeigeoptionen“ lässt sich am einfachsten mit der F4-Bedienhilfe-Taste einen 5250-Dialogs der System i vergleichen.



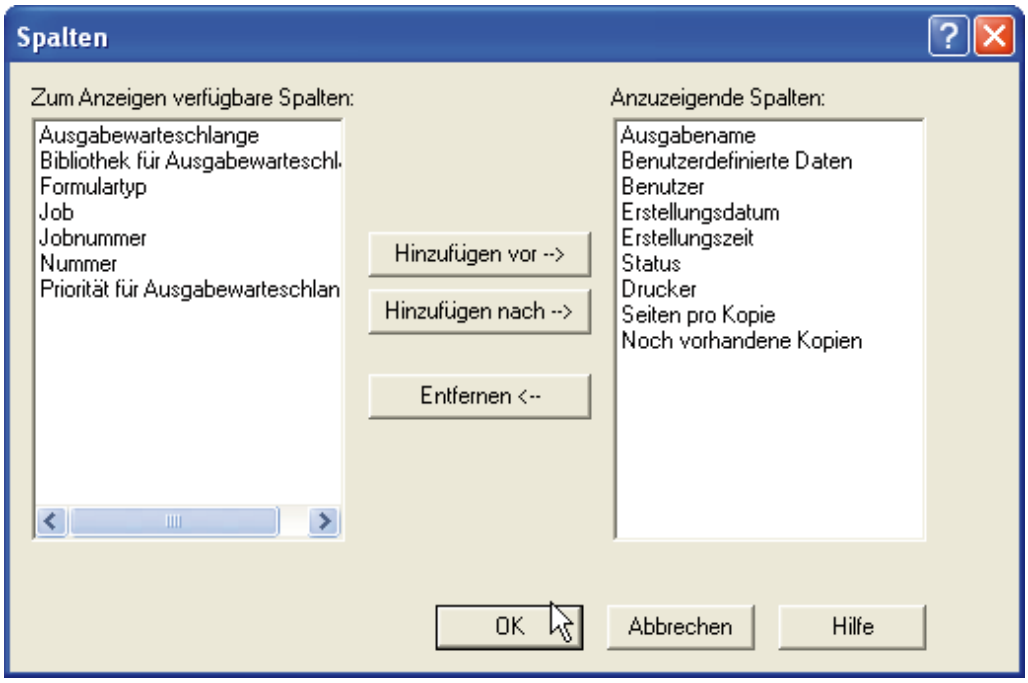
Menü „Anzeigeoptionen“

Haben Sie eine bestimmte Funktion im Funktionsbaum des System i Navigator angewählt und lassen sich dann den Dialog „Anzeigeoptionen“ für diese Funktion anzeigen, so erhalten Sie eine Liste von Parametern. Welche Parameter angezeigt werden, hängt von der jeweils gewählten Funktion im Funktionsbaum ab. In der Abbildung oben sehen Sie die Parameterliste für die Funktion „Druckausgabe“.

Nur zum Spaß gebe ich in einem traditionellen Bildschirm der System i einmal den Befehl WRKSPLF ein und prompte ihn mit F4.

Die Übereinstimmungen sind leicht zu erkennen.

In der Menüauswahl „Spalten“ steuern Sie den dargestellten Inhalt in der Detailanzeige einer System i Navigator-Funktion.



Menü „Spalten“

Grundsätzlich funktioniert dieser Dialog genauso wie der Dialog „Sortieren“. Auf der linken Seite finden Sie die für eine System i Navigator-Funktion möglichen anzuzeigenden Spalten – „Zum Anzeigen verfügbare Spalten“. Mithilfe der Hinzufügen-Knöpfe werden gewünschte Spalten auf die rechte Seite des Dialogs – „Anzuzeigende Spalten“ – transferiert.

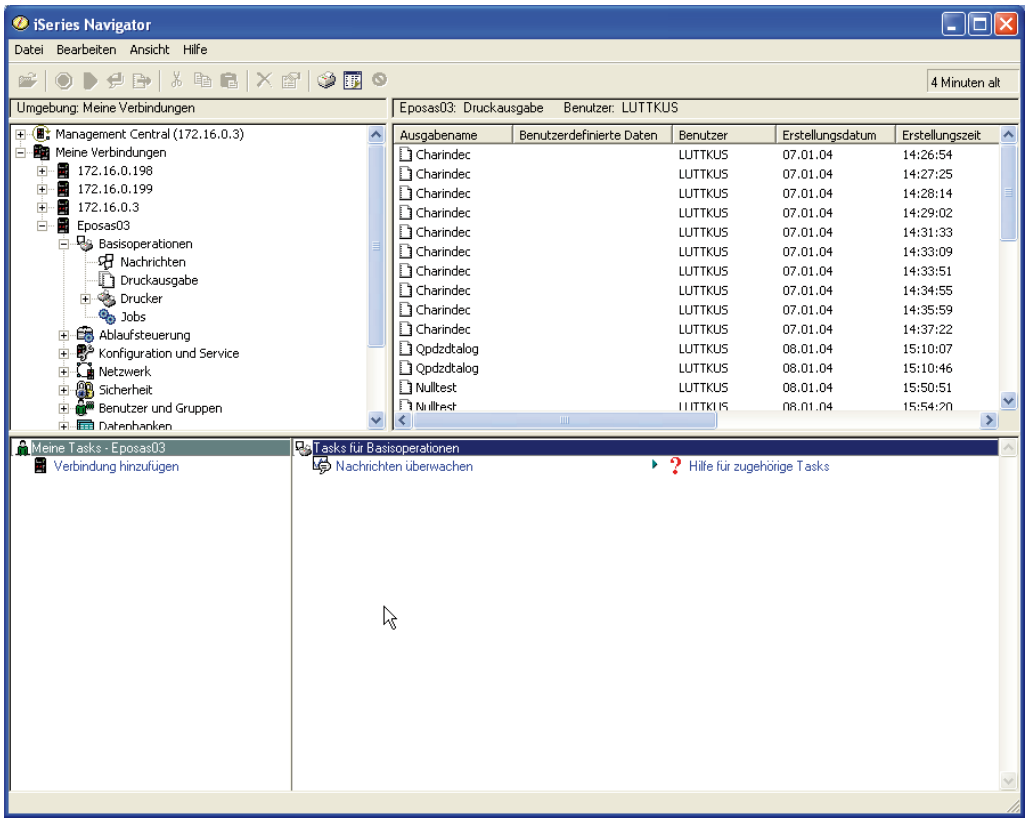
Sie sehen in dem dargestellten Dialog, dass einige Spalten auf die rechte Seite übernommen wurden.

Sollen zusätzliche Spalten in die Anzeige aufgenommen werden, so werden sie links markiert und je nach gewünschter Reihenfolge wird rechts ein Bezugsfeld markiert und dann mit „Hinzufügen vor“ bzw. „Hinzufügen nach“ auf die rechte Seite gebracht.

Entfernen Sie nicht anzuzeigende Spalten durch Markieren eines Spaltennamens auf der rechten Seite und Drücken des Entfernen-Knopfs.

7.8.5
Seite 8

Ein weiterer Hauptbestandteil des System i Navigator ist das sogenannte Taskpad.



Taskpad

Das Taskpad ermöglicht den direkten Zugriff auf oft angewählte Funktionen. Wie für Menüs gilt auch für das Taskpad, dass sich sein Inhalt in Abhängigkeit vom im System i Navigator gewählten Funktionsbereich ändert.

Im Taskpad finden Sie sogenannte Wizards (Zauberer), die u. a. Konfigurationsarbeiten durch zwangsgeführte Dialoge und die Bereitstellung von sinnvollen Default-Werten vereinfachen sollen.

Neben den Wizards finden Sie im Taskpad eine ganze Reihe von hierarchisch geordneten Hilfetexten.

Ein weiteres Bedienungselement des System i Navigator stellen die Kontextmenüs dar, in die wir durch Drücken der rechten Maustaste gelangen.

Kontextmenü

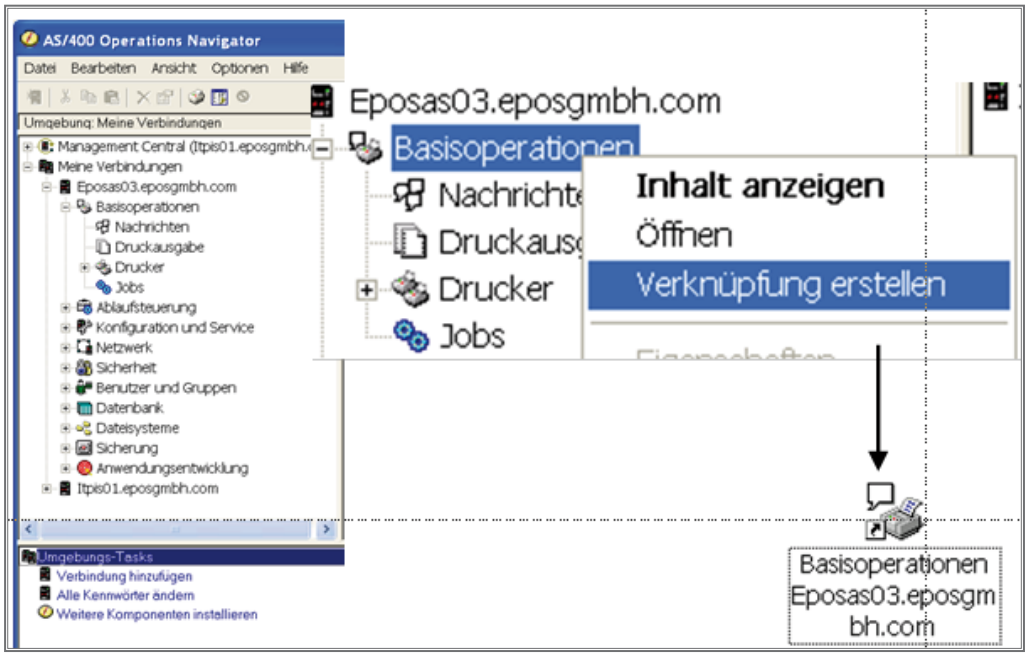
Sie sehen hier, dass ich im Funktionsbaum auf der linken Seite des System i Navigator eine Verbindung markiert habe, durch Drücken der rechten Maustaste wird dann ein Kontextmenü aufgerufen, das mir die entsprechenden Funktionen für das Arbeiten mit Verbindungen anzeigt.

Kontextmenüs existieren für fast alle im Funktionsbaum dargestellten Funktionen, aber auch für die meisten Detailanzeigen.

Zum Schluss möchte ich Ihnen noch eine nützliche kleine Funktion der Kontextmenüs vorstellen. In fast jedem funktionsbezogenen Kontextmenü – nicht in denen der Detailanzeigen! – finden Sie die Menüauswahl „Verknüpfung erstellen“. Diese Funktion basiert auf folgenden Grundgedanken:

Als Administrator benötige ich nicht ständig den kompletten Funktionsumfang des System i Navigator. Meistens wird sich meine tägliche Arbeit auf relativ wenige, jedoch oft benötigte Funktionen reduzieren. Warum sollte ich also immer den gesamten System i Navigator mit seinem gesamten Funktionsumfang öffnen?

7.8.5
Seite 10



Verknüpfung erstellen

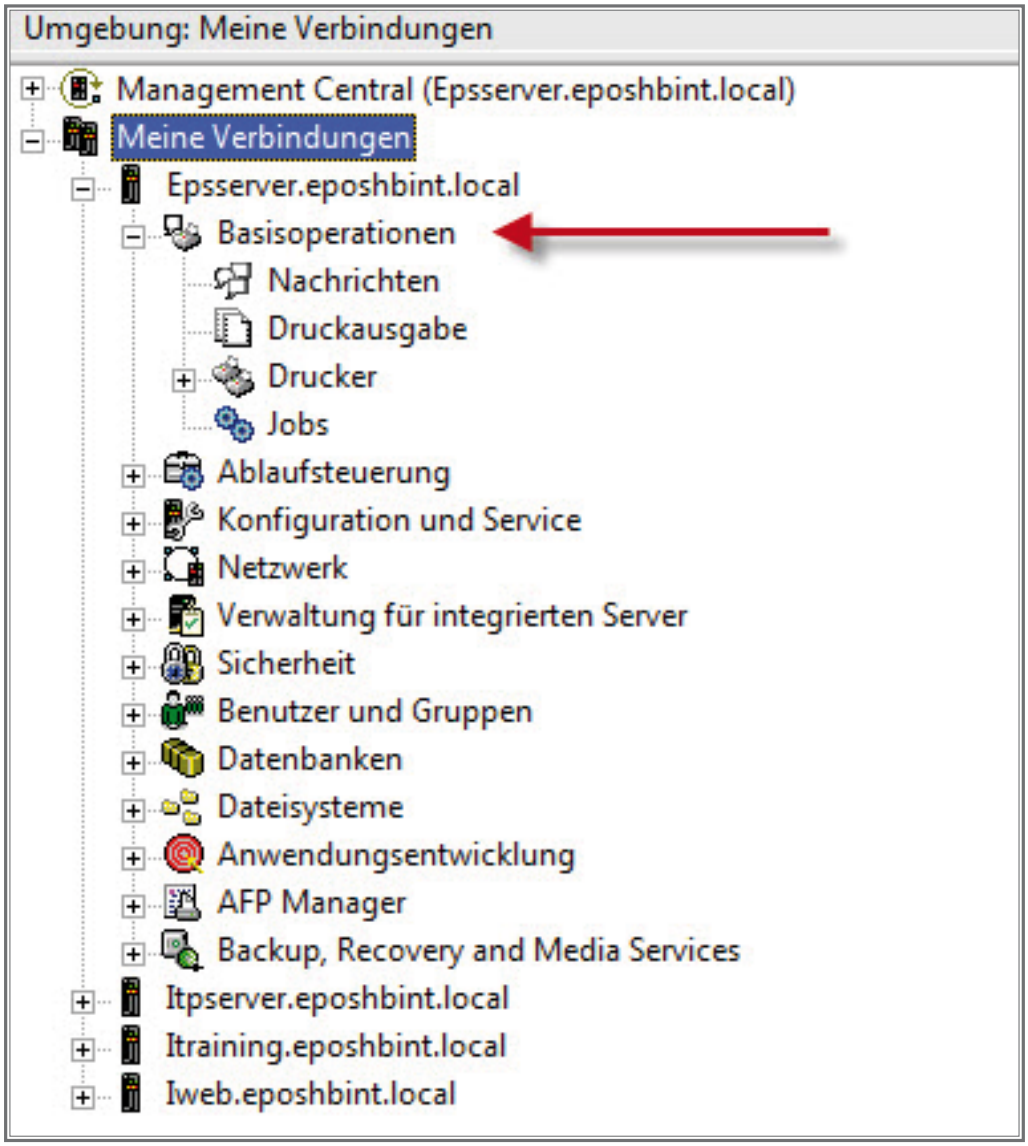
Ist es nicht besser, gezielt auf eine Funktion zuzugreifen?

An dieser Stelle setzt die Auswahl „Verknüpfung erstellen“ an. Wenn ich also weiß, dass ich oft die Funktion „Basisoperationen“ benötige, so erstelle ich für diese Funktion eine Verknüpfung. Die Auswahl dieser Funktion bewirkt, dass auf dem Windows Desktop ein Symbol abgelegt wird, das mir den Direktzugriff auf die Funktion ermöglicht.

Damit beende ich den Überblick über die Bedienungselemente des System i Navigator. Sie sehen, dass eine ganze Reihe dieser Bedienelemente sich an die Vorgaben der Windows-Schnittstelle halten, so dass die Einarbeitungsphase in den System i Navigator insgesamt als erfreulich kurz anzusehen ist.

Basisoperationen

Der Bereich der Basisoperationen umfasst den bedienernahen Teil des System i Navigator. Hier finden wir Systembedienungsfunktionen, die von vielen Benutzern des System i täglich ausgeführt werden.



Basisoperationen

Diese sogenannten Basisoperationen bestehen aus folgenden Funktionen:

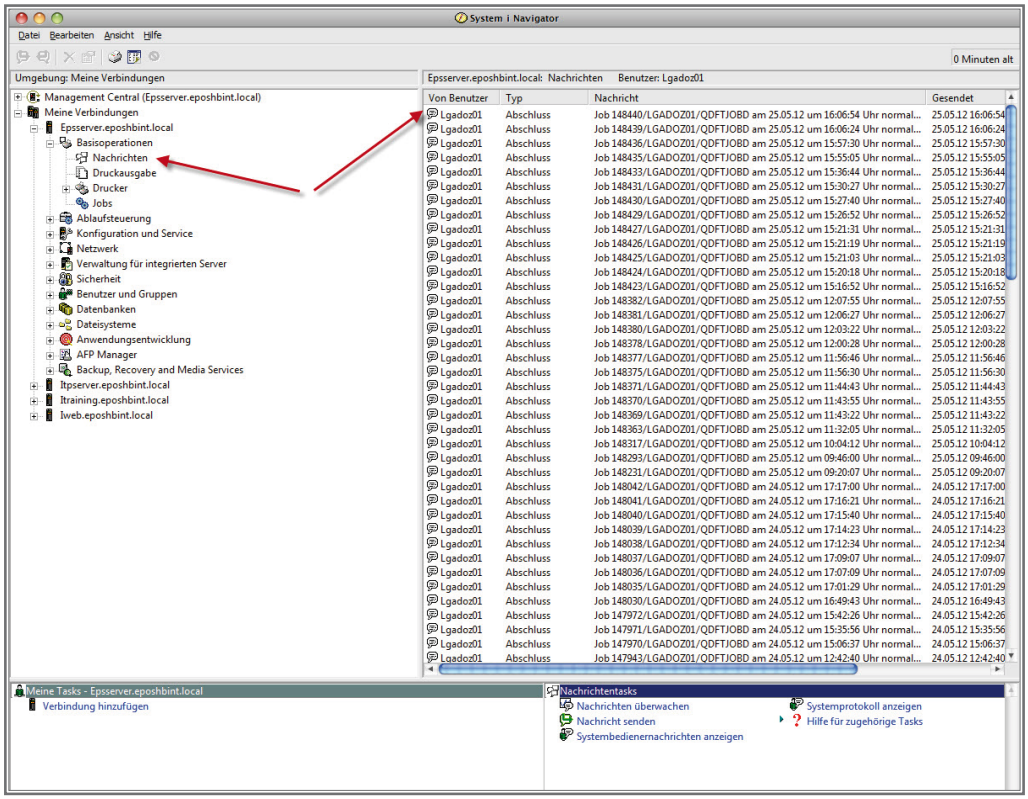
- Nachrichten – Mit Nachrichten arbeiten
- Druckausgabe – Mit Druckausgaben arbeiten
- Drucker – Mit Druckern arbeiten
- Jobs – Mit Benutzerjobs arbeiten

7.8.6
Seite 2

Die in der Folge gezeigten Funktionen können im Prinzip als Alternativen zu den System i CL-Befehlen WRKMSG (Work with Messages), WRKSPL (Work with Spoolfiles) WRKWTR (Work with Writers) und WRKUSRJOB (Work with User Jobs) angesehen werden.

Mit Nachrichten arbeiten

In der folgenden Anzeige des System i Navigator ist der Funktionsbereich „Nachrichten“ markiert.



Nachrichten

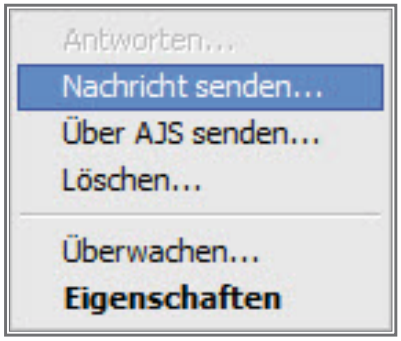
Auf der rechten Seite sehen Sie – nachdem Sie die Nachrichtenfunktion ausgewählt haben – die Nachrichtenanzeigedetails. Für den Benutzer LGADOZ01 ist offenkundig eine Reihe von Nachrichten eingegangen. Die hier gezeigten Nachrichten sind vom Typ „Abschluss“, also Beendigungsnachrichten. Am Nachrichtentext erkennen wir, dass sich die Nachrichten sowohl auf normal als auch auf abnormal beendete Jobs im System i beziehen.

Der hier teilweise sichtbare Nachrichtentext wird als Text der ersten Ebene (first level) bezeichnet. Aber als erfahrener System i-Anwender kennen Sie diese Anzeige ja aus dem Befehl DSPMSG (Display Message).

Ein Doppelklick auf eine der angezeigten Nachrichten würde dann auch den Nachrichtentext der zweite Ebene anzeigen.

7.8.6.1
Seite 2

Öffnen wir mit Hilfe der rechten Maustaste das Kontextmenü für eine Nachricht, so werden wir folgende Abbildung zu sehen bekommen.



Kontextmenü „Nachrichten“

Gehen wir einfach schnell die Menüpunkte durch:

- Antworten:
  Sollte es sich bei der ausgewählten Nachricht um eine Nachricht handeln, die eine Antwort erfordert, so kann sie sofort beantwortet werden.
- Nachricht senden:
  Diese Funktion spricht für sich selbst.
- Über AJS senden:
  Diese Funktion wir nur angezeigt, wenn Sie den IBM Advanced Job Scheduler (AJS) installiert haben. Mit Hilfe des AJS können Sie E-Mails an Systembenutzer versenden.
- Löschen:
  Diese Funktion spricht auch wieder für sich selbst.
- Überwachen:
  Mit Hilfe dieses Eintrags definieren Sie Nachrichtenüberwachungen, so dass Sie Nachrichten an die überwachte Nachrichtenwarteschlage sofort ausgeliefert bekommen.
- Eigenschaften:
  Hier erhalten Sie Einblick in die Nachrichtendetails: Wer hat diese Nachricht geschickt, von welchem Programm/Job kam die Nachricht etc.?

Sie werden ganz richtig vermuten, dass sich hinter dieser Funktion eine Art grafischer SNDMSG (Send Message) -Befehl verbirgt. Aber lassen Sie uns an dieser Stelle ein wenig über die Wirkungsweise der Nachrichtenversendungsbefehle des System i nachdenken.

Im System existieren zwei Nachrichtenversendungsbefehle, die dem Benutzerumfeld zuzuordnen sind:

- SNDMSG (Send Message)
- SNDBRKMSG (Send Breakmessage)

Beide Befehle haben Eigenarten, die ihren Einsatz einschränken können. Der Befehl SNDMSG sendet eine Benutzernachricht an einen Benutzer, wobei es unerheblich ist, an welchem Bildschirm sich der Benutzer angemeldet hat. Die Nachricht erreicht die Nachrichtenwarteschlange des Benutzerprofils auf alle Fälle.

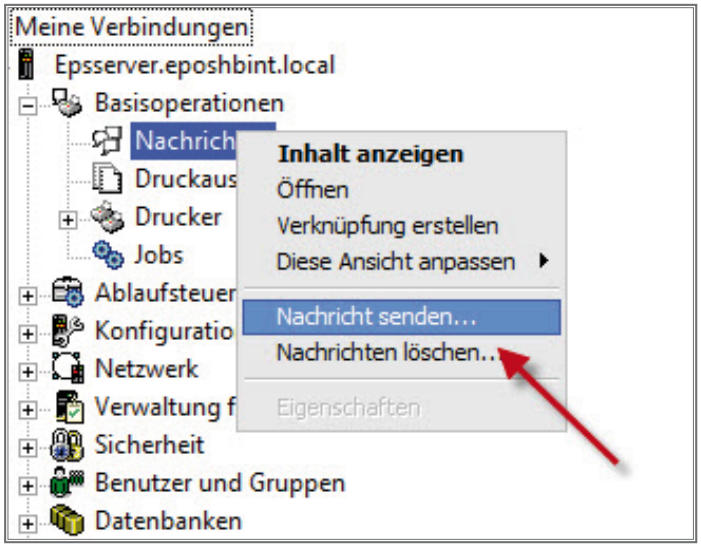
Jedoch hat die Sache einen Haken. Soll der adressierte Benutzer die Nachricht unbedingt lesen und möglicherweise darauf antworten, so hat der Nachrichtenversender dafür keine Garantie, denn wie die Nachricht vom Empfänger wahrgenommen wird, hängt von der Einstellung für die Nachrichtenwarteschlange in dessen Empfängerprofil ab. Wenn für die Nachrichtenwarteschlange des Empfängerprofils nicht der Modus *BREAK aktiviert ist, so führt das Eintreffen der Nachricht bestenfalls zu einem Signalton (Modus *Notify) oder wird schlimmstenfalls überhaupt nicht wahrgenommen (Modus *Hold).

Soll ein Anwender einer System i eine Nachricht unbedingt lesen, so bietet sich hierfür der Systembefehl SNDBRKMSG an. Wie der Name schon sagt, sendet dieser Befehl eine Durchbruchnachricht. Das heißt, dass der Bildschirm des Empfängers auf alle Fälle die Nachricht anzeigt.

Aber auch hier gibt es einen Haken. Als Standard schickt der Befehl SNDBRKMSG seine Nachricht an die Nachrichtenwarteschlange eines Bildschirms – nicht an einen Benutzer. Sende ich also eine Nachricht an einen Bildschirm, so kann es mir passieren, dass der Bildschirm zwar die Nachricht sofort beim Eintreffen anzeigt, ob der adressierte Benutzer aber auch wirklich vor diesem Bildschirm sitzt, dafür habe ich keine Garantie.

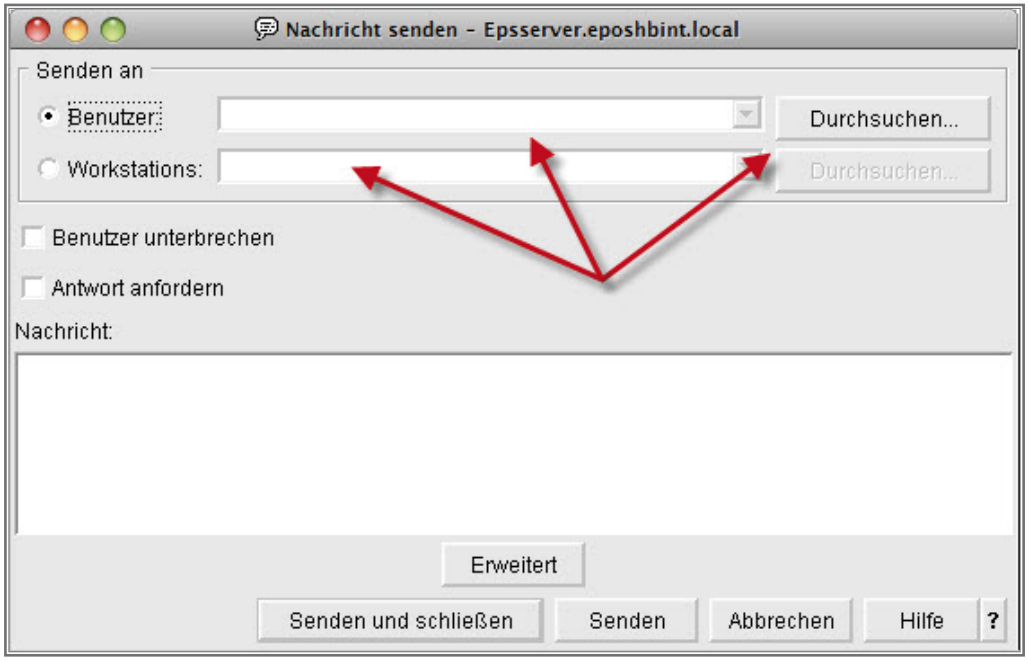
7.8.6.1
Seite 4

An diesem Punkt greift die System i Navigator-Funktion „Nachrichten senden“.



Nachrichten senden

Nachdem Sie die Funktion „Nachrichten senden“ gewählt haben, wird folgenes Fenster angezeigt.



Nachrichten senden

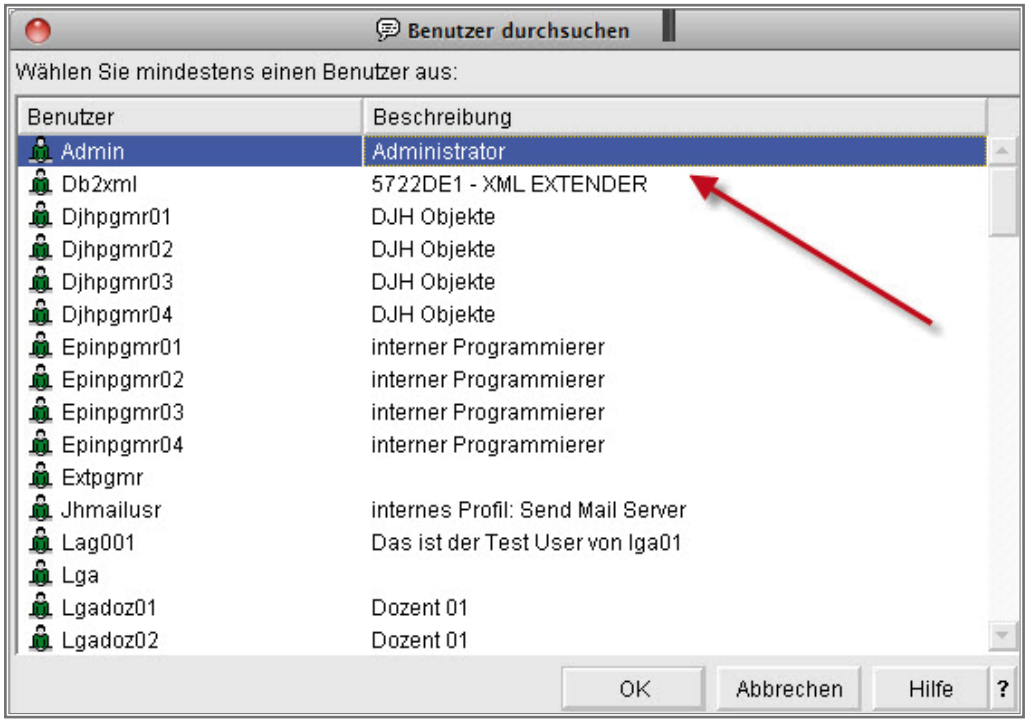
Folgende Parameter steuern diesen Dialog:

Benutzer:
Beabsichtigen Sie die Nachricht an Benutzer zu senden, so können Sie den Namen eines bekannten Benutzerprofils in das Dialogfeld „Benutzer“ direkt eintragen. Darüber hinaus stehen Ihnen über das Pulldown-Menü in diesem Dialogfeld vordefinierte Benutzer bzw. Gruppen zur Auswahl zur Verfügung.

Vordefinierte Benutzer und Gruppen

Wählen Sie an dieser Stelle aus, ob Sie eine Nachricht an alle Benutzer senden wollen oder nur an die derzeit am System angemeldeten Benutzer oder an den Systembediener (QSYSOPR) des System i.

Alternativ ist es auch möglich, den Namen eines System i-Benutzers mit dem „Durchsuchen“-Dialog ausfindig zu machen.

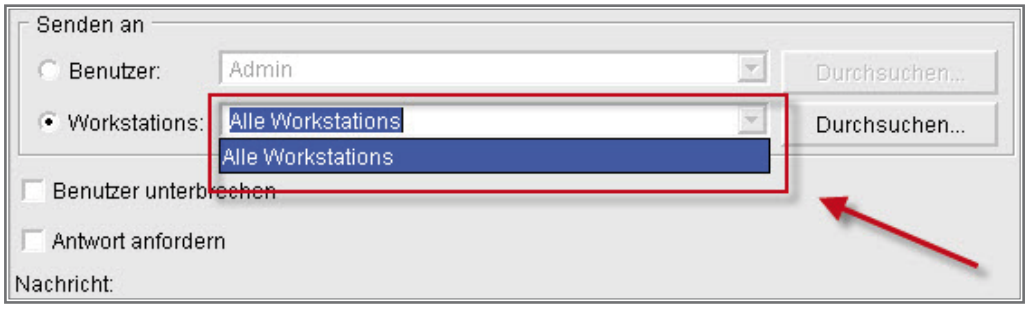


Nach Benutzern der System i suchen

Sie sehen hier den Benutzer ADMIN, dem eine Nachricht gesendet werden soll.

7.8.6.1
Seite 6

Datenstationen:
Es können auch Nachrichten an Datenstations-Nachrichtenwarteschlangen abgesetzt werden.



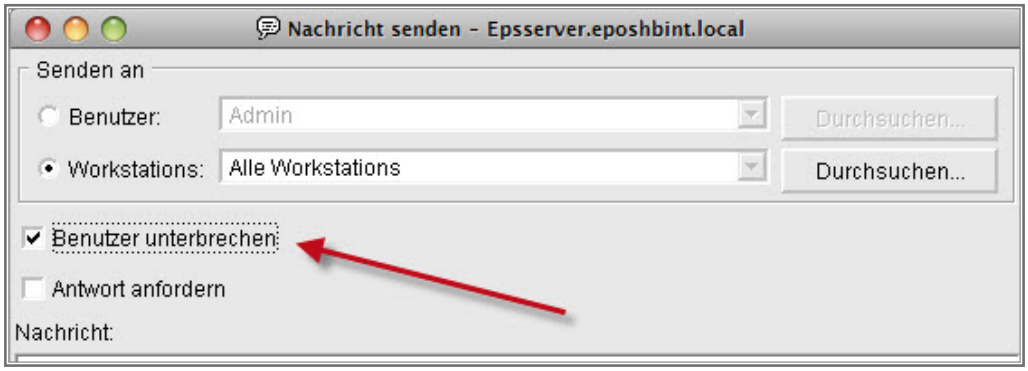
Datenstationsnachrichten

Auch hier ist es möglich, die Nachricht direkt an einen Bildschirm zu senden, sofern der Name der Bildschirmeinheit bekannt ist. Sie haben hier allerdings nur die Möglichkeit, allen Datenstationen (Gruppenadresse) eine Nachricht zu senden.

Achtung!

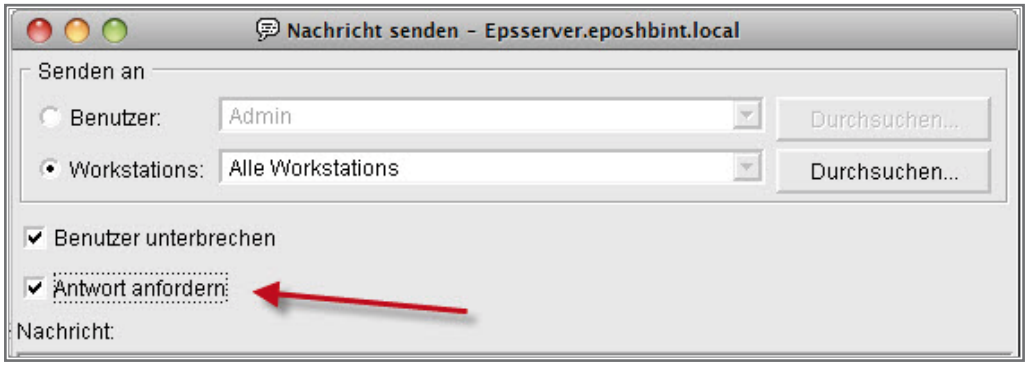
Sowohl der „Durchsuchen"-Dialog für Benutzer als auch der Dialog für Datenstationen haben einen Nachteil. Sie können nicht zwischen aktiven Benutzern/Datenstationen und nicht aktiven Benutzern/Datenstationen unterscheiden. Nehmen wir als Beispiel die Suchfunktion für Datenstationen: Ich als Administrator dieses Systems weiß, dass die beiden Datenstationen DOZENTM1 und ED physisch schon einige Zeit nicht mehr Bestandteil meines Systems sind. Diese Datenstationseinheiten wurden bei Versuchen erstellt, System i Access-Bildschirme unter TCP/IP zu konfigurieren. Danach wurden die Bildschirmeinheiten umbenannt. Der Administrator hat jedoch vergessen, diese nicht mehr benötigten Einheiten zu löschen. Senden wir also eine Nachricht an diese Bildschirme, so landen wir in der Nachrichtenwarteschlage eines nicht mehr aktiven Bildschirms und werden somit auch nicht wahrgenommen.

Benutzer unterbrechen:
Dieser Parameter gilt sowohl für Benutzernachrichten als auch für Datenstationsnachrichten und erweitert das Funktionsspektrum der Systembefehle des System i erheblich.



Durchbruchnachrichten senden

Antwort anfordern:
Benötigen Sie eine Antwort auf Ihre Nachricht, so markieren Sie diesen Parameter mit einem Häkchen.

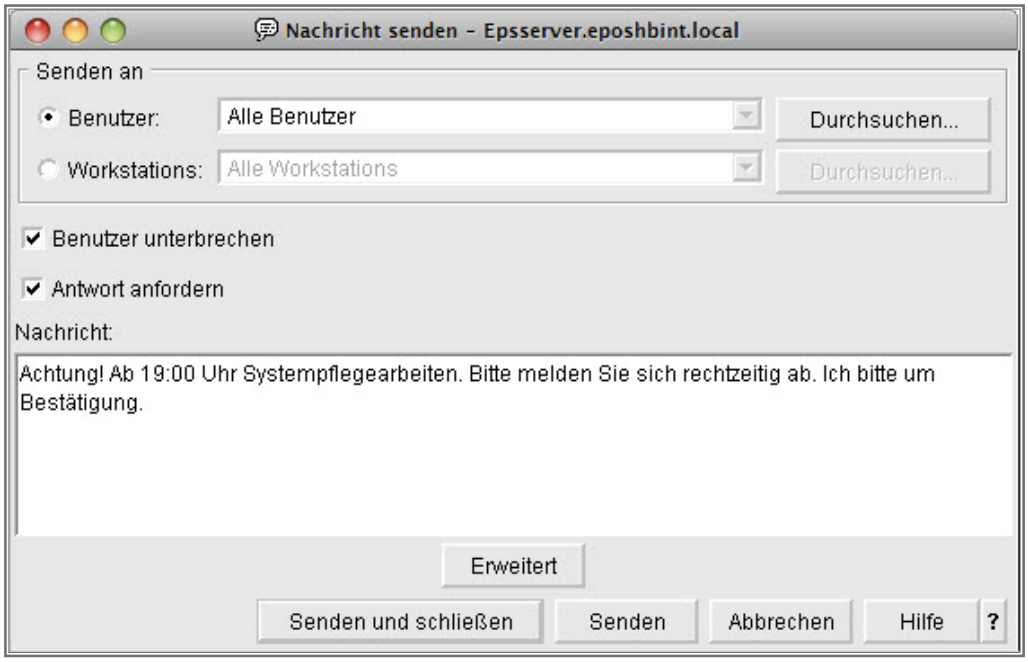


Antwort anfordern

Lassen Sie uns ein kleines Beispiel entwickeln. Sie beabsichtigen, um 19:00 Uhr mit Systempflegearbeiten zu beginnen. Darüber wollen Sie alle Benutzer im System informieren. Sie wollen sicherstellen, dass alle Benutzer diese Nachricht sofort bei Eintreffen lesen. Außerdem erwarten Sie von jedem Benutzer, dass er Ihnen eine Antwort als Bestätigung sendet, dass er die Nachricht gelesen hat.

7.8.6.1
Seite 8

Folgende Eingaben müssen dazu realisiert werden:

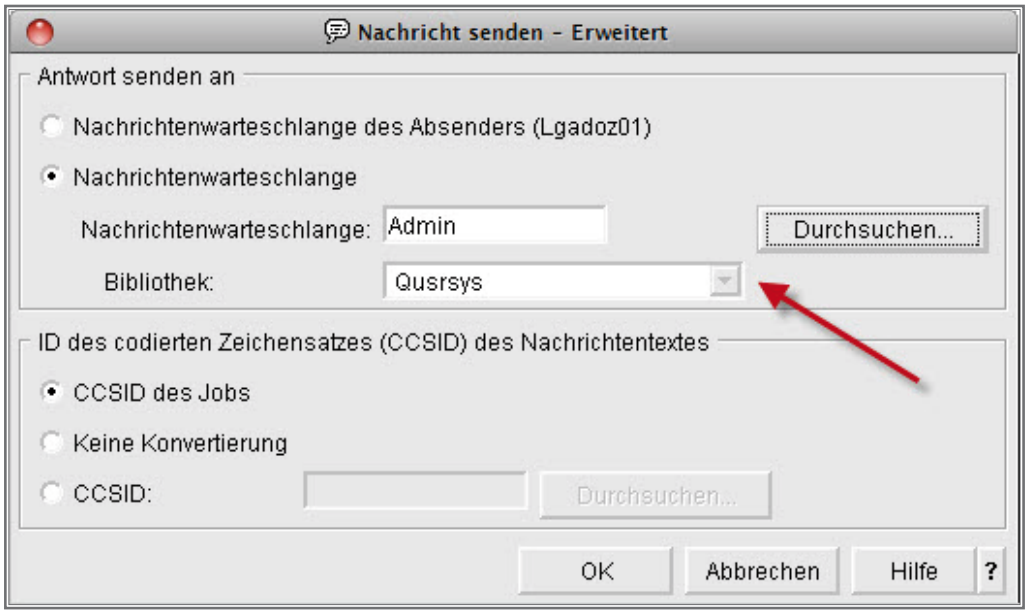


Durchbruchnachricht an alle Benutzer

Sehen Sie sich die in der Hardcopy gezeigten System i Navigator-Parameter an:

- Benutzer: alle Benutzer
- Benutzer unterbrechen: aktiviert
- Antwort anfordern: aktiviert
- Nachricht: der zu sendende Nachrichtentext

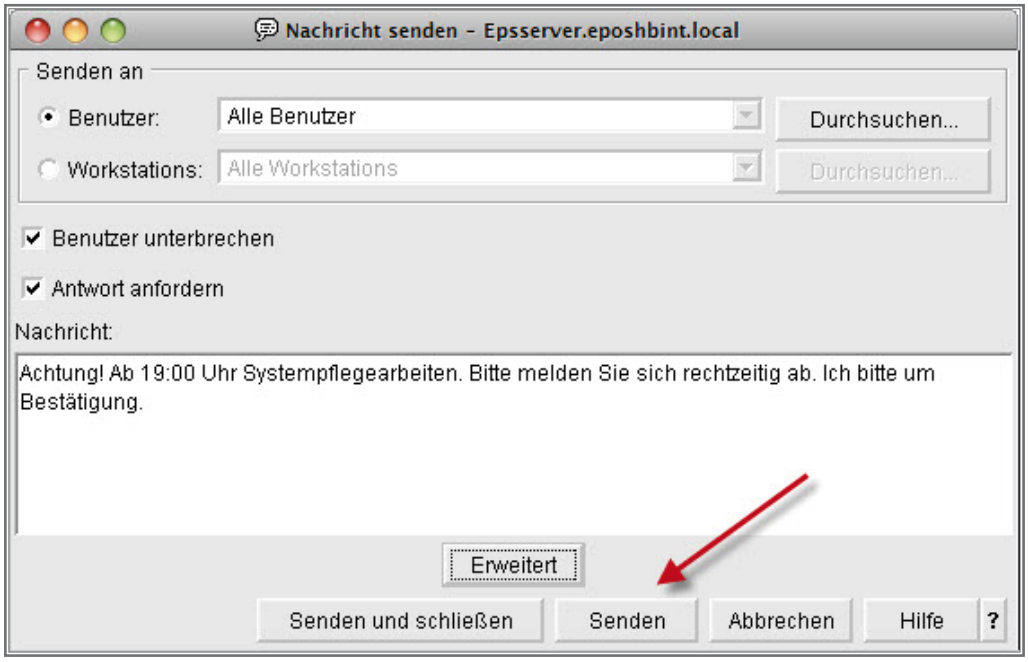
Das reicht eigentlich. Sie haben jetzt noch die Möglichkeit, durch Drücken des Buttons „Erweitert“ zu definieren, an welchen Benutzer die Bestätigungsnachrichten zu senden sind.



Die erweiterten Nachrichtenattribute

Sie sehen hier, dass alle Antworten auf die gesendete Nachricht an das Benutzerprofil ADMIN gesendet werden sollen. Hätte ich diesen Parameter nicht versorgt, würden alle Nachrichten automatisch an den Benutzer zurückgesendet, der die Nachricht versendet hat.

Nachdem der „Erweitert"-Dialog mit „OK" bestätigt und verlassen wurde, bleibt mir nur noch die Möglichkeit, die Nachricht zu versenden.



Die Nachricht senden

7.8.6.1
Seite 10

Um die Betrachtung dieses Funktionsbereichs abzuschließen, möchte ich Ihnen noch einen Hinweis geben. Wie ich schon weiter oben angedeutet habe, sind die Anzeigen des System i Navigator insgesamt als statisch zu betrachten. Anzeigen werden zu folgenden Zeiten aktualisiert:

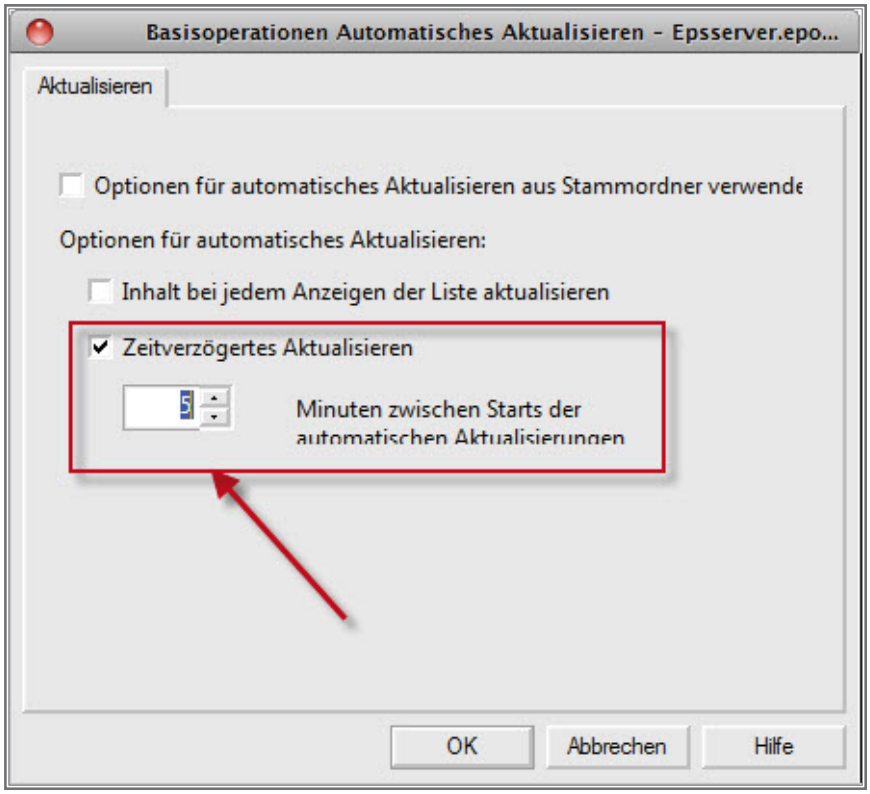
- wenn wir die Detailanzeige eines Funktionsbereichs öffnen oder
- wenn wir die F5-Taste bzw. die „Aktualisieren"-Funktion im Menü „Ansicht" betätigen.

Gerade diese Statik kann bei der Nachrichtenfunktion aber zu Nachteilen führen. Stellen Sie sich einmal vor, Sie bekommen die – wie ich finde – wichtige Nachricht von einem Benutzer, dass er mit der Bedienung seines Druckers nicht zurechtkommt. Die Nachricht entgeht Ihnen aber, da Sie die Aktualisierungsfunktion der Anzeige des System i Navigator nicht auslösen.

Um diesen Nachteil abzufedern, gibt es für Nachrichten die Möglichkeit eine automatische Nachrichtenanzeige-Aktualisierung anzufordern.

Markieren Sie dazu einfach den Funktionsbereich „Nachrichten" auf der rechten Seite des System i Navigator. Drücken Sie dann im weißen Hintergrund die rechte Maustaste. Im Kontextmenü wählen Sie die Option „Diese Ansicht anpassen ➜ automatisch aktualisieren" durch einen Klick an.

Folgender Dialog erscheint:



Nachrichtenanzeige aktualisieren

Der Standardwert für die Anzeigeaktualisierung ist „Optionen für automatisches Aktualisieren aus Stammordner verwenden“. Dieser Parameter bedeutet, dass die Angaben eines übergeordneten Ordners übernommen werden.

Basisoperationen: Nachrichtenanzeige aktualisieren

Ich habe hier den Parameter „Zeitverzögertes Aktualisieren“ markiert und definiert, dass die Nachrichtenanzeige jeweils nach fünf Minuten aktualisiert werden soll.

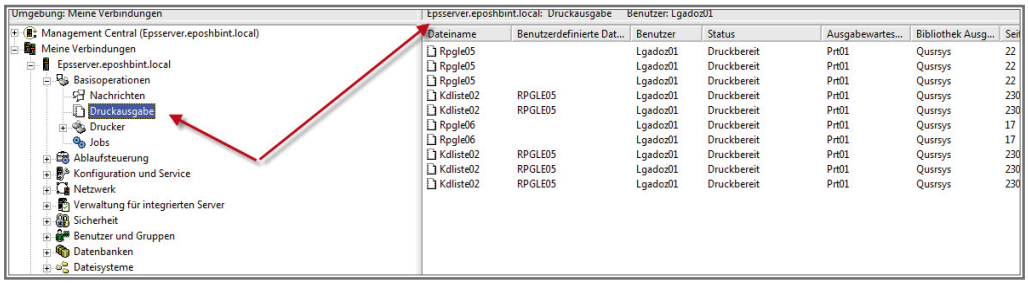
Achtung!

Wählen Sie diesen Wert nicht zu kurz. Durch zu häufige Aktualisierungen erzeugen Sie vermehrt Netzwerkverkehr.

7.8.6.1
Seite 12

Mit Druckausgaben arbeiten

Wenden wir uns jetzt dem Funktionsbereich „Druckausgaben“ zu.

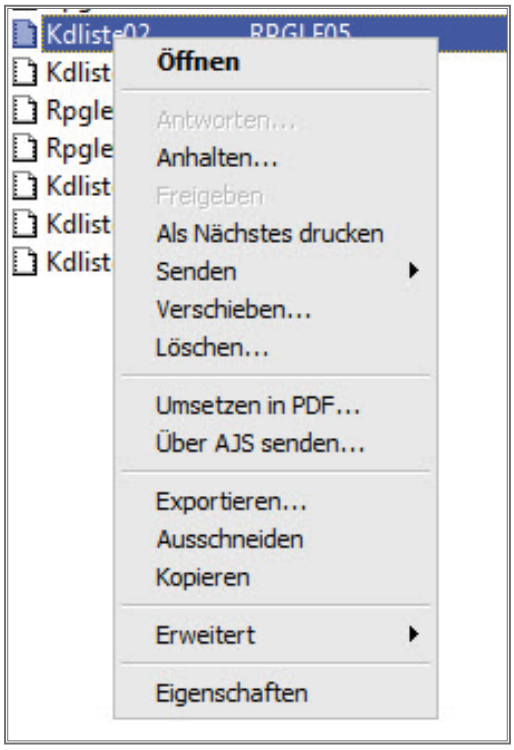


Druckausgabe

Ich werde hier nicht alle Funktionen erklären, denn ich gehe davon aus, dass Sie die Funktionen des System i-Befehls WRKSPLF (Work with Spoolfiles) kennen. Ich möchte mich daher auf die Betrachtung dreier Funktionen beschränken:

- Die Anzeige von Spooldateien in der grafischen Umgebung eines Windows PCs
- Die Änderung von Eigenschaften einer Spooldatei
- Die erweiterten Funktionen

Mit einem Rechtsklick fordern Sie das Kontextmenü für die Arbeit mit einer Druckausgabe an.



Mit Druckausgaben arbeiten

7.8.6.2
Seite 2

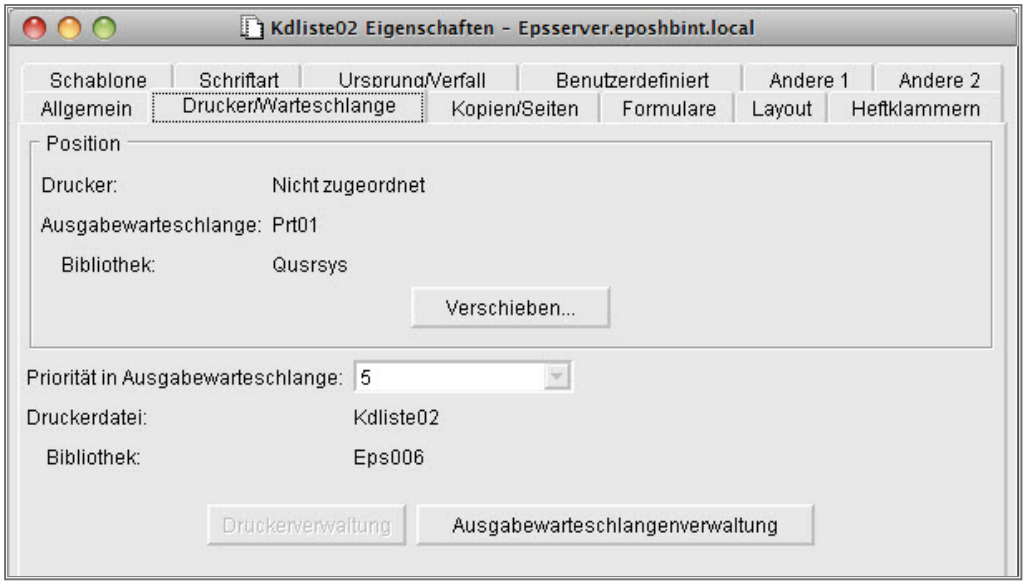
Ich befinde mich jetzt in der Anzeige der Druckausgaben für dieses System, habe eine Druckausgabe (Spooldatei) markiert und die rechte Maustaste betätigt. Dadurch hat sich das angezeigte Kontextmenü geöffnet.

Folgende Menüpunkte sind aktiviert:

- Öffnen
- Anhalten
- Als Nächstes drucken
- Senden
- Ausschneiden
- Kopieren
- Verschieben
- Löschen
- Eigenschaften
- Umsetzen in PDF
- Erweitert
- Über AJS senden (nur bei installiertem Advanced Job Scheduler)

Ich beginne mit dem Eigenschaften-Dialog. In diesem Dialog können die Attribute einer Spooldatei geändert werden. Auf der System i steht zu diesem Zweck der Befehl CHGSPLFA (Change Spoolfile Attributes) zur Verfügung.

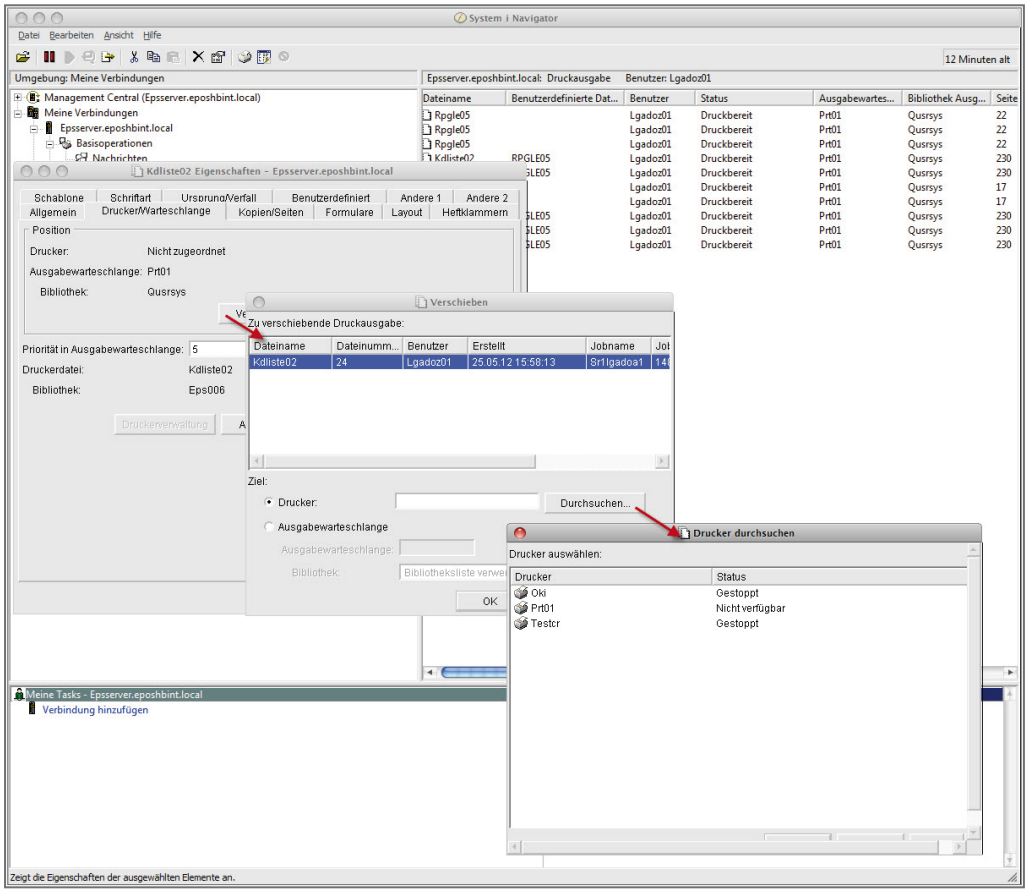
Ich möchte auch hier nur schwerpunktbezogen einige Register betrachten, deren Parameter im Systembetrieb öfter benötigt werden. Ich öffne zunächst das Register „Drucker/Warteschlange".



Register „Drucker/Warteschlange"

Folgende Parameter werden hier eingestellt:

Verschieben: Ordnen Sie mit Hilfe dieses Dialogfelds die Druckausgabe einem anderen Drucker zu.

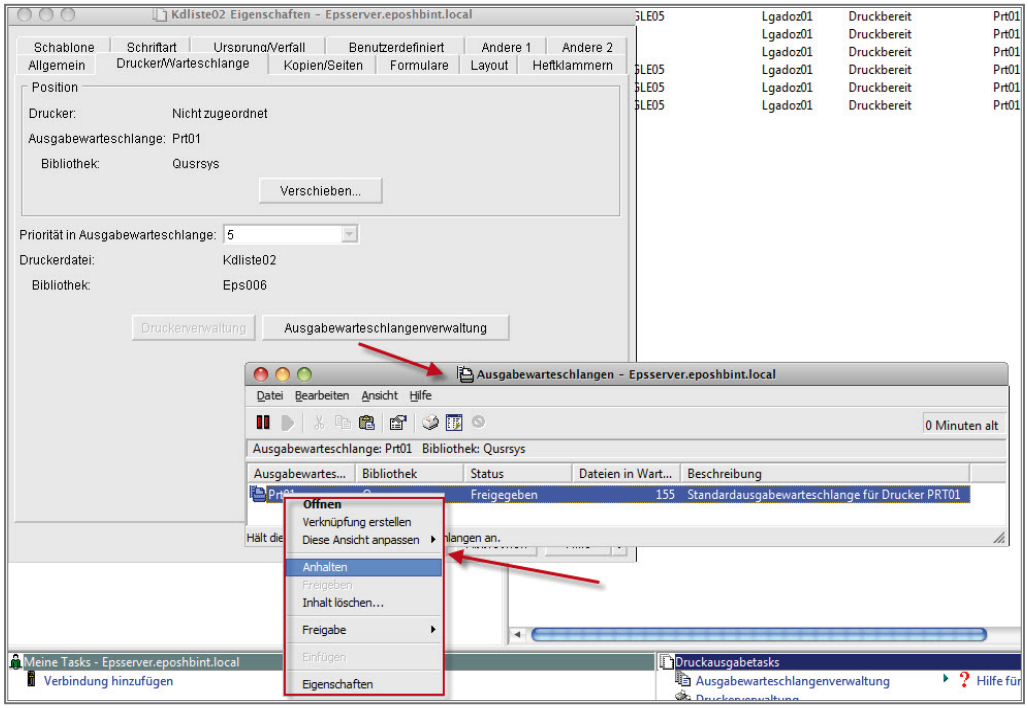


Druckausgabe verschieben

Mittels des Dialogs „Durchsuchen... Drucker durchsuchen“ können Sie der Druckausgabe einen System i-Systemdrucker zuordnen.

7.8.6.2
Seite 4

Warteschlangenverwaltung: Diese Auswahl entspricht einem WRKOUT für die Druckdatei.

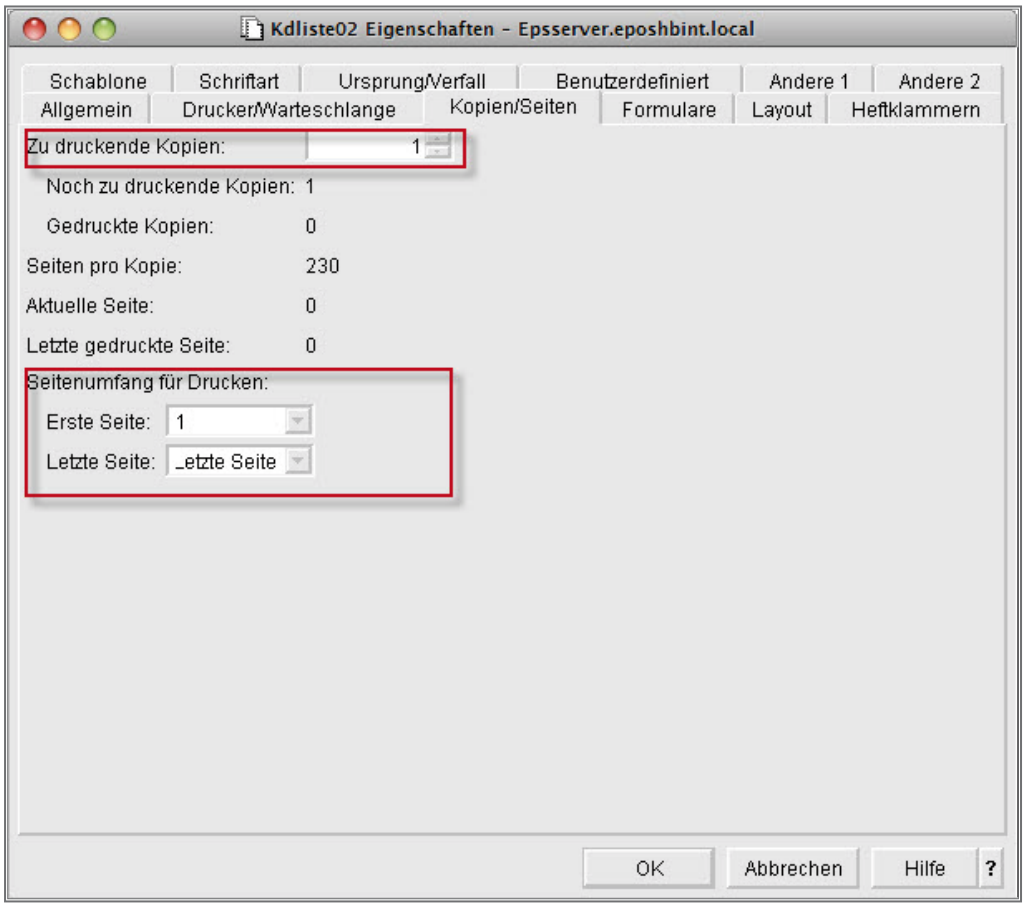


Ausgabewarteschlange bearbeiten

Sie haben so direkten Zugriff auf die Parameter der der Druckausgabe zugeordneten Ausgabewarteschlange.

Priorität für Ausgabewarteschlange: Hier verändern Sie die Priorität der Ausgabe in der Ausgabewarteschlange und können damit sowohl einen Druck in der Warteschlange vorziehen als auch in der Rangfolge weiter nach hinten stellen.

Im Register „Kopien/Seiten“ stellen Sie die Anzahl der Kopien ein, die Sie für eine bestimmte Ausgabe erstellt wissen wollen. Darüber hinaus können Sie Teilbereiche einer Druckdatei auswählen, um diese dann auszugeben.



Register „Kopien/Seiten“

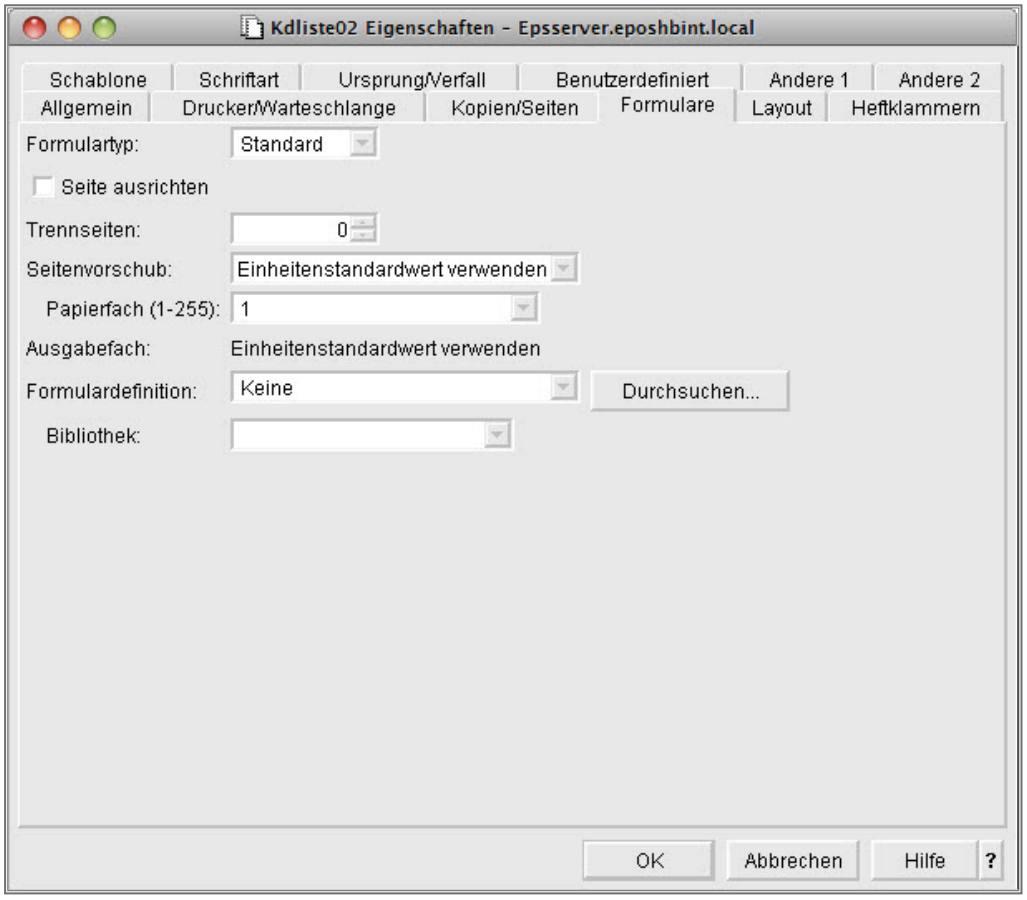
Im Ausgabeparameter „Noch zu druckende Kopien“ wird Ihnen der Fortschritt des Druckens angezeigt. Beabsichtigen Sie z. B. fünf Kopien einer Ausgabe zu drucken und haben bereits zwei Exemplare gedruckt, so werden Sie in diesem Feld den Wert drei (3) sehen. Das ist die Anzahl der noch zum Druck anstehenden Ausgabeexemplare.

Weiterhin können Sie hier einstellen, ob alle Seiten gedruckt werden sollen (das ist die Default-Einstellung) oder ob nur bestimmte Seitenbereiche gedruckt werden sollen.

Um diese Einstellungen zu realisieren, nutzen Sie die Parameter „Erste Seite“ und „Letzte Seite“.

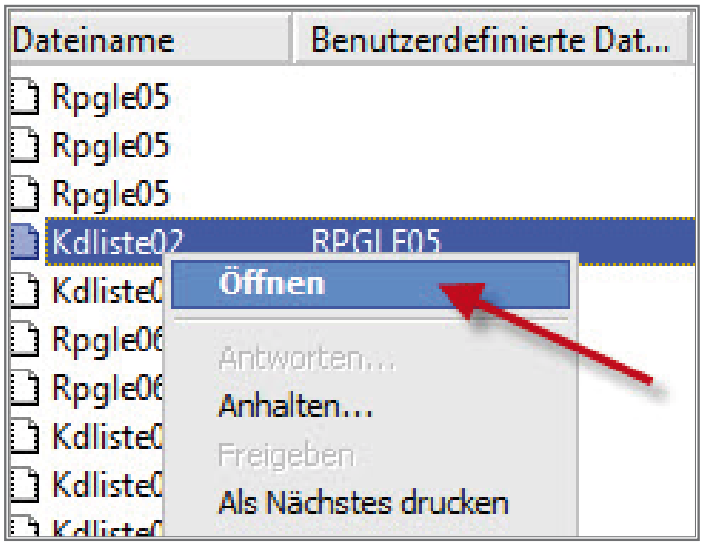
7.8.6.2
Seite 6

Das Register „Formulare“ beschließt die Betrachtung der Spooldateiattribute.



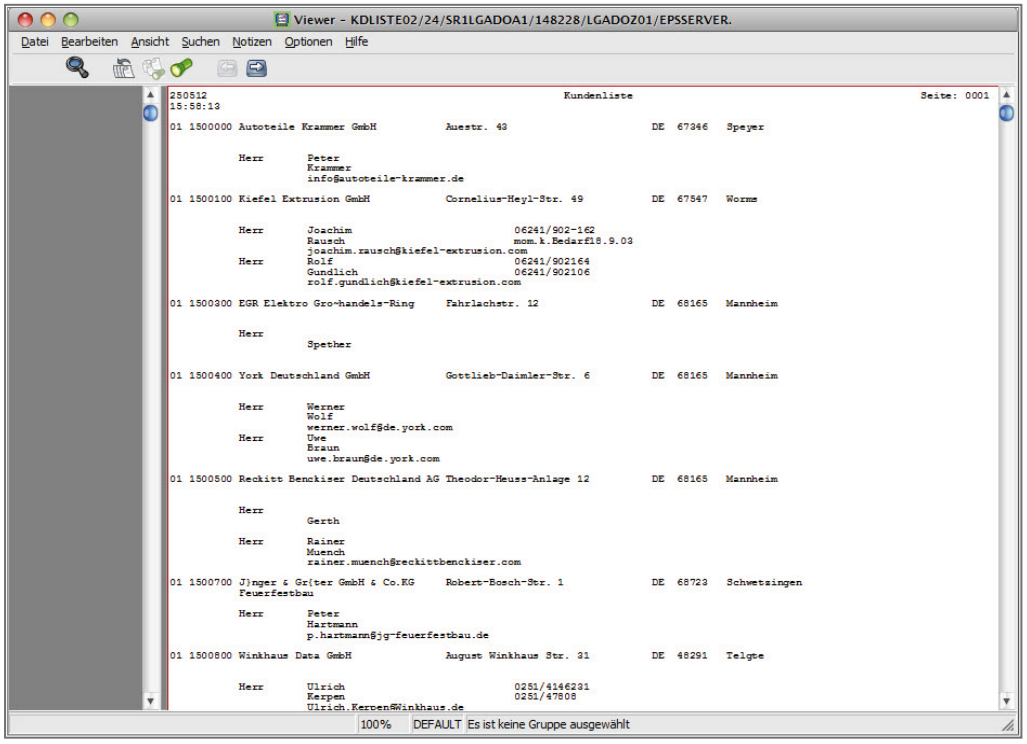
Register „Formulare“

Sehen Sie sich diese Parameterwerte an und Sie werden ihren Sinn aufgrund Ihrer bisherigen Erfahrungen mit dem CHGSPLFA-Befehl leicht erkennen.



Druckausgabe öffnen

Der Kontextmenüpunkt „Öffnen“ ruft den sogenannten AFP-Viewer auf, der System i-Spooldateien im grafischen Umfeld anzeigt.



Der AFP-Viewer

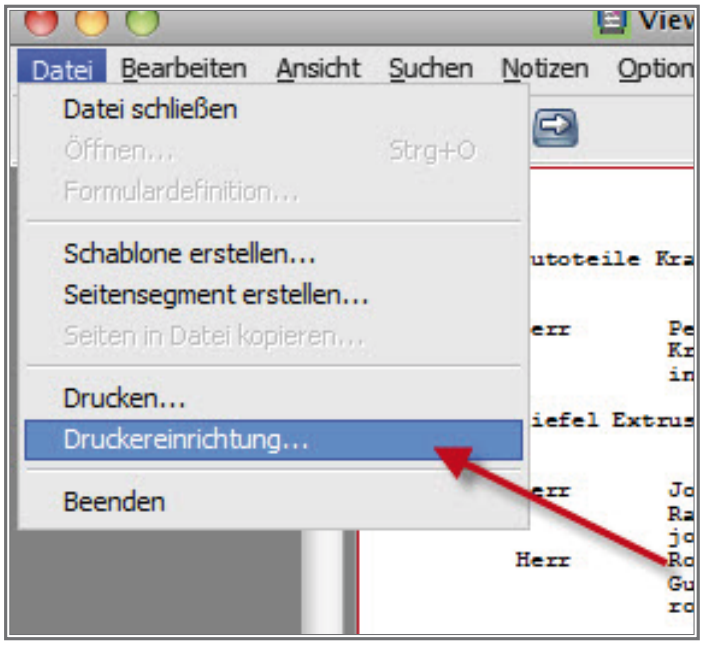
Kurzgesagt ermöglicht diese Komponente des i Access alle Funktionen, die auch der System i-Systembefehl DSPSPLF (Display Spool File) mit sich bringt.

Es muss jedoch gesagt werden, dass der AFP-Viewer weit über die Funktionen des DSPSPLF-Befehls hinausgeht. Somit ist der Viewer nicht nur ein Ersatz für den zitierten Anzeigebefehl, sondern eine erhebliche Erweiterung.

Im AFP-Viewer werden zusätzlich Funktionen realisiert, die die Weitergabe von Spooldateien an PC-Anwendungen oder die Erstellung von AFP-Objekten ermöglichen usw. usf.

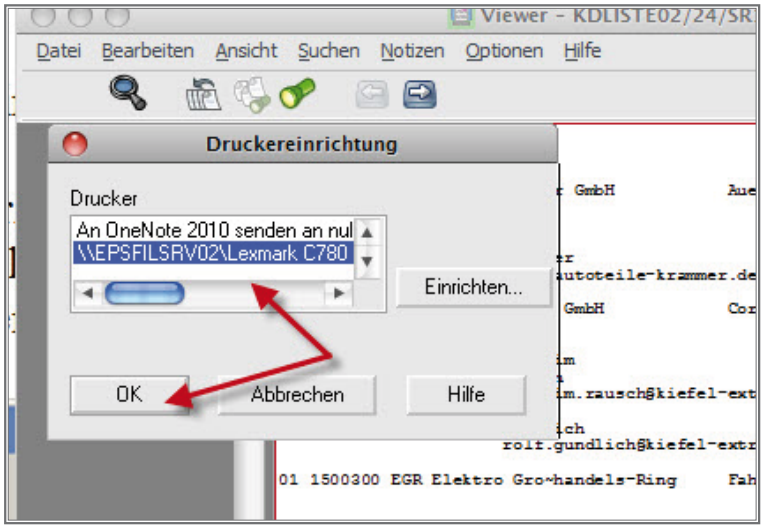
Eine der meines Erachtens bestechendsten Funktionen des AFP-Viewers besteht darin, System i-Spooldateien direkt über einen Windows-Drucker auszugeben.

7.8.6.2
Seite 8



Der AFP-Viewer: Drucker einrichten 1

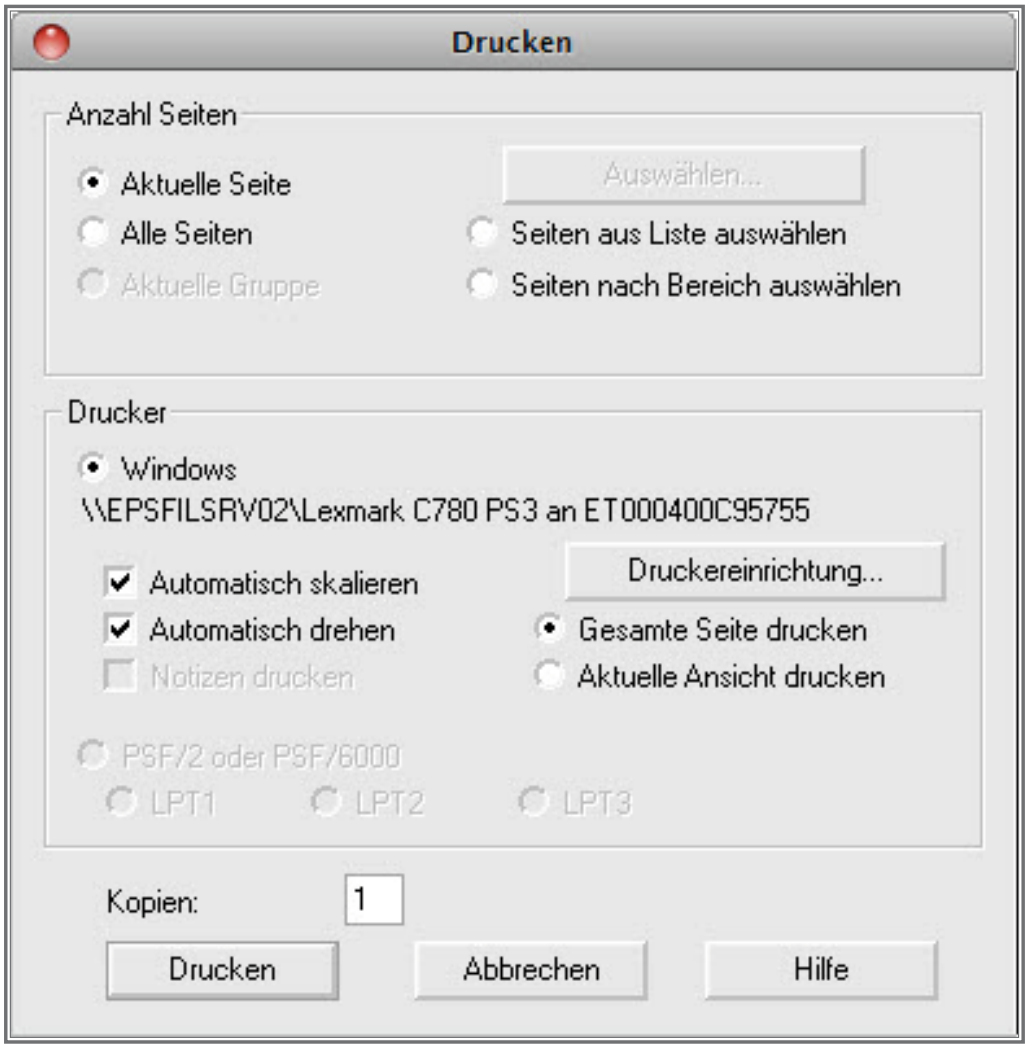
Um eine Druckausgabe vorzubereiten, öffnen Sie das Menü „Datei ➜ Druckereinrichtung“. Hier wählen Sie den Windows-Drucker aus, der die System i-Spooldatei ausgeben soll.



Der AFP-Viewer: Drucker einrichten 2

Sie sehen, dass ich hier auf unserem Druckserver einen Lexmark-Drucker ausgewählt habe. Diese Auswahl bestätige ich mit „OK“.

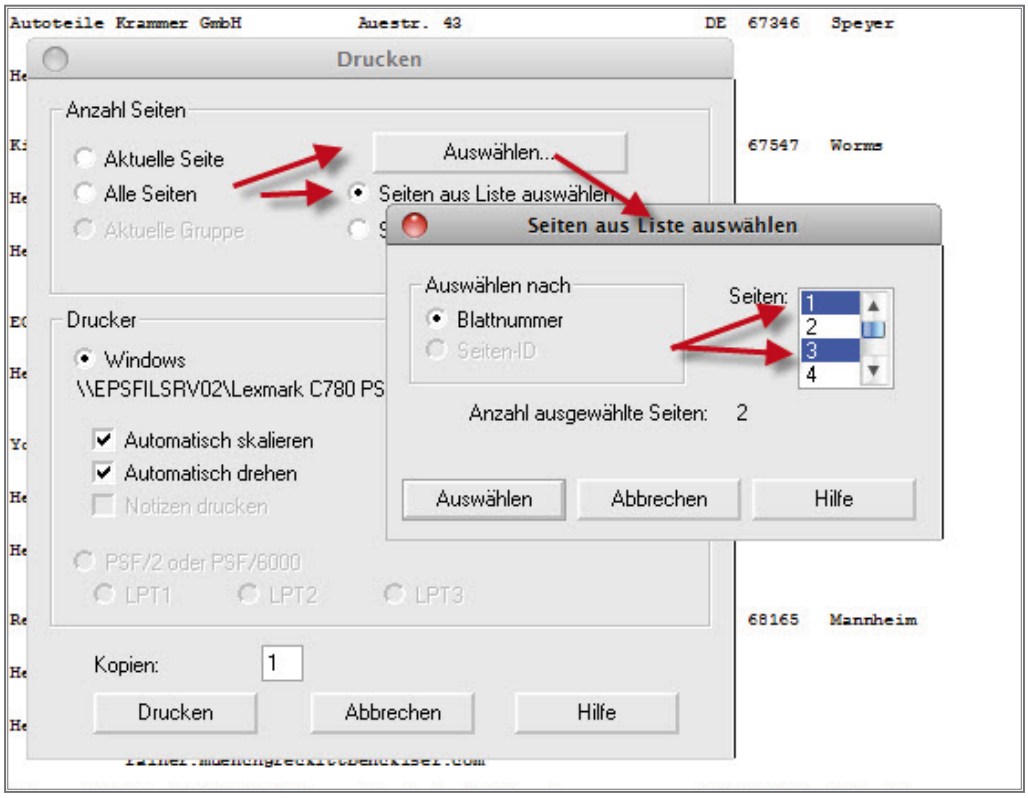
Danach wechsle ich in das Menü „Datei ➜ Drucken“.



Spool drucken

Hier markiere ich im Parameter „Anzahl Seiten“, ob nur die aktuelle Seite oder alle Seiten gedruckt werden sollen. Darüber hinaus können auch Seiten aus einer Liste ausgewählt werden.

7.8.6.2
Seite 10

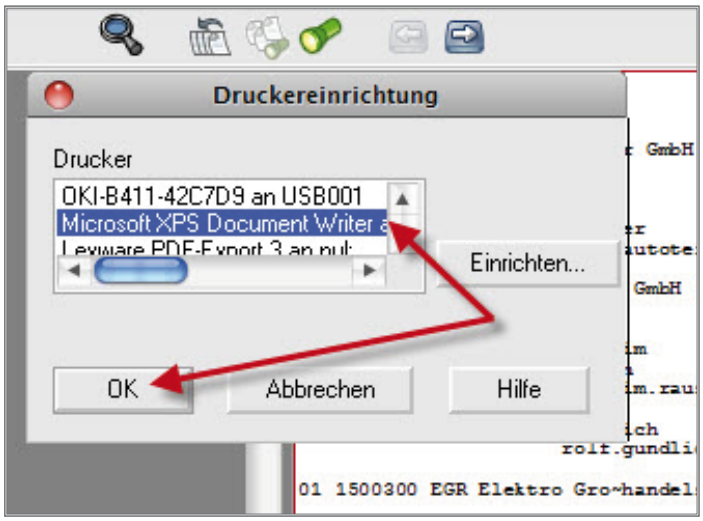


Seiten für Druck

Die Betätigung des Knopfs „Drucken“ startet den Windows-Druckprozess.

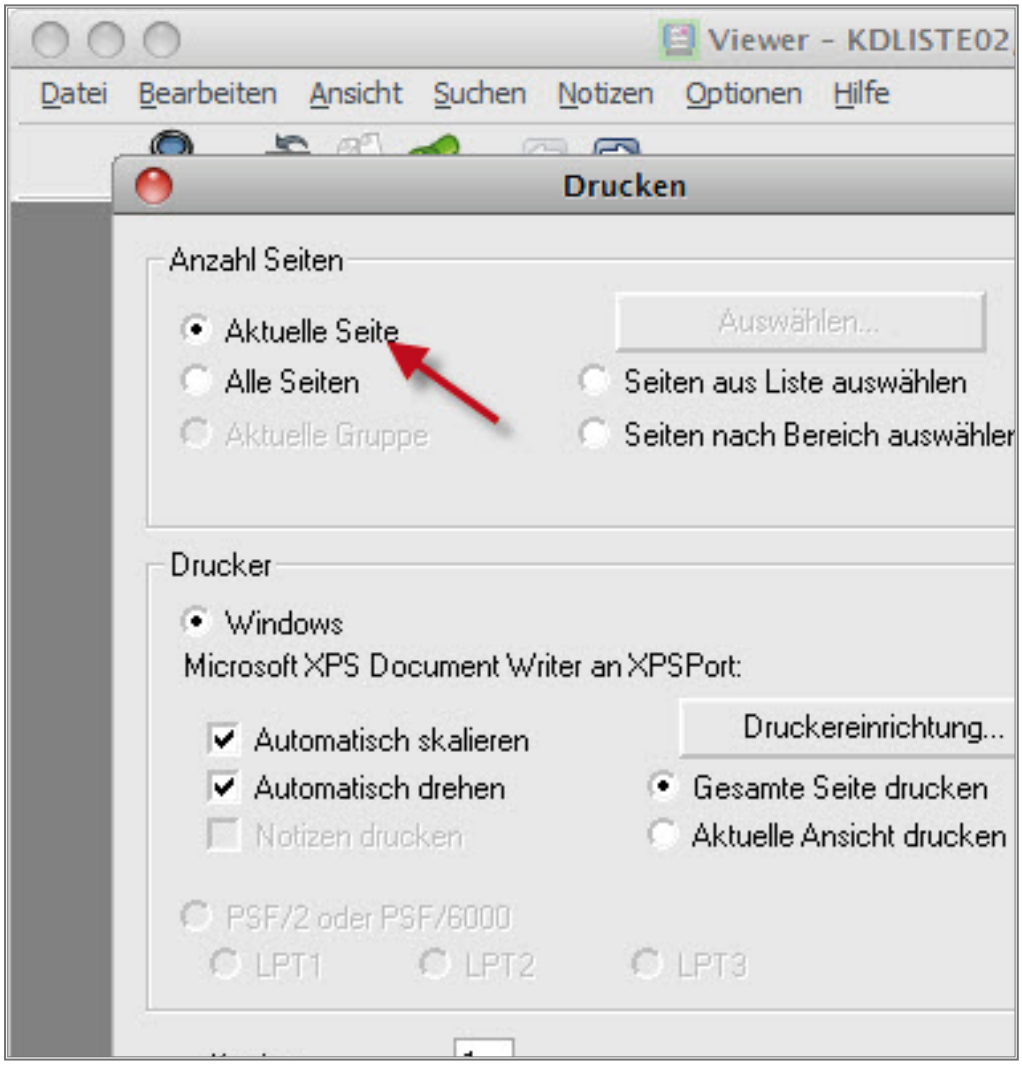
Mit Hilfe dieser Druckfunktion lassen sich aber auch aus jeder System i Spool-Datei PDF-Dokumente oder Microsoft XPS-Dokumente erzeugen. Hierzu benötigen Sie entweder ein PDF-Erstellungsprogramm, wie z. B. das Open Source Tool Ghostwrite oder den Microsoft Office beiliegenden XPS-Writer. Da ich auf meinem System Microsoft Office installiert habe, werde ich Microsoft XPS-Writer zum Erzeugen einer XPS-Datei benutzen.

Ich wechsle hierfür wieder in das Menü „Datei ➜ Druckereinrichtung“.



XPS-Writer einrichten

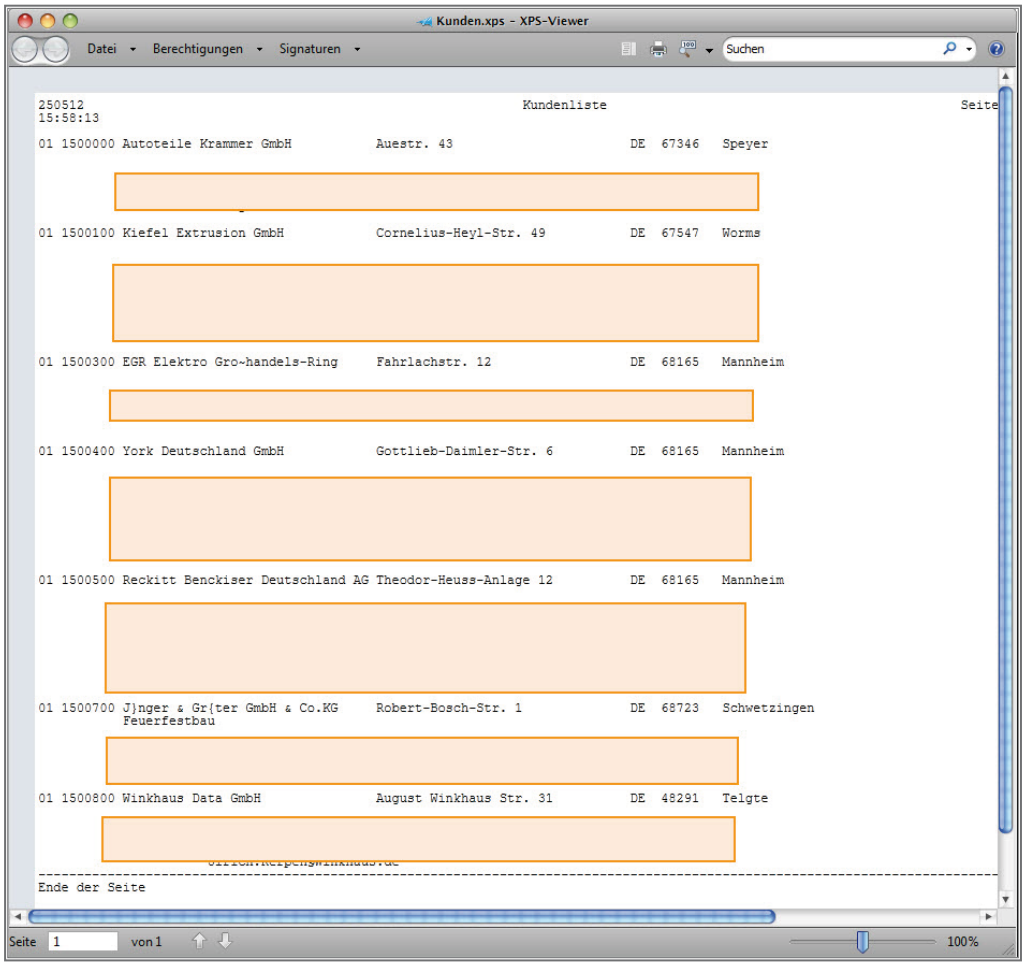
Ich markiere den Eintrag „Microsoft XPS Document Writer“ und bestätige die Auswahl mit „OK“. Danach wechsle ich in das Verzeichnis „Datei ➜ Drucken“.



XPS-Dokument erzeugen

Hier markiere ich im Parameter „Anzahl Seiten“ den Wert „Alle Seiten“. Die Betätigung des Knopfs „Drucken“ startet den Windows-Druckprozess.

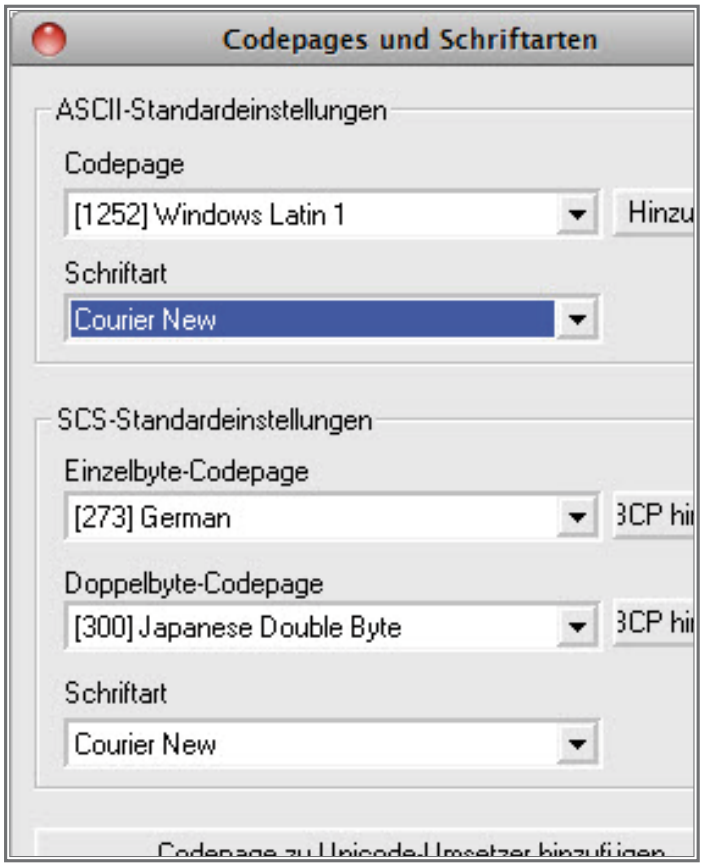
7.8.6.2
Seite 12



Anzeige der Spool in XPS-Viewer

Zum Schluss noch zwei Tipps, mit denen Sie die Anzeigen verbessern können. Mir ist bei einer ganzen Reihe von Arbeitsplätzen aufgefallen, dass die Umlautdarstellung des AFP-Viewers nicht korrekt erfolgt. Dieser Fehler lässt sich leicht beheben.

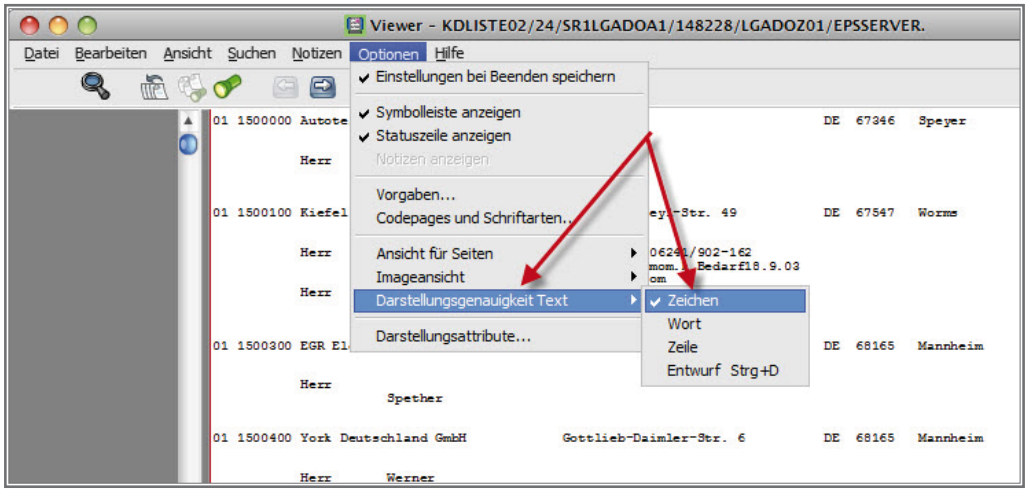
Gehen Sie im AFP-Viewer in das Menü „Optionen ➜ Codepages und Schriftarten“. Es erscheint folgender Dialog:



Weitere Vorgaben

Tragen Sie die für Ihren Arbeitsplatz günstigsten Werte ein.

Haben Sie den Eindruck, dass die Darstellungsgenauigkeit des AFP-Viewers zu wünschen übrig lässt, leicht erkennbar durch rechte „Flatterränder“ in der Druckdarstellung, so können Sie die Darstellungsgenauigkeit im Menü „Optionen ➜ Darstellungsgenauigkeit ➜ Zeichen“ erhöhen.



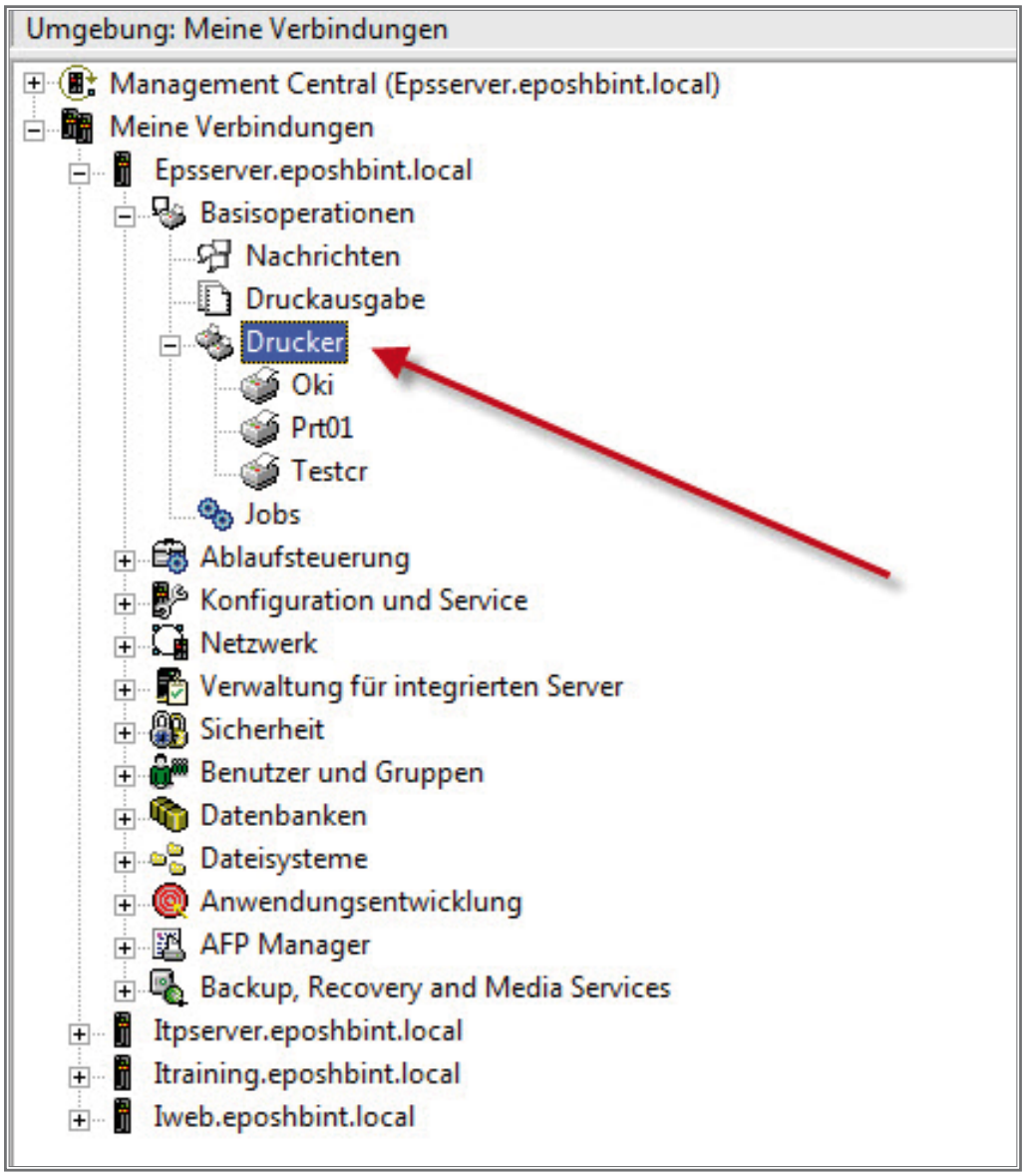
Darstellungsgenauigkeit ändern

7.8.6.2
Seite 14

Im Untermenü finden Sie die Auswahlmöglichkeiten „Zeichen“, „Wort“, „Zeile“ und „Entwurf“. Sie stellen unterschiedliche Präzisionskategorien für die Textdarstellung im AFP-Viewer dar. Markieren Sie die Auswahl „Zeichen“ und die Flatterränder verschwinden.

Mit Druckern arbeiten

Der Funktionsbereich „Drucker“ ermöglicht die Steuerung von System i-Systemdruckern.



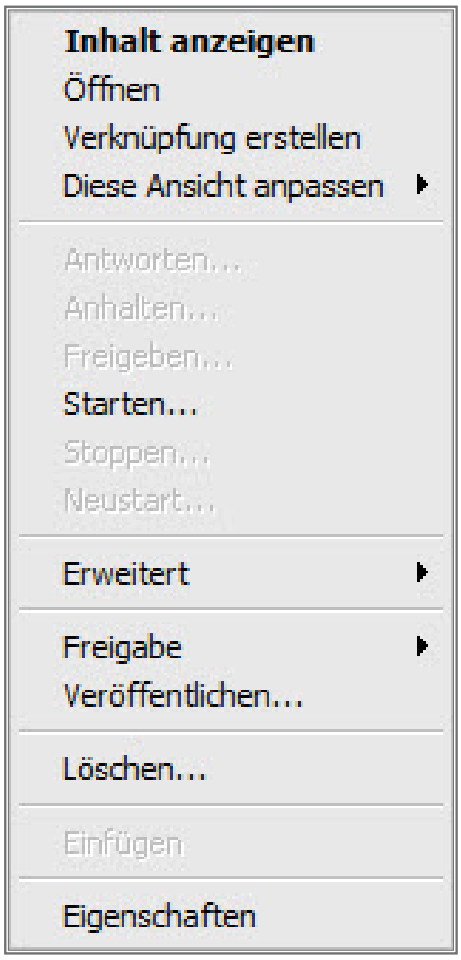
Drucker

Sie sehen in der Hardcopy des System i Navigator alle Systemdrucker, die am geöffneten System Epsserver.eposhbint.local angeschlossen sind.

7.8.6.3
Seite 2

Ein Doppelklick auf einen der ausgewiesenen Drucker zeigt uns wieder alle Verarbeitungsmöglichkeiten des Funktionsbereichs „Druckausgaben“ an. Verglichen mit System i-Systembefehlen bedeutet ein Doppelklick auf einen der angezeigten Drucker soviel wie ein WROUTQ (Work with Outqueues) für die dem Drucker zugeordnete Ausgabewarteschlange. Das bedeutet, dass hier dieselben Ausgabebearbeitungsfunktionen realisiert werden können, die Sie im vorherigen Kapitel kennengelernt haben.

Lassen Sie uns nun noch das Kontextmenü für den Funktionsbereich „Drucker“ betrachten.



Kontextmenü „Drucker“

Folgende Verarbeitungsmöglichkeiten bestehen:

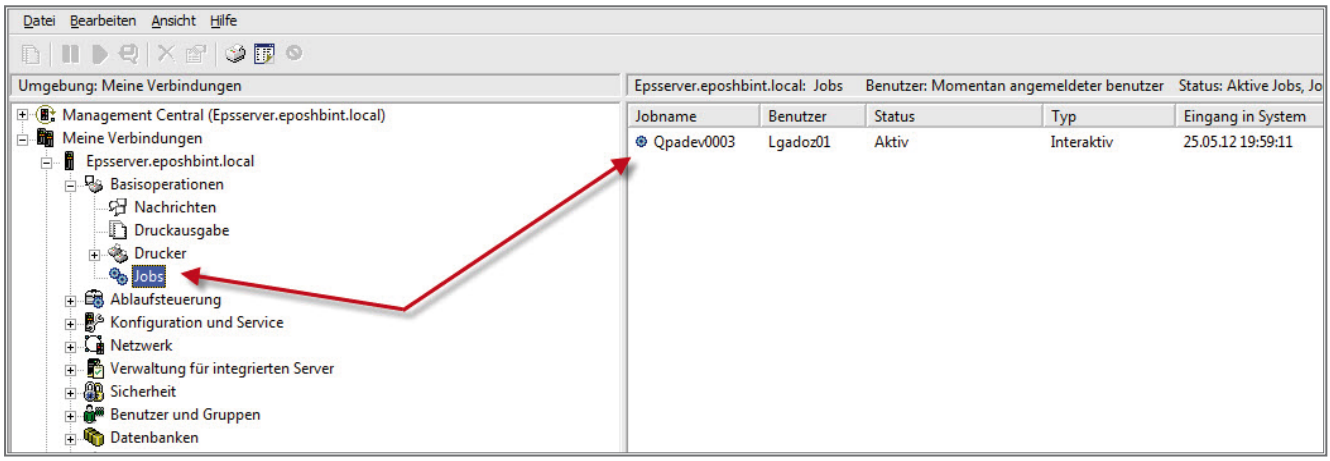
- Inhalt anzeigen:
  Die angeschlossenen Drucker anzeigen
- Öffnen:
  Die angeschlossenen Drucker in einem eigenen Fenster anzeigen
- Verknüpfung erstellen:
  Icon für diese Funktion in einem Ordner oder auf dem Desktop erstellen
- Freigabe:
  Druckerfreigabe für das Microsoft-Netzwerk erstellen
- Veröffentlichen:
  Drucken an einen LDAP-Server publizieren
- Antworten:
  Druckernachrichten beantworten, sofern welche anstehen
- Anhalten:
  Den Drucker anhalten
- Freigeben:
  Einen angehaltenen Drucker wieder freigeben
- Starten:
  Das Druckausgabeprogramm starten
- Stoppen:
  Das Druckausgabeprogramm beenden
- Neustart:
  Das Druckausgabeprogramm nach einem Reset starten
- Verfügbar machen:
  Zugriffsrechte für einen Drucker vergeben
- Eigenschaften:
  Druckereigenschaften ändern/anzeigen
- Erweitert:
  Mit der zugeordneten Ausgabewarteschlange arbeiten
  (Anhalten, Starten…)
  Mit dem zugeordneten Druckausgabeprogramm arbeiten
  (Starten, Beenden)
  Mit der zugeordneten Druckeinheit arbeiten (Anhängen, Abhängen)

Markieren Sie einen Drucker, so sehen Sie auf der rechten Seite alle Ausgaben, die in der zugehörigen Ausgabewarteschlange auf ihre Verarbeitung warten. Die Verarbeitungsmöglichkeiten entsprechen denen, die im vorangegangenen Abschnitt dargestellt wurden.

7.8.6.3
Seite 4

Mit Benutzerjobs arbeiten

Die Funktion „Jobs“ in den Basisoperationen des System i Navigator entspricht weitgehend den Funktionen des CL-Befehls WRKUSRJOB.



Jobs

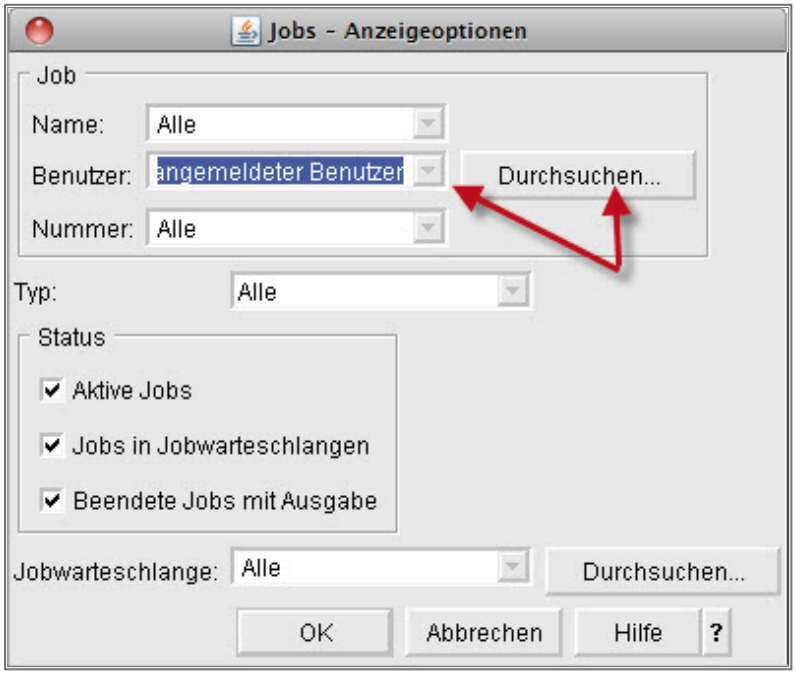
Dieser Befehl dient im Betriebssystem IBM i zur Suche nach im System befindlichen Jobs. Verwechseln Sie diese Funktion nicht mit einem WRKACTJOB (Mit aktiven Jobs arbeiten). Zwar sind viele Anzeigen dieser Funktion ähnlich, doch die komplexen Steuerungsfunktionen eines WRKACTJOB fehlen hier.

Als kleine Fallstudie möchte ich nach allen Jobs des Benutzers „Luttkus“ auf dem System suchen. Die Suche betrifft sowohl Jobs, die in einer Jobwarteschlange auf Verarbeitung warten, solche die derzeit aktiv vom System bearbeitet werden sowie Jobs, die auf ihre Ausgabe in einer Ausgabewarteschlange warten.

Wie gehe ich vor? Zuerst einmal muss ich Sorge dafür tragen, dass ausschließlich die Jobs für den Benutzer LGADOZ01 angezeigt werden. Ich gehe dazu in das Kontextmenü „Anzeigeoptionen“, das sich im Hauptmenü des System i Navigator befindet.

7.8.6.4
Seite 2

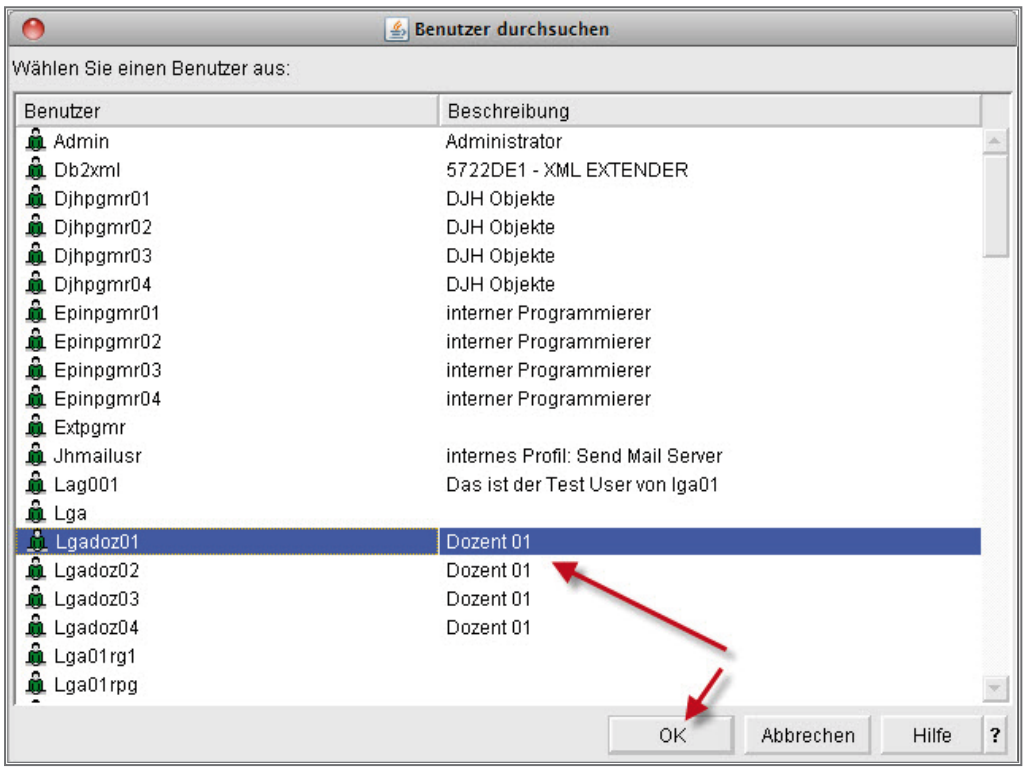
Es öffnet sich der folgende Dialog, der starke Ähnlichkeit mit dem Bedienerführungsdialog des CL-Befehls WRKUSRJOB aufweist.



Anzeigeoptionen für Jobs

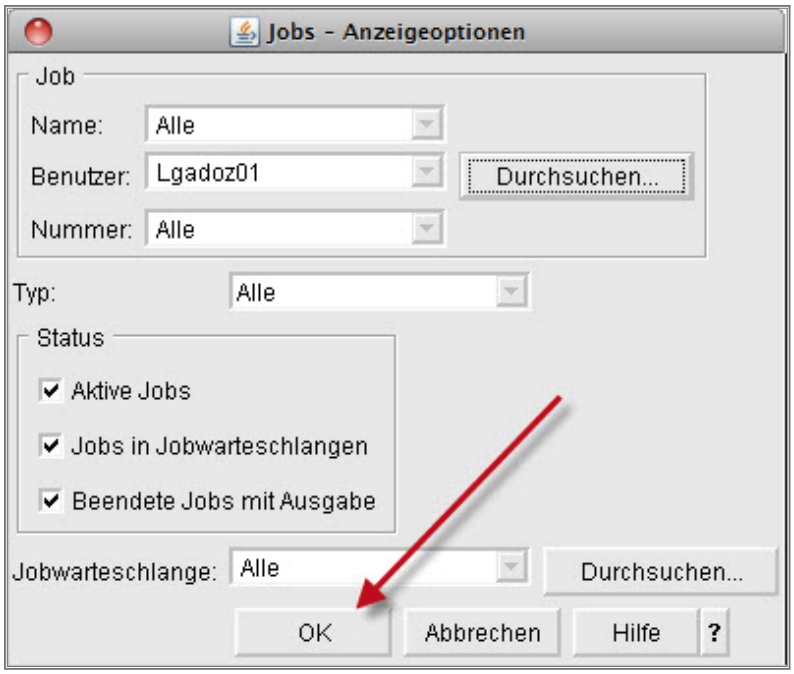
Sie sehen in der Parametergruppe „Job“ den Einzelparameter „Name“, in den der Standardwert „Alle“ eingetragen ist. Das bedeutet, dass als Standard alle Jobs aller Benutzer der System i angezeigt werden. Diesen Wert werde ich ändern, indem ich auf den Knopf „Durchsuchen“ klicke.

Es öffnet sich der Dialog „Benutzer“.



Benutzer auswählen

Hier werden alle für Sie berechtigten Benutzer angezeigt. Ich wähle den Benutzer LGADOZ01 aus. Nach Drücken auf „OK“ komme ich in den Anzeigeoptionen-Dialog zurück.



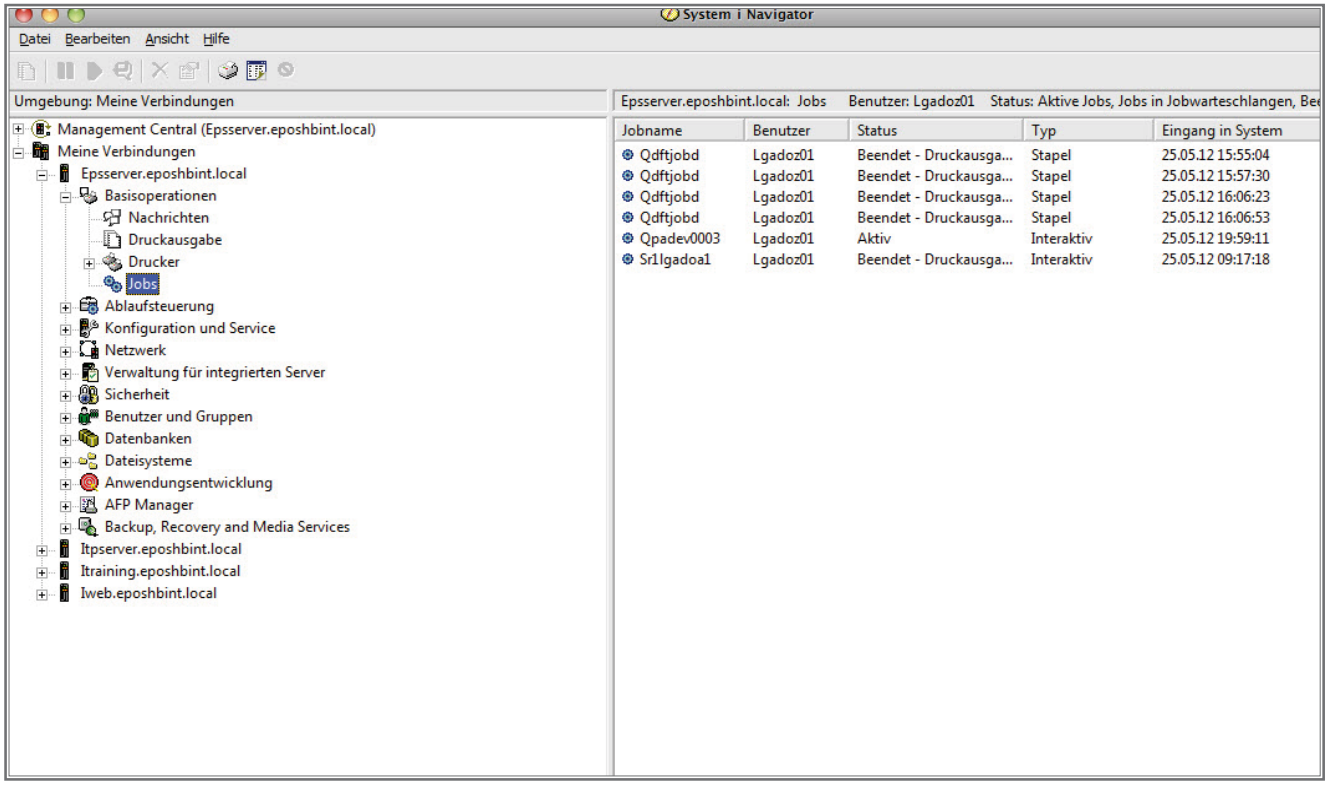
Anzeigeoptionen: mehr Parameter

7.8.6.4
Seite 4

In der Parametergruppe „Jobstatus“ aktiviere ich folgende Parameter:

- Aktive Jobs anzeigen – Ja
- Jobs in Jobwarteschlange anzeigen – Ja
- Beendete Jobs mit Druckausgabe anzeigen – Ja

Danach kehre ich in die System i Navigator-Anzeige zurück.



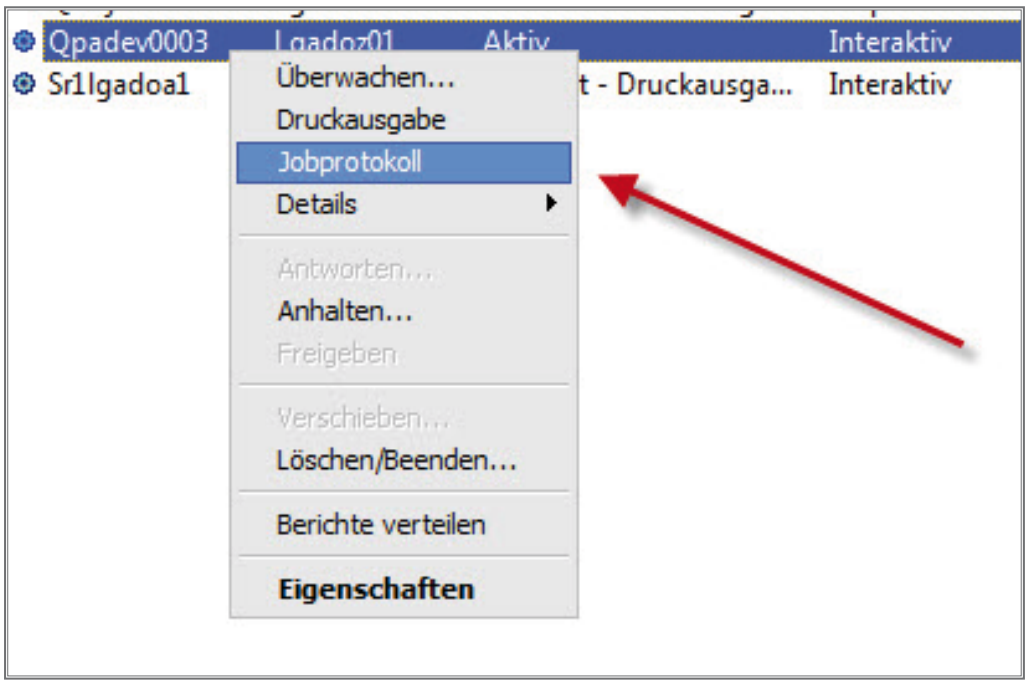
Jobs des Benutzers LGADOZ01

In der Anzeige auf der linken Seite erkennen Sie jetzt die Jobs des Benutzers LGADOZ01 im System.

Außerdem sehen Sie, dass alle angezeigten Jobs zur Druckausgabe anstehen. Einige dieser Jobs sind als Stapeljobs (Batchjobs) durch das System gelaufen, einer der Jobs – Typ „Interaktiv“ – ist ein Bildschirmjob, der noch aktiv ist.

Das ist interessant. Wir schauen uns das Jobprotokoll dieses Jobs an. Dazu markiere ich den Job mit der rechten Maustaste. Es öffnet sich das Kontextmenü für diesen Job.

Ich führe einen Doppelklick auf die Auswahl „Jobprotokoll“ aus.



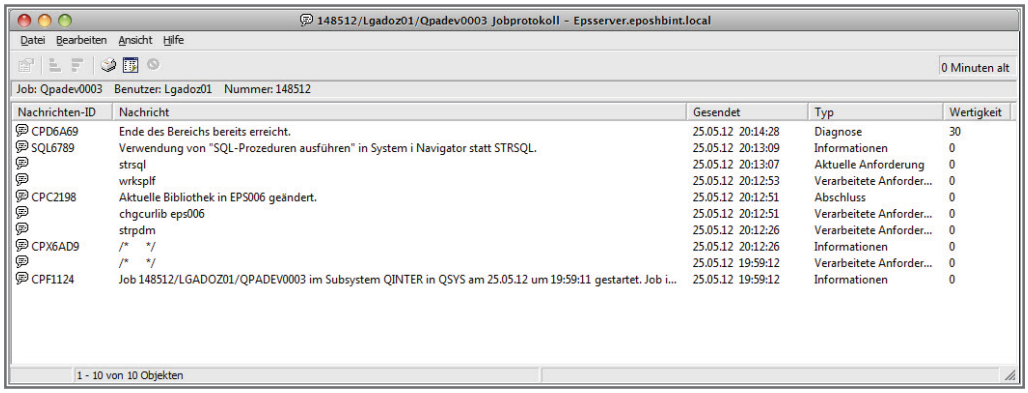
Kontextmenü für einzelnen Job

Folgende zusätzliche Funktionen sind je nach Jobstatus möglich:

- Druckausgabe:
  Druckausgaben für einen Job anzeigen
- Jobprotokoll:
  Jobprotokoll für einen Job anzeigen
- Antworten:
  Anstehende Nachrichten für einen Job beantworten
- Anhalten:
  Laufenden Job anhalten
- Freigeben:
  Angehaltenen Job wieder freigeben
- Verschieben:
  Aktiven Job in ein anderes Subsystem verschieben
- Löschen:
  Job sofort oder kontrolliert beenden
- Eigenschaften:
  Eigenschaften eines Jobs anzeigen
- Überwachen:
  Job auf seinen Ressourcenverbrauch hin überwachen

7.8.6.4
Seite 6

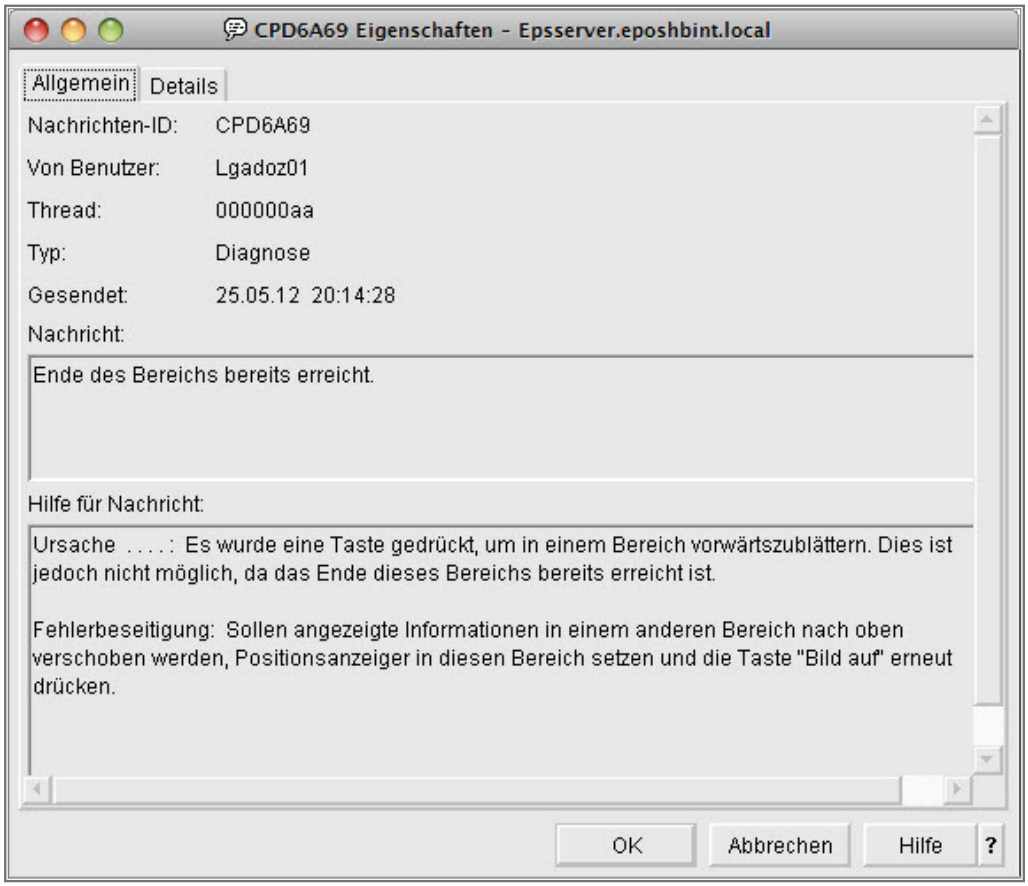
Nachstehend sehen Sie das Jobprotokoll:



Druckausgabe für Job

Wir schauen uns die Nachricht CPD6A69 noch etwas genauer an und führen einen Doppelklick auf sie aus.

Nachstehend sehen Sie die Eigenschaften dieser Nachricht:

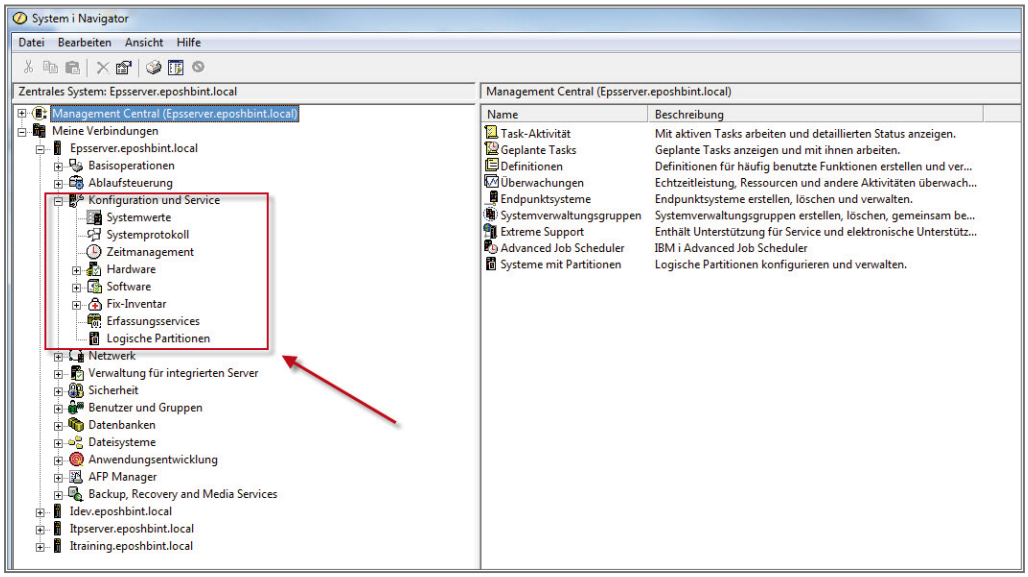


Eigenschaften einer Nachricht

Diesem Dialog entnehmen Sie die genauen Kenndaten der an den Job gesendeten Nachricht.

Konfiguration und Service

„Konfiguration und Service“ stellt eine Vielzahl von Funktionen bereit. Sie können sich beispielsweise eine vollständige Liste der Hardwareressourcen Ihres Systems sowie den Betriebsstatus der einzelnen Hardwareressourcen anzeigen lassen, des Weiteren eine vollständige Liste der Softwareressourcen Ihres Systems sowie den Release-Level der einzelnen Ressourcen und die installierten Komponenten.



Konfiguration und Service

Folgende Funktionsgruppen sind darstellbar. Zum leichteren Verständnis benenne ich in der Folge die CL-Befehle, die diesen Funktionsgruppen am ehesten entsprechen:

- Systemwerte anzeigen und ändern
  WRKSYSVAL – Mit Systemwerten arbeiten
- Hardware-Inventar anzeigen
  WRKHDWRSC – Mit Hardwareressourcen arbeiten
- Software-Inventar anzeigen
  GO LICPPGM: Lizenzprogramm Menü
  Auswahl 10
- Fix-Inventar anzeigen und verwalten
  GO PTF: Mit Program Temporary Fixes arbeiten
- Mit Erfassungsservices arbeiten
  STRPFRMON: Performance-Monitor starten
- Zeitmanagement des System i
  WRKSYSVAL QTIMZON

7.8.7
Seite 2

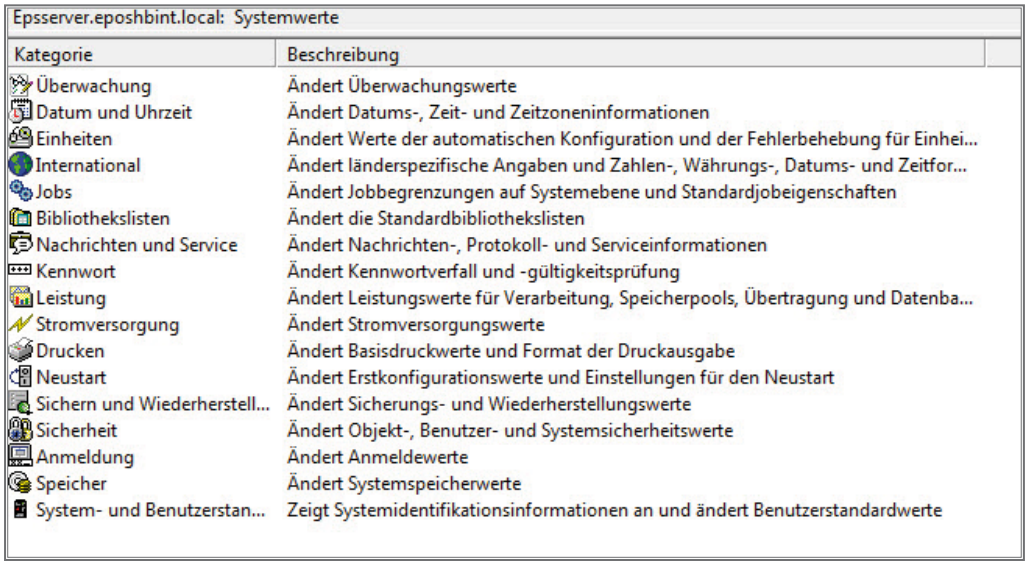
Systemwerte

Systemwerte sind Informationselemente, die die Betriebsumgebung im gesamten System beeinflussen. Ein Systemwert enthält Steuerungsinformationen für den Betrieb bestimmter Komponenten des Systems. Ein Benutzer kann die Systemwerte ändern, um zum Beispiel die Arbeitsumgebung zu definieren. Systemdatum und Bibliotheksliste sind Beispiele für Systemwerte.

Im System i Navigator werden Systemwerte in Funktionsgruppen zusammengefasst. Hierbei geht der System i Navigator weit über das Gruppenraster hinaus, das uns aus dem Befehl WRKSYSVAL bekannt ist:

- *ALC
- *DATTIM
- *EDT
- *LIBL
- *MSG
- *SEC
- *STG
- *SYSCTL

Vergleichen Sie einfach diese Liste mit dem unten gezeigten Dialog.



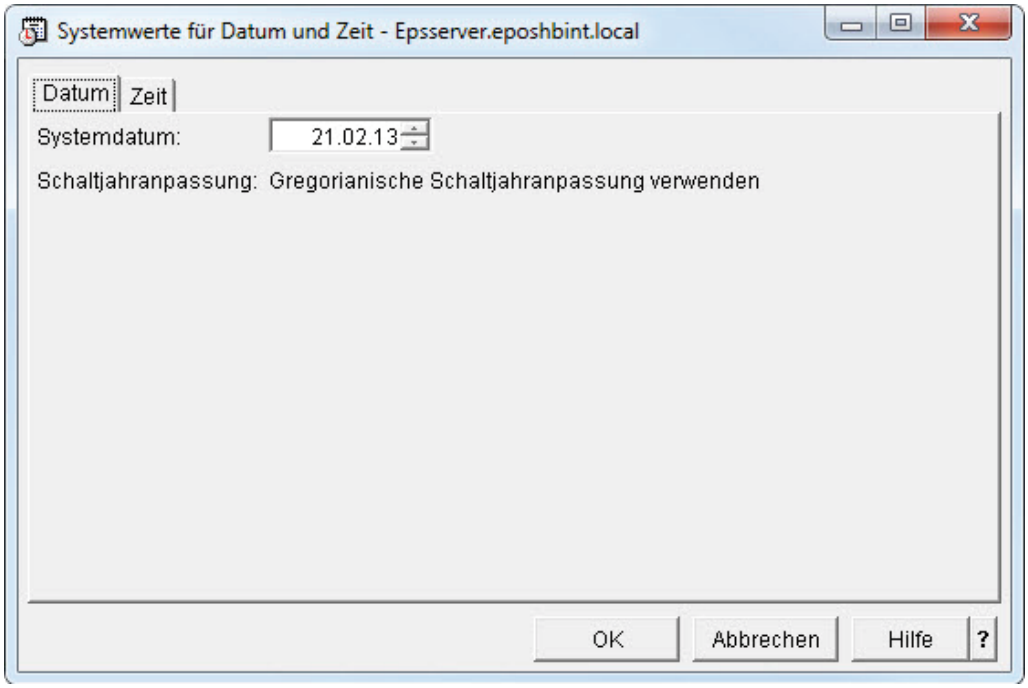
Systemwertgruppen

Es ist sicherlich leicht zu erkennen, dass hier eine differenzierte Gruppierung der Systemwertfunktionen vorgenommen wurde.

7.8.7.1
Seite 2

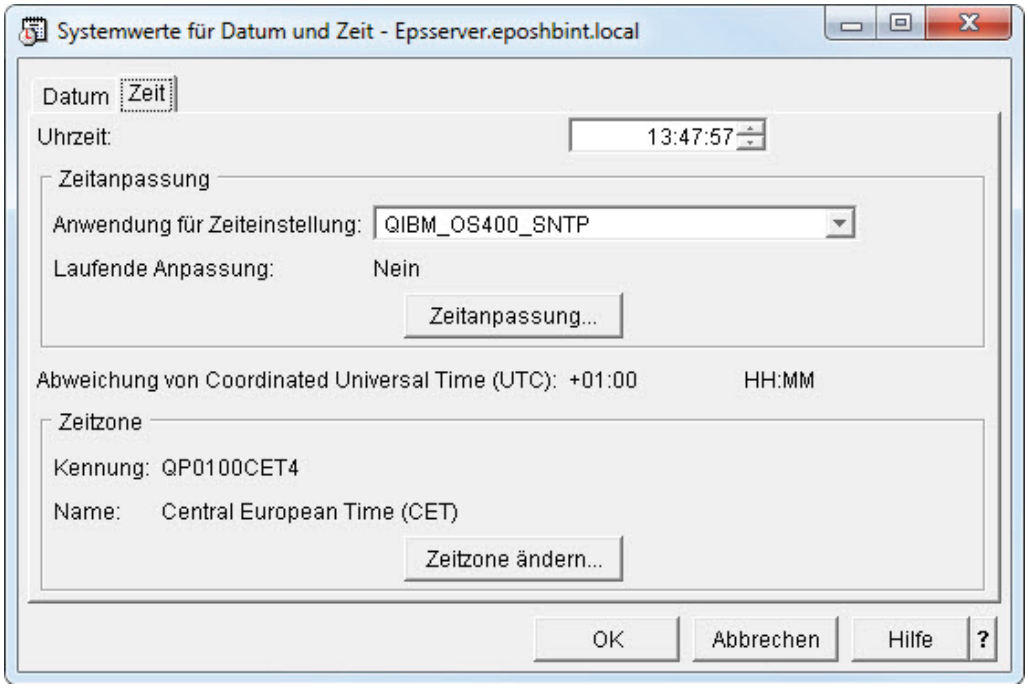
Ich werde jetzt nicht alle oben gezeigten Gruppen mit Ihnen zusammen durchgehen, sondern nur stellvertretend einige Gruppen kursorisch darstellen.

Ich öffne zunächst die Systemwertfunktion „Datums- und Zeitwerte".



Datums- und Zeitwerte: QDATE

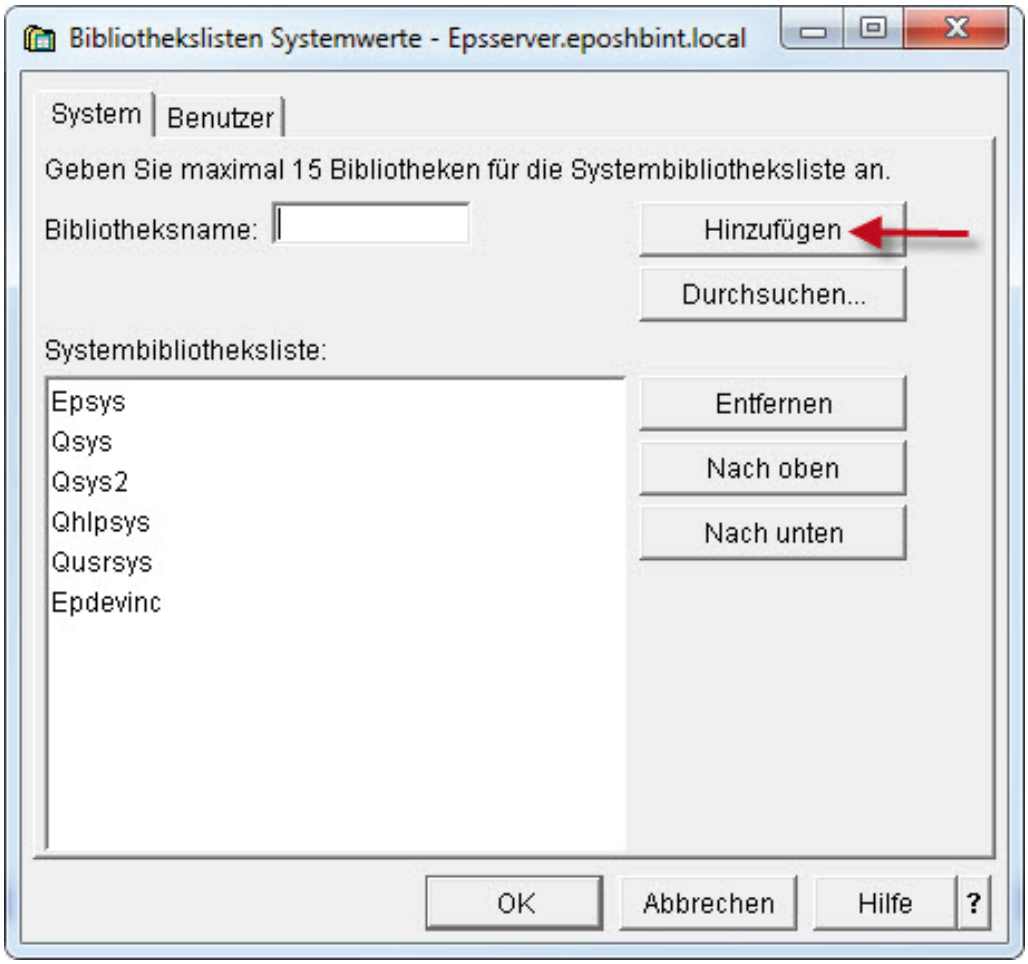
Hier sehen Sie die System i Navigator-Darstellung des Systemwerts QDATE.



Datums- und Zeitwerte: QTIMADJ und QTIMZON

Schauen Sie sich oben den Dialog zur Veränderung der Systemwerte QTIMAD (Zeitanpassung) und QTIMZON (Zeitzone) an. Mit QTIMZON geben Sie die zeitliche Verschiebung zur Greenwich-Standardzeit an und aktivieren die Sommer-/Winterzeitanpassung.

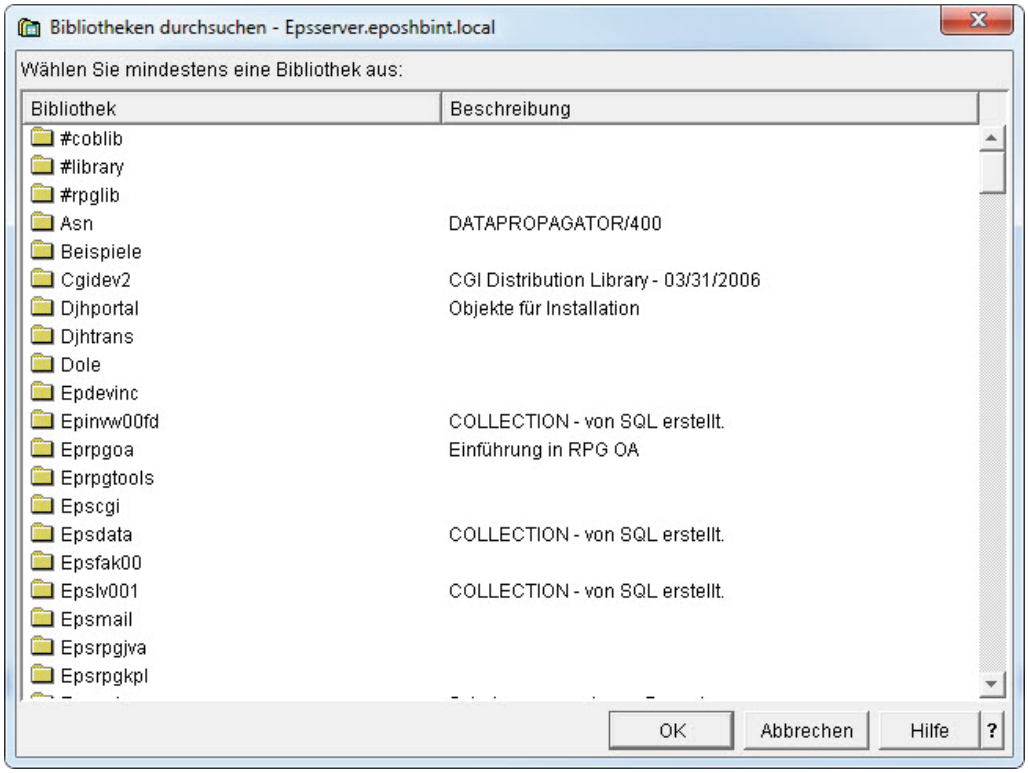
Jetzt bearbeiten wir die Systembibliothekssuchliste.



Systemwert QSYSLIBL

7.8.7.1
Seite 4

Hinter dem gezeigten Dialog steht die Bearbeitung des Systemwerts QSYSLIBL. Soll der Systemteil der Bibliothekssuchliste geändert werden, so geben Sie den Namen der gewünschten Bibliothek in den Parameter „Bibliotheksname“ ein und drücken dann „Hinzufügen“. Ist Ihnen der Name der Bibliothek jedoch nur teilweise oder nicht bekannt, so gehen Sie in „Durchsuchen...“ und folgender Dialog öffnet sich:



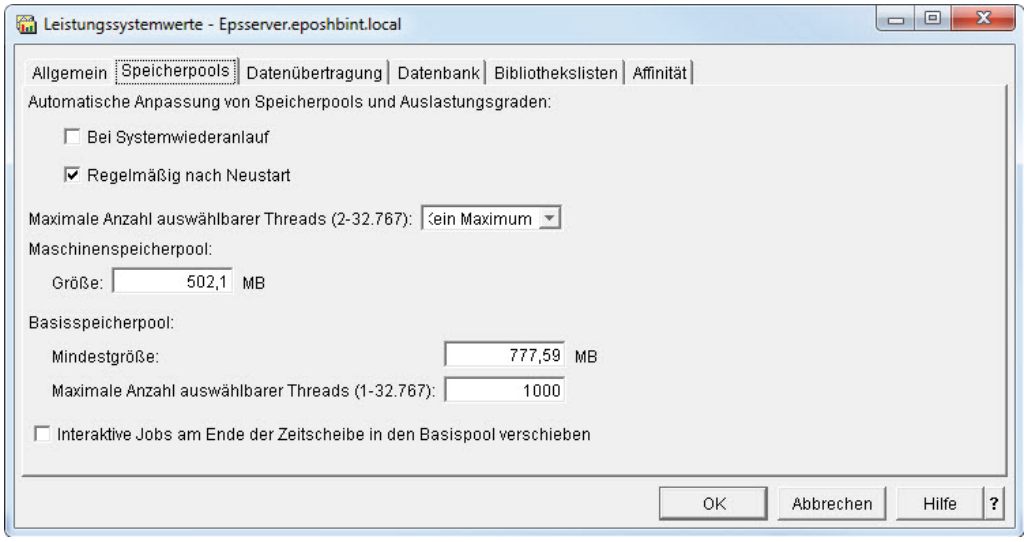
Bibliothek suchen

Wählen Sie aus der gezeigten Liste „Ihre“ Bibliothek aus. Es werden hier nur die Bibliotheken des Systems angezeigt, für die Sie ein Leserecht besitzen.

Nachdem Sie die betreffende Bibliothek markiert haben, beenden Sie den Dialog, indem Sie auf „OK“ klicken.

Danach wird Ihnen die ausgewählte Bibliothek im Feld „Systembibliotheksliste“ angezeigt.

Um mit performancerelevanten Systemwerten zu arbeiten, öffnen Sie die Funktion „Leistungssystemwerte“. Im unten gezeigten Beispieldialog habe ich das Register „Speicherpools“ geöffnet.



Leistungssystemwerte

Hier können unter anderem folgende Systemwerte gesetzt bzw. verändert werden:

- QMAXACTLVL – maximal auswählbare Threads
- QMCHPOO – Größe des Maschinenspeicherpools
- QBASPOOL – Basisspeicherpool

Die Einstellungen für den Systemdrucker können Sie unter der Funktion „Drucken“ ändern.

Druckersystemwerte

7.8.7.1
Seite 6

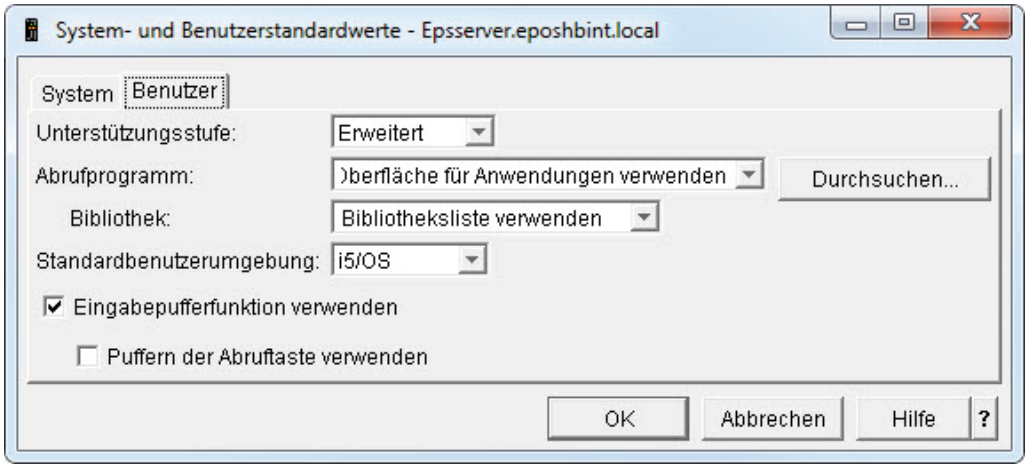
Dieser Dialog entspricht dem Systemwert QPRTDEV.

Die Funktion „System- und Benutzerstandardwerte“ zeigt im Register „System“ die Systemkenndaten Ihres System i an.

Systemkenndaten

Hinter diesem Anzeigedialog verbergen sich Systemwerte wie QMODE (, QSRLNBR etc.

Im Register „Benutzer“ finden Sie Systemwerte wie die Unterstützungsstufe – QASTLVL, das Abrufprogramm – QATNPGM und die Standardbenutzerumgebung – QSPCENV.



Benutzerstandardwerte

Hardware

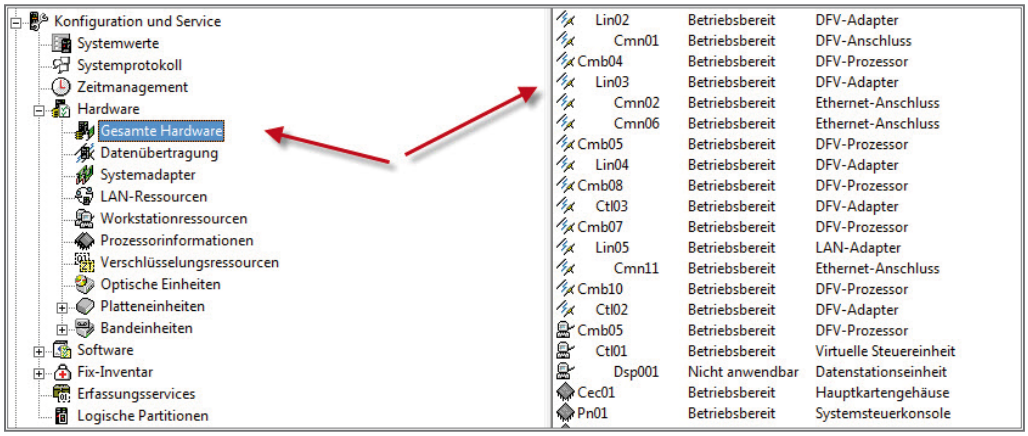
Der Funktionsknoten „Hardware“ dient vor allem dazu, die in einem System i verbauten Hardwarekomponenten in unterschiedlichen Selektionen anzuzeigen.

Folgende Selektionsmöglichkeiten bestehen:

- Gesamte Hardware
- Datenübertragung
- Systemadapter
- LAN-Ressourcen
- Datenstationsressourcen
- Prozessorinformationen
- Verschlüsselungsressourcen
- Optische Einheiten
- Platteneinheiten
- Bandeinheiten

7.8.7.2
Seite 2

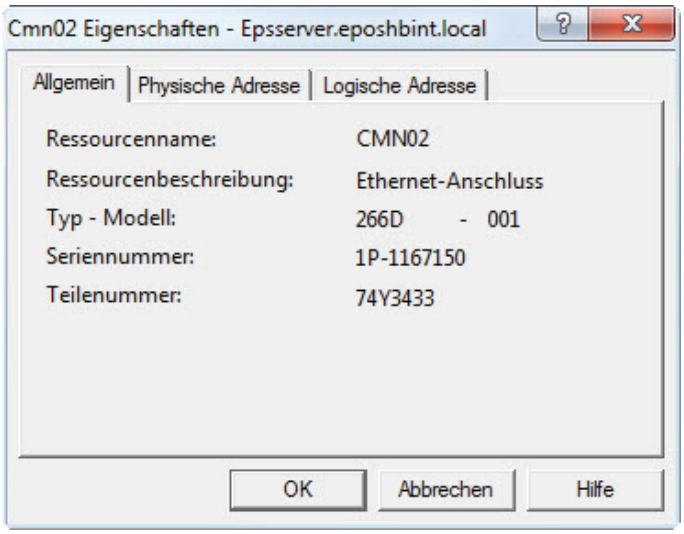
Gesamte Hardware



Gesamte Hardware i

Die Funktion „Gesamte Hardware“ zeigt eine Liste aller Hardwareressourcen Ihres Systems an. Diese Liste enthält keine LAN-Ressourcen (LAN = lokales Netzwerk). Für jede Ressource werden der Ressourcenname, der Status und die Beschreibung angezeigt.

Ein Doppelklick auf eine der dargestellten Ressourcen öffnet einen Detaildialog.



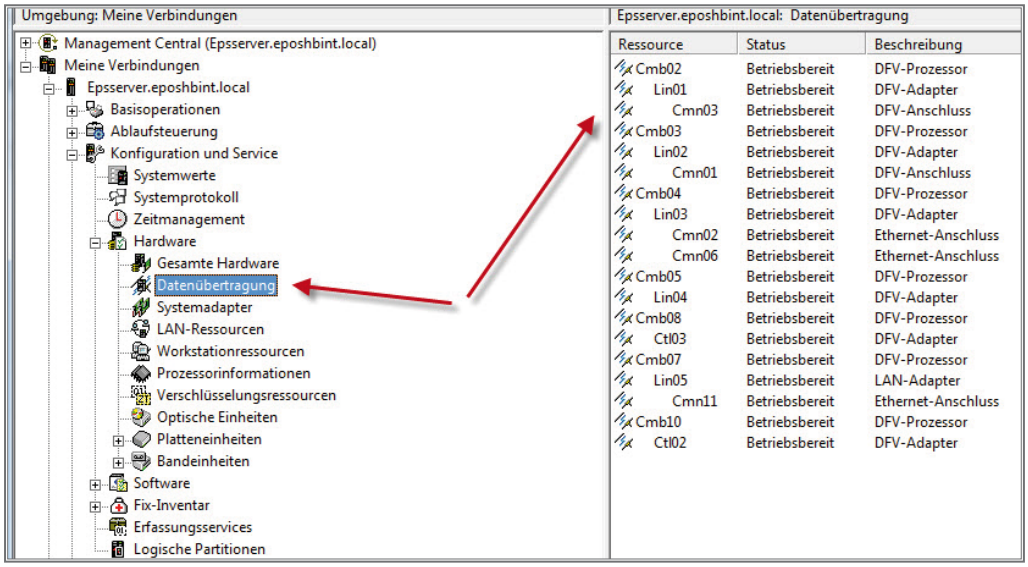
Detaildialog

Der Detaildialog ermöglicht die präzise Identifikation einer Hardwarekomponente.

7.8.7.2.1

Seite 2

Datenübertragungsressourcen



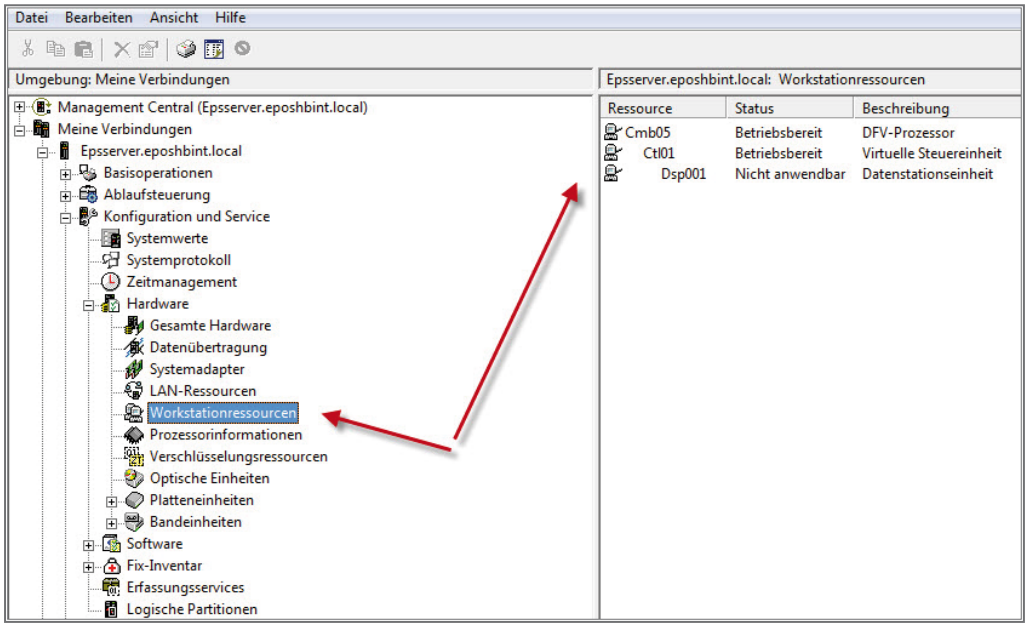
Datenübertragung

Die Funktion „Datenübertragungsressourcen“ zeigt eine Liste der lokalen Datenübertragungsressourcen auf Ihrem System sowie die zugeordneten Konfigurationsobjekte an – einschließlich aller Ein-/Ausgabeprozessoren, Ein-/Ausgabeadapter und Ports für die Datenübertragung, die auf dem System installiert sind. Für jede Ressource werden der Ressourcenname, der Status und die Beschreibung angezeigt.

7.8.7.2.2

Seite 2

Datenstationsressourcen



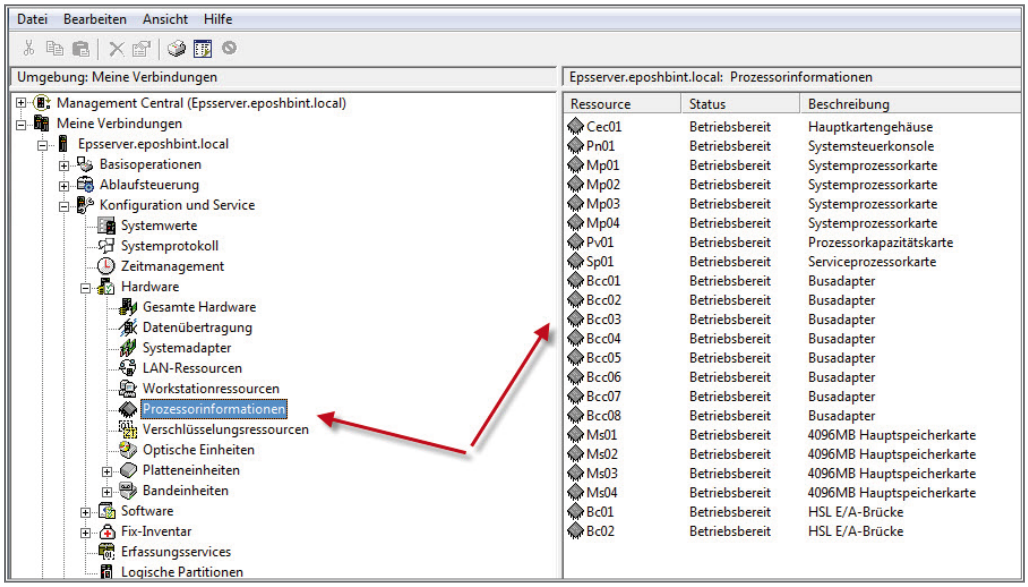
Datenstationsressourcen

Die Funktion „Datenstationsressourcen“ zeigt eine Liste aller Datenstationsprozessoren (Ein-/Ausgabeprozessoren) sowie lokalen Datenstationseinheiten und Steuereinheiten an, die in Ihrem System installiert sind. Für jede Ressource werden der Ressourcenname, der Status und die Beschreibung angezeigt.

7.8.7.2.3

Seite 2

Prozessorinformationen



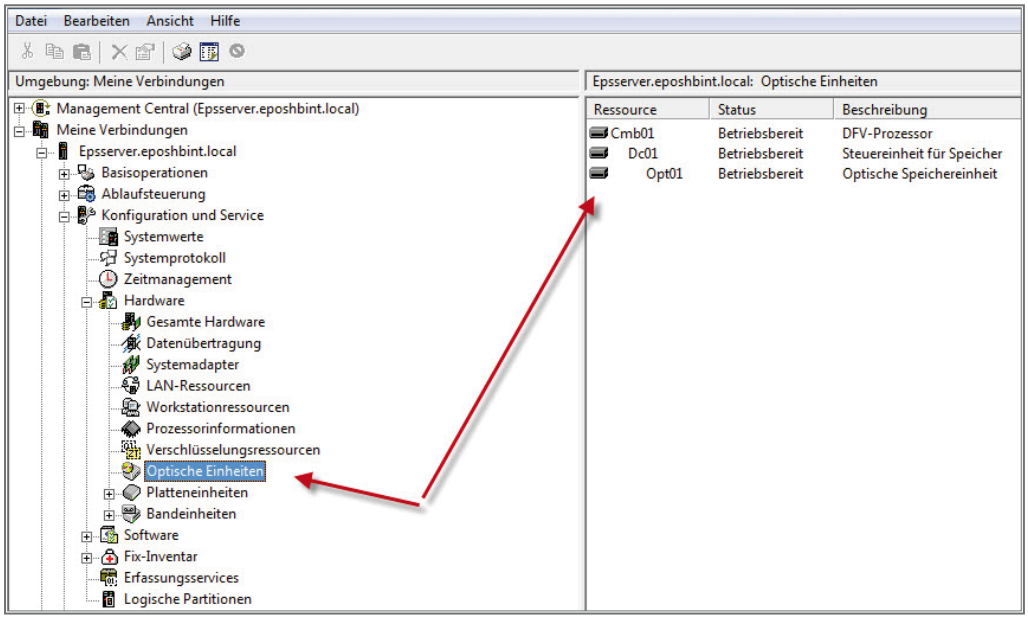
Prozessorinformationen

Die Funktion „Prozessorinformationen“ zeigt eine Liste der Prozessorressourcen Ihres Systems an. Diese Liste führt alle auf dem System installierten Prozessor- und Hauptspeicherkarten auf und enthält Einträge für die Verarbeitungseinheit selbst sowie für die Systemsteuerkonsole und das Hauptkartengehäuse oder das Systemeinheitengehäuse dieser Karten. Für jede Ressource werden der Ressourcenname, der Status und die Beschreibung angezeigt.

7.8.7.2.4

Seite 2

Optische Einheiten



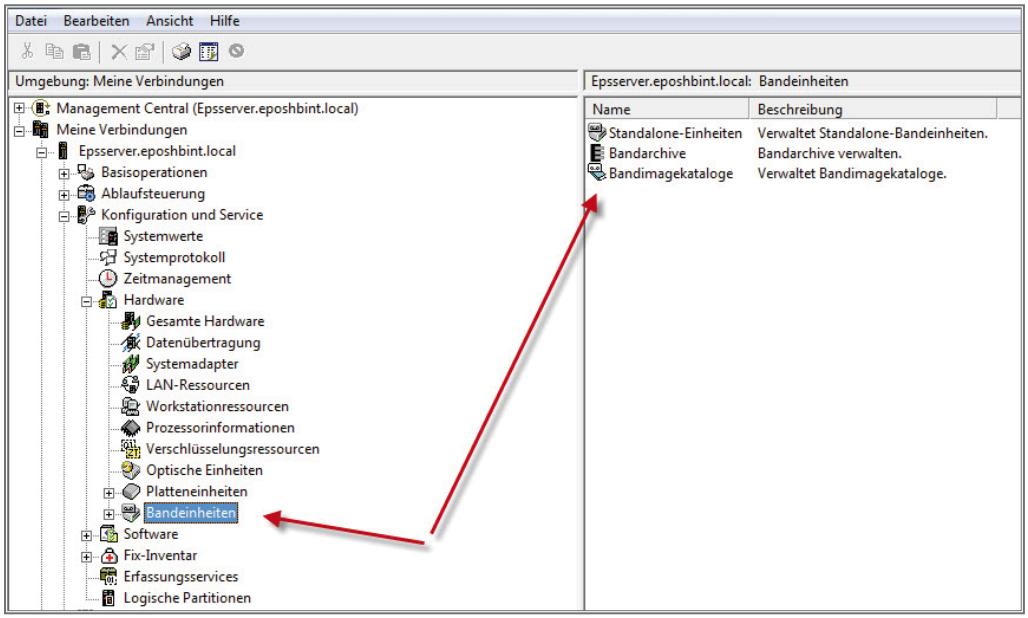
Optische Einheiten

Die Funktion „Optische Einheiten“ zeigt eine Liste aller optischen Einheiten, Steuereinheiten und Ein-/Ausgabeprozessoren Ihres Systems an. Eine optische Einheit wird verwendet, um Daten auf einer optischen Platte zu suchen, zu lesen und zu schreiben. Für jede optische Einheit werden der Ressourcenname, der Status und die Beschreibung angezeigt.

7.8.7.2.5

Seite 2

Bandeinheiten



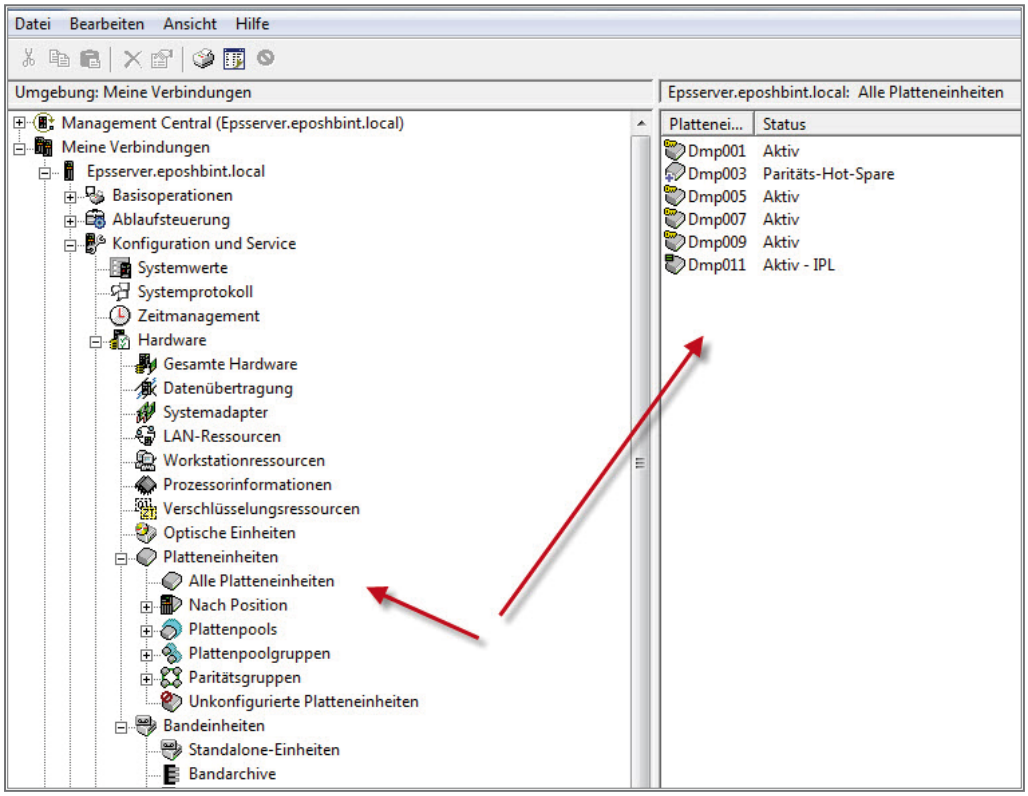
Bandeinheiten

Die Funktion „Bandeinheiten“ zeigt eine Liste aller Bandeinheiten, Steuereinheiten und Ein-/Ausgabeprozessoren Ihres Systems an. Eine Bandeinheit wird verwendet, um Daten auf Magnetbänder zu schreiben, Daten zu lesen. Für jede Bandeinheit werden der Ressourcenname, der Status und die Beschreibung angezeigt.

7.8.7.2.6

Seite 2

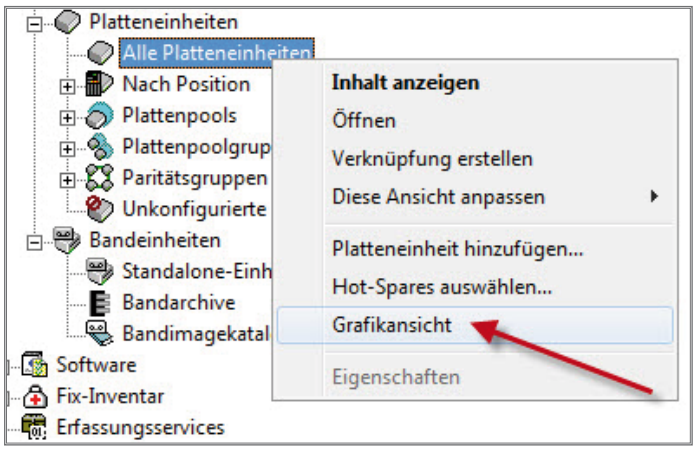
Platteneinheiten



Platteneinheiten

Die Funktion „Platteneinheiten“ ermöglicht das Aufrufen einer Liste aller Platteneinheiten Ihres Systems. Bei einer Platteneinheit werden Daten auf eine Speichereinheit mit Direktzugriff geschrieben und gelesen. Sie können die Liste der Platteneinheiten sortiert nach Namen, nach Standort oder nach Plattenpool anzeigen.

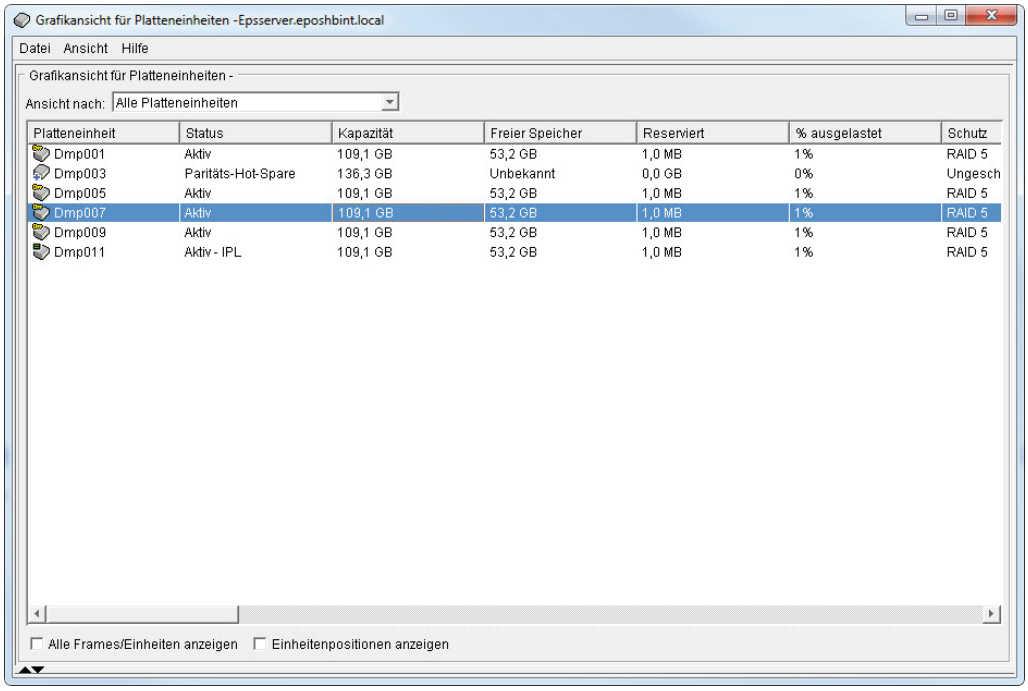
Mit Hilfe der rechten Maustaste gelangen Sie für jede Selektion in die Funktion „Grafikansicht“.



Grafikansicht markieren

7.8.7.2.7
Seite 2

Erweitern Sie den Funktionsknoten „Platteneinheiten“ und klicken Sie mit der rechten Maustaste auf „Alle Platteneinheiten“. Wählen Sie anschließend „Grafikansicht“ aus.

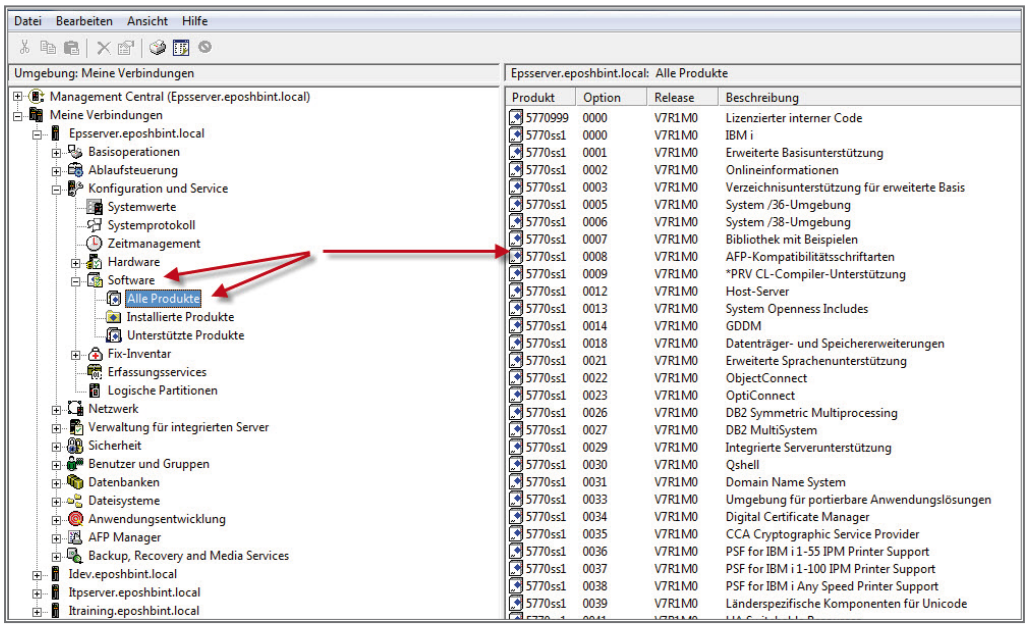


Grafikansicht der Platteneinheiten

Wählen Sie im Parameter „Ansicht nach“ den Parameterwert „Alle Platteneinheiten“ aus. Es wird Ihnen dann rechts daneben eine Liste aller im System befindlichen Platteneinheiten angezeigt.

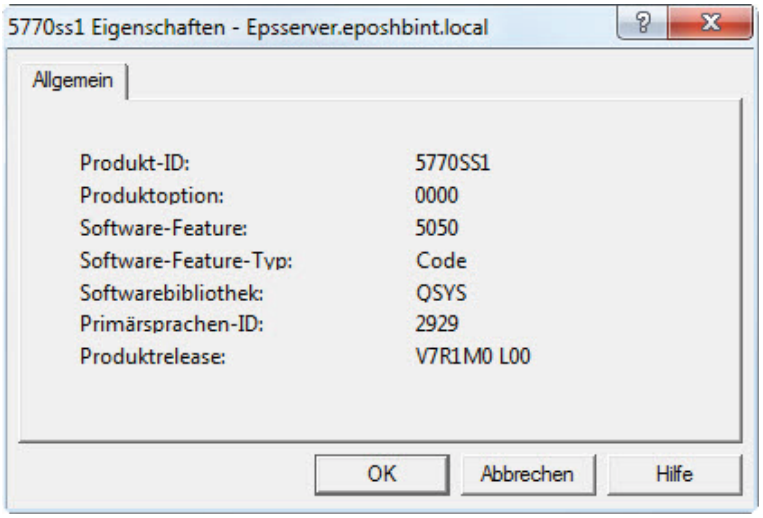
Software

Der Funktionsknoten „Software“ zeigt die auf dem System installierte Software an. Sie können sich hier unter anderem die allgemeinen Eigenschaften eines ausgewählten Softwareprodukts anzeigen lassen. Dazu gehören der Name und eine kurze Beschreibung des Produkts sowie weitere Informationen.



Installierte Software

Das Funktionsverzeichnis „Installierte Produkte“ zeigt eine Liste der zur Zeit auf Ihrem System installierten Softwareprodukte an. Sie können sich auch zusätzliche Informationen beziehungsweise Eigenschaften zu den Produkten anzeigen lassen, etwa die Produkt-ID, die Software-Feature-Nummer und den Software-Feature-Typ oder auch das Produkt-Release. Klicken Sie dazu doppelt auf eine bestimmte Lizenzsoftware-Komponente.



Eigenschaften

7.8.7.3
Seite 2

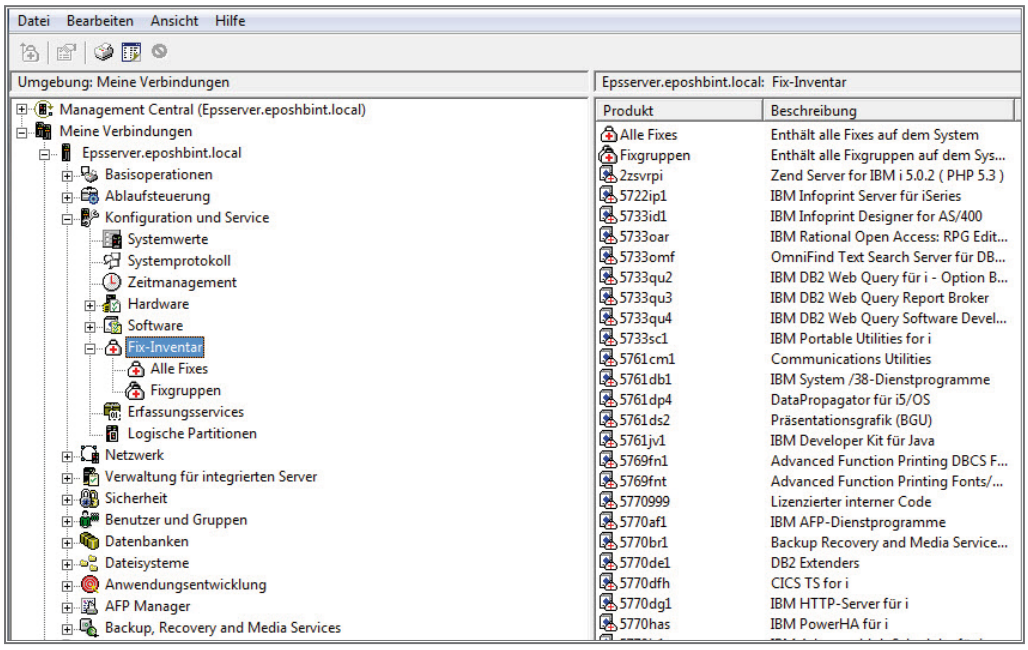
Fix-Inventar

Der Bereich „Fix-Inventar“ dient zur Behebung von Fehlern beziehungsweise potenziellen Fehlern eines oder mehrerer Lizenzprogramme.

Fixes (die in der Vergangenheit als vorläufige Programmkorrekturen (PTFs) bezeichnet wurden) können Probleme lösen, die vermutlich auf Softwarefehler zurückzuführen sind, oder sie können neue Funktionen zur Verfügung stellen. Fixes sind dazu gedacht, eines oder mehrere Objekte im Lizenzprogramm zu ersetzen.

Die Arbeit mit Fixes wurde durch grafische Assistenten vereinfacht. Mit dem Assistenten für „Vergleich und Aktualisierung“ können Sie beispielsweise eine Gruppe von Systemen automatisch mit einem Referenzsystem (Modellsystem) vergleichen lassen, die fehlenden Fixes finden und sie dann an die einzelnen Systeme senden und dort installieren.

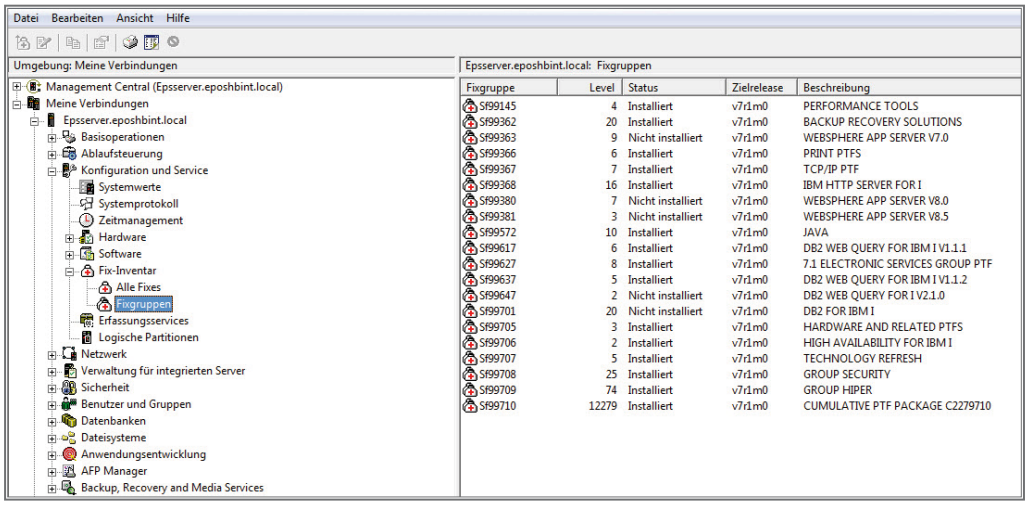
Die Liste mit dem Fix-Inventar zeigt alle Fixes und Produkte mit zugehörigen Fixes an. Zu jedem Fix können Sie sich Informationen wie ID, zugehöriges Produkt, Release oder Typ anzeigen lassen.



Fixes für Lizenzprogramme anzeigen

Ich markiere einmal den Funktionsknoten „Alle Fixes“. Auf der rechten Seite sehen Sie jetzt eine Liste aller Fixes (PTFs), die auf dem System bis zum Zeitpunkt der Anzeige installiert wurden.

7.8.7.4
Seite 2



Fix-Gruppen anzeigen

Damit jedoch eine Anzeige der installierten Fixes überhaupt möglich ist, muss vor der ersten Anzeige das sogenannte Inventar des Systems erfasst werden. Die Funktion „Inventar erfassen“ gehört zum Aufgabenbereich des Central Managements. Das bedeutet für Sie, dass der Management Central Server auf Ihrem System i gestartet worden sein muss. Sollte das nicht der Fall sein, so geben Sie auf der Befehlszeile eines Bildschirms den Befehl STRTCPSVR *MGTC ein.

Der Server besteht aus folgendem Job im Subsystem QSYSWRK:

- QYPSJSVR

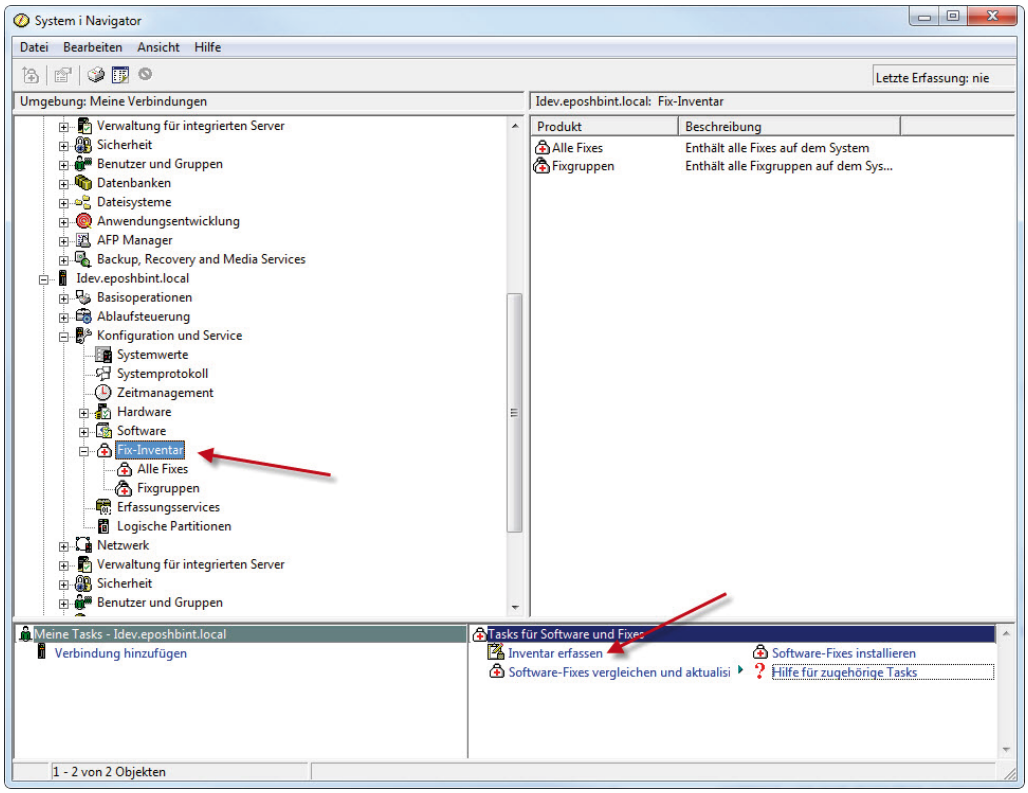
Sehen Sie diese beiden Jobs in besagtem Subsystem, dann läuft Ihr Management Central-Server. Gehen Sie gegebenenfalls in das Jobprotokoll dieser Jobs, sofern Sie bei den folgenden Tasks auf Fehlermeldungen stoßen.

Was muss also getan werden, damit die Fixes angezeigt werden? An dieser Stelle kommt jetzt das erste Mal das Taskpad des System i Navigator ins Spiel. Diese Leiste stellt uns für die Arbeit im Bereich Fixes einige Assistenten zur Verfügung:

- Inventar erfassen
- Assistenten für Vergleich und Aktualisierung von Fixes (Senden und Installieren von Fixes, Installieren und Deinstallieren von Fixes)
- permanentes Installieren von Fixes

Inventar erfassen

Da die Erfassung des Inventars die Voraussetzung für die Anzeige der Fixes ist, führe ich im Taskpad des System i Navigator einen Doppelklick auf die Funktion „Inventar erfassen“ aus.



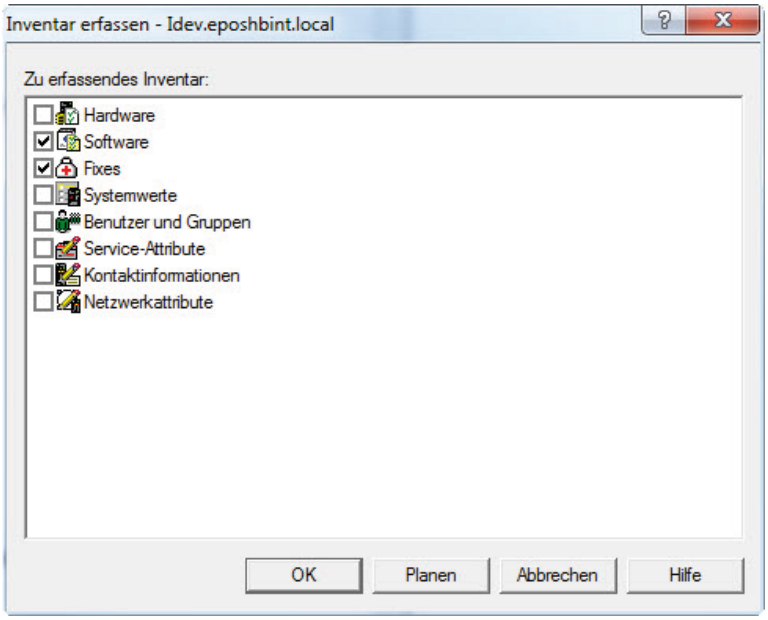
Inventar erfassen

Achtung!

Die hier gezeigten Wizards und Funktionen werden nur dann angezeigt, wenn Sie im System i Navigator die Funktion „Fixes“ oder eine der unterhalb von Fixes befindlichen Funktionen angeklickt haben!

7.8.7.4.1
Seite 2

Jetzt erscheint der Dialog „Inventar erfassen“.

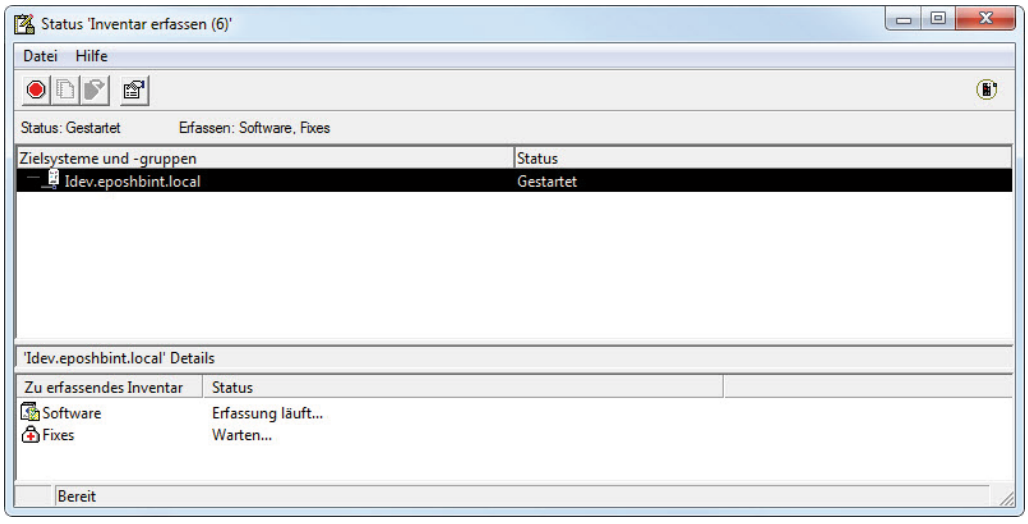


Inventar erfassen 1

In den beiden oben gezeigten Dialogen sehen Sie alle Bestandteile des sogenannten Inventars. Wichtig für den Moment ist, dass die Checkbox vor Fixes ein Häkchen aufweist.

Soll die Inventarisierung sofort und spontan gestartet werden, drücken Sie jetzt einfach auf „OK“.

Mit Hilfe des Dialogs „Planen“ kann die Inventarisierung jedoch auch regelmäßig geplant ausgeführt werden.



Inventar erfassen läuft

Danach erscheint ein Dialog, der Sie darüber informiert, dass die Funktion „Inventar erfassen“ läuft.

Achtung!

Die Gesamterfassung des Inventars ist keine kurzfristige Aktion. Je mehr Komponenten des Systems erfasst werden, desto länger läuft der Erfassungsprozess. Laufzeiten von mehreren Stunden sind keine Seltenheit. Legen Sie deshalb die Inventarerfassung in die Nachtverarbeitung.

7.8.7.4.1

Seite 4

Softwarefixes vergleichen und aktualisieren

Im Taskpad für Fixes steht Ihnen ein Wizard zur Verfügung, der Ihnen die Möglichkeit gibt, Fixes (PTFs) ausgehend von einem Referenzsystem im Netzwerk zu verteilen.

Ich unterhalte im System i Navigator eine Verbindung zum System „epsserver.eposhbint.local – 172.16.110.22“. Dieses System dient mir als Fixes-Referenzsystem, denn der PTF-Stand ist als aktuell und fehlerfrei zu betrachten.

Daneben unterhalte ich im System i Navigator eine weitere Verbindung zum System „itraining.eposhbint.local – 172.16.111.23“.

Achtung!

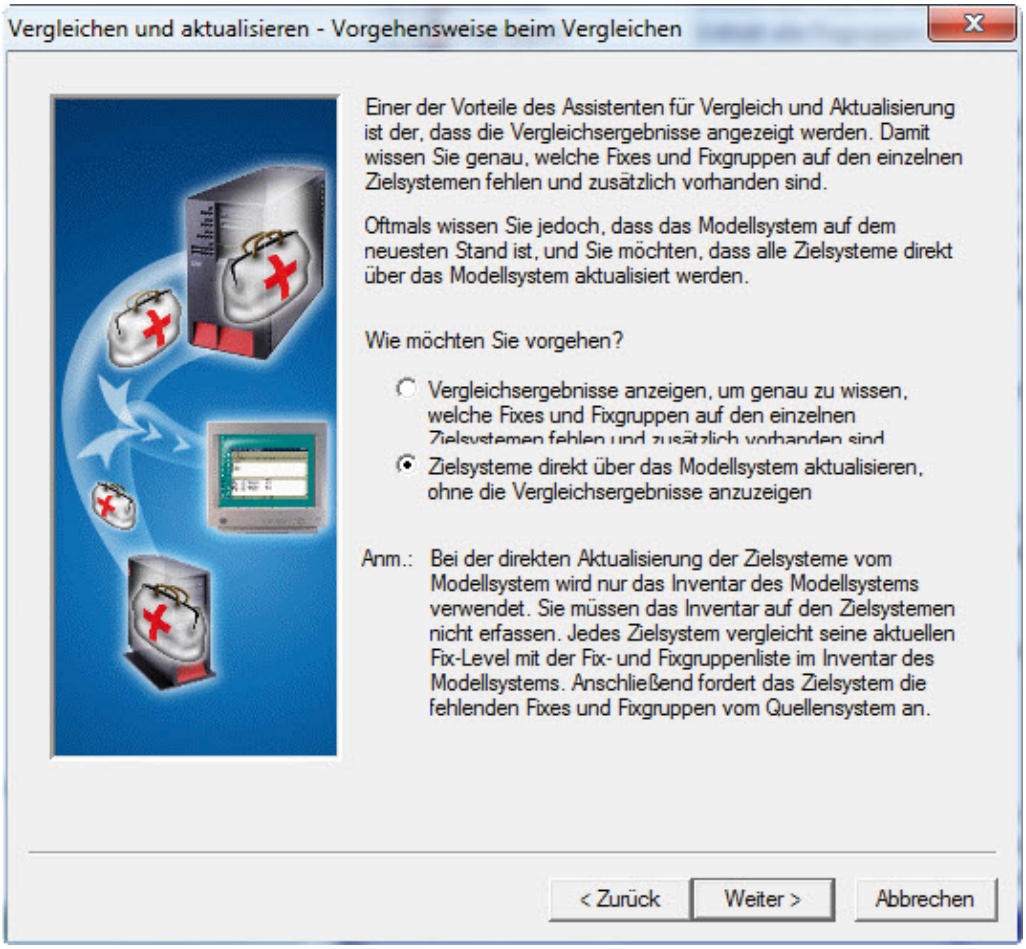
Auch diese Funktion gehört zum Management Central. Achten Sie darauf, dass der Management Central-Server gestartet wurde. Darüber hinaus müssen beide Systeme im Management Central des System i Navigator als sogenannte Endpunktsysteme in der gleichnamigen Management Central-Funktion eingetragen worden sein.

Zur Funktion „Softwarefixes vergleichen und aktualisieren“ gelange ich, indem ich den unten gezeigten Wizard anwähle.

Wizard: Softwarefixes vergleichen…

7.8.7.4.2
Seite 2

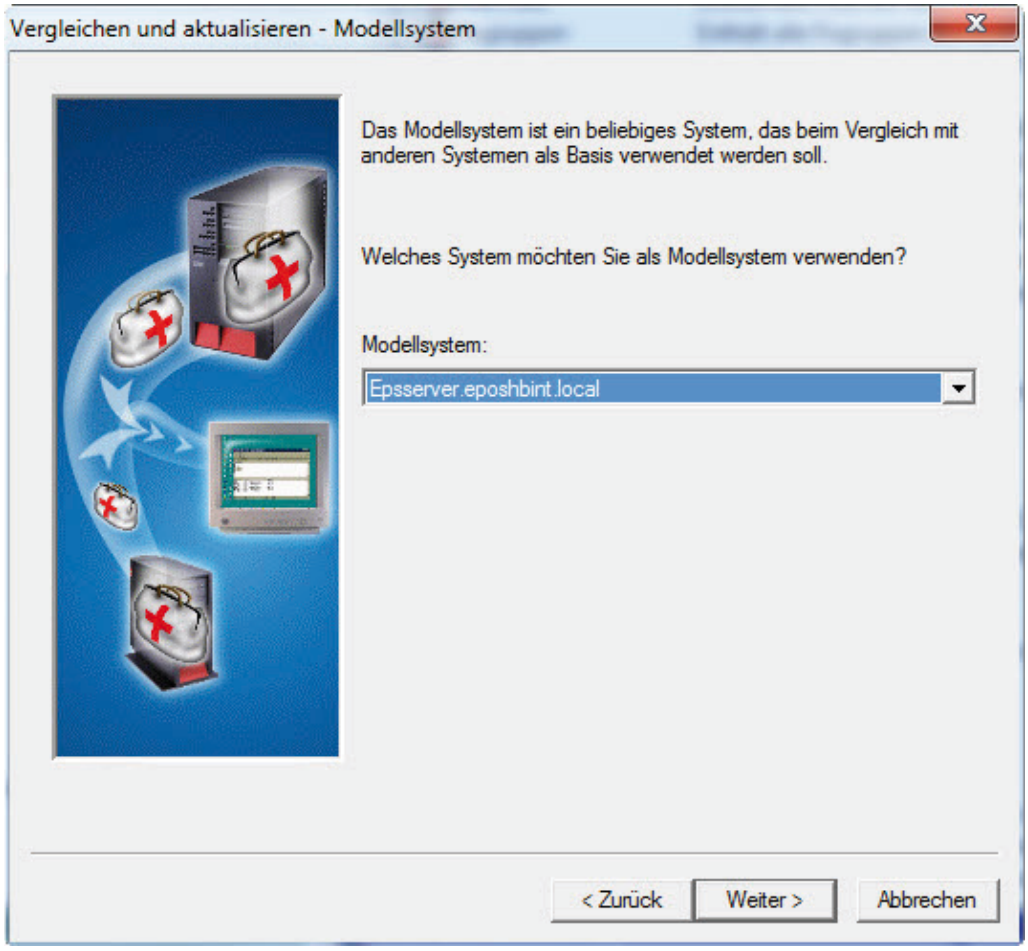
Nach dem obligatorischen Willkommensgruß erscheint der folgende Dialog:



Vorgehensweise beim Vergleichen

Hier wird abgefragt, ob Sie lediglich die PTF-Stände mehrerer Systeme vergleichen möchten oder ob ein PTF sofort auf das Zielsystem übertragen werden sollen. Folgende Möglichkeiten sind gegeben:

- Nur Vergleichsergebnisse anzeigen. Sie beabsichtigen nicht, eine Aktualisierung durchzuführen. Sie wollen lediglich wissen, ob und für welche Produkte ein Ungleichstand der Fixes auf den Systemen gegeben ist.
- Fehlende Fixes senden: Die Fixes werden entsprechend den vorherigen Angaben an das Zielsystem gesendet. Der Administrator des Zielsystems muss die PTF-Installation dann selbst erledigen.
- Fehlende Fixes senden und installieren: Die Fixes werden entsprechend den vorherigen Angaben an das Zielsystem gesendet und installiert. Der Administrator des Zielsystems muss nicht aktiv werden.

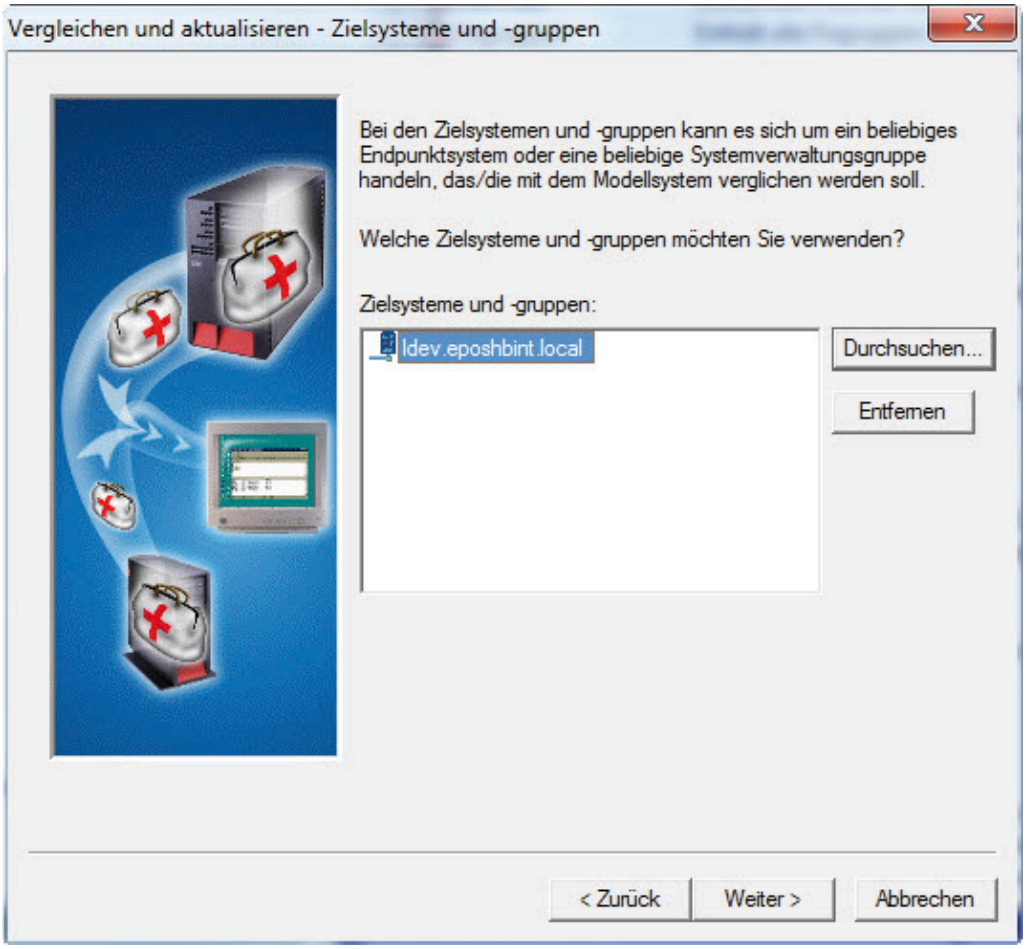


Modellsystem

In diesem Dialog wird das Referenzsystem für die Fixes-Aktualisierung abgefragt. Das Referenzsystem wird hier als Modellsystem bezeichnet. Ich trage dazu die Adresse „epsserver.eposhbint.local“ ein. Sollte die Adresse des Referenzsystems im Pull-Down-Menü nicht auftauchen, so ist Ihr System im Management Central nicht als Endpunktsystem eingepflegt worden.

7.8.7.4.2
Seite 4

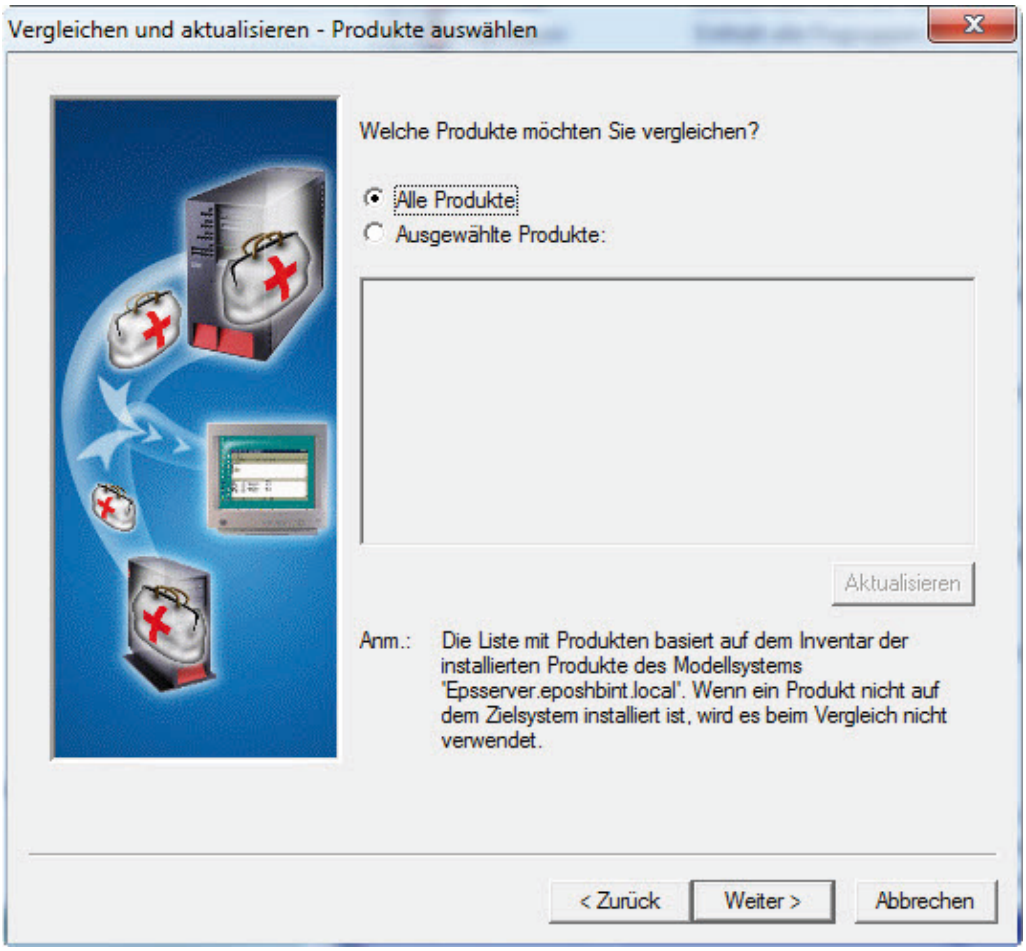
Im nächsten Dialog ist das Zielsystem für das Fixes-Update einzutragen.



Zielsystem

In meinem Beispiel steht hier automatisch die Adresse „epsserver.eposhbint.local“. Diese Adresse entferne ich und ersetze Sie durch „idev.eposhbint.local“. Sollten Sie hier keinen Namen finden, so wurde das beabsichtigte Zielsystem nicht als Endpunktsystem im Management Central deklariert.

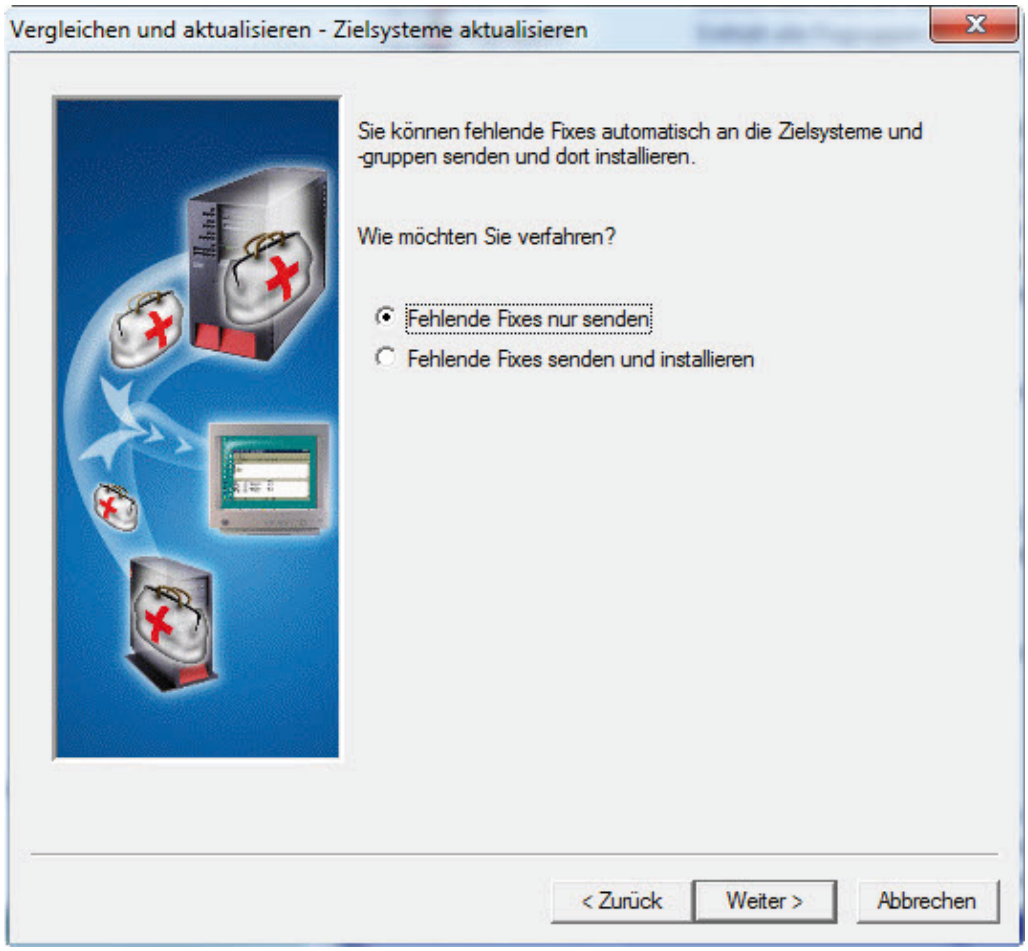
Im nächsten Dialog wird abgefragt, welche Produkte in den Aktualisierungsprozess aufgenommen werden sollen.



Produkte auswählen

7.8.7.4.2
Seite 6

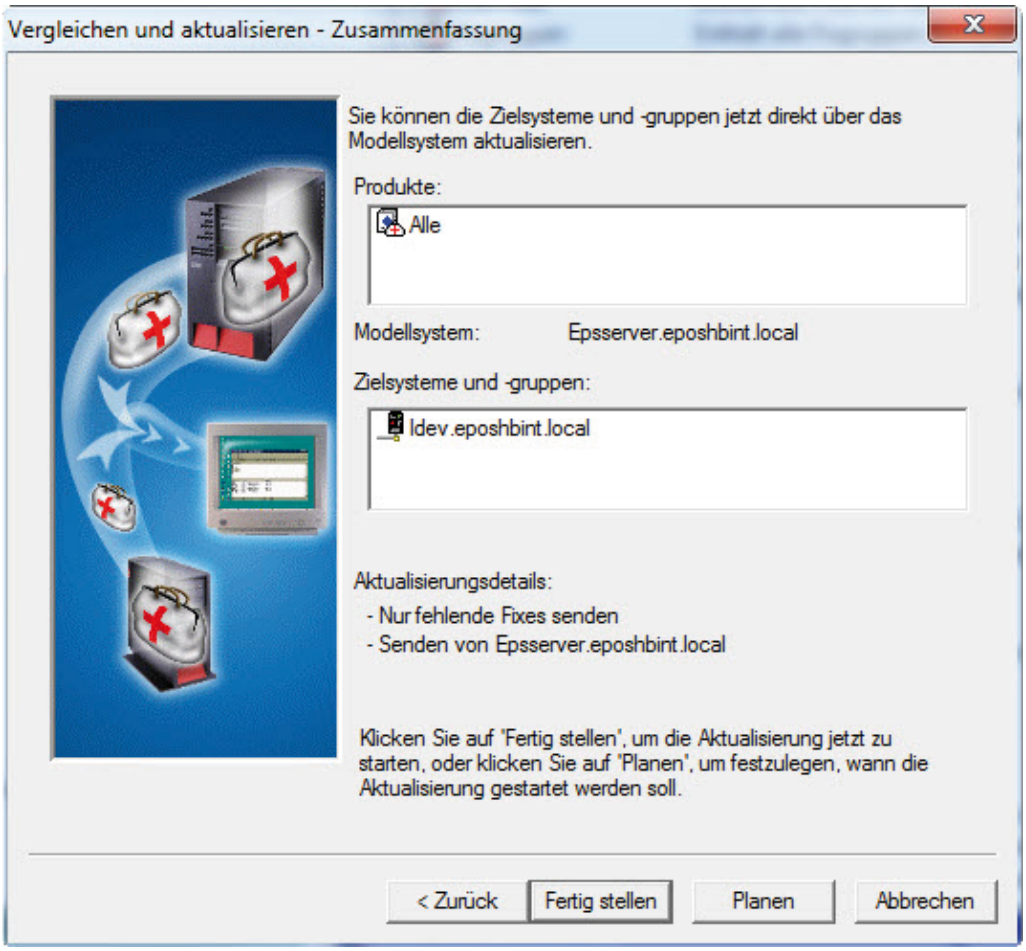
Aktivieren Sie den Parameter „Ausgewählte Produkte“, so erscheint in dem darunter befindlichen Feld eine Liste der auf dem Modellsystem installierten Lizenzprogramme. Durch Anchecken werden das oder die entsprechenden Produkte in den Prozess aufgenommen.



Fixtypen

Im oben gezeigten Dialog wird bestimmt, ob fehlende Fixes vom Modellsystem an das Zielsystem nur gesendet oder dort auch installiert werden sollen.

Im letzten, nachstehend aufgeführten Dialog wird dann der Umfang der Aktualisierung definiert.



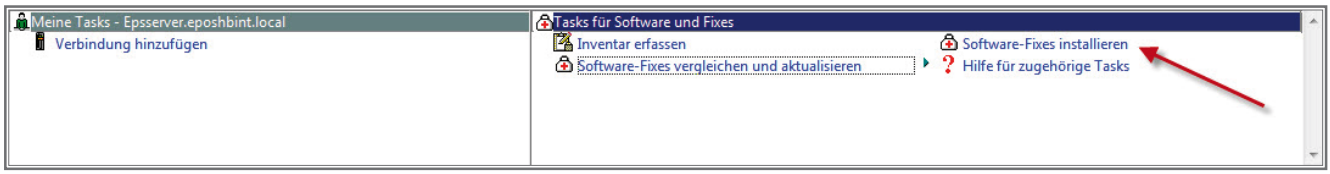
Vorgang fertigstellen

7.8.7.4.2

Seite 8

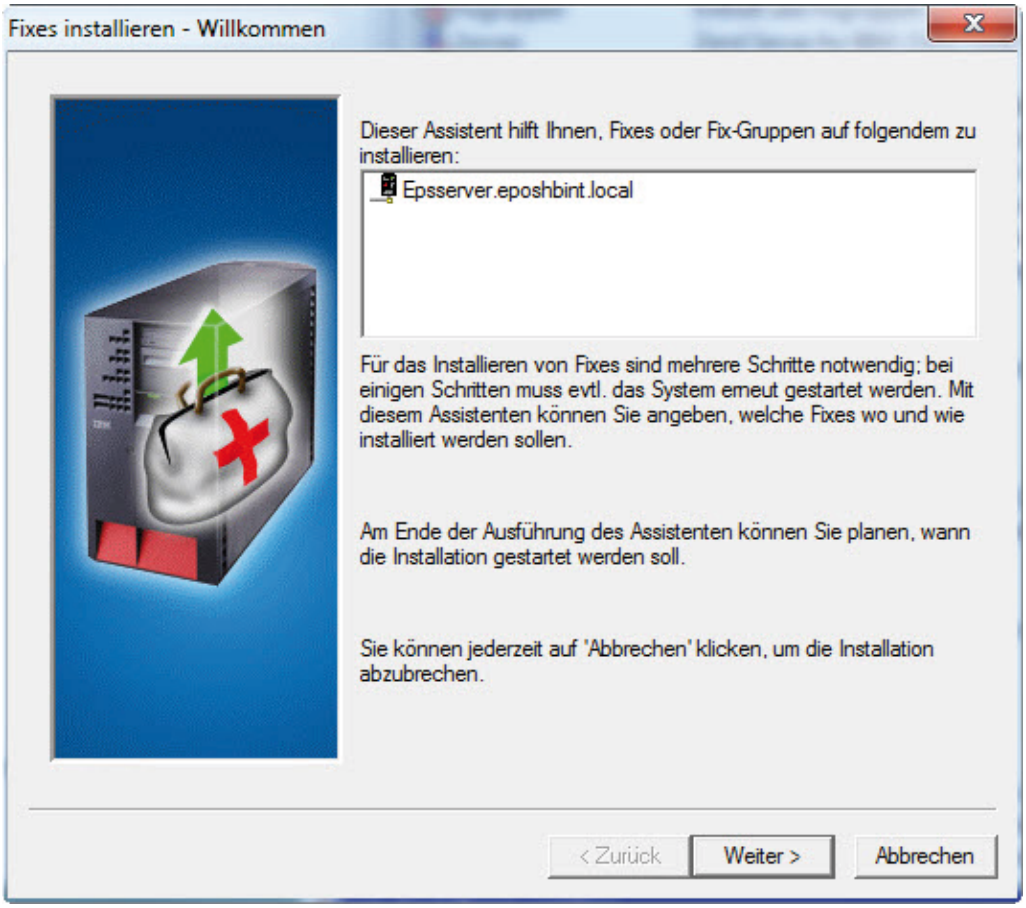
Softwarefixes installieren

Der dritte Wizard, der im Taskpad der Funktion „Fixes“ bereitsteht, steuert die Installation neuer PTFs auf einem System.



Fixes installieren

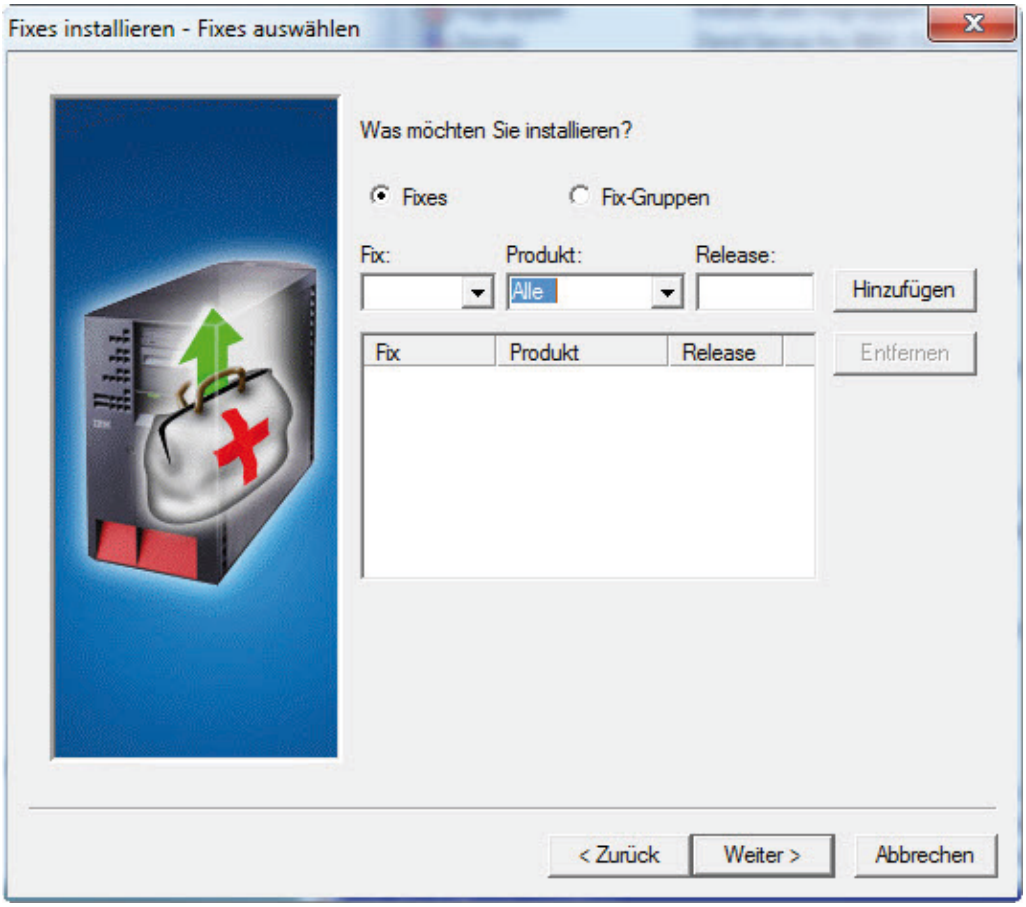
Nach dem Start des Wizards erscheint folgender Dialog:



Willkommen

7.8.7.4.3
Seite 2

In diesem Dialog sehen Sie lediglich die System i Navigator-Verbindung zum lokalen System. Nach der Bestätigung durch „Weiter“ wird folgendes Fenster angezeigt:

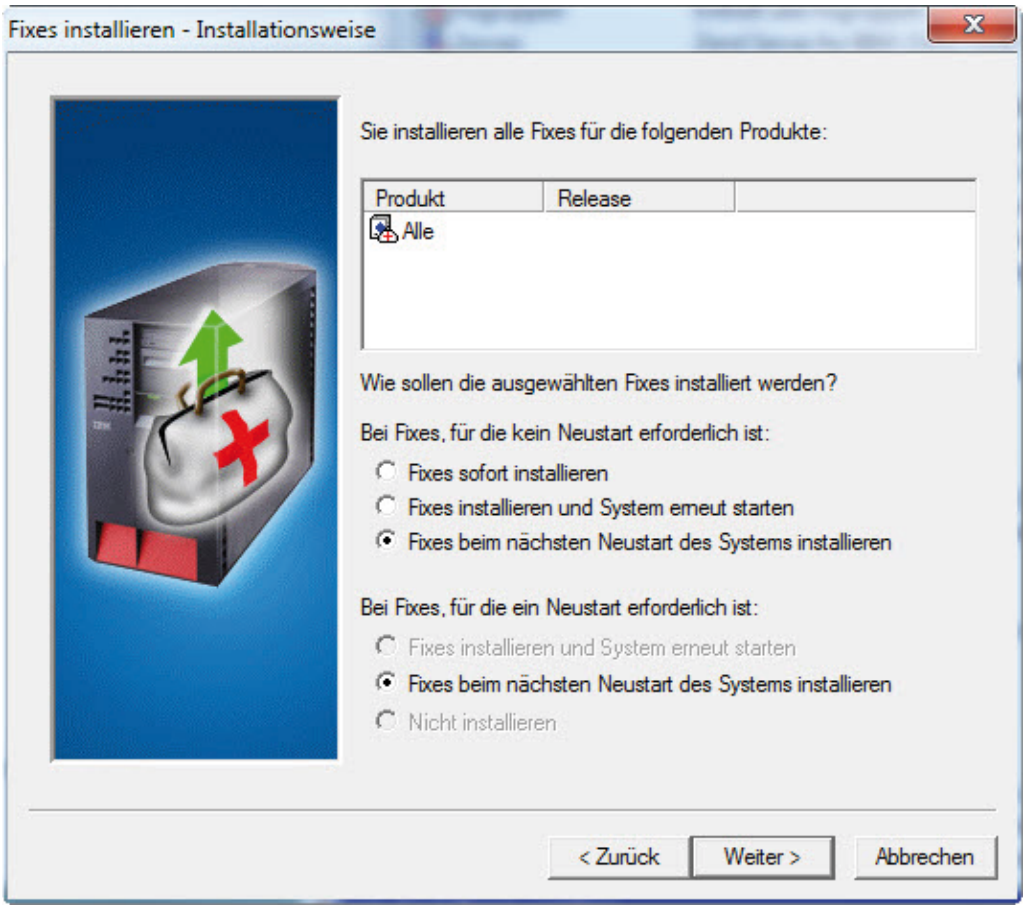


Fixes auswählen

Im oben gezeigten Dialog habe ich die Möglichkeit, sowohl die Fixes als auch die Produkte auszuwählen, für die eine Fixes-Installation vorgenommen werden soll.

Die Auswahl muss mit „Hinzufügen“ bestätigt werden.

Anschließend muss die Art und Weise der Fixes-Installation bestimmt werden.



Installationsweise

Letztendlich geht es beim oben gezeigten Dialog darum, ob das System gegebenenfalls beendet und neu gestartet werden soll.

Danach wird lediglich eine Zusammenfassung des bisher Definierten angezeigt.

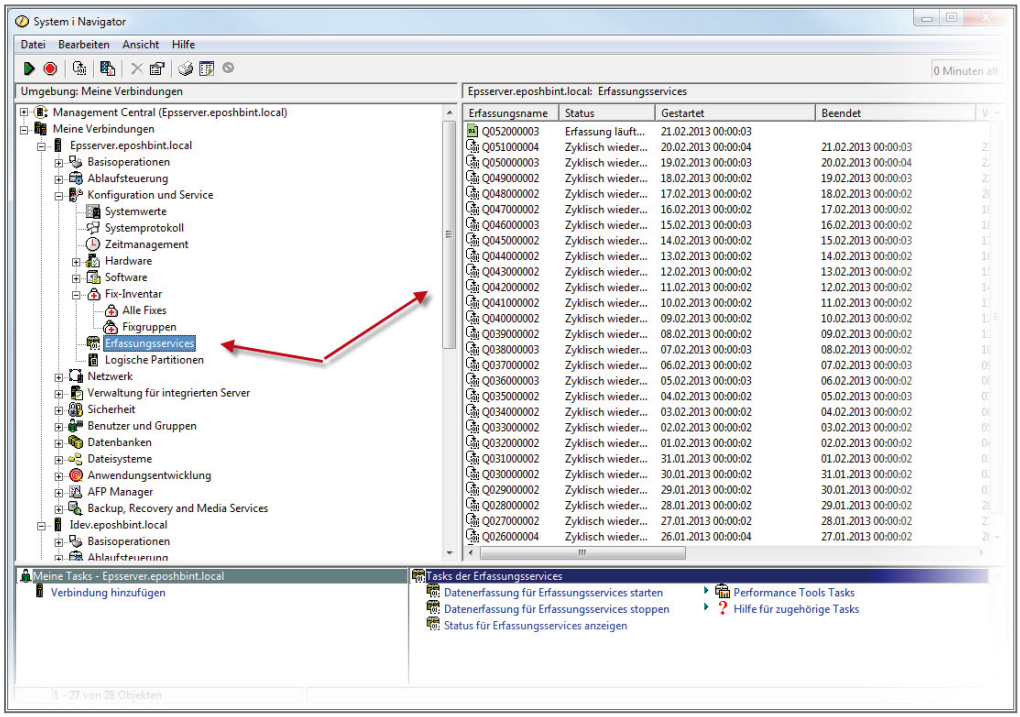
7.8.7.4.3

Seite 4

Erfassungsservice

Auch die Erfassungsservices sind eine Management Central-Funktion, die nur dann realisiert werden kann, wenn der entsprechende Server gestartet wurde.

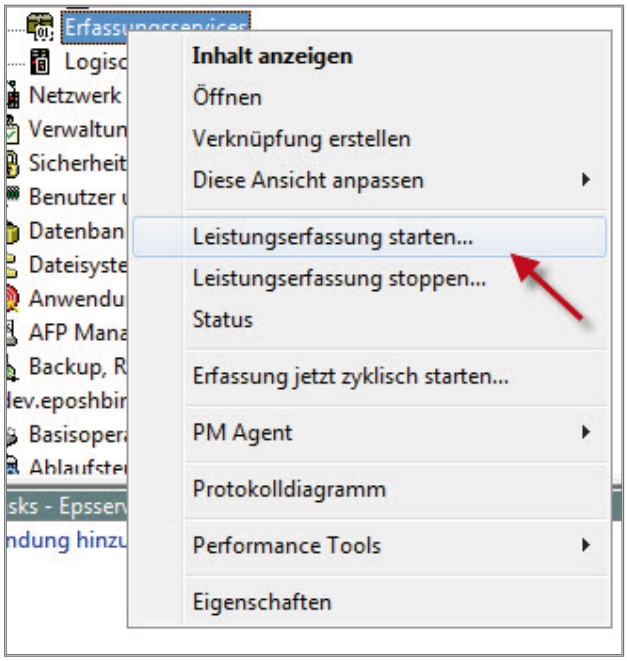
Die Erfassungsservices steuern die Erfassung von Leistungsdaten auf einem oder mehreren System i-Systemen. Die erfassten Daten werden sodann in Datenbankdateien geschrieben, um sie später mit Hilfe des Lizenzprogramms „Performance Tools for SYSTEM I“, dem System i Navigator oder anderen Anwendungen für Leistungsberichte zu analysieren.



Die Erfassungsservices

Indem Sie mit der rechten Maustaste den Funktionsknoten „Erfassungsservices“ markieren, erhalten Sie Zugriff auf das Kontextmenü für diese Funktion.

7.8.7.5
Seite 2

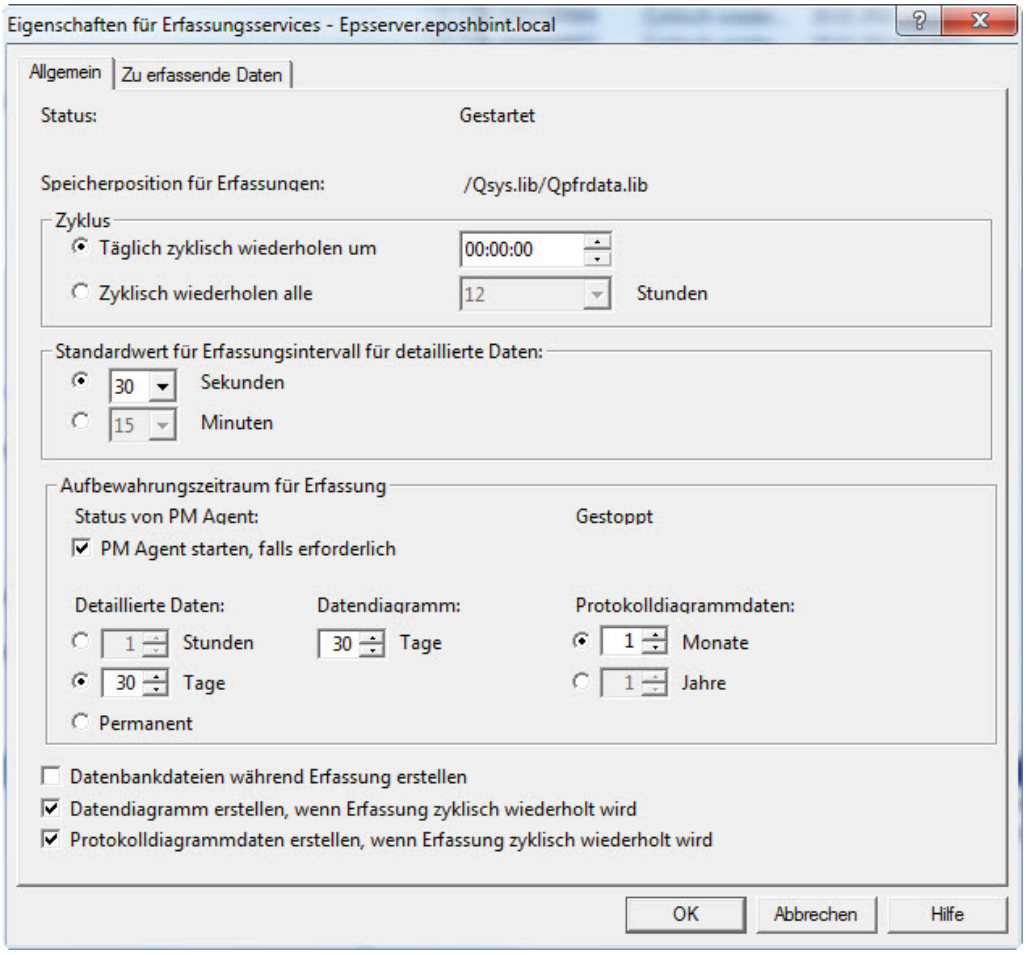


Kontextmenü

Starten Sie die Erfassungsservices, so öffnet sich ein Dialog, mit dem Sie die deren Arbeit steuern können.

Folgende Parameter- bzw. Parametergruppen sind gegeben:

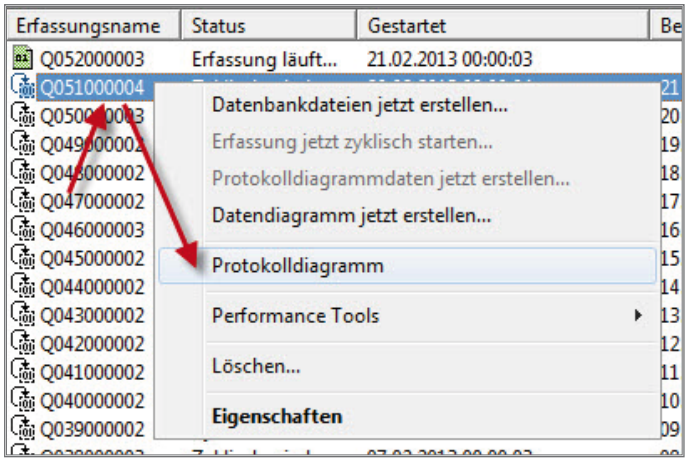
- Zyklisch wiederholen...: Sollen die Erfassungsservices regelmäßig gestartet werden?
- Zyklus: Zeitparameter der zyklischen Ausführung definieren.
- Standardwert...: In welchen Zeitintervallen sollen Detaildaten im System erhoben werden?
- Aufbewahrungszeitraum: Wie lange sollen bestimmte Datengruppen im System gespeichert werden?



Startparameter

Sind die Erfassungsservices gestartet, so werden die Performancedaten in diverse Dateien der Bibliothek QMPGDATA geschrieben.

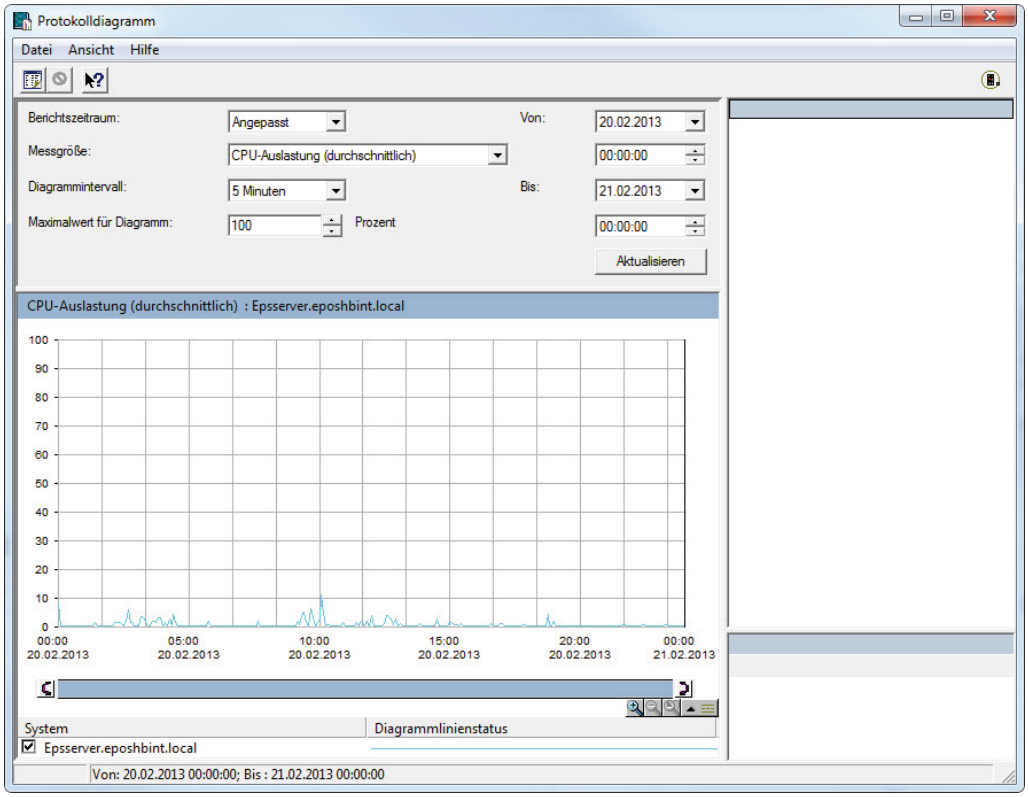
Um eine historische Analyse der erfassten Daten zu ermöglichen, öffne ich einmal mehr ein Kontextmenü und wähle den Punkt „Protokolldiagramm“ an.



Protokolldiagramm anfordern

7.8.7.5
Seite 4

Daraufhin öffnet sich folgender Dialog:



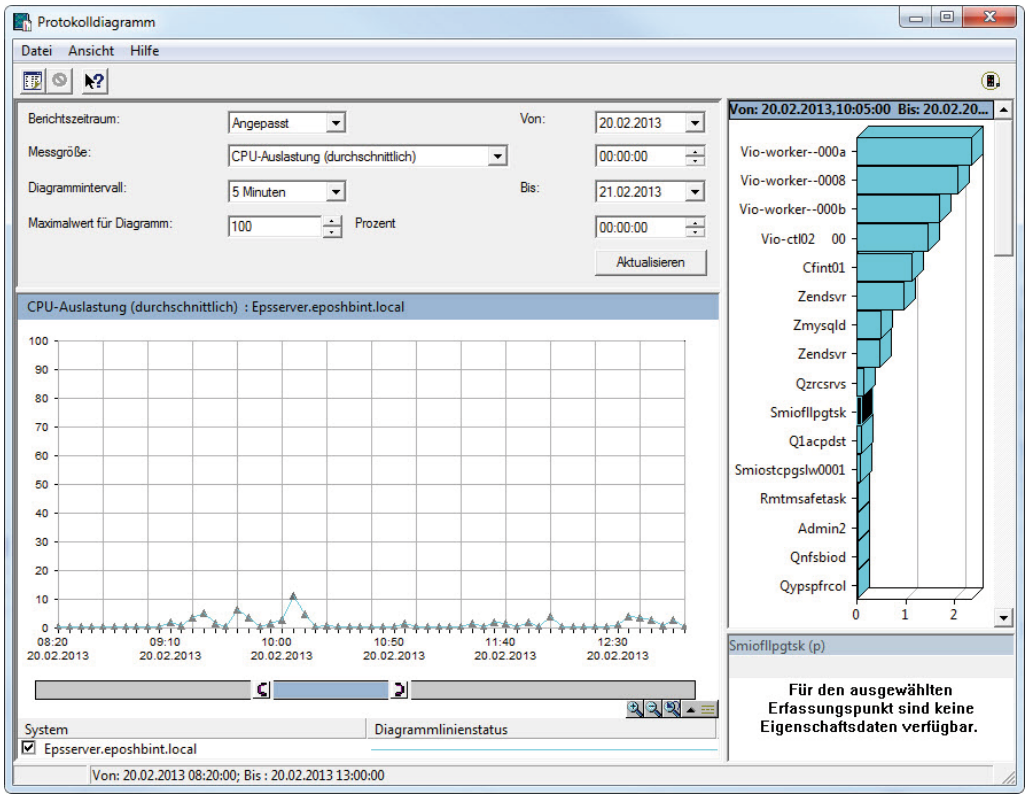
Protokolldiagramm 1

Im oberen linken Bereich der Darstellung sehen Sie Eingabeparameter, die folgende Anzeigebereiche steuern:

- Berichtszeitraum
- Messgröße: überwachte Hardwarekomponente
- Dialogintervall
- Maximalwerte im Diagramm
- Darstellung der Daten: Von- und Bis-Werte

Darunter sehen Sie eine Matrix, die gegebenenfalls die Messpunkte für die überwachte technische Komponente anzeigt.

Nachdem ich auf den Knopf „Aktualisieren“ gedrückt habe, wird die Anzeige mit Daten populiert.



Protokolldiagramm 2

Mit der Schiebeleiste unter der Tabelle lässt sich der betrachtete Zeitraum eingrenzen.

7.8.7.5
Seite 6

Logische Partitionen

Die Funktion „Logische Partitionen“ (LPAR) ermöglicht es Benutzern eines Systems, dieses Einzelsystem in mehrere Systeme aufzuspalten. Jede logische Partition arbeitet als unabhängiges logisches System. Sie können damit Systemvoraussetzungen für Mehrmaschinenbetrieb mit einer einzelnen Maschine erfüllen, etwa um eine Server-Konsolidierung, die Konsolidierung von Geschäftseinheiten, die Aufteilung in Produktions- und Testumgebungen sowie integrierte Cluster zu erreichen.

Eine Diskussion dieser Funktionen setzt entsprechendes Fachwissen bezüglich LPAR voraus. Eine Erläuterung hierzu würde den Rahmen dieses Buches allerdings erheblich sprengen.

Diese Funktion ist nur dann aktiv, wenn keine HMC für die Verwaltung von logischen Partitionen verwendet wird.

7.8.7.6
Seite 2

8 Kapitel

Sicherung und Wiederherstellung

Sicherung und Wiederherstellung

Der Betrieb einer IT Landschaft, die Verfügbarkeit der Systeme und der permanente Zugriff auf die Daten eines Unternehmens sind ein wesentlicher Bereich der Anforderungen an IT Abteilungen und die von diesen betreuten Systemen.

Das IBM i stellt basierend auf der einzigartigen Architektur eine ideale Basis für den sicheren und störungsfreien Betrieb der Anwendungen und bildet damit das Rückgrat einer stabilen Systemumgebung. Aber so gut ein System auch aufgebaut ist, so sehr trifft unter Umständen der Zufall den Betreiber, wenn wider Erwarten das System einmal nicht verfügbar sein sollte. Datenverlust ist heute in einem Unternehmen ein Katastrophenfall. Datenschutz spielt aus diesem Grund eine immens wichtige Rolle.

Das Absichern des Systems gegen einen, wie auch immer gearteten, Datenverlust ist eine der Hauptaufgaben der IT Verantwortlichen in Zusammenarbeit mit den Administratoren.

Von Hause aus bietet das IBM i unterschiedliche Möglichkeiten der Sicherstellung der Sicherung von Daten. Die Datensicherungen und die zugehörigen Wiederherstellungsverfahren sind dabei in der Basis des Betriebssystems integriert und können ohne die Installation zusätzlicher Lizenzprogramme oder Funktionen genutzt und eingesetzt werden.

Datensicherungen sind klassisch in Form von Sicherungen auf externen Datenträgern möglich. Dabei kommen sowohl sogenannte interne Bandlaufwerke als auch externe Bandstationen zum Einsatz. Die Form der genutzten Bandlaufwerke (und damit verbunden auch der verwendeten Bänder) richtet sich nach unterschiedlichen Kriterien wie zum Beispiel:

- Menge der zu sichernden Daten
- Für die Datensicherung zur Verfügung stehende Zeit
- Art der zu sichernden Daten
- Form des Zugriffs und die Verfügbarkeit der zu sichernden Daten
- Anforderungen an die Wiederherstellbarkeit
- Gesetzliche Anforderungen
- Sicherungsrichtlinien des Unternehmens
- Komplexität der Datensicherungen
- Komplexität der zu verwaltenden Bänder (Anzahl der Bänder)
- Anzahl der für die Datensicherung notwendigen Bänder

Von Hause aus bietet das IBM i integrierte Datensicherungswerkzeuge, mit denen man in der Lage ist, das gesamte System auf externen Datenträgern zu sichern. Diese Sicherungsmöglichkeiten können über Menüs oder die zugehörigen Befehle ausgeführt werden.

Welches Sicherungsverfahren letztlich eingesetzt wird, richtet sich nach speziellen Überlegungen, welche zusammen mit den zuvor genannten Kriterien abgewägt werden müssen. Dabei sollte man sich vor der Entscheidung pro/contra eines Sicherungsverfahrens unter anderem die folgenden Fragen stellen:

- In welcher Zeit muss die Datensicherung abgeschlossen sein (Sicherungszeitfenster)?
- Welches sind die kritischen Daten, die täglich gesichert werden müssen?
- Wann sind welche Daten zu sichern (täglich, wöchentlich, monatlich, sporadisch)?
- Wie lange darf die Wiederherstellung dauern?
- Steht das System in einem eingeschränkten Zustand zur Verfügung oder muss die Datensicherung erfolgen, während mit den zu sichernden Informationen gearbeitet wird?
- Ist die Journalaufzeichnung auf dem System aktiv?
- Ist das System in eine Hochverfügbarkeitslösung integriert?
- Werden nur native IBM i Daten gesichert oder müssen beispielsweise auch integrierte Server mit gesichert werden?
- Wird das System lokal oder zentral gesichert (Netzwerksicherung)?

Wenn diese grundsätzlichen Fragen beantwortet sind, kann mit der Einrichtung und Ausführung der Datensicherung begonnen werden.

Aber halt: ein wichtiger Punkt muss zudem noch geklärt werden:

Um welche Form von Daten handelt es sich bei den zu sichernden Informationen?

Diese Frage steht in einem direkten Zusammenhang mit der Architektur des Systems i und der zu sichernden Daten.

Die Grobdifferenzierung ist an dieser Stelle getrennt nach diesen Bereichen durchzuführen:

- Systemdaten
- Benutzerdaten

20. Ergänzung 9/2009

Die verschiedenen Befehle, die auf dem IBM i zur Durchführung der Datensicherung eingesetzt werden können, sind zum Teil nur für Benutzerdaten, zum Teil nur für Systemdaten einsetzbar. Deshalb bestimmen diese beiden Teilklassifizierungen die möglichen Datensicherungsbefehle bzw. auch die Datensicherungsverfahren. Damit einhergehend ist auch die Entscheidung, wann die Daten im Einzelnen gesichert werden können. Während Benutzerdaten meist dann gesichert werden können, wenn diese zum Teil von Anwendungen oder Anwender genutzt werden, müssen die Sicherungen von Systemdaten meist in einem eingeschränkten Zustand erfolgen – in einem solchen Fall darf kein Anwender mit dem System arbeiten. Diese Sachverhalte werden wir uns im weiteren Verlauf noch genauer anschauen.

Als Folge dieser Fragestellungen kommen meist Datensicherungsverfahren zum Einsatz, welche eine tägliche, wöchentliche, monatliche, zyklische und individuelle Sicherung nutzen. Die beiden nachfolgenden Tabellen enthalten die Sicherungsbereiche, die typisch für tägliche und wöchentliche Sicherungen sind.

Beschreibung des Systembestandteils	Wann Änderungen auftreten
Sicherheitsinformationen (Benutzerprofile, persönliche Berechtigungen, Berechtigungslisten)	Regelmäßig: Wenn neue Benutzer und Objekte hinzugefügt werden oder Berechtigungen sich ändern
Konfigurationsobjekte in QSYS	Regelmäßig: Wenn Einheitenbeschreibungen hinzugefügt oder geändert werden
Von IBM gelieferte Bibliotheken, die Benutzerdaten enthalten (QGPL, QUSRSYS)	Regelmäßig
Benutzerbibliotheken, die Benutzerdaten und Programme enthalten	Regelmäßig
Ordner und Dokumente	Regelmäßig, wenn Sie diese Objekte verwenden
Verteilungsoperationen	Regelmäßig, wenn Sie die Verteilungsfunktion verwenden
Benutzerverzeichnisse	Regelmäßig

Tabelle 1: Täglich zu sichernde Daten

Beschreibung des Systembestandteils	Wann Änderungen auftreten
Lizenzierter interner Code	PTFs oder ein neues Release des Betriebssystems
Betriebssystemobjekte in der Bibliothek QSYS	PTFs oder ein neues Release des Betriebssystems
Optionale Bibliotheken von Betriebssystem IBM OS/400 (QHLPSYS, QUSRTOOL)	PTFs oder ein neues Release des Betriebssystems
Lizenzprogrammbibliotheken (QRPG, QCBL, Qxxxx)	Änderungen an Lizenzprogrammen
Lizenzprogrammordner (Qxxxxxxx)	Änderungen an Lizenzprogrammen
Lizenzprogrammverzeichnisse (/QIBM/ProdData, /QOpenSys/QIBM/ProdData)	Änderungen an Lizenzprogrammen

Tabelle 2: Wöchentlich zu sichernde Daten

Achtung!

Es ist nicht nur ein Spruch: „Jede Datensicherung ist so gut wie die damit mögliche Wiederherstellung." Bei jeder Überlegung hinsichtlich der zu sichernden Informationen eines IT Systems muss natürlich auch zwangsläufig die Wiederherstellung bedacht werden.

Wann und wie können die Informationen im Notfall wieder hergestellt werden? Wie viel Zeit steht zur Verfügung? Welche Voraussetzungen müssen für eine Wiederherstellung gegeben sein?

Dies sind nur einige Fragen, die in diesem Zusammenhang zu stellen sind. Letztlich gilt aber: ein gut getestetes Sicherungsverfahren beinhaltet auch ein gut getestetes Wiederherstellungsverfahren!

Jedes IT System muss im Hinblick auf die Verfügbarkeit der Daten und Anwendungen und auch deren Wiederherstellung gewappnet sein. Sicherungsverfahren sind deshalb genauso akribisch einzurichten und in deren Ausführung einzuhalten wie Wiederherstellungspläne und getestete Wiederherstellungsverfahren.

In diesem Kapitel finden Sie Informationen und Tipps, um Ihr IBM i richtig sichern und letztlich auch im Bedarfsfall wieder herstellen zu können.

20. Ergänzung 9/2009

8

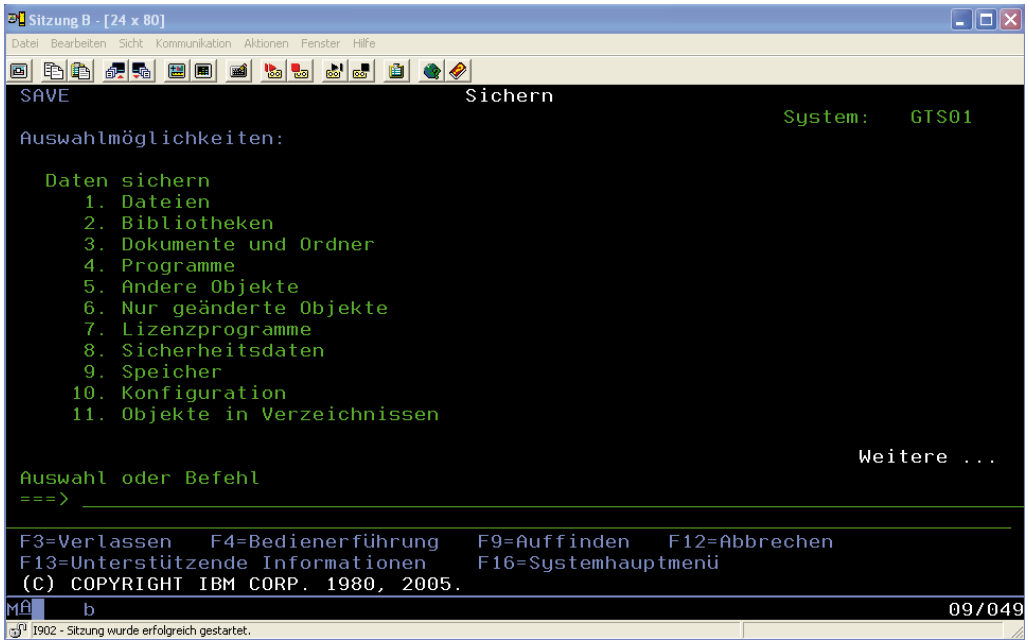
Seite 6

Sicherungsverfahren – Menüs

Wie bereits erwähnt, bietet IBM mit dem IBM i Datensicherungsverfahren (und auch Wiederherstellungsfunktionen) in Form von speziellen Menüs an.

Ungeübte Systembediener haben damit die Möglichkeit, effizient und zielgerichtet die jeweils erforderlichen Datensicherungsverfahren einsetzen zu können.

Die nachfolgende Abbildung zeigt exemplarisch den Inhalt des Menüs „GO SAVE“ – das Menü, in dem die meisten der verfügbaren Datensicherungsmöglichkeiten enthalten sind.



GO SAVE

Wie man unschwer erkennen kann, bietet das Menü GO SAVE unterschiedliche Datensicherungsverfahren an. Es kommt letztlich auf die jeweils durchzuführende Form der Sicherung an, welches der vorhandenen Verfahren geeignet ist, um die Sicherung der Daten und Informationen durchzuführen.

Auf der ersten Seite der GO SAVE Anzeige finden wir unterschiedliche Sicherungsverfahren für Teilbereiche der IBM i Installation. Die zu sichernden Bereiche werden hier im Wesentlichen nach den zu sichernden Objektarten differenziert.

8.1 Seite 2

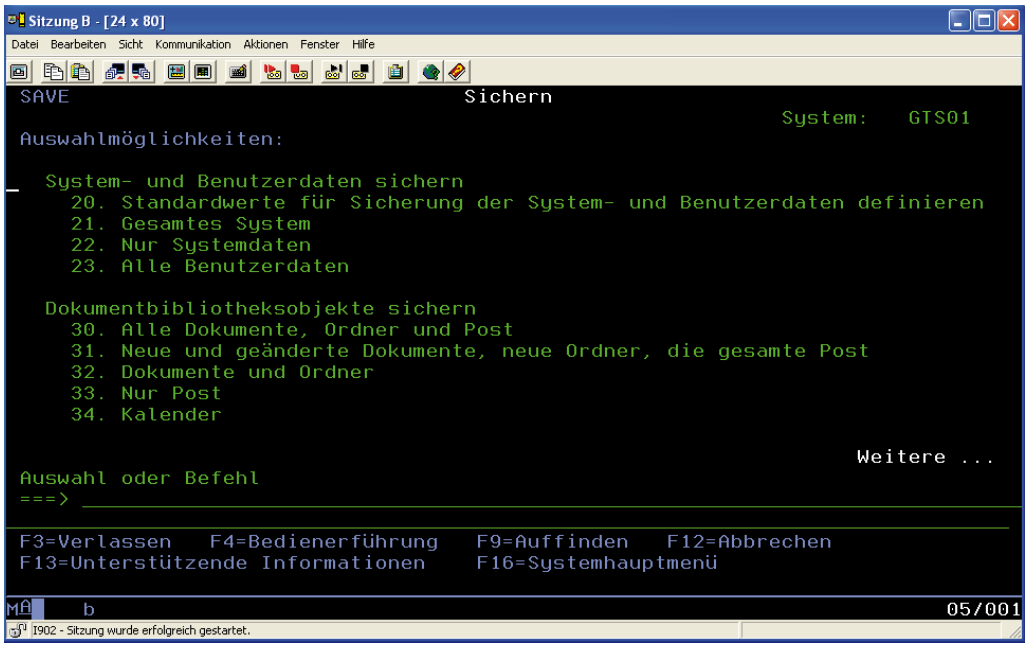
Auf diese Weise können wir mit der ersten Seite der GO SAVE Anzeige zum Beispiel die folgenden Sicherungen durchführen:

- Sicherung von Dateien
  - Verwendung des Befehls SAVOBJ mit dem Parameter *FILE
- Sicherung von Bibliotheken
  - Verwendung des Befehls SAVLIB
- Sicherung von Dokumenten
  - Verwendung des Befehls SAVDLO
- Sicherung von Programmen
  - Verwendung des Befehls SAVOBJ mit dem Parameter *PGM
- Sicherung von anderen Objekten
  - Verwendung des Befehls SAVOBJ mit dem Parameter *ALL
- Sicherung von geänderten Objekten
  - Verwendung des Befehls SAVCHGOBJ
- Sicherung von Lizenzprogrammen
  - Verwendung des Menüs LICPGM
- Sicherung von Sicherheitsdaten
  - Verwendung des Befehls SAVSECDTA
- Sicherung von Speicher
  - Verwendung des Befehls SAVSTG
- Sicherung von Konfiguration
  - Verwendung des Befehls SAVCFG
- Sicherung von Objekten
  - Verwendung des Befehls SAV

Mit diesen Sicherungsfunktionen lassen sich weite Bereiche des IBM i individuell sichern. Auf diese Weise ist ein Administrator in der Lage, spezielle Objekte zielgerichtet zu sichern und damit nicht nur die tägliche, wöchentliche oder monatliche Datensicherung durchzuführen, sondern auch besondere Datensicherungen durchzuführen, die zum Beispiel dann notwendig sind, wenn neue Objekte auf das System eingespielt werden sollen, man jedoch den bisherigen Stand dieser Objekte sichern will.

Wie wir später noch sehen werden, finden wir einige der zuvor genannten Sicherungsverfahren in den weiteren Datensicherungsverfahren wieder, die auf den weiteren Seiten des Menüs „GO SAVE“ zu finden sind. IBM hat diese so zusammengestellt und kombiniert, dass damit eine weitestgehend automatisierte Sicherung spezieller Bereiche ermöglicht wird.

Auf der zweiten Seite des Menüs GO SAVE finden wir erweiterte Sicherungsverfahren. Diese können zum Beispiel eingesetzt werden, um spezielle Gruppen oder Bereiche von Informationen zu sichern. Auf diese Weise lassen sich zum Beispiel reine Systemdaten (Betriebssystem) oder alle Benutzerdaten sichern. Die nachfolgende Abbildung zeigt die zur Verfügung stehenden Sicherungsoptionen.



GO SAVE -2

Während einige der in dieser Anzeige befindlichen Menüoptionen nur selten genutzt werden, möchte ich die wohl wichtigste Menüoption bereits an dieser Stelle hervorheben: Die Sicherung des gesamten Systems (die Auswahl 21).

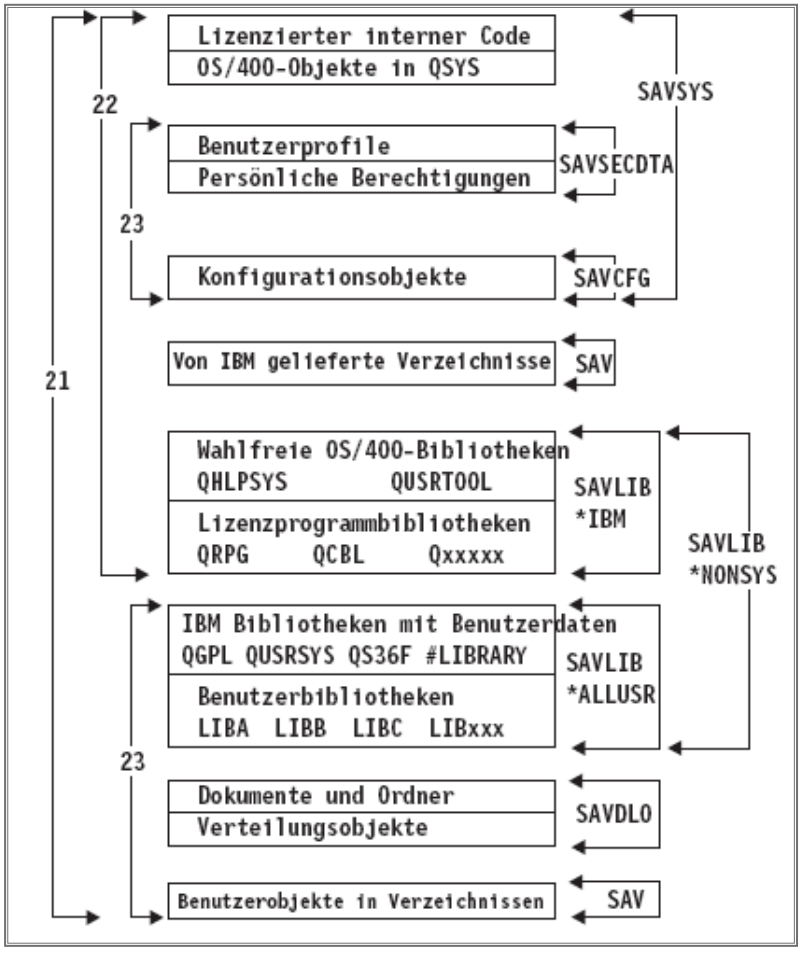
Erfahrene Systembediener kennen diese besondere Form der Sicherung des „gesamten Systems“. Spricht man von einer „Gesamtsicherung“, dann ist damit die Auswahl 21 des Menüs „GO SAVE“ gemeint. Darin enthalten ist die nahezu vollständige Sicherung des gesamten Systems. Damit werden zum Beispiel Systeminformationen, Betriebssystem, Microcode, Konfigurationsdaten, Sicherheitsdaten, Benutzerinformationen und unterschiedliche Arten von speziellen Objekten gesichert.

8.1
Seite 4

Achtung!

Auch wenn der Menüpunkt 21 die Bezeichnung „Sichern des gesamten Systems“ trägt – lIBM suggeriert damit, dass wirklich das gesamte System gesichert wird, ist das leider NICHT der Fall! Auch wenn es den Anschein erweckt – aber es gibt durchaus einige wichtige Bereiche, die nicht mit dieser Auswahl gesichert werden können. So kommt es beispielsweise auf das Release an, das auf dem System betrieben wird. Bei älteren Releaseversionen (bis V5R4) kann zum Beispiel keine Druckausgabe gesichert werden. Der Verlust von Druckausgaben (Inhalten von Ausgabewarteschlangen) kann jedoch durchaus ungewollt sein! Weiterhin kann der Inhalt von unterschiedlichen Warteschlangen mit der Gesamtsicherung (und auch keinen anderen Sicherungsverfahren) gesichert werden. Dies muss jedem verantwortlichen IT Betreiber klar sein – denn in einem Katastrophenfall sind unter Umständen diese Daten nicht wieder herstellbar!

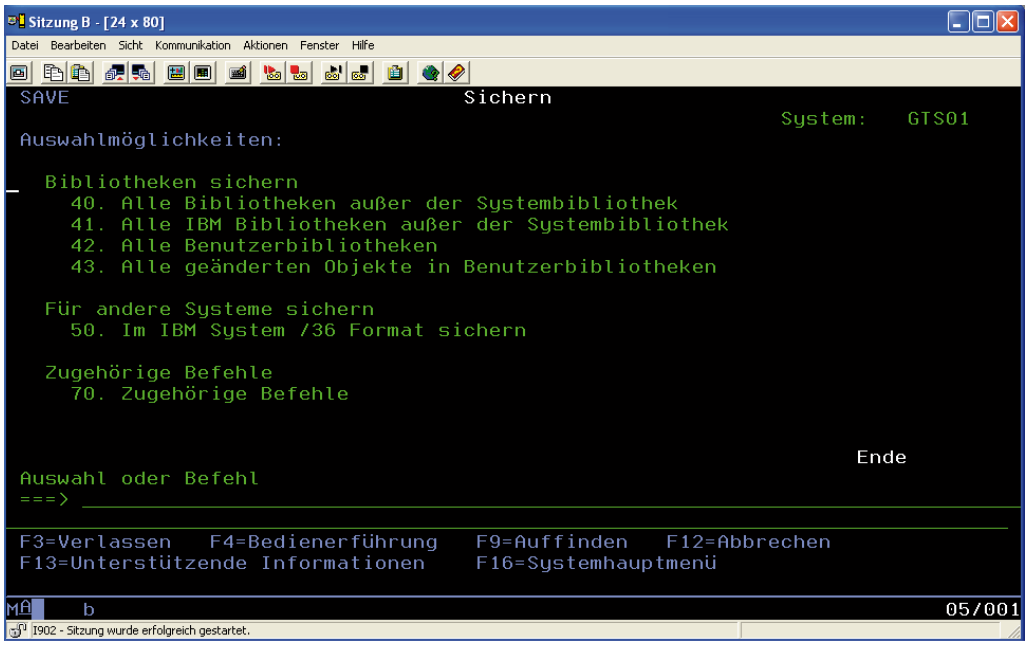
Die nachfolgende Abbildung zeigt die verschiedenen Sicherungsbereiche des IBM i:



Sicherungsbereiche Quelle:IBM

Die in der vorhergehenden Abbildung enthaltenen Sicherungsbereiche sind mit Nummern versehen. Diese Nummern stehen für die Menüauswahloptionen in dem Menü GO SAVE. Wie man unschwer erkennen kann, beinhaltet die Option 21 die Unterbereiche 22 (Systemdaten) und 23 (Benutzerdaten).

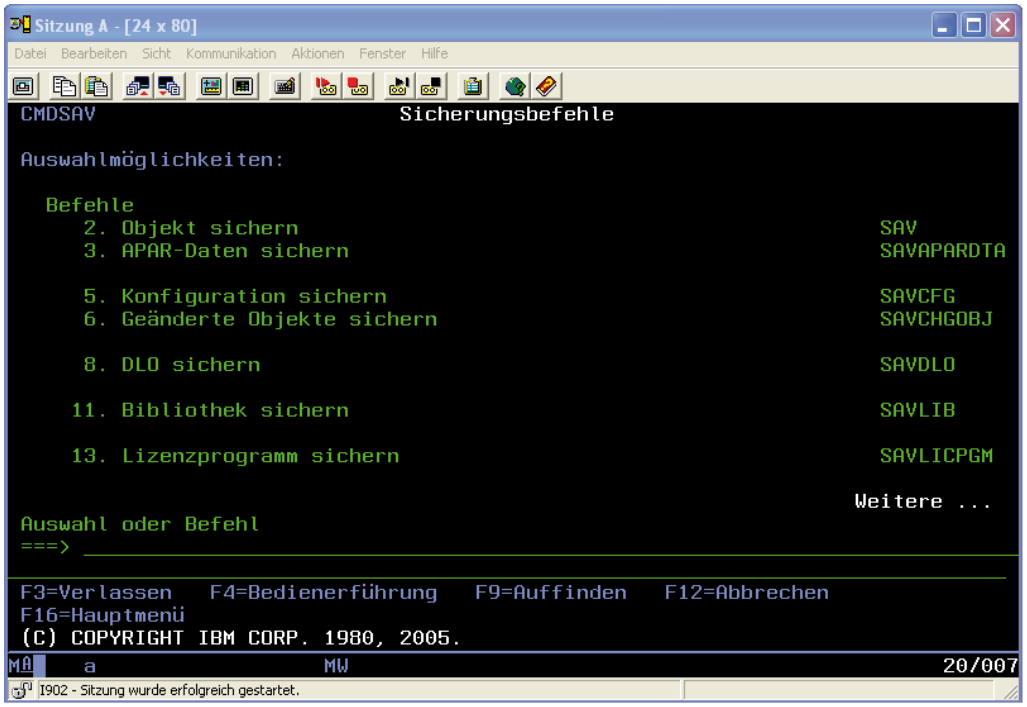
Die letzte Seite des Menüs GO SAVE beinhaltet zum einen den wichtigen Bereich der Bibliotheken und der damit verbundenen Datensicherungsverfahren. Aus der Historie begründet lassen sich auch noch spezielle Sicherungen für das /36 System durchführen, das noch auf dem IBM i unterstützt wird.



GO SAVE

8.1
Seite 6

Wie es bei den meisten Menüs auf dem IBM i der Fall ist, so finden wir auch hier die „zugehörigen Befehle“. Damit erhalten wir eine Übersicht der verfügbaren Datensicherungsbefehle.



Sicherungsbefehle

Wie Sie sehen, bietet das Menü GO SAVE einen umfassenden Fundus an Sicherungsverfahren, die geeignet sind, einzelne Teilbereich zu sichern, aber auch das gesamte System mit einem einzelnen Aufruf sichern zu können.

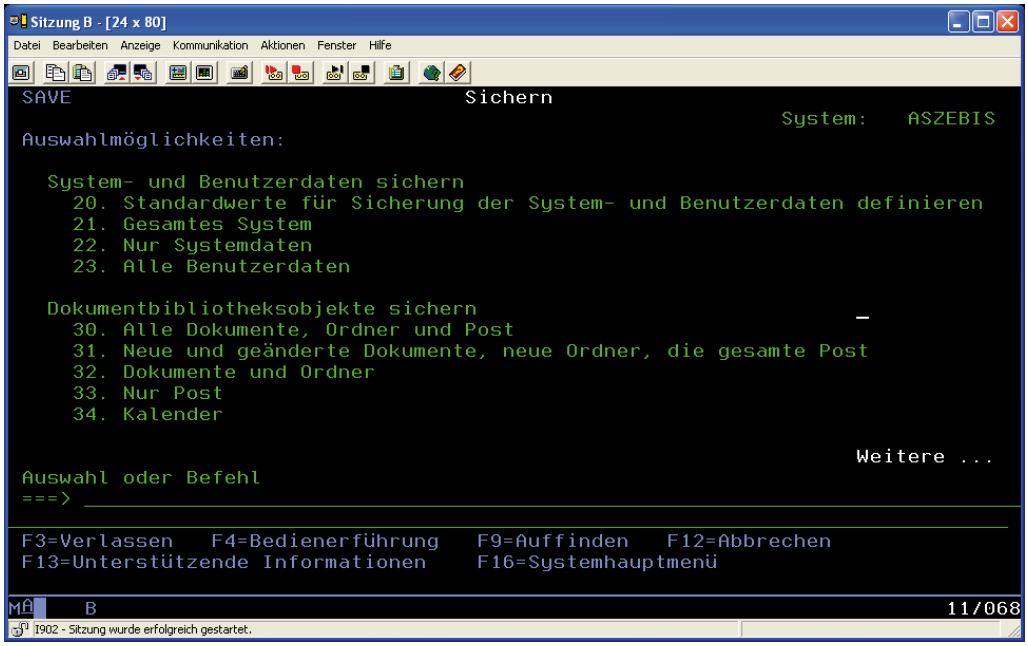
System- und Benutzerdaten sichern

Die Sicherung eines IBM i gliedert sich in die beiden Hauptbereiche Systemsicherung und Sicherung der Benutzerdaten.

Die Systemdatensicherung umfasst dabei die von IBM gelieferten Daten, die neben Microcode das Betriebssystem, Lizenzprogramme und noch einige weitere Bereiche umfassen. In einem Notfall lassen sich diese Informationen natürlich auch von den Auslieferungsdatenträgern der IBM (CDs/DVDs) wiederherstellen. Allerdings sind dann die einzelnen Teilschritte des Aufbaus eines Systems komplexer und es bedarf beispielsweise auch des zusätzlichen Einspielens der PTFs, die in einer Systemsicherung bereits enthalten sind. Deshalb sollte man auch bei der Wiederherstellung von Systemdaten versuchen, stets eine geeignete Datensicherung zu nutzen, eine Datensicherung also, welche die Spezifika des jeweiligen Systems beinhaltet. Dazu gehören im Systembereich u.a. die bereits beschriebenen PTFs, die für Microcode, Betriebssystem und Lizenzprogramme auf der Maschine angelegt werden.

Für die Datensicherung bietet IBM verschiedene voreingestellte Prozesse an, die wir in Form von einzelnen Befehlen oder als Menüaufrufe einsetzen können. Die einfachste Variante einer vollständigen Datensicherung ist die Nutzung des Menüs SAVE mit den darin enthaltenen Optionen.

Die folgende Abbildung zeigt ausschnittweise einige der verfügbaren Optionen.



GO SAVE

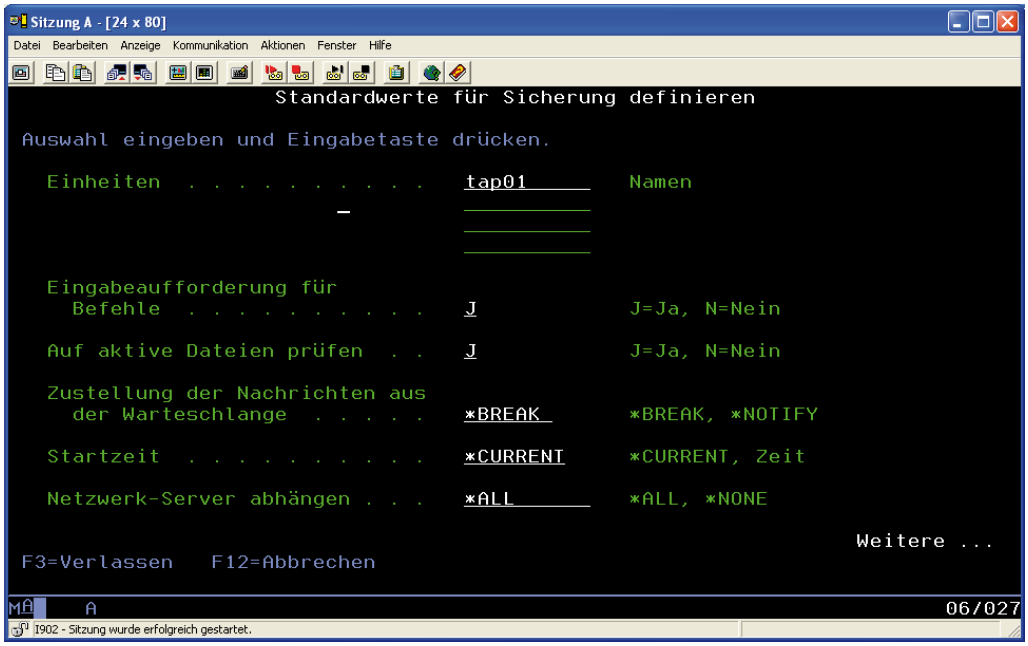
8.2
Seite 2

Schauen wir uns die unterschiedlichen Optionen für die globalen Sicherungsprozesse einmal genauer an.

Die Auflistung der Auswahlmöglichkeiten beginnt in der Abbildung oben mit Auswahl 20, mit der wir Grundeinstellungen bzw. Vorgaben für Sicherungsprozesse definieren können. Diese Grundeinstellungen bzw. Vorgaben kommen bei den verschiedenen Sicherungsaufrufen zum Einsatz und können dort bei Bedarf jederzeit individuell überschrieben werden.

Unter anderem kann festgelegt werden, welche Datensicherungseinheit als Unterlassungswert bei den Sicherungsverfahren eingestellt oder vorgeschlagen werden soll. Einer der hier verfügbaren Parameter definiert z. B., welche Datensicherungseinheit als Standard für Datensicherungen voreingestellt werden soll.

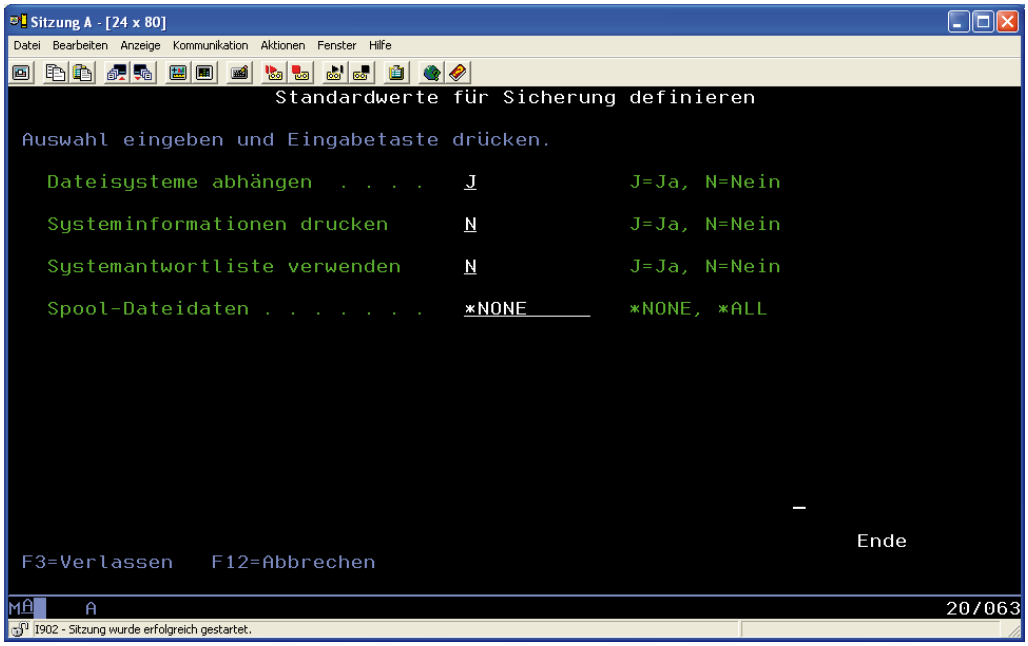
Mit dem Parameter „Eingabeaufforderung für Befehle“ können wir die einzelnen Sicherungsteilschritte beeinflussen. Der Wert „N“ bedeutet, dass keine Meldung vor den einzelnen Sicherungsteilschritten ausgegeben wird. Achtung! Wird dieser Wert auf „J“ gesetzt, dann muss während der Datensicherung die Fortsetzung der einzelnen Sicherungsteilschritte vom Administrator bestätigt werden. Damit ist dann keine unbeaufsichtigte Datensicherung mehr möglich!



Standardwerte 1

Über den Parameter „Startzeit“ lässt sich beispielsweise auch eine versetzte Datensicherung ausführen. Dabei muss aber beachtet werden, dass die Sicherungsanforderungen bezüglich der Objektverfügbarkeit gewährleistet sind. Wenn z. B. eine Bibliothek gesichert werden soll, dann muss dabei beachtet werden, dass diese zum Zeitpunkt der Datensicherung nicht in Verwendung ist. Der Sonderwert *CURRENT definiert, dass per Standard die Sicherungsprozedur sofort startet. In der Regel sollte man diesen Parameter nicht verändern! Eine zeitlich versetzte Datensicherung lässt sich idealerweise mit einem CL und entsprechenden Zeitverzögerungen in dem CL selbst oder der Verwendung von Jobplanungen realisieren. Der Gebrauch der Startzeit innerhalb der Vorgabewerte bzw. der einzelnen Datensicherungsbefehle ist in der Regel nicht zu empfehlen.

Wenn wir weiterblättern, gelangen wir in den 2. Anzeigenbereich der Standardwerte für die Datensicherung. Diese finden Sie in der nachfolgenden Abbildung.



Standardwerte 2

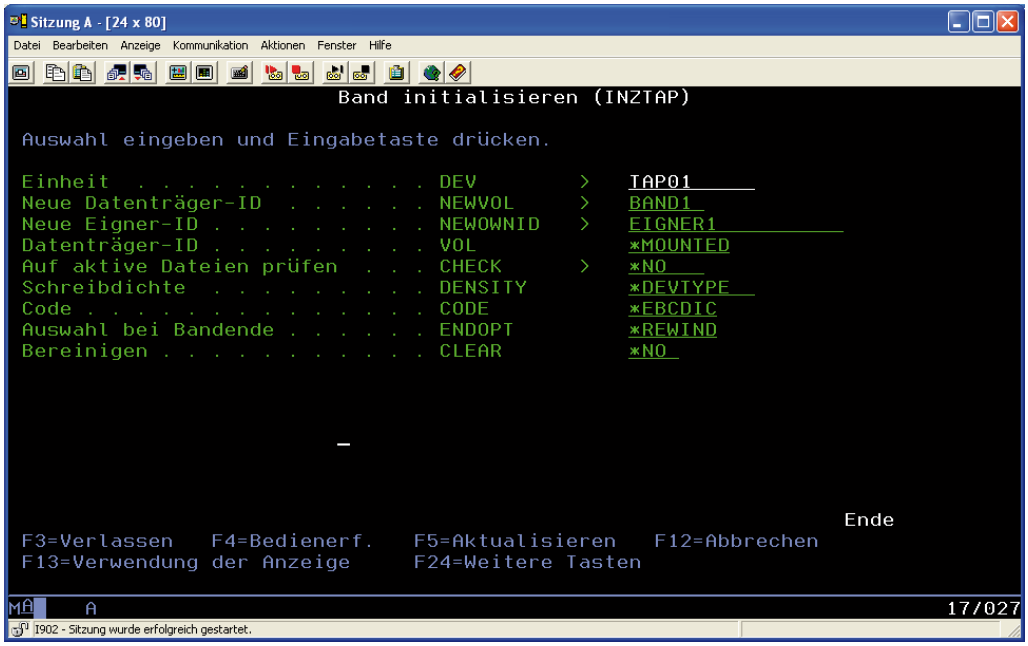
Der Parameter „Dateisysteme abhängen“ definiert, ob die dynamischen Dateisysteme vor der Datensicherung abgehängt werden sollen.

Mit dem Wert „Systemwerte drucken“ kann im Zuge der Datensicherung ein Ausdruck der wichtigsten Systemeinstellungen ausgelöst werden. Solche Ausdrucke können im Fall einer Systemwiederherstellung sehr wichtig sein! Auch wenn nicht bei jeder Datensicherung ein solcher Ausdruck sinnvoll erscheint, so ist es doch empfehlenswert, zumindest mehrmals im Jahr einen Ausdruck der Systemdaten zu erstellen. Sollte einmal der unwahrscheinliche Fall eintreten, dass eine vollständige Systemwiederherstellung notwendig ist, dann kann man basierend auf den ausgedruckten Daten Fehlerquellen ausschließen und nicht zuletzt wichtige Zeit gewinnen.

Bandinitialisierung

Bevor wir eine Datensicherung durchführen, sollte sichergestellt sein, dass die zu verwendenden Datenträger, in der Regel sind das Bänder, vorbereitet und in ausreichender Menge vorhanden sind. Bänder können mehrfach für Datensicherungen verwendet werden. Dabei müssen wir jedoch bedenken, wie die Daten auf den Bändern gespeichert werden. Werden neue zu sichernde Daten an bereits bestehende Daten angehängt? Werden die Daten auf dem Band vielleicht noch benötigt? Beinhaltet ein Band also „aktive" Daten? Diese Überlegungen sind natürlich nicht nur für die Systemsicherung relevant, sondern für jede Art der Datensicherung. Beachten Sie deshalb die Bedeutung schon vorhandener Bandinhalte bei einer Datenwiederherstellung.

In der Regel empfiehlt es sich, vor einer Datensicherung eine Initialisierung der Bänder durchzuführen, wobei einige Angaben zu machen sind. Mit dem Initialisieren werden auch die Kennsätze bzw. das Inhaltsverzeichnis des Bandes initialisiert und bestehende Bandinhalte nachhaltig gelöscht. Eine solche Initialisierung kann mit dem Befehl INZTAP durchgeführt werden.



INZTAP

Die Parameter für den Befehl INZTAP sind zum Teil optional. Meist werden die folgenden Parameter genutzt:

- Einheit (DEV)
  Dieser Parameter muss angegeben werden. Hier geben wir den Namen des Bandlaufwerks bzw. der Datensicherungseinheit an.

8.2.1
Seite 2

- Neue Datenträger ID (NEWVOL)
  Dieser eigentlich als optional anzusehende Parameter kann, wenn er hier nicht angegeben wird, bei einer Datensicherung zu einem Problem führen. Ein nicht benannter Datenträger kann zusammen mit einer fehlenden Eigner-ID dazu führen, dass beim Starten der Datensicherung eine Meldung ausgegeben wird, die ein erneutes Initialisieren erfordert. Deshalb sollte bei der Initialisierung in jedem Fall eine Datenträger-ID sowie eine Eigner-ID angegeben werden.
- Neue Eigner-ID (NEWOWNID)
  Zusammen mit der Datenträger-ID definiert die Eigner-ID das Band. Mit diesen Angaben können in komplexen Umgebungen die Bänder besser identifiziert und verwaltet werden.
- Datenträger-ID (VOL)
  Dieser Parameter spielt insbesondere bei der Verwendung von Library-Systemen eine Rolle. Einige Datenträger verfügen zudem über eindeutige Kennzeichnungen in Form von Barcodes. Diese können als eindeutige Datenträger-Idents verwendet werden. Auch dieser Parameter spielt bei der Nutzung einer Vielzahl von Bändern eine Rolle, kann aber bei kleineren Installationen entfallen.
- Auf aktive Dateien prüfen (CHECK)
  Datensicherungen können in der Gesamtheit mit anderen Bändern dazu dienen, eine vollständige Systemwiederherstellung durchzuführen. Deshalb sprechen wir auch von „aktiven“ Dateien. Diese werden in einem Wiederherstellungsprozess benötigt. Allerdings ist dazu eine entsprechende Verwaltung der Datenträger erforderlich. Wenn keine entsprechenden Systeme und Verfahren verwendet werden, dann kann der Standardwert *NO gesetzt werden.
- Schreibdichte (DENSITY)
  Die Schreibdichte variiert je nach verwendetem Datenträger bzw. auch nach verwendeter Datensicherungseinheit. Unter bestimmten Voraussetzungen können die zu sichernden Daten auf dem Datenträger komprimiert und damit eine erhöhte Menge an Sicherungsdaten erreicht werden. Die gültigen Werte richten sich nach dem verwendeten Band bzw. der Datensicherungseinheit. Der Sonderwert *DEVTYPE sagt aus, dass die von der Datensicherungseinheit unterstützten Maximalwerte genutzt werden.
- Code (CODE)
  Das IBM i arbeitet mit einem speziellen IBM-Zeichensatz, der in der Regel auch als Standardcode für die Datensicherungen genutzt wird. In seltenen Fällen kann dieser Standardwert EBCDIC auf den ANSI-ASCII-Wert umgestellt werden.

- Auswahl bei Bandende (ENDOPT)
  Mit dieser Angabe kann festgelegt werden, wie das Band nach dem Initialisieren behandelt werden soll. Möglichkeiten sind, das Band an den Anfang zurückzuspulen und so direkt für die Verwendung nutzbar zu machen, es an der Stelle stehen zu lassen, an der die Initialisierung geendet hat, oder es aus der Sicherungseinheit auszuwerfen. (Letztere Option wird jedoch nicht von allen Sicherungseinheiten unterstützt.)
- Bereinigen (CLEAR)
  Unter der Initialisierung eines Bandes verstehen wir im Allgemeinen ein Löschen des Inhaltsverzeichnisses. Damit ist der Zugriff auf den eigentlichen Bandinhalt nicht mehr möglich – zumindest nicht mit Standardmitteln des Betriebssystems. Allerdings: Die eigentlichen Daten stehen auf dem Band immer noch gespeichert zur Verfügung und können mit entsprechenden Fachmitteln auch verwendet werden. Wenn man den gesamten Bandinhalt löschen will, dann muss dieser Parameter mit *YES angegeben werden. Damit wird nicht nur das Inhaltsverzeichnis des Bandes (die Kennsätze), sondern der gesamte Inhalt gelöscht.

Bezogen auf die vorhergehende Abbildung bedeuten diese Parameter, dass wir ein Band in der Einheit TAP01 initialisieren, dessen Datenträger-ID „Band 1“ lautet; die Eigner-ID wurde mit EIGNER1 angegeben. Wir wollen die optimale Schreibdichte und prüfen vor dem Initialisieren nicht, ob das Band aktive Daten beinhaltet. Nach dem Initialisieren soll das Band an den Anfang zurückgespult werden. Außerdem wird lediglich der Kennsatz gelöscht, der Rest bleibt auf dem Band nicht mehr verwendbar erhalten und wird bei der neuen Datensicherung überschrieben.

Mit einem auf diese Weise vorbereiteten Band kann nun eine beliebige Datensicherung erfolgen.

Anmerkung

Bereiten Sie im Vorfeld einer Datensicherung stets so viele Bänder vor, wie für die Datensicherung voraussichtlich benötigt werden. Damit sparen Sie nicht nur Zeit während der Sicherung selbst, sondern vermeiden auch Unsicherheiten!

8.2.1
Seite 4

Gesamtes System sichern

Beide Teilbereiche der Datensicherung (Systemdaten und Benutzerdaten) können in Form der sogenannten Gesamtsicherung (GO SAVE – Auswahl 21) ausgeführt werden. Diese Form der Datensicherung bietet sich z. B. dann an, wenn entscheidende Änderungen an der Installation oder den Anwendungen vorgenommen werden (etwa Releasewechsel des Betriebssystems oder markante Änderungen an der Anwendungssoftware). Als Empfehlung gilt, mindestens 1-2 Mal pro Jahr eine Gesamtsicherung durchzuführen, außerdem vor und nach entscheidenden Veränderungen an Betriebssystemobjekten, wie sie z. B. beim Einspielen von PTFs oder bei der Installation eines neuen Release bzw. einer neuen Version vorgenommen werden.

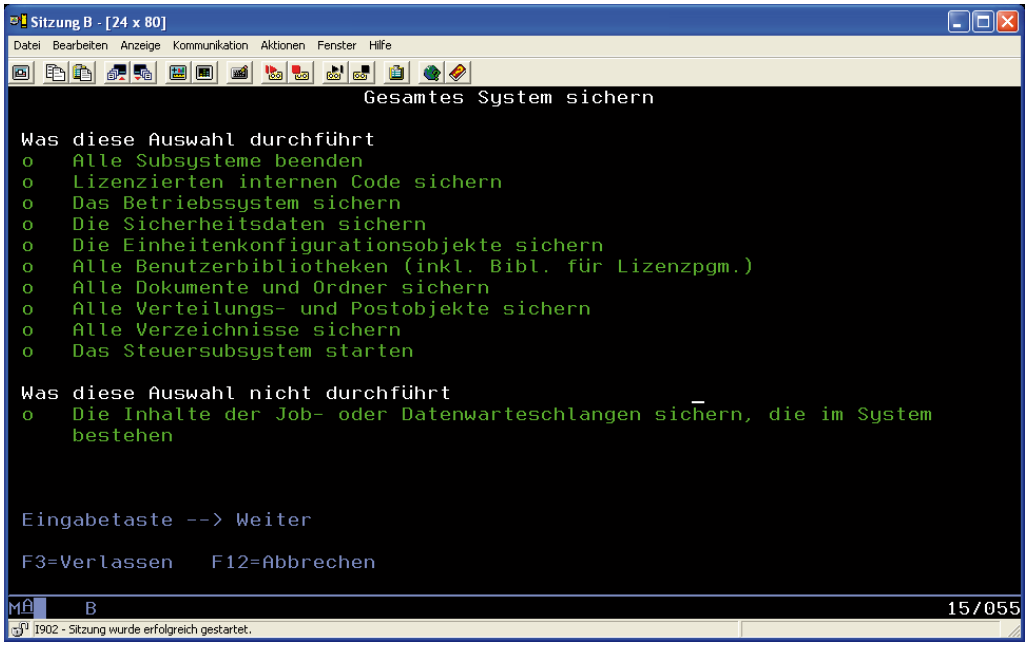
Die einzelnen Sicherungsteilschritte der Gesamtsicherung sind:

- Lizenzierter Interner Code (LIC)
- Betriebssystem
- Sicherheitsdaten
- Einheitenbeschreibungen
- Alle Benutzerbibliotheken (auch Bibliotheken mit Lizenzprogrammen)
- Dokumente und Ordner
- Verteilungsobjekte
- Verzeichnisse

Bei einer Gesamtsicherung ist zu beachten, dass diese nur von der Konsole des IBM i ausgeführt werden kann. Sollte der Versuch unternommen werden, die Gesamtsicherung von einer anderen Arbeitsstation aus aufzurufen, so lässt sich der Sicherungsjob nicht starten. Im Jobprotokoll findet man dann Hinweise darauf, dass in der aktuellen Umgebung die Datensicherung nicht durchgeführt werden kann. Unter anderem liegt der Grund dafür darin, dass der für eine Gesamtsicherung notwendige Befehl ENDSBS *ALL (Beenden aller Subsysteme mit Ausnahme des steuernden Subsystems) nur vom Konsolbildschirm aus eingegeben werden kann.

Ausgehend davon, dass eine Gesamtsicherung durchgeführt werden soll, melden wir uns am Konsolbildschirm an. Wichtig ist dabei, dass der Benutzer über die notwendigen Berechtigungen verfügt, um die für die Gesamtsicherung notwendigen Teilschritte auch wirklich durchführen zu können. Wie bereits zuvor erwähnt, müssen alle Subsysteme beendet worden sein und die zuvor genannten Sicherungsfunktionen aufgerufen werden können. Idealerweise verwendet man ein dem SECOFR nahekommendes Benutzerprofil.

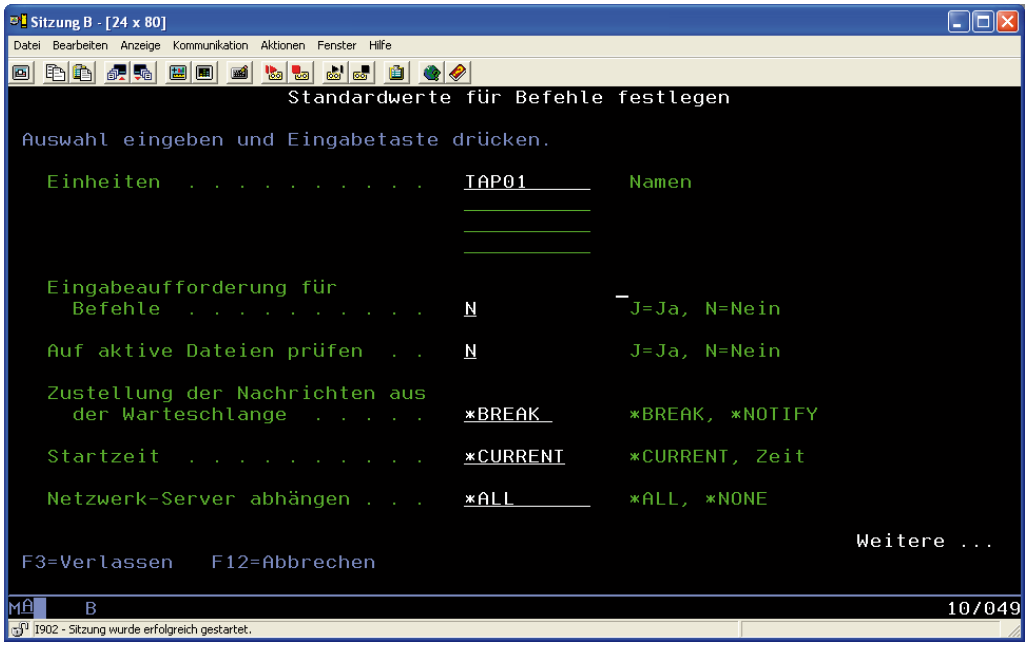
8.2.2
Seite 2



Teilschritte der Gesamtsicherung

In der Abbildung oben sehen Sie die einzelnen Bestandteile der Gesamtsicherung des IBM i. Beachten Sie dabei den letzten Eintrag! Auch wenn IBM mit der Bezeichnung „Gesamtsicherung“ suggeriert, dass alles auf dem System i gesichert wird – das ist NICHT der Fall! Inhalte von Job- und Datenwarteschlangen können nicht mitgesichert werden. Zudem muss auch beachtet werden, dass der Inhalt von Ausgabewarteschlangen (Druckausgaben) erst ab der Version V5R4 gesichert werden kann – und auch nur dann, wenn dies in den Voreinstellungen so definiert wurde!

Nach Drücken der Eingabetaste gelangen wir in den Definitionsbildschirm der Gesamtsicherung. Dort können wir die in den Vorgaben definierten Standardwerte bei Bedarf überschreiben.



Angaben zur Gesamtsicherung

Beachten Sie in dieser Anzeige den Parameter „Startzeit“. Damit kann quasi der Start der Gesamtsicherung geplant werden – allerdings ist auch in diesem Fall die Eingabe des Startbefehls am Konsolbildschirm notwendig.

8.2.2
Seite 4

Systemdaten sichern

Es macht durchaus Sinn, die Sicherungen in den Bereichen Systemdaten und Benutzerdaten getrennt voneinander auszuführen. Mit einer solchen Sicherungsmethode kann man gewährleisten, dass die Benutzerdaten täglich – und die Systemdaten beispielsweise einmal am Wochenende gesichert werden.

Auch für diese Form der Datensicherung finden wir im Menü SAVE vorgefertigte Sicherungsaufrufe.

Diese typische Form der Trennung der Sicherungsbereiche finden wir in den meisten IBM-i-Umgebungen. Dies ist meist historisch gewachsen – leider, muss man sagen.

Während in den früheren Zeiten der AS/400- und iSeries-Systeme die Datensicherungen mit relativ langsamen Bandlaufwerken ausgeführt wurden, die zum Teil noch nicht einmal in der Lage waren, ohne einen Benutzereingriff alle Daten des Systems auf ein Band zu sichern, verfügen wir heute dank moderner Technologien über leistungsstarke Bandlaufwerke, die meist in der Lage sind, 400 GB und mehr auf ein einzelnes Band zu sichern – und das auch noch in einer recht kurzen Zeit.

Deshalb bietet sich heute die Überlegung an, ob man mittels der geeigneten Sicherungstechniken täglich eine zumindest annähernde Gesamtsicherung mitsamt dem Betriebssystem und den übrigen Systemdaten durchführen sollte. Natürlich lässt sich die Frage „Wie oft und wann soll eine Systemsicherung durchgeführt werden?“ nicht global beantworten. Am einfachsten, Sie sichern das System täglich neben den Benutzerdaten. Sind die Ressourcen (in Form eines geeigneten Bandlaufwerks und eines ausreichenden Zeitfensters) vorhanden, dann ist dies sicher eine Ideallösung. Ansonsten gilt, dass man zumindest einmal im Quartal (besser jedoch monatlich oder wöchentlich) eine Systemsicherung durchführt. Zusätzlich sollte vor jedem größeren Eingriff in die Hardware oder beim Einspielen neuer Software (neue Version oder PTFs) eine Systemsicherung erfolgen.

Achtung!

Systemsicherungen sind wichtig. Jeder, der einmal in der Situation war, sein System nach einem Ausfall wiederherstellen zu müssen, wird das Vorhandensein einer Systemsicherung zu schätzen wissen. Eine Systemsicherung versetzt uns in die Lage, die Maschine wieder von Grund auf aufbauen zu können. Auch hier nochmals der Hinweis: Natürlich kann ein System mit den Originaldatenträgern von IBM (CD/DVDss) wiederhergestellt werden. Allerdings müssen dazu zusätzlich PTFs installiert, Konfigurationseinstellungen angepasst und weitere Einstellungen vorgenommen werden.

8.2.3
Seite 2

Betrachten wir ein Beispiel:
Nach einem Hardwaredefekt ist das System neu aufzusetzen. Die Daten auf den Plattensystemen sind verloren. Sicher eine Katastrophe, wenn man nicht über eine geeignete Datensicherung verfügt. Aufgrund einer Systemsicherung haben wir allerdings die notwendigen Daten, um das System wieder komplett neu aufsetzen zu können. In einem solchen Fall wird das Band mit der Systemsicherung zum Hochfahren der Maschine genutzt. Der lizenzierte interne Code (die Basis der Maschine, die quasi das dumme Blech mit Leben versieht) wird dabei auf die Load-Source gespielt, das Betriebssystem vom Band geladen und ebenfalls auf die Platte übertragen. Damit lässt sich eine komplett leere Maschine mit dem letzten gesicherten Stand von Microcode und Betriebssystem hochfahren. Sicher: Man kann im Katastrophenfall auch die von IBM ausgelieferten Installations-CDs oder -DVDs verwenden – doch muss dabei sichergestellt werden, dass auch die PTFs auf der Maschine nachinstalliert werden. Ein zusätzlicher Arbeitsschritt, der beim Wiederaufbau der Maschine über eine Systemsicherung entfällt, denn: Die Systemsicherung beinhaltet u. a. auch die PTFs, so dass diese nach der Installation auf dem System automatisch wieder zur Verfügung stehen.

Sie sehen, es macht durchaus Sinn, die Systemsicherung in gewissen Zeitabständen durchzuführen, um jederzeit die Möglichkeit zu haben, das System mit Hilfe dieser Daten wiederherstellen zu können.

Doch was genau gehört zu einer Sicherung der Systemdaten?

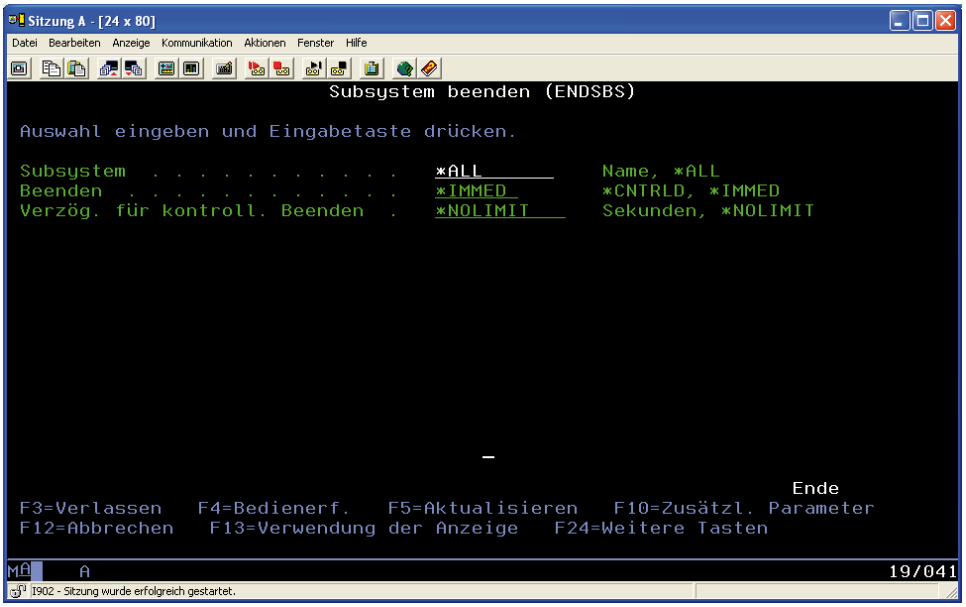
Bedingt durch die Architektur des IBM i ist die Struktur der Systemdaten wie folgt vorgegeben:

- Microcode
- Betriebssystem
- Lizenzprogramme
- Sicherheitsdaten
- Konfigurationsdaten
- Alle von IBM gelieferten Bibliotheken
- Alle von IBM gelieferten Verzeichnisse

Die verschiedenen Bereiche einer Systemsicherung können wahlweise manuell einzeln nacheinander, automatisiert über ein CL-Programm oder mittels der von IBM zur Verfügung gestellten Sicherungsprozeduren gesichert werden. Sollten Sie sich für eine manuelle Sicherung des Systems entscheiden – sei es durch die Angabe der einzelnen Sicherungsbefehle für die Systembereiche oder deren Angabe in einem CL-Programm –, so müssen Sie selbst sicherstellen, dass auch tatsächlich alle Bereiche des Systems gesichert werden.

Achtung!

Eine Sicherung der Systemdaten bedingt, dass gerade niemand mit dem System arbeitet – sprich: Das System muss sich in einem eingeschränkten Zustand befinden. Dieser eingeschränkte Zustand kann wahlweise automatisch erreicht werden – etwa bei Verwendung einer der Menüauswahlen im GO SAVE – oder manuell gesetzt werden durch Verwendung des Befehls ENDSBS. Es kann in Abhängigkeit von den Anwendungen durchaus sinnvoll sein, dass das Beenden der Subsysteme manuell durchgeführt wird. Mit dem Befehl ENDSBS *ALL werden alle Subsysteme auf dem System i beendet (mit Ausnahme des steuernden Subsystems). Wichtig ist, dass dieser Befehl an der Konsole oder einer anderen Einheit abgesetzt wird, die sich im steuernden Subsystem befindet. Durch das Beenden der Subsysteme werden alle bestehenden Jobs auf dem System beendet. Ein Beispiel für das Beenden aller Subsysteme finden Sie in der folgenden Abbildung.



ENDSBS

Das manuelle Beenden aller Subsysteme ist nicht notwendig, wenn Sie die Menüoptionen innerhalb des Menüs SAVE verwenden. Darin sind alle für das erfolgreiche Ausführen der Datensicherung erforderlichen Schritte bereits implementiert. Die Ausführung des Befehls ENDSBS – wie zuvor gezeigt – ist daher optional.

8.2.3
Seite 4

Wenn die Systemdaten des IBM i gesichert werden, sprechen wir auch von einer SAVSYS (Save System). Dieser Begriff steht für den Befehl, der zum Sichern des Systems dient und auch an Stelle des Menüeintrags SAVE in der Befehlszeile eingegeben werden kann, um die Systemsicherung zu starten.

Die Sicherung der Systemdaten kann über das schon vorgestellte Menü „GO SAVE“ ausgeführt werden. Dort finden wir auf der zweiten Seite die Sicherungsoptionen für das gesamte System oder auch nur die Systemdaten.

Achtung!

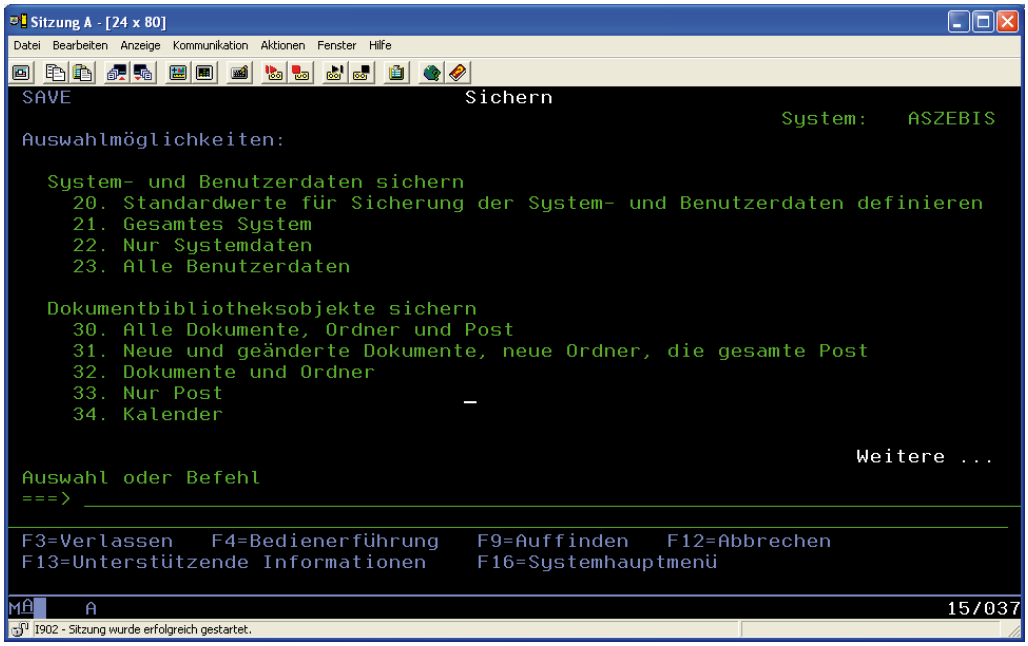
Sicherungen allgemeiner Art können auf dem System i auf unterschiedliche Bänder, Sicherungsdateien oder virtuelle Einheiten erfolgen. Bei der Systemsicherung aber ist diese Auswahl nicht gegeben. Vielmehr besteht hier nur die Möglichkeit, die Systemdaten auf einem Bandlaufwerk zu sichern. Aber auch hier heißt es Vorsicht! Nicht jedes Bandlaufwerk eignet sich auch wirklich für eine Systemsicherung bzw. später für die Wiederherstellung des Systems. Es muss im Wiederherstellungsfall sichergestellt sein, dass die Bandstation, die zum Zurückspeichern des Systems genutzt wird, auch IPL-fähig ist. Die Voraussetzungen dafür variieren und hängen vom vorhandenen Systemanschluss ab. Im Zweifelsfall sollte hier ein IBM-Techniker zu Rate gezogen werden, der Auskunft darüber geben kann, ob die Bandeinheit als alternative IPL-Quelle genutzt werden kann!

Ist sichergestellt, dass die Systemsicherung auf die Bandeinheit auch später erfolgen kann, dann können wir dafür eines der zur Verfügung stehenden Verfahren verwenden.

Systemdaten mit dem Menü GO SAVE sichern

Es gibt unterschiedliche Verfahren, um die Systemdaten des System i zu sichern. Etwa die Nutzung der Auswahl „Nur Systemdaten“ innerhalb des Menüs GO SAVE.

Ein Beispiel hierzu finden Sie in der nächsten Abbildung.

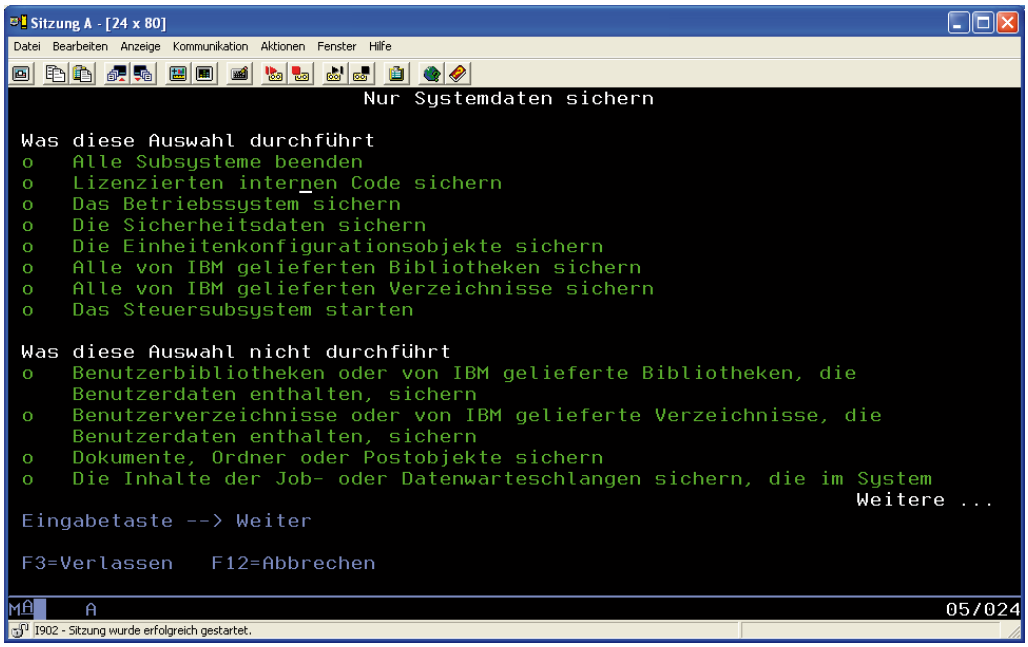


Systemsicherung über GO SAVE

Über die Auswahl 22 („Nur Systemdaten“) wird der Systemteil des IBM i gesichert.

8.2.3.1
Seite 2

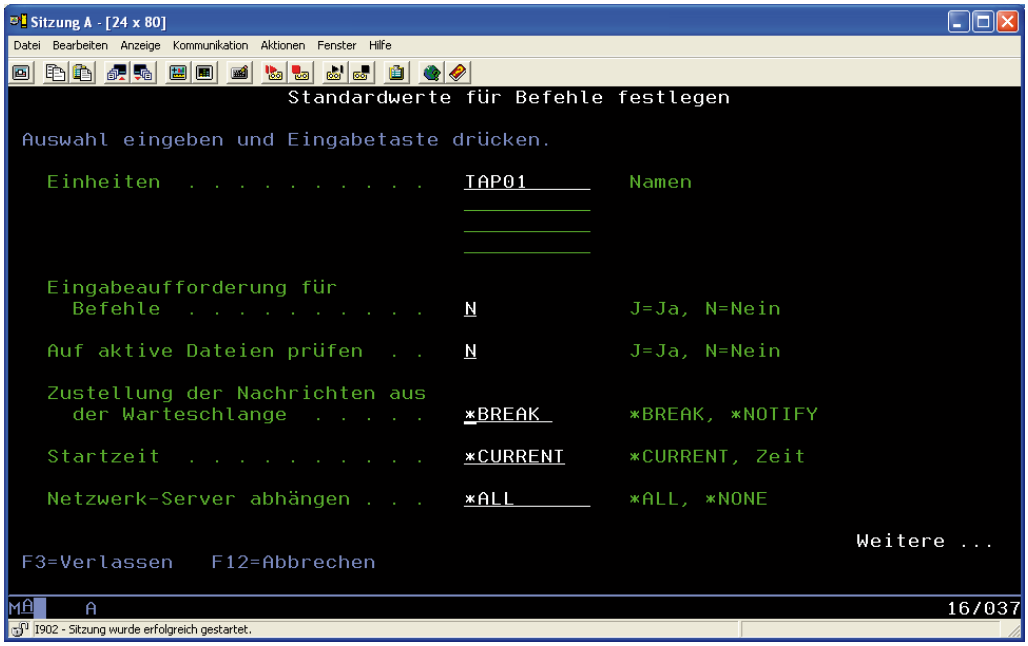
Wenn Sie diese Auswahl vornehmen, gelangen Sie in eine Anzeige, wie sie in der folgenden Abbildung zu sehen ist.



Teilschritte der Systemsicherung

In dieser Informationsanzeige erhalten wir Auskunft über die einzelnen Teilbereiche der Systemsicherung.

Die aufgelisteten Schritte werden im Zuge der Systemsicherung ohne Ausnahme ausgeführt. Es kommt letztlich auf die Einstellungen an, ob das System die verschiedenen Teilschritte automatisch ohne weiteren Benutzereingriff ausführt oder überwacht als Einzelschritte nach Bestätigung durch den Administrator.



Systemdaten sichern

Die Sicherung über das Menü erlaubt die Angabe des Bandlaufwerks und einiger weiterer Parameter. Diese haben die folgende Bedeutung:

- Eingabeaufforderung für Befehle
  Eine Systemsicherung besteht aus verschiedenen Schritten. Wenn dieser Parameter mit „J“ vorbelegt wird, dann werden Sie vor jedem Schritt zur Bestätigung aufgefordert. Es wird empfohlen, den Wert auf „N“ zu ändern, damit die Systemsicherung quasi „unbeaufsichtigt“ erfolgen kann. Das System führt in diesem Fall jeden der erforderlichen Schritte ohne zusätzliche Bestätigung aus.
- Auf aktive Dateien prüfen
  Wenn dieser Parameter auf „J“ gesetzt ist, dann wird vor der Sicherung geprüft, ob sich auf dem Band aktive Dateien befinden, die beispielsweise für einen anderen Wiederherstellungsprozess benötigt werden. Sind solche aktiven Daten vorhanden und ist der Wert auf „J“ gesetzt, so wird der Datensicherungsprozess mit einem Fehlerhinweis beendet und keine Sicherung auf dem Band durchgeführt.
- Zustellung der Nachrichten aus der Warteschlange
  Mit diesem Parameter kann angegeben werden, wie ggf. Nachrichten, die den Datensicherungsprozess betreffen, dem Systembediener zugestellt werden sollen. Mit der Einstellung *BREAK werden alle Nachrichten im Durchbruchmodus zugestellt. Das kann zwar hilfreich sein, denn man vermeidet damit, dass man wichtige Nachrichten nicht sofort lesen und darauf reagieren kann. Wenn die Datensicherung jedoch unbeaufsichtigt ausgeführt werden soll, dann kann diese Form der Durchbruchnachrichten auch zu

einer Unterbrechung des Sicherungsprozesses führen. Wenn man die Nachricht dann nicht sofort sieht und entsprechend reagiert, pausiert der Sicherungsprozess so lange, bis die Nachricht beantwortet bzw. bestätigt wird.

- Startzeit
  Man kann die Systemsicherung bei Bedarf für einen zukünftigen Zeitpunkt planen. Wenn die Sicherung nicht sofort starten soll, kann man diesen Parameter mit der gewünschten Startzeit angeben. Die Sicherung startet dann bei Erreichen des angegebenen Zeitpunkts. Achtung! Sollte damit eine „automatische Sicherung" initiiert werden, so sollten Sie darauf achten, dass der Parameter „Eingabeaufforderung für Befehle" auf „N" steht!
- Netzwerkserver abhängen
  Das System i unterstützt den Betrieb von Netzwerkservern. Diese dürfen bei einer Systemsicherung nicht aktiv sein. Deshalb besteht hier die Option, laufende Netzwerkserver vor der Sicherung zu beenden. Aber Vorsicht: Je nach eingesetztem Netzwerkserver und den darauf laufenden Anwendungen kann es sinnvoll sein, den Server vor der Systemsicherung manuell zu beenden!

Die Voreinstellungen für diese Parameter lassen sich global im Bereich „Vorgabewerte für Sicherung festlegen" definieren. Soll z. B. die Eingabeaufforderung für Befehle stets auf „N" stehen, so kann man dies im Voreinstellungsbereich definieren.

Nachdem die gewünschten Eingaben vorgenommen wurden, führt das System die für eine Systemsicherung benötigten Schritte aus. Prüfen Sie die Systemsicherung unbedingt, damit Sie im Fall der Fälle wirklich sicher sein können, dass die Sicherung zu einer Wiederherstellung nutzbar ist.

Systemdaten manuell sichern

Im vorangegangenen Kapitel haben wir die Sicherung der Systemdaten mittels des Menüs GO SAVE und der Auswahl 22 durchgeführt. Alternativ stehen uns noch andere Sicherungsmethoden zur Verfügung, die ebenso dazu geeignet sind, die einzelnen Systembestandteile zu sichern. Auch wenn es keine wirklich gängige Methode ist, so sei an dieser Stelle doch erwähnt, dass die Systemdaten des System i auch im Zuge der Gesamtsicherung (Auswahl 21 des Menüs GO SAVE) mitgesichert werden.

Eine Systemsicherung umfasst verschiedene Schritte:

- Microcode
- Betriebssystem
- Lizenzprogramme
- Sicherheitsdaten
- Konfigurationsdaten
- Alle von IBM gelieferten Bibliotheken
- Alle von IBM gelieferten Verzeichnisse

Die einzelnen Sicherungsbefehle können auch manuell auf der Befehlszeile eingegeben, oder in einem CL automatisiert ausgeführt werden.

Damit die Sicherung der Systemdaten – ob manuell als Befehl eingegeben oder in einem CL abgesetzt – erfolgreich durchgeführt werden kann, sind dieselben Voraussetzungen zu erfüllen, wie sie bereits bei der Verwendung der menügesteuerten Systemsicherung beschrieben wurden. Stellen Sie deshalb sicher, dass sich das System in einem eingeschränkten Zustand befindet und die Sicherung auf einem IPL-fähigen Band ausgeführt wird.

Nachfolgend finden Sie die einzelnen Sicherungsbefehle, die in Summe eine Systemsicherung ausmachen.

1. Beenden der aktiven Subsysteme (mit Ausnahme des steuernden Subsystems)

```
ENDSBS SBS(*ALL) OPTION(*IMMED)
```

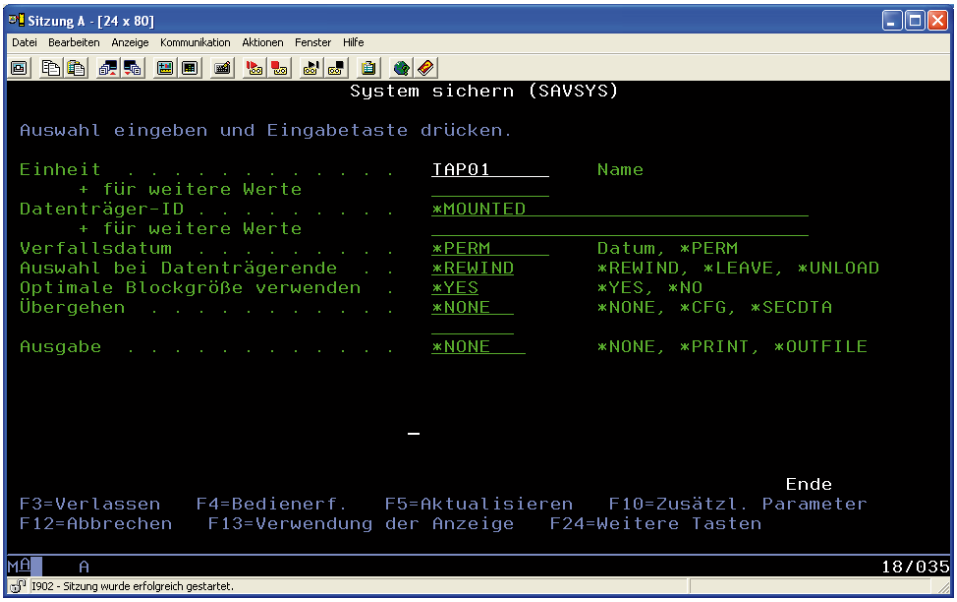
2. Anpassen der Zustellung der Nachrichten

```
CHGMSGQ MSGQ(QSYSOPR) DLVRY(*BREAK oder *NOTIFY)
```

3. Durchführung der Systemsicherung

```
SAVSYS
```

8.2.3.2
Seite 2



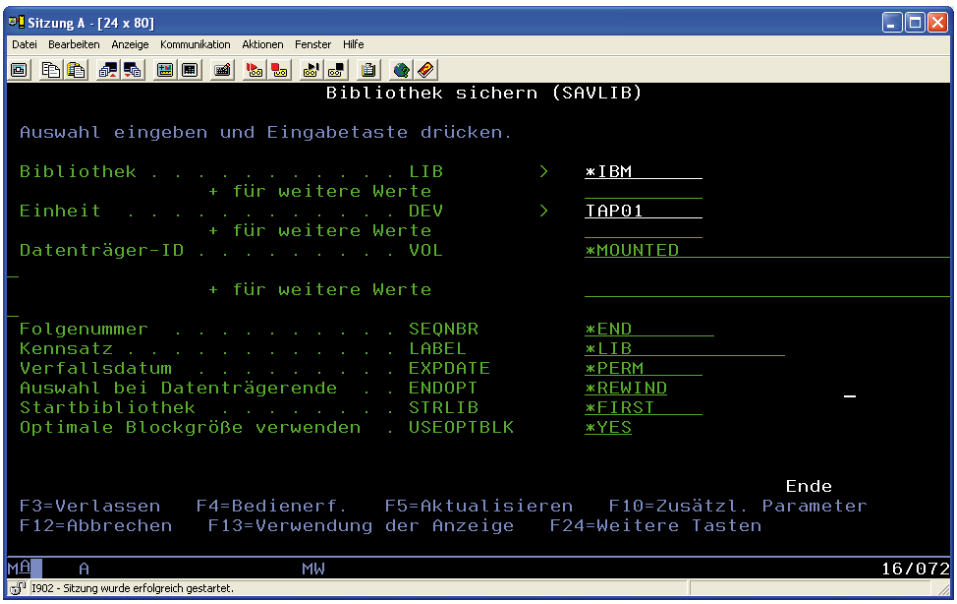
SAVSYS

Bei der Verwendung des Befehls SAVSYS stehen noch einige weitere Parameter zur Verfügung. Diese sehen wir erst nach Betätigen der Taste F10 (Zusätzliche Parameter):

ASP-Einheit (ASPDEV)
Dieser Parameter definiert die zu sichernden persönlichen Berechtigungen

4. Sicherung der IBM-Bibliotheken

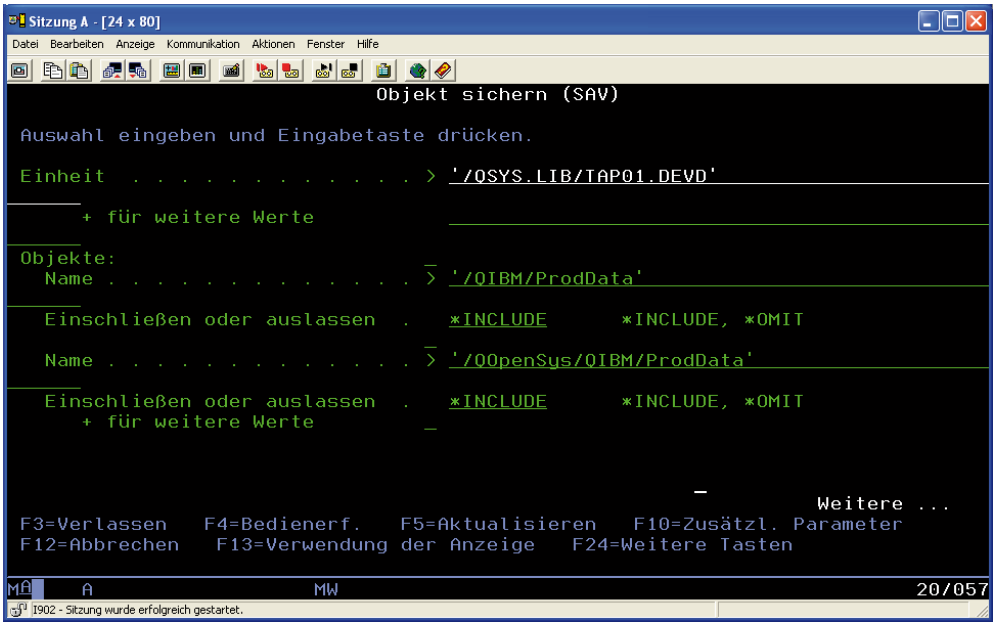
```
SAVLIB LIB(*IBM) DEV(TAP01) ACCPTH(*YES)
```



SAVLIB IBM

5. Sicherung der IBM-Verzeichnisse

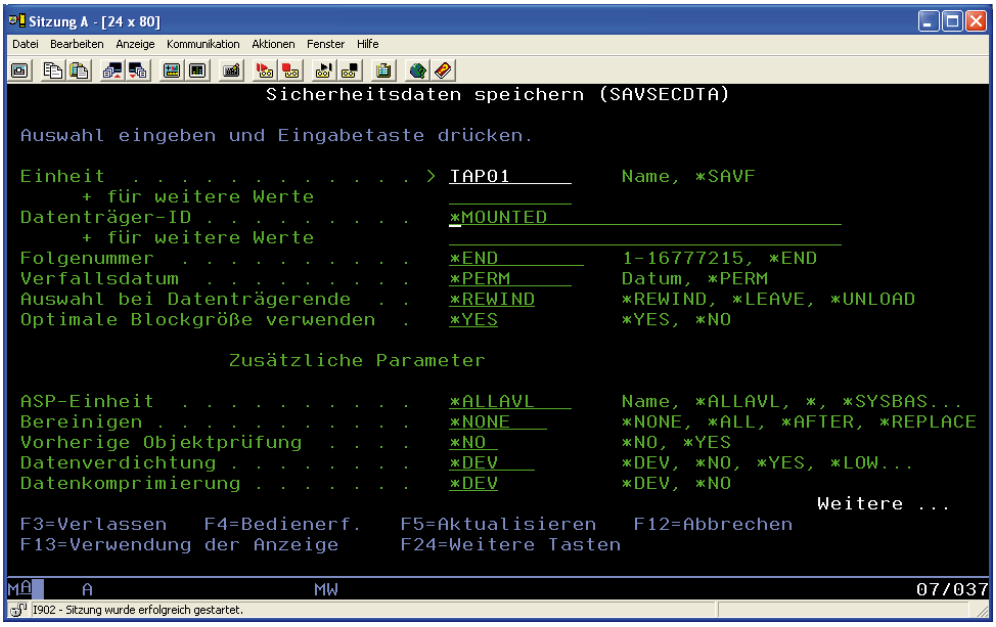
```
SAV DEV(,/QSYS.LIB/Name der Datensicherungseinheit.DEVD‘) +
OBJ((,/QIBM/ProdData‘) +(,/QOpenSys/QIBM/ProdData‘)) + UPDHST(*YES)
```



SAV

6. Sichern der Sicherheitsdaten

SAVSECDTA

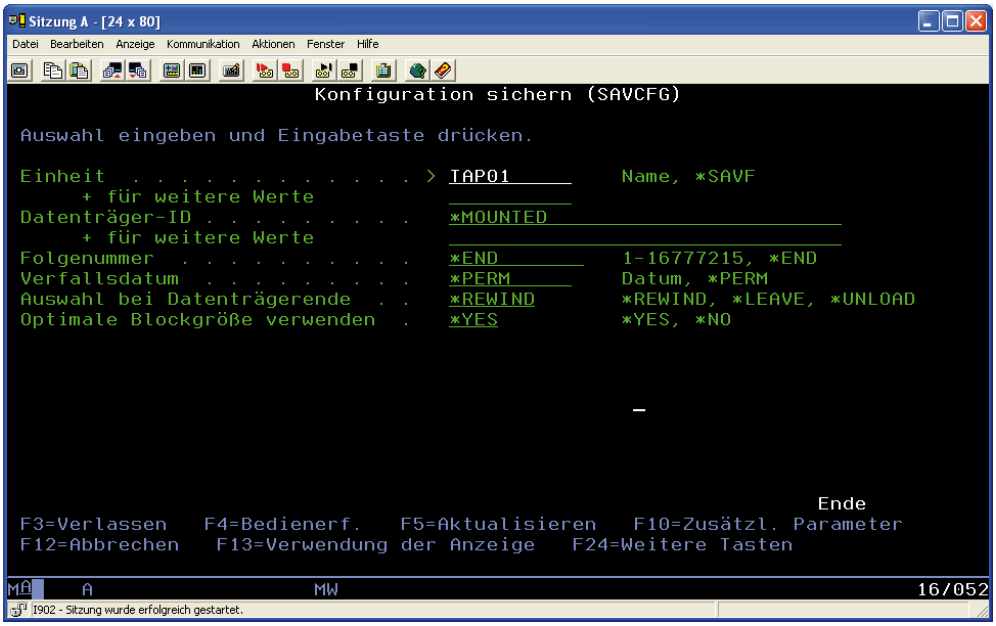


SAVSECDTA

8.2.3.2
Seite 4

7. Sichern der Konfigurationsdaten

```
SAVCFG
```



SAVCFG

8. Starten des steuernden Subsystems

```
STRSBS QCTL
```

Verschlüsselung von Bändern

Das regelmäßige Sichern der Daten ist ein MUSS für jedes Unternehmen. Und das Sicherstellen des Zugriffs auf alle unternehmensrelevanten Daten ist letztlich sogar eine Aufgabe, die von der Geschäftsleitung eines Unternehmens permanent zu gewährleisten ist.

Das Ziel einer Datensicherung ist in der Regel die Sicherung der Daten, die z. B. bei einem Systemausfall beschädigt werden könnten. Datensicherung hat aber nicht nur etwas mit der Sicherheit zu tun, Daten bei Bedarf wiederherstellen zu können. Datensicherung hat auch etwas mit Datenschutz zu tun. Haben Sie einmal daran gedacht, was wohl passiere würde, wenn Ihre Datensicherungsbänder in falsche Hände gelangen?

Immer ausgefallener werden die Varianten des Datenklaus. Und es ist ein Leichtes, ein Band (z. B. LTO) mal eben in eine Tasche zu stecken. So wunderbar die Tatsache auch ist, dass heute ein Band eine enorme Speicherkapazität aufweist – so nachteilig kann das auch sein, wenn Informationen unrechtmäßig entwendet werden. Nicht selten genügt nämlich ein einziges LTO-Band, auf dem alle Informationen eines Unternehmens gespeichert sind.

In der Tat – die Erfordernisse, den Inhalt des Bandes dann einzulesen und auszuwerten sind relativ umfangreich – aber letztlich genügt ein System i (oder ein Vorgänger) und ein Bandlaufwerk, das die Bänder lesen kann, und schon hat man auf Bibliotheks-, Objekt- oder Dokumentenebene den Zugriff auf die Daten. Dass man dazu lediglich auf dem Zielsystem die Berechtigung zum Zurückspeichern benötigt und dann mit dem QSECOFR mutmaßlich geschützte Objekte (ausgehend vom Einsatz einer Objektsicherheit) zugänglich machen kann, verdeutlicht, wie leicht ein solches Unterfangen ist.

Meist – und das verwundert – befinden sich die Daten auf den Datensicherungsbändern in ungeschützter Form, sprich unverschlüsselt. Dabei sind wir heute in der Lage, mit Hardware- und Software-Verschlüsselungsverfahren den Inhalt der Banddatenträger gegen unberechtigten Zugriff abzusichern.

Verschlüsselungen werden von IBM insbesondere mit den jüngsten Versionen des Systems IBM i angeboten. Diese Verschlüsselungen sind nicht nur für Datenbankinhalte selbst, sondern auch für Datensicherungen sinnvoll. Dabei liefern einige Datensicherungseinheiten schon seit geraumer Zeit die Möglichkeit, die Daten auf den Bändern bzw. Datenträgern zu verschlüsseln und damit sicherer zu machen. Nur, allzu wenige Unternehmen nutzen diese Funktion.

8.2.5
Seite 2

Seit Version 6.1 des IBM i besteht nun auch die Möglichkeit der Softwareverschlüsselung unabhängig von der Datensicherungseinheit. Während Hardwareverschlüsselung direkt an die Einheiten gebunden ist und nur von einigen Sicherungseinheiten unterstützt wird, steht die softwaregestützte Verschlüsselung für alle Datensicherungen zum Einsatz bereit.

Zu beachten ist allerdings, dass die Inhalte von Sicherungsdateien (SAVF) und optischen Einheiten nicht verschlüsselt werden können. Virtuelle Einheiten werden, wie Bandlaufwerke auch, voll unterstützt. Ebenso lassen sich die Bestandteile des Betriebssystems nicht verschlüsseln. Folglich können wir keine SAVSYS in verschlüsselter Form durchführen. Als Faustregel gilt, dass keine Q-Bibliotheken verschlüsselt werden können.

Außerdem zu bedenken ist: Der Einsatz von Verschlüsselung benötigt Rechenzeit, da die Verschlüsselungen vom Prozessor durchgeführt werden müssen. Das kann unter Umständen zu Performance-Beeinflussungen führen und die Laufzeit der Datensicherung nachteilig beeinflussen. Hier machen sich im Übrigen die neuen Prozessorgenerationen spürbar bemerkbar. Auch wenn Power 5 durchaus in der Lage ist, die Verschlüsselungen akzeptabel durchzuführen, sind die Power 6-Prozessoren deutlich leistungsfähiger und verkürzen die Sicherungs- bzw. Verschlüsselungszeit zum Teil erheblich! Bedenken Sie also die möglichen langen Laufzeiten einer Verschlüsselung, die insbesondere bei großen Datenmengen zum Tragen kommen!

Grundsätzlich gilt: Man sollte klar planen und festlegen, welche Daten und Informationen verschlüsselt gesichert werden sollen. Eine globale Verschlüsselung von Datensicherungen macht wirklich nur dann Sinn, wenn die Menge der zu sichernden Informationen überschaubar ist, das Zeitfenster für die Sicherung ausreichend ist und die Kapazität der Bänder ebenfalls ausreicht.

Bei der Entscheidungsfindung, welche Daten verschlüsselt gesichert werden sollen, sollten folgende Kriterien bedacht werden:

- Sensibilität der Daten.
  Müssen Daten geschützt werden, weil sie sensibel sind?
- Gibt es rechtliche Vorschriften, die einzuhalten sind?
- Ist die zur Verfügung stehende Zeit ausreichend, um die Daten verschlüsselt sichern zu können?
- Sind genügend Sicherungsdatenträger verfügbar, um die verschlüsselten Informationen sichern zu können?

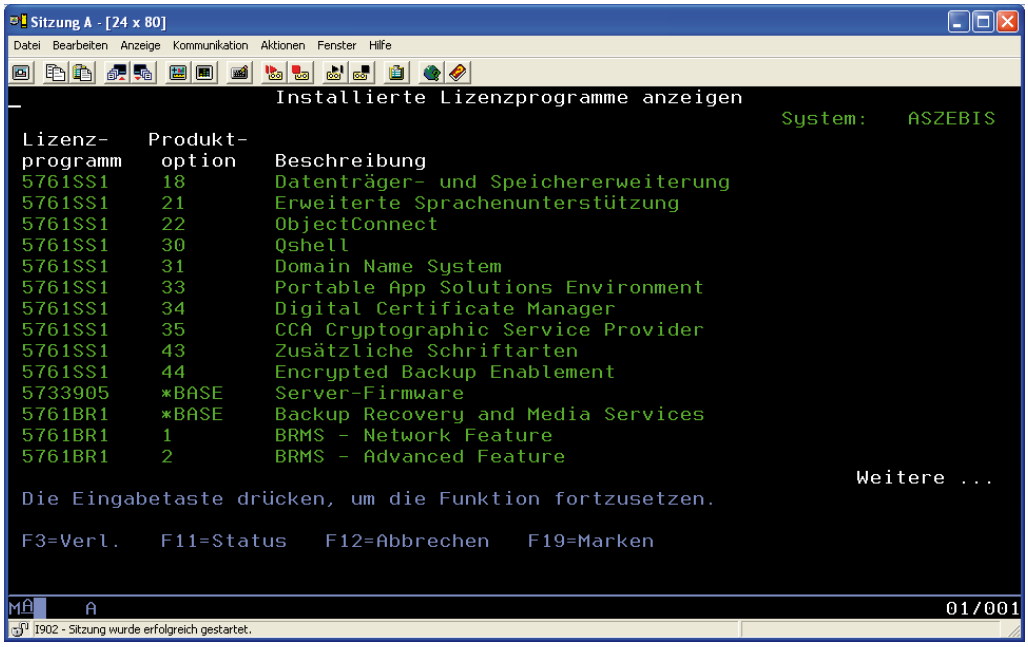
In der Praxis hat es sich bewährt, Inhalte von Dateien und Tabellen zu verschlüsseln. In der Regel enthalten diese Objekte die Daten und Informationen, die gegen einen unberechtigten Zugriff geschützt werden müssen. Programmobjekte hingegen sind in der Regel weniger sensibel – stellen sie doch nur ein Werkzeug zur Verarbeitung der Daten dar. Sind die Daten nicht vorhanden, dann ist meist auch das Programmobjekt wertlos.

Leider wird mit der Verschlüsselung ein Mechanismus nachteilig beeinflusst, der bei der Datensicherung dienlich sein kann, die Komprimierung der Daten auf dem Band. Aufgrund der Verschlüsselungsfunktionen kann die Komprimierung meist nur mit einem sehr niedrigen Faktor umgesetzt werden. Deshalb kann es durchaus sein, dass ein Datensicherungsprozess mit Verschlüsselung deutlich mehr Platz benötigt als eine unverschlüsselte Variante. Das kann besonders dann zu einem Problem werden, wenn nur ein Einzelbandlaufwerk zur Verfügung steht bzw. die Kapazität eines Laufwerks bzw. Bandes ausgeschöpft ist.

Sind diese Kriterien bedacht und ist der Entschluss gefasst, die Verschlüsselung der Datensicherung einsetzen zu wollen oder zu müssen, dann können Sie mit der Prüfung der Voraussetzungen beginnen.

Die Verschlüsselung basiert auf folgenden Lizenzprogrammen

– 5761-SS1 Option 18 (Media and Storage Extensions)
– 5761-SS1 Option 44 (Encrypted Backup Enablement)
– 5761-BR1 Option 2



Installierte Lizenzprogramme

Mit diesen Lizenzprogrammen fallen leider auch zusätzliche Kosten an. Kosten, die sich allerdings im Extremfall lohnen, denn damit werden die Datensicherungen auf den Banddatenträgern mit einem 256-Bit-Schlüssel gegen unberechtigten Zugriff abgesichert.

Ein Band, das mit einem solchen Schlüssel versehen ist, kann dann natürlich immer noch entwendet werden, ohne den entsprechenden Schlüssel zum Decodieren der Daten ist es dann jedoch quasi wertlos.

ACHTUNG!

Damit verbunden ist natürlich auch der Hinweis, dass der Schlüssel, der für die Verschlüsselung verwendet wurde, nicht verloren gehen darf. Anderenfalls sind die Datensicherungsbänder wertlos!

Aber wie funktioniert nun das Verschlüsseln mit Softwaremitteln?

Die Basis bildet der Exit-Point QIBM_QTA_TAPE_TMS oder innerhalb von BRMS das Programm Q1ATRMS. Mit dem Exit Point kann definiert werden, ob ein Objekt verschlüsselt gesichert werden soll oder nicht.

Bei einer verschlüsselten Sicherung wird dem Exit Point der Schlüssel bekannt gemacht, mit dem die zu sichernden Objekte verschlüsselt werden sollen. Das Bekanntmachen des Schlüssels besteht dabei aus den folgenden Informationen:

– Bibliothek, die den Schlüssel enthält.
– Name der Datei, in der sich der Schlüssel befindet.

Anmerkung

Der Exit Point ermöglicht es, dass man sich eigene Verschlüsselungsprogramme schreiben kann, mit denen die Verschlüsselung aktiviert und ausgeführt werden kann.

Wenn man die Verschlüsselung zusammen mit BRMS einsetzt, dann definieren wir die Schlüsselinformationen auf Basis der Einheitenrichtlinien.

Die Schlüsselinformationen werden innerhalb von BRMS in der Datei Q1AKEYFILE abgelegt, die sich in der BRMS-Benutzerbibliothek QUSRBRM befindet. Diese Datei kann nur mit speziellen Methoden verwaltet werden. Damit ist ein unberechtigter Zugriff auf den Inhalt der Schlüsseldatei nicht oder nur sehr schwer möglich. Um den Inhalt einsehen und auch verwalten zu können, benötigt man einen sogenannten Masterkey, der im LIC des Systems zu finden ist. Bis zu acht solcher Masterkeys werden vom System unterstützt.

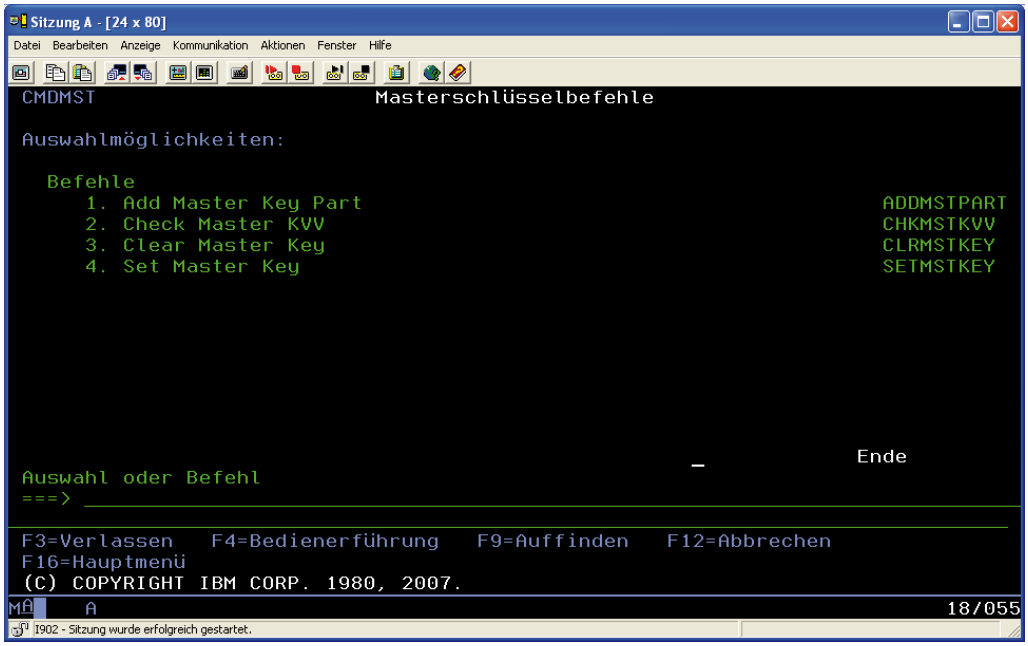
Zu beachten: Mit Version V5R4 lassen sich die Masterkeys nicht sichern! Diese Funktion wurde von IBM erst ab Version 6.1 eingeführt. Deshalb ist es unbedingt wichtig, darauf zu achten, dass beim Einsatz der Verschlüsselung unter V5R4 die Schlüssel sicher aufbewahrt werden und bei einem Wiederherstellungsszenario bedacht werden!

Die Bestandteile der Schlüsseldatei sind die einzelnen Schlüssel, die für unterschiedlichste Sicherungsprozesse genutzt werden können.

Schauen wir uns einmal die Verwaltung der Schlüsseldatei ein wenig genauer an.

Die Verwaltungsmöglichkeiten finden wir sowohl in der 5250-Umgebung als auch im System i Navigator.

Das Menü GO CMDMST sowie das Menü GO CMDCKM beinhalten die Befehle für die Schlüsselverwaltung.

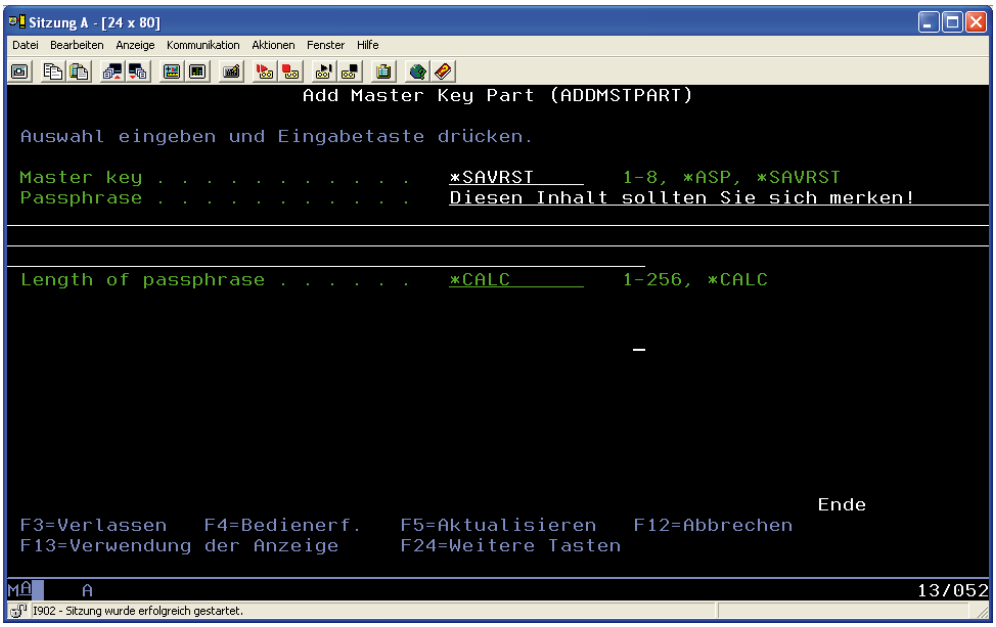


Menü CMDMST

8.2.5
Seite 6

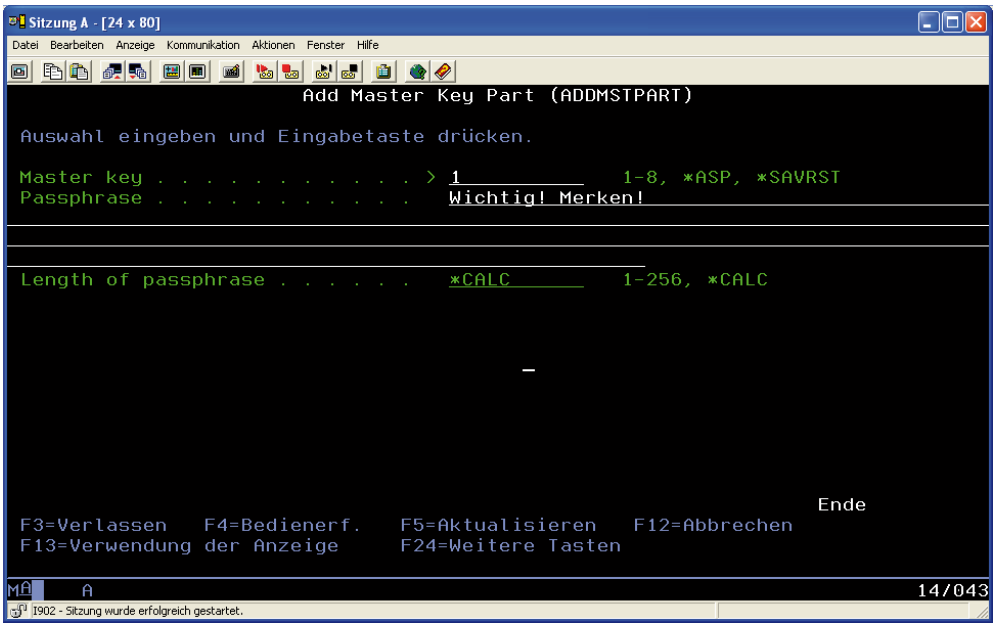
Es sind nur wenige Schritte, die für eine erfolgreiche Einrichtung der Softwareverschlüsselung erforderlich sind. Im Zusammenhang mit dem Einsatz von BRMS sind es folgende:

– Erstellen des Masterkeys
  Befehl ADDMSTPART



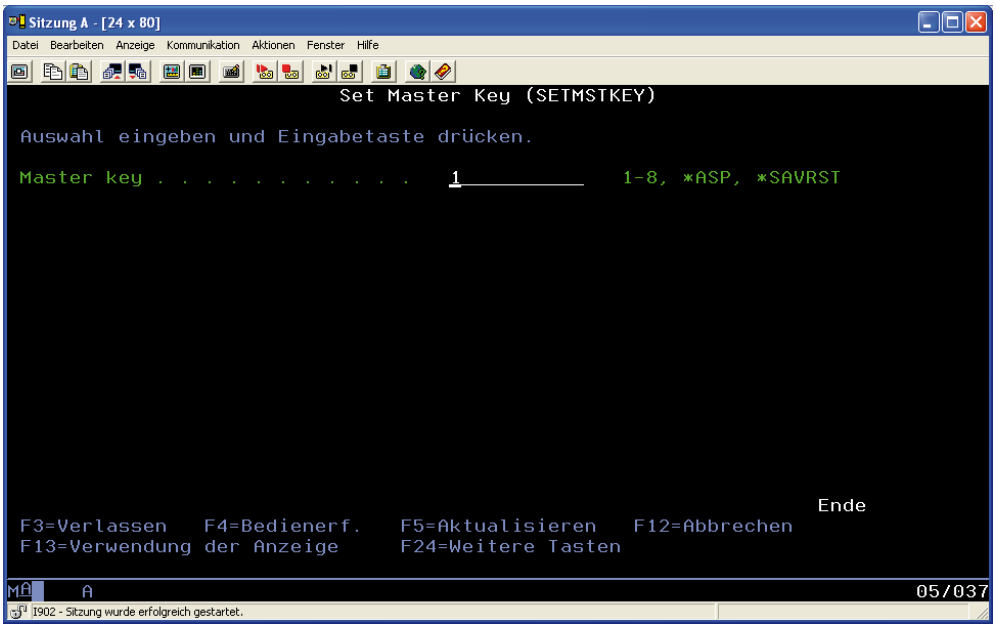
ADDMSTPART

– Jetzt definieren wir den Masterkey für die Schlüsseldatei. Dazu verwenden wir erneut den Befehl ADDMSTPART. Schauen Sie sich das folgende Beispiel dazu an:



ADDMSTPART 2. Ausführung

– Nun legen wir den Masterkey fest. Dazu nutzen wir den Befehl SETMSTKEY:



SETMSTKEY

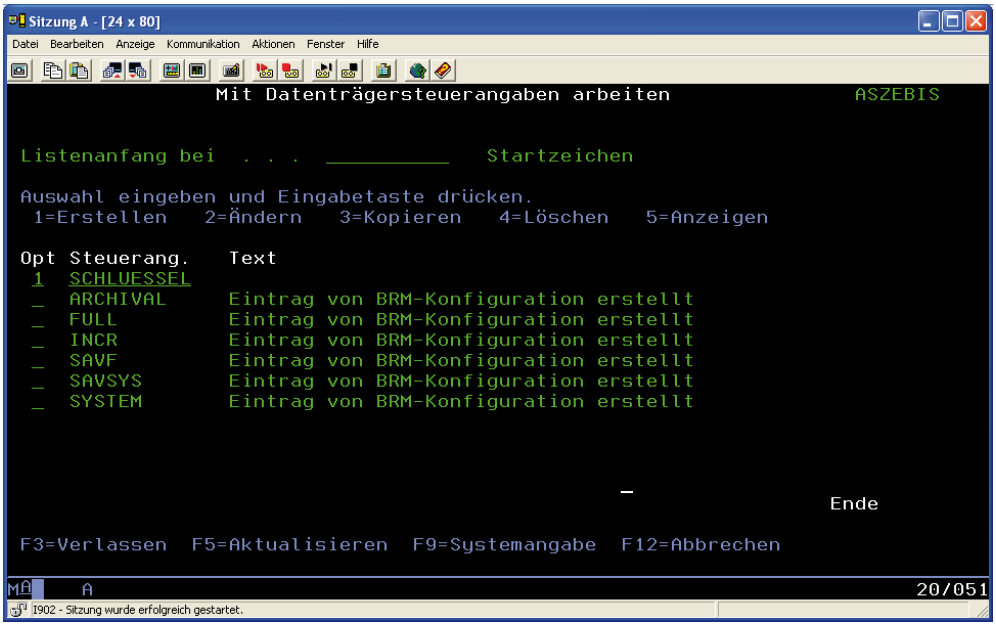
– Erstellen der Datei A1AKEYFILE in der Bibliothek QUSRBRM. Dabei handelt es sich um die Schlüsseldatei.
CRTCKMKSF KEYSTORE(QUSRBRM/Q1AKEYFILE) MSTKEY(1) AUT(*EXCLUDE) TEXT(‚Datei für Sicherungsschlüssel’)

– Erstellen eines neuen 256-Bit-AES-Schlüssels.
GENCKMKSFE KEYSTORE(QUSRBRM/Q1AKEYFILE) RCDLBL(KEY1) KEYTYPE(*AES) KEYSIZE(32)

  Im Parameter RCDLBL wird der Name des Schlüssels eingegeben. In unserem Beispiel habe ich „Key1“ verwendet.

– Anpassen der Einheitenrichtlinie.
WRKPCYBRM TYPE(*MED)

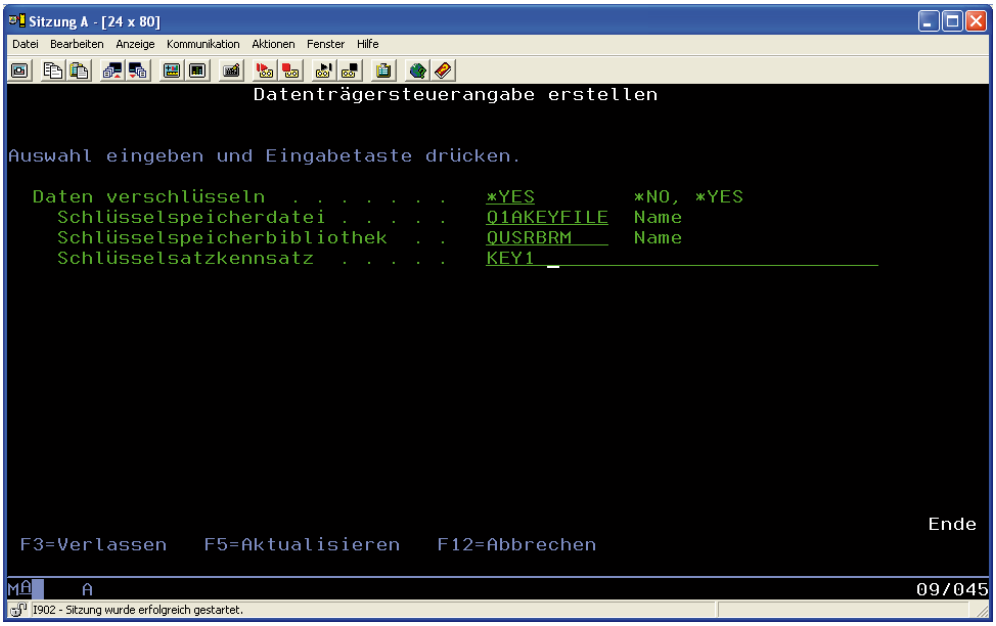
8.2.5
Seite 8



WRKPCYBRM

– In der Policy müssen die Verschlüsselungsinformationen in den dafür vorgesehenen Parametern eingegeben werden:

Encrypt Data . . . . . *YES
Key store file . . . . . Q1AKEYFILE Name
Key store library. . . . . QUSRBRM Name
Key record label KEY1



Schlüsselparameter

– Jetzt muss die BRMS-Sicherung noch an die neuen Einstellungen angepasst werden. Dazu muss in den Sicherungsrichtlinien die Angabe für das Verschlüsseln der Objekte definiert werden. Hier empfiehlt sich beispielsweise für die Verschlüsselung die Angabe *MEDPCY.

Die Verschlüsselungsinformationen in BRMS lassen sich mit dem Befehl WRKMEDIBRM einsehen. Dort finden wir nach Betätigen der F11-Taste u. a. auch den Hinweis auf den Schlüssel.

Bei der Arbeit mit den Schlüsseln ist natürlich eine entsprechende Sorgfalt wichtig. Nicht nur, das die Schlüssel nicht in fremde Hände gelangen sollten, denn dann wären zusammen mit den verschlüsselten Sicherungen auch die Informationen auf den Bändern wieder einsehbar – Sie sollten auch in gewissen Zeitabständen die Masterkeys ändern. Hier muss aber UNBEDINGT bedacht werden, dass dann alle Verschlüsselungen auf den neuen Masterkey abgestimmt werden müssen. Sind also Bandsicherungen mit einem Masterkey geschützt, der durch einen neuen Masterkey ersetzt wurde, dann müssen auch die Schlüssel für die Verschlüsselung mit dem neuen Masterkey versehen werden. Hört sich kompliziert an, ist es aber nicht – denn die Neuzuordnung wird über den Befehl TRNCKMKSF durchgeführt.

Solche Masterkey-Verwaltungsaktionen lassen sich natürlich auch automatisch durchführen, da alle Befehle auch mit APIs abgebildet werden können.

8.2.5
Seite 10

Wiederherstellung

Wie heißt es doch so schön:

„Eine Sicherung ist nur so gut wie die damit mögliche Wiederherstellung."

Wer einmal in der Verlegenheit gewesen ist, einen Teil eines Systems oder gar eine gesamte Installation mit Hilfe einer Datensicherung wieder aufspielen zu müssen, wird diese Aussage sicher bestätigen können.

Das IBM i bietet mit Bordmitteln verschiedene Sicherungsverfahren an, mit denen einzelne Teile einer Installation, aber auch das gesamte System wiederhergestellt werden können.

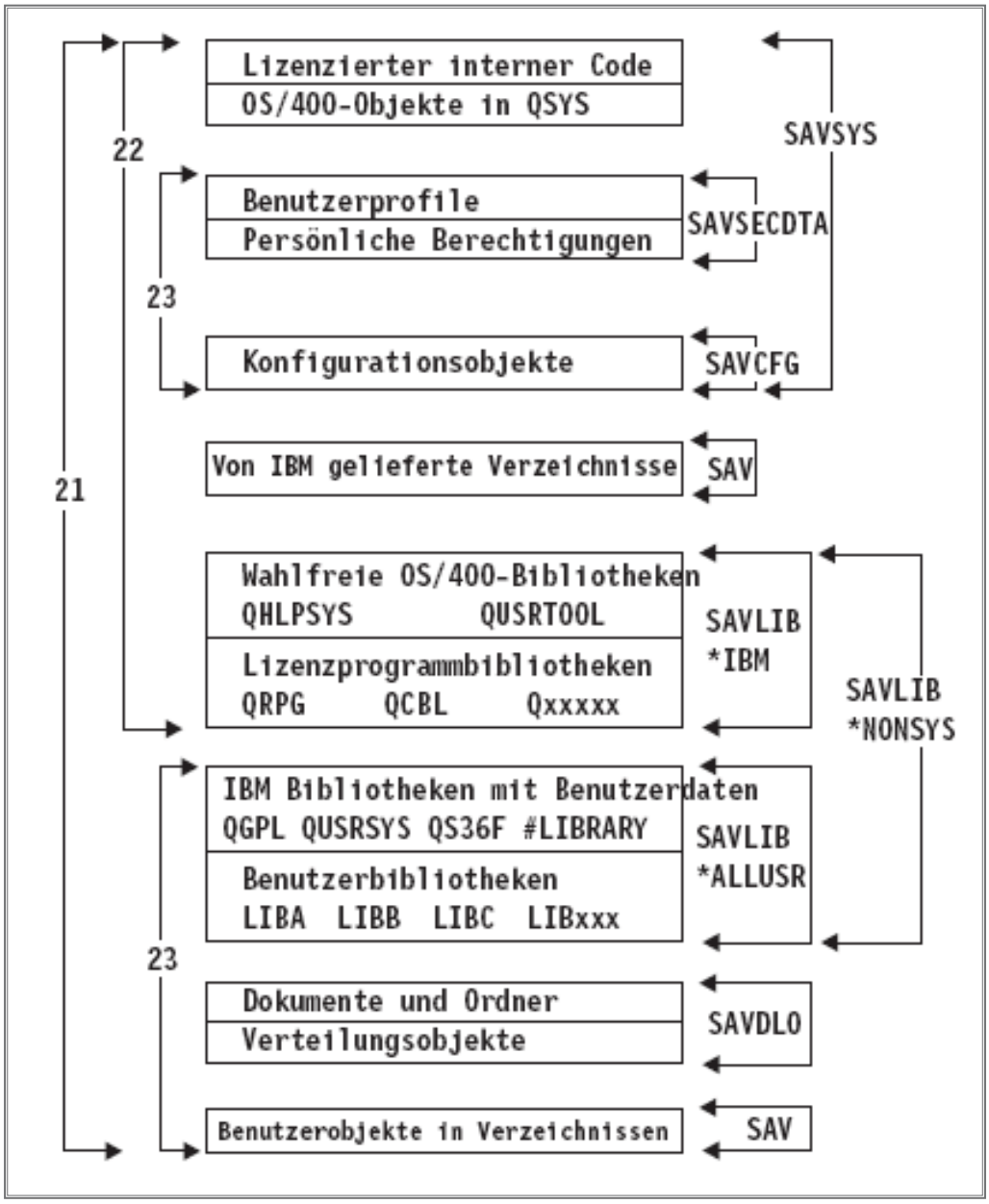
Grundsätzlich muss im Wiederherstellungsfall entschieden werden, welche Informationen zurückgespeichert werden müssen. Dabei sind natürlich die Abhängigkeiten der einzelnen Bereiche unbedingt zu beachten. Leider ist es nur selten der Fall, dass Informationen, Daten oder Programme in einer einzelnen Bibliothek enthalten sind. Vielmehr sind Abhängigkeiten zu anderen Bibliotheken, Inhalten im IFS, Berechtigungen und vieles mehr zu beachten.

Im Folgenden finden Sie Informationen zur Wiederherstellung eines gesamten Systems mit Hilfe einer Datensicherung. Voraussetzung dafür ist natürlich das Vorhandensein einer entsprechenden Datensicherung. Diese muss ALLE Teilbereiche abdecken, die für das Aufbauen der Maschine und der Anwendungen benötigt werden. Dabei ist es zunächst unerheblich, ob die Informationen in einer oder mehreren Sicherungen enthalten sind. Wichtig sind die Vollständigkeit und das Zusammenpassen der einzelnen Sicherungsbereiche. In dem gezeigten Beispiel gehen wir davon aus, dass das IBM i mit Hilfe einer Gesamtsicherung (GO SAVE – Auswahl 21) gesichert wurde. Damit können nahezu alle Informationen der Maschine gesichert werden.

Diese Funktion kann nur von der Systemkonsole ausgeführt werden. Dabei ist darauf zu achten, dass sich das System in einem eingeschränkten Zustand befindet.

23. Ergänzung 6/2010

8.3
Seite 2



Wiederherstellung (Quelle: IBM)

Analog zu den Sicherungsbefehlen liefert IBM im Wiederherstellungsmenü GO RESTORE die erforderlichen Befehle.

Für die Wiederherstellung einer Maschine ist zunächst zu entscheiden, ab welcher Stufe der Architektur des Systems die Wiederherstellung erfolgen soll. Sind lediglich Benutzerdaten wiederherzustellen, dann können dazu die Befehle RSTLIB, RSTOBJ, RSTDLO oder RST genutzt werden. Diese sind in den Sicherungsprozeduren der Gesamtsicherung enthalten.

Systemwiederherstellung

Nachfolgend nun eine Anleitung für die vollständige Wiederherstellung eines Systems mit Hilfe einer Datensicherung. Diese Form der Sicherung bedingt eine vorangegangene Gesamtsicherung. Bevor dieses Wiederherstellungsverfahren verwendet wird, muss man sich über die Auswirkungen wirklich im Klaren sein. Das Wiederherstellen einiger IBM-i-Teilbereiche initialisiert das System – damit verbunden ist dann der Verlust ALLER auf den Platten befindlichen Informationen. In den Sicherungs- und Wiederherstellungshandbüchern der IBM finden wir ausreichend Hinweise und Möglichkeiten zur Wiederherstellung des Systems. Wenn wirklich der außergewöhnliche Fall der vollständigen Systemwiederherstellung erforderlich sein sollte, dann kann das in diesem Kapitel beschriebene Verfahren angewendet werden.

Mögliche Gründe für eine vollständige Systemwiederherstellung sind:

- Hardwaredefekt
- Fehler im Betriebssystem
- Fehler in Lizenzprogrammen
- Datenbankprobleme
- Installation einer neuen Hardware
- Disaster Recovery

Es ist den Eigenschaften des IBM i zu verdanken, dass eine vollständige Systemwiederherstellung zu den wirklichen Ausnahmen im Leben eines IT-Verantwortlichen bzw. eines Administrators gehören. Deshalb sind die hier aufgeführten Teilschritte auch keine gewöhnlichen Funktionen, die im täglichen IBM-i-Leben ausgeführt werden müssen.

23. Ergänzung 6/2010

8.3.1
Seite 2

Vollständige Systemwiederherstellung

Bei der vollständigen Wiederherstellung muss zunächst ein IPL vom Band durchgeführt werden. Dabei ist es wichtig, dass die Bandeinheit als alternative Loadsource angeschlossen ist. Dies ist durch den IBM-Servicegeber zu prüfen.

Die alternative Load Source kann im DST/SST wie folgt kontrolliert werden:

- Starten Sie SST
- Wählen Sie Option 1 (SST)
- Wählen Sie Option 7 (Hardware Service Manager)
- Betätigen Sie in der Anzeige „Hardware Service Manager“ die Taste F6 = Drucken
- Übernehmen Sie die Voreinstellungen
- Suchen Sie in dem erzeugten Ausdruck nach dem %-Zeichen. Dies ist die alternative Load Source.

Es kann Fälle geben, in denen das Ermitteln auf diese Weise nicht funktioniert. Dann gilt die Regel, dass die Einheit am Bus 1 des Systems angeschlossen sein muss. Diese Informationen erhalten Sie auch im SST/Hardware Service Manager.

1. Durchführen eines manuellen D-IPL.
   Stellen Sie das Display der Maschine auf 01 D M. Je nach eingesetzter Maschine ist das anzuwendende Einstellungsverfahren unterschiedlich. Beispielhaft dazu das Einstellen bei einer Nicht-LPAR 520:
   - Betätigen Sie die Pfeil-auf-Taste am Display der Maschine – es erscheint die Anzeige „02“.
   - Bestätigen Sie mit der Eingabetaste.
   - Blättern Sie mit einer der Pfeiltasten soweit, bis die Anzeige „D“ erscheint.
   - Bestätigen Sie mit der Eingabetaste.
   - Blättern Sie mit einer der Pfeiltasten soweit, bis „M“ in der Anzeige steht.
   - Bestätigen Sie mit der Eingabetaste.
   - Blättern Sie mit der Pfeiltaste in der Anzeige „02“ zurück auf den Wert „01“ und bestätigen Sie mit der Eingabetaste.
2. Legen Sie das Band mit der Gesamtsicherung ein.
3. Schalten Sie die Konsole ein.
4. Starten Sie die Maschine mit dem Einschaltknopf.

8.3.1.1
Seite 2

5. An der Konsole erscheint nach einigen Minuten die folgende Anzeige:

```
                    Lizenzierten internen Code (LIC) installieren

 Ausgewählte Platte, auf die der LIC geschrieben werden soll:
     Seriennummer      Typ      Modell    E/A-Bus      Steuereinheit  Einheit
      xx-xxxxxxx       xxxx      xxx         x             x           x

 Auswahlmöglichkeiten:

      1. LIC zurückspeichern
      2. LIC installieren und System initialisieren
      3. LIC installieren und Konfiguration wiederherstellen
      4. LIC installieren und Platteneinheitendaten zurückspeichern
      5. LIC installieren und Ladequelle erweitern
```

LIC installieren

6. Wählen Sie Option 2 aus. ACHTUNG!!! Damit werden alle Daten auf dem System zerstört!!!!
7. Wählen Sie in der Folgeanzeige die Platte für das Installieren des LIC. Diese wird voreingestellt. Die Installation kann längere Zeit dauern!
8. Nach Installation des LIC müssen im DST alle Platten dem System hinzugefügt werden. Dazu wählen wir im DST die Option 4 „Work with disk units“. Fügen Sie dem System alle Platteneinheiten hinzu. Achten Sie dabei auf den zu verwendenden Plattenschutz (RAID, Spiegelung).
9. Sollte ein Hinweis kommen, dass die Platteneinheiten bereits anders konfiguriert wurden, können Sie diesen mit der Taste F10 übergehen.
10. Nach Abschluss des Hinzufügens der Platten folgt nun die Installation des Betriebssystems. Dazu wählen wir die Auswahl „Install the operating system“. Damit wird das Basisbetriebssystem zurückgespielt. Befolgen Sie die Anweisungen auf dem Bildschirm. Beachten Sie bei der Installation die korrekte Sprache. Die deutsche Version wird mit dem Sprachcode 2929 installiert!
11. Weitere Angaben sind u. a. das aktuelle Datum. Geben Sie die erforderlichen Werte ein.

```
                        Betriebssystem installieren

Auswahl eingeben und Eingabetaste drücken.

     Installations-
       auswahl  . . .  __              1=Verwenden der Standardwerte (keine wei-
                                         teren Auswahlmöglichkeiten ...
                                       2=Ändern der Auswahlmöglichkeiten

Datum:
  Jahr . . . . .  __                   00-99
  Monat  . . . .  __                   01-12
  Tag  . . . . .  __                   01-31

Uhrzeit:
  Stunde . . . .  __                   00-23
  Minute . . . .  __                   00-59
  Sekunde  . . .  __                   00-59
```

Betriebssystem installieren

Das System speichert nun die Basisdaten zurück.

```
Nachrichten-ID  :   CPI2070                    OS/400-Installationsstatus

Ab-        +-----------------------------------------------------+
schnitt    |XXXXXXX                                              |
           +-----------------------------------------------------+
           0       20        40        60        80        100

Installations-                                                      Objekte
abschnitt                                              Beendet      zurückgesp.

   1  Erstellen der Profile und Bibliotheken . . . . :     X

>> 2  Zurückspeichern der Programme in QSYS  . . . . :                XXXX

   3  Zurückspeichern der Sprachenobjekte in QSYS  . :

   4  Aktualisieren der Programmtabelle  . . . . . :

   5  Installieren der Datenbankdateien  . . . . . :

   6  Beenden der iSeries-Installation  . . . . . . . . :
```

Installationsstatus

8.3.1.1
Seite 4

12. Sobald der Anmeldebildschirm erscheint, melden Sie sich mit QSECOFR an. Dieser ist NEU und hat kein Kennwort (bzw. das Standardkennwort QSECOFR, das geändert werden muss).

```
                    Kennwort ändern

Kennwort zuletzt geändert . . . . xx.xx.xx

Auswahl eingeben und Eingabetaste drücken.

   Aktuelles Kennwort  . . . . . . . QSECOFR

   Neues Kennwort  . . . . . . . . . _______

   Neues Kennwort (zur Überprüfung)  _______
```

Kennwort ändern

13. Nun erscheint ein Bildschirm, der die Möglichkeit bietet, mit PTFs zu arbeiten. Diesen können wir mit der F3-Taste verlassen.

```
            Produkte zum Arbeiten mit PTFs auswählen

Listenanfang bei . . . . . .  ____________ Produkt


Auswahlmöglichk. eing., Eingabetaste drücken. Mit F21 alle auswählen.
   1=Auswählen

Aus-          Produkt-
wahl Produkt  option  Release
_    5769999  *BASE   V4R5M0
_    5769SS1  *BASE   V4R3M0
```

Mit PTFs arbeiten

14. Die Folgeanzeige „IPL-Auswahlmöglichkeiten“ erlaubt Definitionen für den IPL. Hier kann festgelegt werden, ob Informationen automatisch angelegt oder später von der Sicherung zurückgespeichert werden sollen. Bestätigen Sie die Anzeige mit der Datenfreigabe. Es empfiehlt sich die folgende Einstellung:
    - Druckausgabe starten „N“
    - System starten eingeschränkter Betrieb „J“
    - Hauptangaben bei IPL festlegen „J“
    - System bei IPL definieren „J“
15. Unter Umständen wird eine Anzeige „Zugriffspfadwiederherstellung“ angezeigt. Diese können Sie mit der Eingabetaste übergehen.

16. Nachdem die Anmeldeanzeige ausgegeben wird, ist das Basisbetriebssystem wiederhergestellt. Nun können Sie wahlweise die einzelnen Wiederherstellungsschritte manuell ausführen, oder mit Hilfe des Menüs GO RESTORE und der Auswahl 21 das gesamte System zurückspeichern.

    Damit werden die folgenden Wiederherstellungsschritte ausgeführt:
    - ENDSBS SBS(*ALL) OPTION(*IMMED)
    - RSTUSRPRF USRPRF(*ALL)
    - RSTCFG OBJ(*ALL)
    - RSTLIB SAVLIB(*NONSYS)
    - RSTDLO DLO(*ALL) SAVFLR(*ANY)
    - RST DEV('/QSYS.LIB/Name der externen Einheit.DEVD')
      OBJ(('/*') ('/QSYS.LIB' *OMIT) ('/QDLS' *OMIT))
    - RSTAUT
    - STRSBS SBSD(steuerndes Subsystem)

Nach jedem einzelnen Wiederherstellungsschritt sollte eine Prüfung erfolgen, ob die Wiederherstellung auch wirklich erfolgreich war. Dazu bedienen wir uns in der Regel der Einträge aus dem Jobprotokoll, die automatisch während des Rücksicherns erstellt werden. Fehlerhafte Rücksicherungsprozeduren werden im Protokoll neben dem erfolgreichen Abschluss von Teilaktionen aufgenommen.

Abschließend muss das Systempanel wieder auf 01 B N gestellt werden. Dabei ist dasselbe Verfahren anzuwenden, das zuvor beim Einstellen des manuellen Modus beschrieben wurde.

8.3.1.1
Seite 6

BRMS

Die Sicherung der Daten eines Systems ist die Grundlage für jede Wiederherstellung. Sei es in einem geplanten Fall, bei dem beispielsweise die Daten von einem System auf ein anderes übertragen werden sollen, oder wenn Informationen in eine Testumgebung gespielt werden müssen.

Die mit dem Betriebssystem gelieferten Datensicherungsmöglichkeiten sind weitreichend und erlauben es dem Administrator, individuell die Bestandteile der Maschine so zu sichern, dass in jedem Fall eine Wiederherstellung ganz oder teilweise ermöglicht wird.

Die steigende Komplexität der Systeme, die Integration verschiedener Betriebssysteme auf einer physischen Maschine und die Problematik der für eine Datensicherung zur Verfügung stehenden Zeit sind Faktoren, die den Einsatz der Bordmittel des IBM i in Bezug auf die Sicherung und Wiederherstellung an die Grenzen gelangen lassen. Nicht, dass damit die Möglichkeiten der Datensicherung erschöpft seien (man kann durchaus mit Bordmitteln auch in hochkomplexen Installationen die Sicherungs- und Wiederherstellungsprozesse ausführen), aber die Prüfbarkeit und die komplette Administration der verschiedenen Datensicherungsprozesse und deren Abstimmung auf die vorhandene Datensicherungsinfrastruktur stellen häufig eine große Herausforderung an die Verantwortlichen dar.

Nehmen wir nur eine banale, aber immens wichtige Frage: „Sichern Sie alles Notwendige so, dass Sie in der kürzest möglichen Zeit in der Lage sind, das gesamte System wieder aufbauen zu können?“ oder: „Sind Sie in der Lage, Bestandteile des Systems auf Anforderung zurück zu speichern?“

Die Antworten auf beide Fragen sollten in jedem Fall „Ja“ lauten.

Nun finden wir heute in zahlreichen System i-Umgebungen größere Datenmengen. Auch das Anwachsen der Kapazitäten der Datensicherungsmedien verhindert nicht, dass für die Sicherung der Datenmengen komplexere Systeme zur Sicherung zum Einsatz kommen. Bandroboter oder Bibliothekssysteme bieten heute die Möglichkeit, elegant und effektiv die Sicherungsmedien so einzusetzen, dass diese für die verschiedensten Systeme genutzt werden können. Dazu erforderlich ist dann allerdings eine gewisse Managementsoftware, die es erlaubt, die Bänder den Systemen zuzuordnen und Informationen darüber zu liefern, welche Bestandteile eines Systems wann und wo gesichert wurden. Dass solche Systeme dann auch Informationen darüber liefern, welche Bestandteile des Systems gesichert wurden und welche unter Umständen nicht in einem Sicherungsverfahren enthalten sind, ist eine Funktion, die sicher den einen oder anderen Verantwortlichen besser schlafen lässt.

IBM bietet für die Durchführung von komplexen Sicherungen oder auch Sicherungen in heterogenen Umgebungen ein Lizenzprogramm an, mit dem

24. Ergänzung 9/2010

8.5
Seite 2

der Betreiber eines System i in der Lage ist, basierend auf den klassischen IBM i Sicherungs- und Wiederherstellungsbefehlen Zusatzfunktionen zu nutzen, welche die Administration in dem Bereich Sicherung und Wiederherstellung vereinfachen. BRMS – die Abkürzung für Backup Recovery and Media Services für IBM i.

Berichte und Assistenten, die einem im Fall der Fälle detaillierte Auskunft über die durchzuführenden Aufgaben geben, sind in BRMS ein wichtiger Bestandteil. Basierend auf den BRMS-Protokolldaten, die bei jedem Sicherungslauf innerhalb von BRMS je nach vorgegebener Detailstufe mitgeschrieben werden, lassen sich unterschiedliche Berichte drucken, die zum Beispiel jeden einzelnen Teilschritt erläutern, der bei dem Neuaufbau eines kompletten Systems mit den BRMS-Datensicherungen durchzuführen ist.

Ebenso sind besondere Sicherungen in Form von zum Beispiel Save-While-Active oder inkrementelle Sicherungsverfahren mit BRMS möglich, zielgerichtet auf moderne Unternehmen, bei denen das Sicherungszeitfenster wichtig ist und Ausfallzeiten für Datensicherungen so gering als möglich ausfallen dürfen. Doch dazu später mehr.

BRMS „Backup Recovery and Media Services für System i“ ist ein kostenpflichtiges Lizenzprogramm, das auf dem System i in Form von drei unterschiedlichen Optionen angeboten wird:

57xx-BR1	*BASE	Backup Recovery and Media Services (Basis)
57xx-BR1	1	BRMS Network Feature
57xx-BR1	2	BRMS Advanced Feature

```
                    Installierte Lizenzprogramme anzeigen
                                                          System:   ASZEBIS
Lizenz-    Status der
programm   Install.     Beschreibung
5722SS1    *COMPATIBLE  Erweiterte Sprachenunterstützung
5722SS1    *COMPATIBLE  ObjectConnect
5722SS1    *COMPATIBLE  Integrated Server Support
5722SS1    *COMPATIBLE  Qshell
5722SS1    *COMPATIBLE  Domain Name System
5722SS1    *COMPATIBLE  Portable App Solutions Environment
5722SS1    *COMPATIBLE  Digital Certificate Manager
5722SS1    *COMPATIBLE  CCA Cryptographic Service Provider
5722SS1    *COMPATIBLE  International Components for Unicode
5722SS1    *COMPATIBLE  Zusätzliche Schriftarten
5722BR1    *COMPATIBLE  Backup Recovery and Media Services
5722BR1    *COMPATIBLE  BRMS - Network Feature
5722BR1    *COMPATIBLE  BRMS - Advanced Feature
5722BZ1    *COMPATIBLE  IBM Business Solutions
                                                          Weitere ...
Die Eingabetaste drücken, um die Funktion fortzusetzen.

F3=Verl.    F11=Release    F12=Abbrechen    F19=Marken
```

LICPGM

Zusammen mit der Option 18 des 57xx-SS1 (Media and Storage Extensions) bieten diese Lizenzprogrammversionen unter anderem folgende Funktionen:

- Durchführung von Sicherung und Wiederherstellung
- Bandmanagement
- Lokationsmanagement
- Prüffunktionen
- Save While Active
- Sichern von Spooldateien (nicht nur auf Basis von OUTQ)
- Basierend auf BRMS-Datenbankkomponenten Wiederherstellungsleitfaden
- Systemübergreifende Sicherung

Schauen wir uns einmal die drei verschiedenen BRMS-Optionen und deren Inhalte an:

57xx-BR1 *BASE
Mit dieser in jedem Fall zu installierenden Option stehen die Sicherungs- und Wiederherstellungsfunktionen von BRMS für den Einsatz auf einem System i zur Verfügung. In dieser Option sind keine Archivfunktionen enthalten.

57xx-BR1 Option 1 (BRMS Network Feature)
Mit dieser Option kann BRMS in einer Installation mit mehreren Maschinen (oder auch LPAR) eingesetzt werden. Dabei bildet eines der Systeme die zentrale Datenbank für die BRMS-Informationen (QUSRBRM), welche dann von den übrigen Maschinen unter dem Namen des jeweiligen Systems dort eingestellt werden. Mit dieser Funktion lassen sich zum Beispiel Einheiten von mehreren Maschinen nutzen und auch Datenträger über die Systeme hinweg administrieren.

57xx-BR1 Option 2 (Advanced Function)
Die wesentlichen Funktionen beziehen sich auf die hierarchische Speicherverwaltung (kurz: HSM). Diese Funktion wird im Zusammenhang mit der „Archivierung“ eingesetzt und erlaubt es, Daten in Abhängigkeit einer vorherigen Definition auf weniger leistungsstarken oder billigeren Speichervarianten abzulegen. Denkbar wäre solch ein Einsatz für die Bereitstellung von Inventurdaten oder historischen Daten, die nicht mehr auf den eigentlichen Systemplatten bereitstehen müssen, sondern durchaus auch auf einem langsameren externen ASP vorgehalten werden können. Die Werkzeuge für diese Speicherverwaltung finden wir in dieser BRMS-Option. Diese Option sollte nicht mit der Archivierung im klassischen Sinne verwechselt werden!

24. Ergänzung 9/2010

Im Nachfolgenden möchte ich Ihnen BRMS vorstellen.

Anmerkung

Diese Erläuterung zu dem Einsatz von BRMS soll keinen Ersatz für ein Datensicherungs- und Wiederherstellungskonzept darstellen, sondern lediglich die wesentlichen Funktionen von BRMS beschreiben. Welche Daten wann, wie, wo und wie oft gesichert werden sollen, bedarf natürlich auch bei der Nutzung von BRMS einer entsprechenden Planung!

Bevor wir uns den Einsatz von BRMS anschauen, noch einige Hinweise.

BRMS erweitert die Datensicherungs- und Wiederherstellungsmöglichkeiten auf dem System i. Anforderungen und Einschränkungen, wie sie bei nativen Datensicherungsmethoden zum Einsatz kommen, gelten auch in vielen Bereichen von BRMS.

Mit dem Einspielen von BRMS wird unter anderem eine Bibliothek QUSRBRM auf dem System installiert. Diese Bibliothek beinhaltet eine Vielzahl an Tabellen, welche für BRMS-Protokolle und -Einstellungen genutzt werden. Diese Bibliothek ist unter anderem Bestandteil einer jeden BRMS-Datensicherung – beinhaltet sie doch Informationen, die für eine auf BRMS-basierte Wiederherstellung bzw. Rücksicherung von Bedeutung sind.

BRMS-Datensicherungen basieren auf nativen Sicherungsbefehlen, die um bestimmte Funktionen erweitert wurden. Meist stehen für native Sicherungs- und Wiederherstellungsbefehle BRMS-Versionen zur Verfügung. Diese sind in der Regel um „BRM“ erweitert.

Beispielsweise können wir mit BRMS Bibliotheken mitsamt deren Inhalt mit dem Befehl SAVLIBBRM sichern. Wie Sie sehen, wurde der native Sicherungsbefehl SAVLIB um „BRM“ erweitert. Dieser „neue“ Befehl beinhaltet unter anderem Protokollfunktionen, mit denen die BRMS-Tabellen bestückt werden.

BRMS beinhaltet Erweiterungen zu den bestehenden Datensicherungsoptionen des IBM i. Dabei wird leider zu oft übersehen, dass diese Optionen auch konfiguriert werden müssen. In der Tat beinhaltet BRMS schon seit geraumer Zeit die Möglichkeit, Inhalte von Spooldateien zu sichern (nicht nur den Inhalt von OUTQs, wie es seit V5R4 auch nativ möglich ist). Leider ist diese Funktion aber nicht mit der Installation von BRMS aktiviert, sondern muss mit nur wenigen Einstellungsanpassungen eingeschaltet werden. Diese und noch einige andere Tipps und Hinweise finden Sie in dieser Beschreibung.

Wenn man sich für den Einsatz von BRMS entschlossen hat, dann kommt man nicht umhin, die bisherigen Datensicherungspraktiken zu analysieren. Sicher wird man dann auch im praktischen Einsatz hier und da Situationen haben, bei denen „mal schnell“ eine Sicherung außerhalb von BRMS durch-

geführt werden könnte. Aber: Der Einsatz von BRMS unter dem Aspekt, dass das System bei der Sicherung und bei einer möglichen Wiederherstellung einen sicheren und lückenlosen Leitfaden liefert, bedarf einer eben solchen lückenlosen Nutzung von BRMS und keiner „mal eben Nicht-BRMS-Sicherung". Das ist in der Praxis kein wirkliches Problem – stehen in BRMS doch die notwendigen „Schnellsicherungsfunktionen" ebenso zur Verfügung, wie nativ mit dem IBM i.

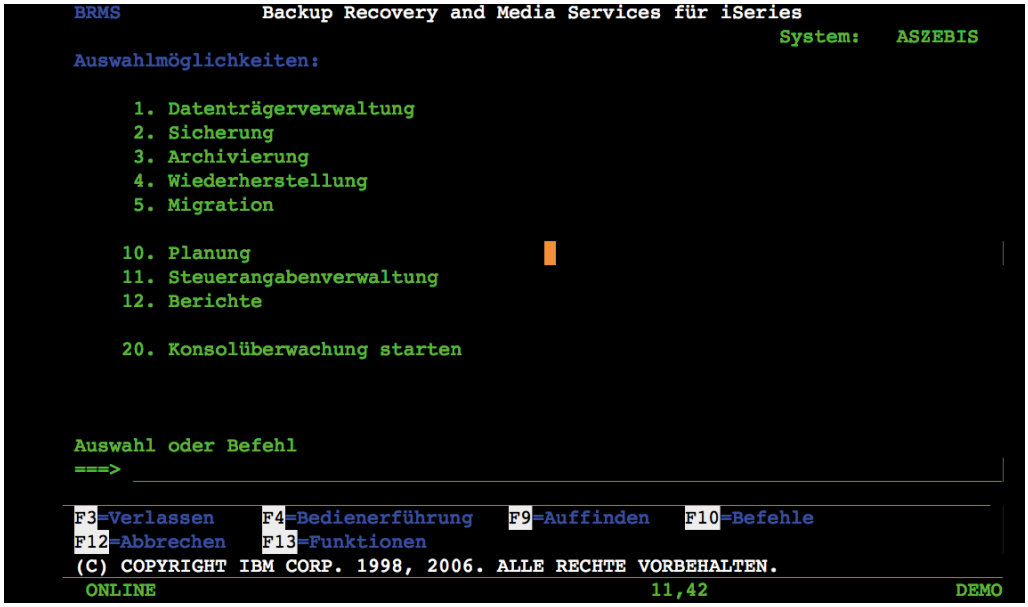
Machen Sie es sich deshalb unbedingt zur Regel, die Sicherungsprozeduren ausschließlich über BRMS durchzuführen!

Die Bedienung von BRMS erfolgt wahlweise über eine 5250-Eingabe oder den System i Navigator. Leider gibt es Unterschiede in der Verwendung der beiden Benutzerschnittstellen, die zum Teil ärgerliche Nebeneffekte nach sich ziehen, wenn man einmal mit 5250, das andere Mal mit dem System i Navigator arbeitet. Aber dazu später mehr.

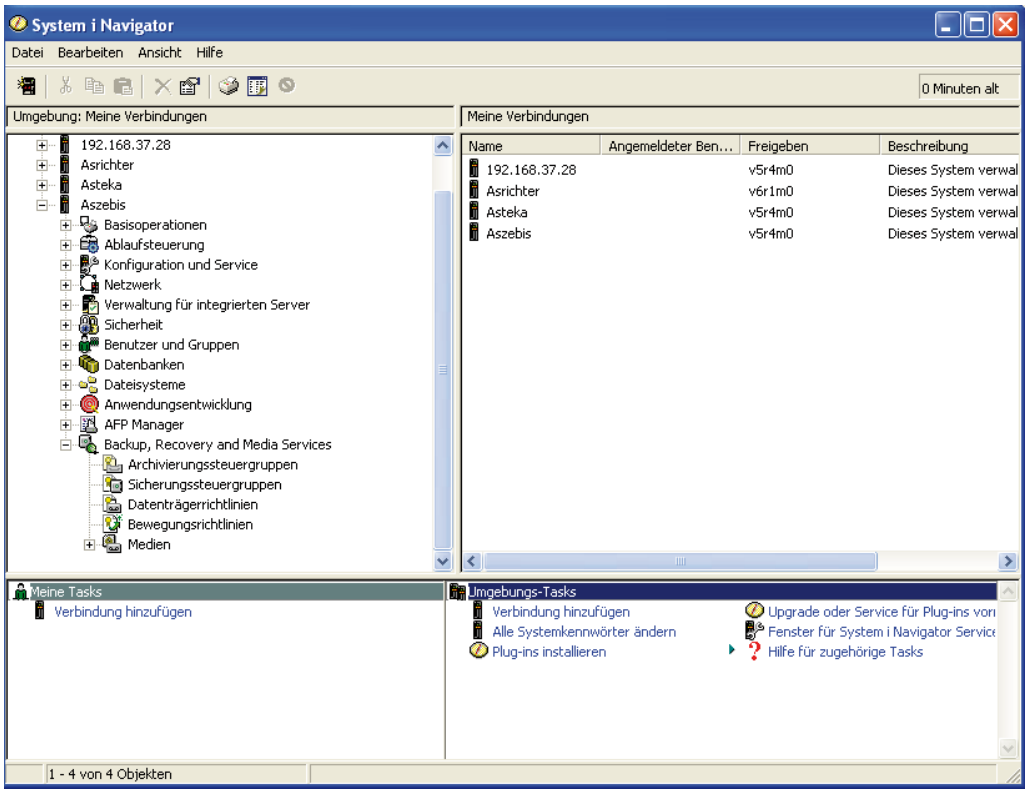
Das Verwalten der Grundeinstellungen von BRMS bietet sich in der 5250-Welt an. Die Einrichtung von Richtlinien hingegen bietet sich mit dem System i Navigator an, der in diesem Bereich über wirklich hilfreiche Assistenten verfügt, die durch die notwendigen Schritte führen. Zudem finden wir einige Funktionen in dem System i Navigator, zu denen es keine direkten 5250-Gegenstücke bzw. -Befehle gibt. Allerdings lassen sich diese „fehlenden" 5250-Funktionen bei Bedarf auch unter Verwendung der auf dem System i vorhandenen APIs nachbauen. Aber das ist sicher nicht der Weisheit letzter Schluss. Ein gezielter Einsatz beider Oberflächen ist sicher in der Praxis die beste Lösung.

Die beiden nachfolgenden Abbildungen zeigen beispielhaft das Menü BRMS, von dem aus die wichtigsten 5250-BRMS-Bereiche und -Funktionen aufgerufen werden können, und einen Verwaltungsbereich für BRMS in dem System i Navigator.

24. Ergänzung 9/2010



BRMS-Hauptmenü in 5250-Form



System i Navigator und BRMS

Die meisten BRMS-Funktionen sind über spezielle Menüs ausführbar. Entgegen dem Einsatz des grafischen Assistenten in dem System i Navigator, verfügen die Menüs und die Funktionen in der 5250-Umgebung nicht über Hilfestellungen, welche Funktionen oder Teilfunktionen zu einem gesamten Lösungsabbild eingesetzt werden müssen. Es ist letztlich dem Anwender bzw.

Administrator selbst überlassen, ob er lieber den klassischen 5250-Verwaltungsbereich nutzen oder die grafisch gestützten Funktionen für BRMS im System i Navigator einsetzen will. Zwar sind die Funktionen annähernd gleich, differieren aber leider in dem einen oder anderen Fall. Dies hat zur Folge, dass ein Mischbetrieb der Administration im BRMS-Umfeld mit 5250 und System i Navigator zu Irritationen führen kann. Ich werde diese an entsprechender Stelle aufzeigen.

Als Tipp für die BRMS-Neulinge gilt: Richten Sie die Grundeinstellungen mit Hilfe der BRMS-Einrichtungsassistenten über den System i Navigator ein. Damit wird weitestgehend sichergestellt, dass alle notwendigen Teilschritte ausgeführt wurden.

Die folgende Abbildung zeigt exemplarisch das BRMS-Sicherungsmenü, das als Einstieg für die Verwaltung der Sicherungseinstellungen eingesetzt wird.

BRMS-Sicherungsrichtlinien

Grundsätzlich bietet BRMS eine Fülle an Funktionen, die beispielsweise neben den Sicherungs- und Wiederherstellungsfunktionen auch Archivkomponenten beinhalten. Damit ist allerdings nicht gemeint, dass BRMS als Archivsystem im klassischen Sinne zum Einsatz kommen kann, sondern die Speicherung von Informationen auf verschiedenen Speicherkomponenten gesteuert wird.

Beschränken wir uns zunächst einmal auf die Sicherungs- und Wiederherstellungsfunktionen. Diese bieten für kleinere Installationen genauso Werkzeuge, wie sie auch bei sehr komplexen Umgebungen zum Einsatz kommen. Ob diese Funktionen in Fülle nützlich und sinnvoll sind, muss im Einzelfall betrachtet werden.

Sicher werden die Datensicherungsfunktionen in jedem Fall zum Einsatz kommen, ebenso mögliche Wiederherstellungsszenarien. Was jedoch hier und da in Bezug auf einen möglichen Einsatz hin abgewägt werden muss, ist zum Beispiel die Verwendung von Speicherlokationen oder Bewegungsrichtlinien. Ohne an dieser Stelle bereits zu sehr ins Detail gehen zu wollen, eine kurze Erläuterung zu diesen BRMS-Komponenten:

Lokationen beschreiben im Sinne von BRMS Lagerorte für Datenträger. Bei dem Einsatz einer Vielzahl von Datenträgern an unterschiedlichen Orten kann die Definition von Lokationen durchaus hilfreich sein. So lassen sich beispielsweise sogar Container, Regale oder Schließfächer in Banken als Lokation definieren. Ergänzt um wichtige Informationen wie Ansprechpartner und wichtige Telefonnummern, wer in einem Notfall unter welcher Nummer erreicht werden kann, um an die Bänder zu gelangen, sind dies wichtige Ergänzungen zu den Sicherungs- und Wiederherstellungsinformationen. Allerdings machen diese für „überschaubare“ BRMS-Bereiche keinen wirklichen Sinn. Wenn in einem Unternehmen zum Beispiel zwei Generationen von Wochensicherungen, mehrere Monatssicherungen, Jahressicherungen etc. an nur wenigen Lagerorten (Tresor, Bank, Geschäftsleitung) gelagert werden, dann sollte man den Einsatz und die Definition der Bewegungsrichtlinien abwägen, welche den Administrator anweisen, Bänder an einen bestimmten Ort zu bringen.

BRMS-Einrichtung

Bevor wir die BRMS-Funktionen nutzen können, muss das kostenpflichtige Lizenzprogramm 57xx-BR1 auf dem System installiert werden. BRMS wird mit dem Befehl RSTLICPGM auf dem System eingespielt. Es kommt letztlich auf die Anforderungen an, welche der drei BRMS-Optionen installiert werden müssen.

Auch wenn BRMS in einer weniger komplexen Umgebung mit nur einem einfachen Bandlaufwerk eingesetzt werden kann, ist die Nutzung von Media und Storage Extensions in der Regel sinnvoll. Diese ebenfalls kostenpflichtige Lizenzprogrammversion (57xx SS1 Option 18) dient dem Erkennen und Verwalten der Bänder aus einer Datensicherungslibrary und muss zusätzlich auf dem System i installiert werden.

Solche Datensicherungslibraries oder Sicherungssysteme, auch Sicherungsroboter genannt, verfügen meist über einen Barcodereader, der die Bänder erkennt und die Informationen an das Betriebssystem weiterleitet. Auf diese Weise lassen sich die Bänder hinsichtlich unterschiedlicher Kriterien verwalten. Beispiele dafür sind:

- Maschinenzuordnung
- Verwendung
- Aktive Daten
- Frei für Sicherungsprozesse
- Wichtig für Wiederherstellungsverfahren
- Alter
- Lebensdauer
- Etc.

Tipp!

Wenn Sie die Bandverwaltung mit BRMS nutzen wollen (was ich bis auf wenige Ausnahmefälle auf jeden Fall empfehlen würde), dann muss zusätzlich zur Installation von 57xx-BR1 auch die Option 18 des 57xx-SS1 installiert werden!

Zudem sind die Media and Storage Extensions auch eine Voraussetzung, wenn es um die Verwendung von Verschlüsselungen geht, die seit der Version 6.1 des IBM i zur Verfügung stehen.

Bedenken Sie zudem, dass für BRMS besondere Gruppen PTFs zur Installation angeboten werden. Diese sollten auf jeden Fall installiert werden, bevor BRMS eingesetzt und auch initialisiert wird!

Nachdem die gewünschten Optionen von BRMS installiert wurden (mindestens notwendig: Option *BASE), dann kann mit der Initialisierung von BRMS und der Grundeinrichtung begonnen werden. Diese Initialisierung analysiert

24. Ergänzung 9/2010

8.5.1
Seite 2

das System im Hinblick auf die zu sichernden Teilbereiche, die zur Verfügung stehenden Sicherungseinheiten und weitere Ressourcen, die mit BRMS genutzt werden. Das Ergebnis dieser Analyse und Initialisierung wird in die BRMS-eigenen Tabellen in der Bibliothek QUSRBRM gestellt. Diese Bibliothek wird uns auch später bei der Ausführung der verschiedenen Sicherungsschritte immer wieder begegnen – beinhaltet sie doch alle BRMS-Informationen.

INZBRM

Die Grundinitialisierung der BRMS-Umgebung kann mit dem Aufruf des Befehls INZBRM durchgeführt werden. Für diesen Befehl stehen unterschiedliche Parameter zur Verfügung:

Mit dem Ausführen von INZBRM werden unter anderem folgende Aktionen ausgelöst:

- Initialisieren der BRMS-Tabellen in der Bibliothek QUSRBRM.
- Initialisieren und Einrichten der Standard Richtlinien bzw. Steuergruppen für BRMS.
- Starten des Subsystems für die Netzwerkkomponenten (nur bei Installation des Netzwerk Features).

INZBRM

Bevor man diese Initialisierung durchführt, muss man bedenken, dass während der Initialisierungsphase keine BRMS-Aktionen durchgeführt werden können. Unter Umständen sind die Laufzeiten für die Initialisierungsschritte außerordentlich lang und können durchaus auch mehrere Stunden dauern. Das sollte vor dem Aufruf von INZBRM bedacht werden!

IBM bietet mit einer Sonderoption auch die Schätzung der notwendigen Zeit für die BRMS-Initialisierung. INZBRM(*ESTPRDINZ). Das Ergebnis dieses Laufes ist eine Zeitschätzung, die als Leitlinie für die Erstinstallation herangezogen werden kann.

Weiterhin sind für einige Aktionen weitreichende Berechtigungen erforderlich. Deshalb empfiehlt es sich, die Ausführung mit einem Benutzerprofil zu starten, das über die Sonderberechtigungen *ALLOBJ und *IOSYSCFG verfügt.

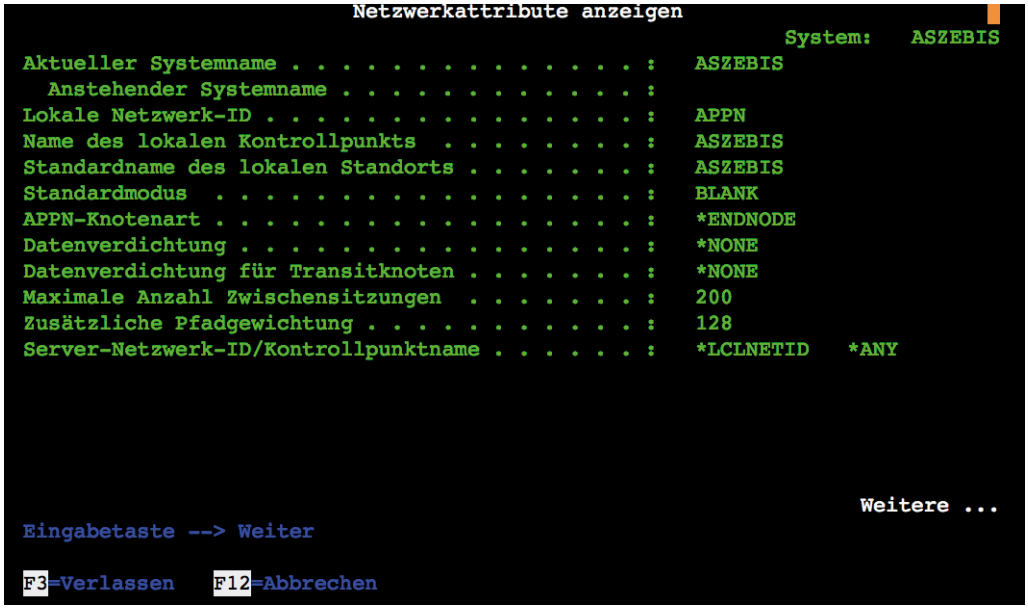
Weitere Optionen für den Aufruf von INZBRM:

INZBRM(*ALL)
Wenn BRMS nach der Installation der Lizenzprogramme genutzt werden soll, dann führen Sie den Befehl INZBRM (*ALL) durch. Achten Sie dabei darauf, dass keine weiteren BRMS-Aktivitäten auf dem System vorhanden sind. Anderenfalls können Satzsperren das erfolgreiche Initialisieren verhindern.
Bei der Verwendung des Parameter *ALL werden alle wesentlichen BRMS-Teilbereiche initialisiert.

INZBRM(*DATA)
Mit dieser Option werden die Standardrichtlinien und die Basis-Tabellen initialisiert. Dabei werden bereits bestehende Daten nicht beeinflusst und die BRMS-Historie bleibt bestehen!
Der Befehl kann auch nach dem Einrichten von BRMS genutzt werden. Zusätzlich werden mit diesem Befehl neue Einheiten zur Verwendung mit BRMS kenntlich gemacht. Dies kann zum Beispiel dann der Fall sein, wenn ein neues Bandlaufwerk angeschlossen wurde.

INZBRM(*CHGSYSNAM)
BRMS arbeitet auf dem System i auf Basis des Systemnamens. Dieser Name wird unter anderem auch in den Tabellen von BRMS dazu verwendet, die Historie zu pflegen und ein System in einem BRMS-Netzwerk den übrigen Systemen kenntlich zu machen. Als Systemname wird der lokale Name des Systems verwendet, den man in dem Befehl DSPNETA ermitteln kann:

8.5.1
Seite 4



DSPNETA

Das Ändern dieses Systemnamens kann dann notwendig sein, wenn mit BRMS Informationen von einem System auf ein anderes übertragen werden. Wenn zum Beispiel im Zuge einer Neuinstallation eines Systems die Daten von einem alten System übertragen werden sollen und der Name sich ändert, dann kann mit diesem Parameter ein neuer BRMS-Systemname vergeben werden. Dieser wird dann in den Datenbanktabellen nachgezogen.

Zudem bildet der Systemname einen der Eckpfeiler bei der Verwendung von BRMS mit dem Netzwerkfeature. Dabei werden auf einem zentralen BRMS-System die Daten der anderen Systeme in der Datenbank gespeichert. Die Zuordnung, um welche Daten es sich letztlich handelt, erfolgt dann unter anderem über den Systemnamen.

Anmerkung!

Der Einsatz des Netzwerkfeatures speichert die BRMS-Daten aller Systeme zentral auf einem System. Dieses bildet damit quasi die BRMS-Datenbank. Das bedeutet aber nicht, dass damit auch die Sicherungs- und Wiederherstellungsprozeduren auf dem zentralen System ausgeführt werden müssen. Auch bei dem Einsatz des Netzwerkfeatures können die Sicherungen natürlich auf dem jeweiligen System selbst durchgeführt werden.

INZBRM(*DEVICE)
BRMS arbeitet mit Einheitenbeschreibungen von Bandlaufwerken und sonstigen Einheiten, die sich zur Sicherung und Wiederherstellung eignen. Die für die Nutzung erforderlichen Informationen werden durch das Initialisieren der BRMS-Umgebung mit INZBRM(*DEVICE) bereitgestellt. Sollten diese

Einstellungen fehlerhaft sein oder eine neue Einheit angeschlossen werden, dann können die Basiseinstellungen für diese Einheiten mit dem Befehl INZBRM(*DEVICE) erneuert werden. Dieser Befehl löscht nicht mehr existierende Einheiten und fügt neue hinzu. Netzwerkkomponenten, die bei der Verwendung des Netzwerkfeatures von BRMS zum Einsatz kommen können, werden nicht gelöscht.
Wie auch bei den übrigen Formen der INZBRM-Ausführung sind die historischen BRMS-Informationen nicht betroffen.

INZBRM(*ESTPRDINZ)
Mit dieser Option weisen wir BRMS an, eine Schätzung der Laufzeit der Initialisierung der BRMS-Umgebung durchzuführen. Das kann unter anderem dann Sinn machen, wenn eine BRMS-Initialisierung unter Umständen Sicherungsprozesse beeinflussen könnte. Je nach Datenmenge und Umgebung kann die Initialisierung durchaus einige Stunden laufen!

INZBRM(*ESTPRDUPD) TGTRLS(Zielrelease)
Diese Option wird nur zum Schätzen der Konvertierungszeit bei einem Wechsel auf eine neue Version von BRMS eingesetzt. Das Umsetzen der Informationen der BRMS-Datenbanken kann, in Abhängigkeit von der Größe der Datenbanken, sehr lange dauern. Um Probleme durch solch einen Langläuferjob zu vermeiden, kann mit einem Analyseprogramm die ungefähre Umstellungszeit ermittelt werden.

INZBRM(*MERGE)
BRMS kann in heterogenen Umgebungen eingesetzt werden. Dabei können verschiedene Formen der Installation zum Einsatz kommen. Während unter der Verwendung des Netzwerk Features ein System als zentrales BRMS-System gekennzeichnet wird, das unter anderem die Informationen und Daten der übrigen Systeme im Bereich von BRMS speichert, können natürlich auf jedem System i oder jeder einzelnen Partition BRMS-Einzelinstallationen durchgeführt werden. Wenn mehrere BRMS-Einzelinstallationen zu einer Installation zusammengefasst werden sollen (zum Beispiel im Zuge einer Konsolidierung), dann lassen sich mit dieser Anweisung die Informationen mehrerer BRMS-Datenbanken mischen.

INZBRM(*NETSYS)
Das Einrichten einer Netzwerkumgebung von BRMS kann unter Verwendung des Mischens von Informationen anderer Systeme durchgeführt werden. Voraussetzung dafür ist die Option 1 des BRMS-Lizenzprogramms. Dabei sollten die einzubindenden Informationen zuvor mit dem Befehl CPYMEDIBRM auf das zentrale System gespielt werden, um diese dann mit INZBRM(*NETSYS) zu mischen.
Wenn Sie zum Beispiel die Netzwerkfunktionen von BRMS aufsetzen wollen, dann muss auf dem Hauptsystem folgender Befehl abgesetzt werden:
INZBRM OPTION(*NETSYS) FROMSYS(*NetzID.SystemID*)

24. Ergänzung 9/2010

8.5.1
Seite 6

Der Platzhalter NetzID ist durch den Namen des BRMS-Netzwerks und der Wert SystemID durch den Systemnamen des „fernen“ Systems zu ersetzen!

INZBRM(*RESET)
Mit dieser Option lassen sich die BRMS-Tabellen initialisieren. Achtung! Dabei gehen auch die historischen Daten verloren, mit denen eine Systemwiederherstellung durchgeführt werden kann (nicht die Sicherungsdaten, sondern die Basis für die Erstellung des Wiederherstellungsleitfadens!). Diese Option sollte mit Bedacht nur dann angewendet werden, wenn wirklich ALLE BRMS-Informationen auf den Auslieferungsstand zurückgesetzt werden sollen!
Ein Beispiel für diese Option ist das Initialisieren der BRMS-Installation:
INZBRM OPTION(*RESET)

INZBRM(*RUNPRDINZ)
Diese Option stellt den Abschluss der Installation von BRMS dar und führt die Erstinitialisierung durch.

INZBRM(*SETAUT)
Diese Option ist im Falle einer vollständigen Systemwiederherstellung erforderlich, um die entsprechenden Berechtigungen für die Benutzerprofile wieder einzustellen.

Jeder Initialisierungsschritt wird in dem BRMS-Protokoll aufgezeichnet. Dabei handelt es sich genau genommen um das Standard-Jobprotokoll, dessen Einträge auf die BRMS-Bereiche eingeschränkt sind.

Folgende Abbildung zeigt einen solchen BRMS-Protokollbereich.

BRMS-Protokoll

Steuerangaben

BRMS-Definitionen basieren auf so genannten Steuerangaben, auch Policies genannt. Diese Steuerangaben beschreiben das Verhalten und die Funktionen des BRMS.

Es gibt Steuerangaben, die auf anderen Steuerangaben basieren. Ein Beispiel dafür ist die Definition von Sicherungssteuerangaben. Diese können nur angelegt werden, wenn beispielsweise auch Datenträgersteuerangaben existieren.

Steuerangaben lassen sich zudem in zwei Gruppen unterteilen:

- Unbedingt erforderliche Steuerangaben
  – z. B. Systemsteuerangaben, Datenträgersteuerangaben
- Optionale Steuerangaben
  – Bewegungssteuerangaben

Mit den Steuerangaben legen wir also fest, wie Sicherungen und Wiederherstellungen mit BRMS durchzuführen sind. Für den Anfänger mögen die Vielfalt und auch die Abhängigkeit der BRMS-Steuerangaben auf den ersten Blick verwirren. Deshalb hier der Rat, Steuerangaben mit den Funktionen des System i Navigators zu erstellen. Dieser bietet für die Verwaltung der Steuerangaben Assistenten, die uns durch die notwendigen Schritte leiten.

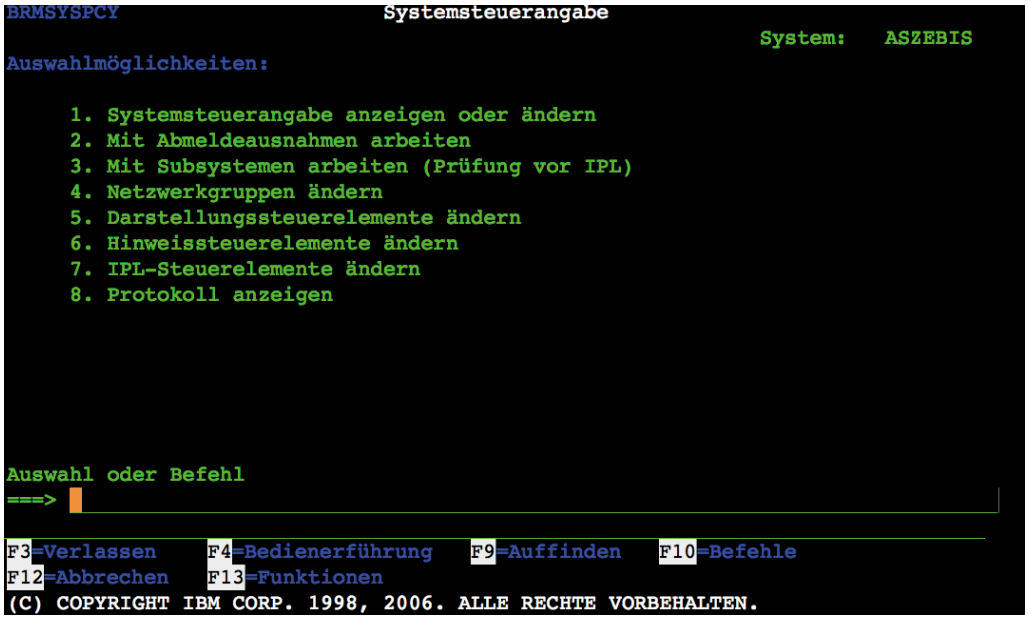
Die meisten Steuerangaben verfügen über Standardwerte. So werden zum Beispiel die Datenträger bereits anhand der während der Initialisierung von BRMS angeschlossenen Bandeinheiten zum Teil angelegt. Lokationen für die Aufbewahrung von Datenträgern werden ebenfalls mit Standardwerten (z. B. Offsite und Vault) angelegt. Diese Unterlassungswerte können genutzt werden. Meist sind diese aber nur globaler Art und es empfiehlt sich, individuelle Einstellungen zu definieren.

Schauen wir uns zum allgemeinen Verständnis einige der Steuerangaben einmal an:

8.5.2
Seite 2

Systemsteuerangaben

Systemsteuerangaben definieren globale Einstellungen für die BRMS-Installation und bilden die oberste Hierarchieebene von BRMS. Diese Hauptform der Steuerangaben kann grob mit den Systemwerten verglichen werden, die wir von IBM i kennen.



Systemsteuerangaben

Mit der Systemsteuerangabe legt man fest, wie BRMS allgemein ausgeführt werden soll. Unter anderem befinden sich dort folgende Informationen:

- Einheitensteuerangaben
- Zu verwendende Einheiten
- Homelokation
- Abmeldebedingungen für Sicherungszwecke
  - Auch Ausnahmen für Abmeldebedingungen
- Anfangstag der Woche
- Art und Form der Datensicherung (vollständig, inkrementell)
- Beenden der Subsysteme
- Für BRMS-Ausführung steuerndes Subsystem
- Netzwerkeinstellungen
- IPL-Definitionen
- Etc.

Die auf der Ebene der Systemrichtlinien getätigten Angaben sind allgemeingültig, können jedoch durch Richtlinien tieferer Ebenen individuell überschrieben werden. Auf diese Weise lassen sich bei geschickter Verwendung der Angaben die notwendigen Definitionen auf ein Minimum beschränken.

8.5.2.1
Seite 2

```
V5R4M0                  Systemsteuerangabe ändern                    ASZEBIS

Auswahl eingeben und Eingabetaste drücken.

Datenträgersteuerangabe . . . . . FULL          Name, F4=Liste
Einheiten . . . . . TAP01          Name, F4=Liste
                                   _____
                                   _____
                                   _____
Basislokation für Datenträger . . . . . *HOME      Name, F4=Liste
Datenträgerklasse . . . . . QIC4DC      Name, F4=Liste
Interaktive Benutzer abmelden . . . . . *NO       *YES, *NO
Grenzwert für Abmeldung . . . . . 30       0-999 Minuten
Ausgabewarteschlange . . . . . *PRTF      Name, *PRTF
  Bibliothek . . . . . _____      Name, *LIBL
Startzeit . . . . . 0:00:00    Uhrzeit
Datenträgerüberwachung . . . . . *YES      *YES, *NO
Verzögerung bei gemeinsam gen. Bestand . 60       30-9999 Sekunden
Datenträger automatisch registrieren . . *NO       *NO, *YES
Standardverwendung . . . . . *YES      *NO, *YES
                                                              Weitere...
F3=Verlassen    F4=Bedienerführung    F5=Aktualisieren    F12=Abbrechen
```

Systemsteuerangaben

Die Systemsteuerangaben beinhalten zudem noch weitere Einträge, die exemplarisch in der folgenden Abbildung zu sehen sind:

```
BRMSYSPCY                    Systemsteuerangabe
                                                    System:   ASZEBIS
Auswahlmöglichkeiten:

     1. Systemsteuerangabe anzeigen oder ändern
     2. Mit Abmeldeausnahmen arbeiten
     3. Mit Subsystemen arbeiten (Prüfung vor IPL)
     4. Netzwerkgruppen ändern
     5. Darstellungssteuerelemente ändern
     6. Hinweissteuerelemente ändern
     7. IPL-Steuerelemente ändern
     8. Protokoll anzeigen

Auswahl oder Befehl
===>

F3=Verlassen    F4=Bedienerführung    F9=Auffinden    F10=Befehle
F12=Abbrechen   F13=Funktionen
(C) COPYRIGHT IBM CORP. 1998, 2006. ALLE RECHTE VORBEHALTEN.
```

Weitere Optionen im Systemsteuergruppen-Menü

Ein Beispiel für weitere Systemsteuerangaben sind die Abmeldeausnahmen. Dabei handelt es sich um die Festlegung von Benutzerprofilen oder Einheitenbeschreibungen, die für einen Datensicherungsprozess aktiv bleiben dürfen oder müssen. Wenn Sie zum Beispiel für eine Tagessicherung sicherstellen wollen, dass kein normaler Anwender mehr auf dem System arbeitet, dann können Sie mit der Einstellung „interaktive Anwender abmelden“ das Been-

den der Jobs erzwingen. Soll ein solches erzwungenes Ende für zum Beispiel den QSECOFR oder einen an der Konsole (DSP01) angemeldeten Benutzer entfallen, dann nehmen Sie die gewünschten Einträge in die Abmeldeausnahmen auf.

Mit Abmeldeausnahmen arbeiten

In dem vorhergehenden Beispiel brauchen sich QSECOFR und QSYSOPR sowie unter Umständen an dem Bildschirm DSP01 angemeldete Benutzer für die Datensicherung nicht abmelden.

Ein weiteres Beispiel für zusätzliche Sicherungssteuerangaben ist die Festlegung der Behandlung von Subsystemen. Manche Subsysteme dürfen während einer Datensicherung nicht laufen, während andere problemlos aktiv sein können, da sie den zu sichernden Bereich nicht beeinflussen. Letztere werden in den Einstellungen für die Subsysteme hinterlegt. Dabei ist zu beachten, dass die Angaben auf Existenz geprüft werden. In folgendem Beispiel kann das Subsystem QBATCH in der Bibliothek ZEIG für die Datensicherung aktiv bleiben.

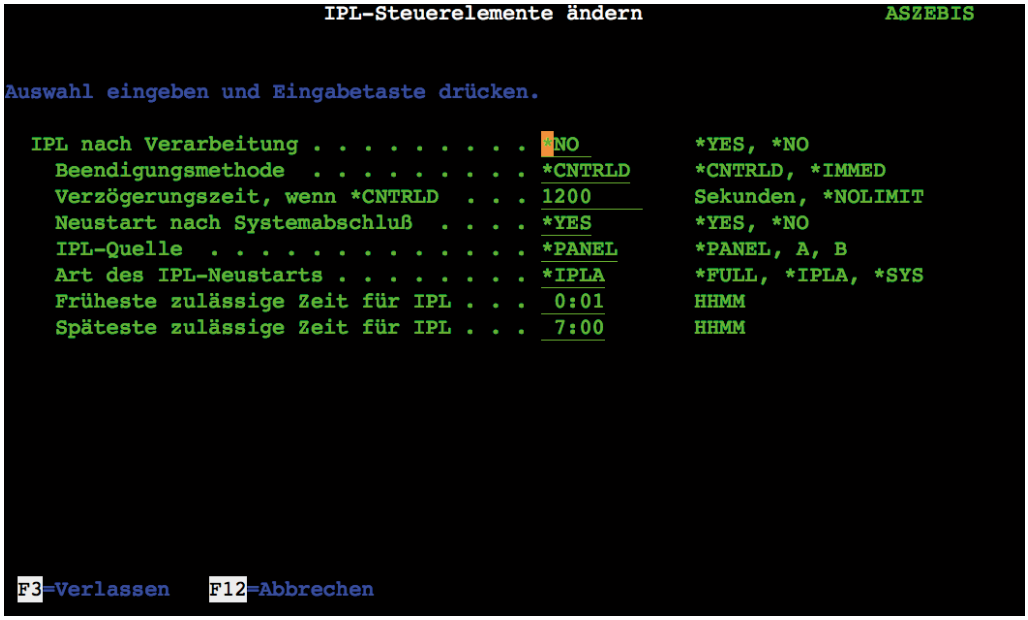
8.5.2.1
Seite 4

Mit Subsystemen arbeiten

ACHTUNG!!

Bedenken Sie, dass laufende Jobs in dem Subsystem die Datensicherung bei fehlerhafter Definition im BRMS-Bereich natürlich beeinflussen können. Deshalb sollten Eintragungen in dem Bereich der Subsysteme mit Kenntnis der Anwendungen und Jobs durchgeführt werden!!

Mit der Durchführung von Datensicherungen ist zum Teil auch ein Neustart des Systems verbunden. Diese Aktion kommt dem Parameter „Restart“ der Sicherungsbefehle gleich. Über den Menüpunkt „IPL Steuerelemente ändern“ können wir Einfluss auf den IPL nach der Datensicherung nehmen. Unter anderem sind hier die früheste und späteste IPL-Zeit besonders wichtig. Damit lässt sich zum Beispiel vermeiden, dass das System aufgrund von Problemen bei der Datensicherung am Folgetag nicht zeitgerecht zur Verfügung steht. Dauert beispielsweise eine Datensicherung wegen unvorhersehbarer Locksituationen zu lange, dann kann mit dieser Einstellung sichergestellt werden, dass das System in jedem Fall hochgefahren wird.



IPL Steuerelemente ändern

8.5.2.1

Seite 6

Sicherungssteuerangaben

8.5.2.2
Seite 1

Sicherungssteuerangaben definieren die Sicherungsaktionen, die mit BRMS durchgeführt werden.

Unter anderem wird auf dieser Ebene festgelegt, welche Datenträger für eine Sicherung genutzt werden, ob die Datensicherung vollständig oder inkrementell erfolgt, oder beispielsweise interaktive Benutzer angemeldet bleiben dürfen. Sind es Benutzerdaten, Systemdaten oder beide Bereiche, die gesichert und dementsprechend unterschiedlich behandelt werden müssen?

Sicherungssteuerangaben definieren die unterschiedlichen Formen der Datensicherungen auf dem System i. Mit den Sicherungssteuerangaben werden unter anderem folgende Eckdaten definiert:

- Art der Sicherung
  - vollständig, inkrementell, kumulativ
- Zu verwendende Sicherungseinheiten
- Abmeldeangaben
  - inkl. Ausnahmen von Abmeldedefinitionen
- Journalbehandlung
- Sicherung von Zugriffspfaden
- Komprimierung
- Aktionen bei Ende des Bandes

Folgende Abbildung zeigt exemplarisch den Definitionsbereich für die Sicherungssteuerangaben.

```
                  Sicherungssteuerangabe ändern                  ASZEBIS


Auswahl eingeben und Eingabetaste drücken.

Datenträgerst. für Gesamtsicherungen . . *SYSPCY     Name, F4=Liste
Datenträgersteuerangabe für
   Teilsicherungen . . . . . . . . . . . *SYSPCY     Name, F4=Liste
Sicherungseinheiten  . . . . . . . . . . *SYSPCY     Name, F4=Liste
                                         ___________
                                         ___________
                                         ___________
Standardmäßige wöchentliche Aktivität  . FFFFFFF     SMDMDFS(F/I)
Art der Teilsicherung  . . . . . . . . . *CUML       *CUML, *INCR
Gesamtsicherung erzwingen (Anzahl Tage)  *NOMAX      0-365, *NOMAX
Interaktive Benutzer abmelden  . . . . . *SYSPCY     *YES, *NO, *SYSPCY
Grenzwert für Abmeldung  . . . . . . . . *SYSPCY     0-999 Minuten, *SYSPCY
Journalobjekte beim Sichern
  geänderter Objekte sichern . . . . . . *NO         *YES, *NO

                                                                 Weitere...
F3=Verlassen  F4=Bedienerführung  F5=Aktualisieren  F9=Systemsteuerangabe
F12=Abbrechen
```

Sicherungssteuerangaben

8.5.2.2

Seite 2

Wiederherstellungssteuerangaben

Mit den Wiederherstellungssteuerangaben definieren wir die Wiederherstellungsaktionen im BRMS-Umfeld.

```
                Wiederherstellungssteuerangabe ändern              ASZEBIS

Auswahl eingeben und Eingabetaste drücken.

  Wiederherstellungseinheit  . . . . *MEDCLS      Name, F4=Liste
                                     ________
                                     ________
                                     ________

  Parallele Einheitenressourcen:
    Mindestanzahl Ressourcen . . .   *SAV         1-32, *SAV, *NONE, *AVAIL
    Maximale Anzahl Ressourcen . .   ____        1-32, *AVAIL, *MIN
  Auswahl bei Bandende . . . . . . . *REWIND     *REWIND, *LEAVE, *UNLOAD
  Auswahl  . . . . . . . . . . . . . *ALL        *ALL, *NEW, *OLD, *FREE
  Auswahl für Datenbankteildatei . . *ALL        *MATCH, *ALL, *NEW, *OLD
  Spooldateidaten zurückspeichern  . *NONE       *NONE, *NEW
  Objektunterschiede zulassen  . . . *NONE       *NONE, *ALL, *AUTL,
                                     ________    *FILELVL, *OWNER, *PGP
                                     ________
                                     ________

                                                                  Weitere...
F3=Verlassen  F4=Bedienerführung  F5=Aktualisieren  F9=Systemsteuerangabe
F12=Abbrechen
```

Wiederherstellungssteuerangaben

Viele der Parameter, die für die Wiederherstellungsangaben getätigt werden können, sind Gegenstücke der Restore-Befehle, die nativ auf dem System eingesetzt werden können. Die Behandlung von Abweichungen oder das Zurückspeichern von Spooldateien sind nur einige Beispiele.

Die Wiederherstellungssteuerangaben werden in der Regel für die einzelnen möglichen Wiederherstellungsverfahren definiert.

8.5.2.3

Seite 2

Bewegungssteuerangaben

Bewegungssteuerangaben definieren die Aktionen, die zum Bewegen der Sicherungsdatenträger benötigt werden. ACHTUNG! Damit ist in der Regel kein automatisches Bewegen der Datenträger gemeint, sondern durchaus auch Anweisungen, wo Bänder zu entnehmen sind und an welche Orte sie gebracht werden müssen.

Während die bisher genannten Steuergruppen alle definiert werden müssen, können die Bewegungssteuerangaben optional genutzt werden. Diese machen eigentlich nur in größeren Umgebungen wirklich Sinn, bei denen unterschiedliche Lokationen genutzt werden und man dann anhand der Bewegungsrichtlinien analog den übrigen Steuerangaben festlegt, wann welches Band an welchen Lagerort gebracht werden muss.

Installationen mit einer überschaubaren Anzahl an Sicherungsdatenträgern können meist auf die Definition der Bewegungssteuerangaben verzichten, da man eh weiß, dass die aktuellen Datensicherungsbänder zum Beispiel in den Tresor im Nebengebäude gebracht werden müssen.

```
                    Mit Bewegungssteuerangabe arbeiten          ASZEBIS

Listenanfang bei  . . .  __________   Startzeichen

Auswahl eingeben und Eingabetaste drücken.
  1=Erstellen   2=Ändern   3=Kopieren   4=Löschen   5=Anzeigen

Ausw.  Steuerangabe  Text
 _     __________
 _     BANK          Sicherung in Bank Tresor
 _     OFFSITE       Eintrag von BRM-Konfiguration erstellt
 _     TRESOR        TRESOR

                                                              Ende

F3=Verlassen   F5=Aktualisieren   F12=Abbrechen
```

Bewegungssteuerangaben

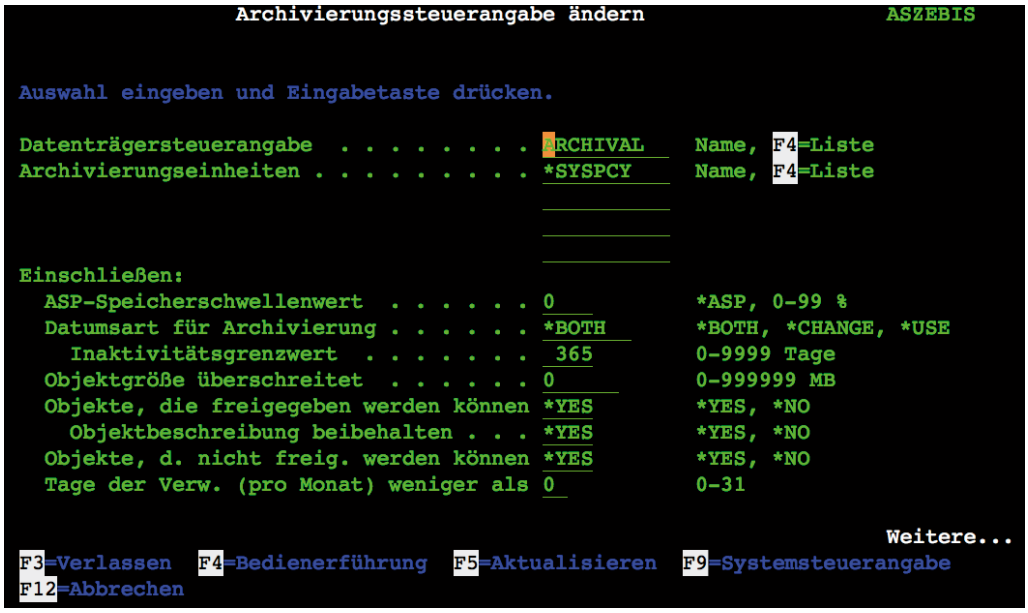
8.5.2.4

Seite 2

Archivsteuerangaben

Mit den Archivsteuerangaben definiert man, wann und wo welche Daten archiviert werden sollen. BRMS bietet neben den Sicherungs- und Wiederherstellungsfunktionen auch in einem gewissen Rahmen Archivfunktionen an. Diese sind nicht mit einem klassischen Archivsystem zu verwechseln, wie man es heute oft kennt. Die Archivierung bezieht sich vielmehr auf das Zuordnen von Speicherressourcen zu den Daten.

Nehmen wir einmal ein Beispiel: In einem Unternehmen werden die Inventurdaten jedes Jahr gespeichert, um bei Bedarf für interne Zwecke oder Prüfungen eingesehen werden zu können. Es macht meist keinen Sinn, die Daten, die nur selten, wenn überhaupt einmal, wieder benötigt werden, auf den „teuren“ Sicherungsressourcen vorzuhalten. Also kann man durchaus in Erwägung ziehen, die Daten auf ein günstigeres Medium zu verlagern, beispielsweise auf langsame interne Platten. Dieses Verschieben kann mit den Archivfunktionen von BRMS vorgenommen werden. Da sich der Inhalt dieses Kapitels mit den Sicherungs- und Wiederherstellungsfunktionen beschäftigt, werden wir an dieser Stelle nicht weiter in die Details der Archivfunktionen von BRMS gehen.



Archivsteuerangaben

8.5.2.5

Seite 2

9 Kapitel

Sicherheit für Power i

Sicherheit für Power i

„Die AS/400“ – sorry, jetzt heißt sie IBM Power i – ist das sicherste System der Welt?! Immer wieder wird dem Anwender diese Aussage unter die Nase gerieben. Fragen dazu – wie „Ist dem tatsächlich so?“, „Gehört das eventuell in die Vergangenheit?“ oder „Wie hat sich die Sicherheit im Laufe der Zeit verändert?“ – gilt es zu beantworten. Solche Antworten, in denen es um die Sicherheit der IBM Power i-Systeme geht, finden Sie in den folgenden Kapiteln.

Dieses Fachbuch bietet einen sehr komplexen Informationsbereich zu diesem Thema an. Niemand, der sich mit Security beschäftigt, kann von sich behaupten, darin perfekt zu sein. D.h. – eine perfekte Security gibt es nicht. Falls doch, dann ist der Umgang mit den Systemen so kompliziert geworden, dass man kaum noch damit arbeiten kann. Darum sollte man Kompromisse schließen – und zwar unter Einbeziehung eines praktischen Ansatzes und Einbeziehung einer ausreichenden, bestmöglichen und optimalen Sicherheit, so dass man letztendlich damit auch arbeiten kann.

Aber wo fängt man beim Thema Security an? Am Besten von vorn! Sorry, aber hier werden wir Informationen für Laien und fortgeschrittene Anwender gleichermaßen anbieten. Daher bitte ich den Spezialisten für Power i Systeme ein wenig um Geduld, da wir zuerst einmal mit den Basisinformationen beginnen werden, die allerdings für den Anfänger sehr wichtig sind. Die Fortgeschrittenen können an dieser Stelle diese wenigen Kapitel einfach überspringen.

Grundlegende Konzepte

Die Power i mit dem Betriebssystem IBM i wurde schon bei der Grundsteinlegung 1988 mit einem starken Fokus auf Sicherheit entwickelt. Ging es doch bei diesem System schon immer – auch als es noch AS/400 hieß – um unternehmenskritische Transaktionen. Millionen von Anwendern, die mit Milliarden von Transaktionen agieren, benötigen eben eine gewisse Stabilität und Sicherheit. Daher wurde auch der Ruf laut, dass die Power i Server zu den sichersten Systemen gehören, die man kaufen kann.

Damit wurde von Anfang an die Sicherheit als fester Bestandteil des Systems mit eingeplant und integriert. Sicherheit war somit keine nachträgliche Idee, die oberhalb des Betriebssystems angesiedelt wurde, wie es bei vielen anderen Rechnersystemen der Fall ist. Das objektorientierte Design machte einen IBM i Server resistent gegenüber Angriffen – wie z. B. denen von Viren.

Allein durch die Installation des Betriebssystems wird ein Basisschutz mitinstalliert. Viele einfach zu handhabende Sicherheitsmechanismen und Dienste erleichtern es Administratoren, den gewünschten Schutz ihres Systems zu erreichen.

Objekte
Die Power i ist ein objektorientiertes System. Wikipedia sagt dazu:

„Das IBM i System ist nach dem Prinzip der Objektbasiertheit aufgebaut, wodurch grundsätzlich alles im Betriebssystem, egal ob Benutzerprofil oder Programm, als Objekt mit Eigenschaften und Funktionen angesehen wird; dieses Prinzip ist jedoch nicht zu verwechseln mit Objektorientierung, wie sie bei Programmiersprachen zu verstehen ist. Gemäß Konvention beginnen alle Systemobjekte mit dem Buchstaben Q (z. B. QSYS oder QSECOFR).

Diese konsequente Objektbasiertheit bewirkt, dass Objekte (Programme, Dateien, Spools, Subsysteme etc.) ausschließlich über einen Satz von abschließend definierten Funktionen angesprochen werden können. Damit ist es beispielsweise nicht möglich, den Binärcode eines Programms frei zu verändern, da diese Art von Schnittstelle zwar für Objekte des Typs Datei (*FILE), aber nicht für Objekte des Typs Programm (*PGM) zur Verfügung steht. Dies unterscheidet OS/400 bzw. i5/OS – wie es neu heißt – grundlegend von den meisten anderen Betriebssystemen, wo vom Dateisystem lediglich Dateien verwaltet werden, deren Verwendungszweck jeweils nur durch eine Dateiendung und eventuell eine User-Zuordnung festgelegt wird.

9.1
Seite 2

Objekte werden in Bibliotheken verwaltet. Diese Bibliotheken können dabei keine Unterbibliotheken enthalten, sondern nur Objekte. Eine Ausnahme ist die Bibliothek QSYS, die sämtliche Bibliotheken enthält. Dateien selbst können in Members unterteilt werden, die man Teildateien nennt. In den Teildateien vom Dateityp PF-SRC können z. B. Quellcodes abgelegt werden, in PF-DTA werden Members als indizierbare Datentabellen abgelegt. In OS/400 hat jedes Member eine logische Satzlänge (LRECL). OS/400 kennt ursprünglich keine strukturfreien Dateien wie Unix. Dies wurde erst mit der Einführung des integrierten Dateisystems (IFS) möglich."

Was bietet das Power i System an Sicherheit

Es gibt drei Sicherheitsebenen:

– Systemebene

– Benutzerebene

– Objektebene

Die Sicherheit auf der Systemebene wird im Systemwert QSECURITY eingestellt. Dabei existieren fünf Sicherheitsstufen, die von keiner Sicherheit bis zur so genannten C2-Sicherheit – eine von der US-Regierung zertifizierte Sicherheitsstufe – reichen. Die Benutzerebene ist notwendig für das Anmelden an das System, wobei hier bereits diverse Berechtigungen festgelegt sind. Auf der Objektebene können Berechtigungen explizit für jedes Objekt vergeben werden. Zum Systemwert QSECURITY kommen wir nach Erläuterung der Grundlagen.

Des Weiteren sind Systemobjekte durch das Domain-Attribut des Objektes vor Manipulation geschützt. Ab einem bestimmten Wert in QSECURITY kann trotz Objektberechtigung nicht von einem Programmcode, der in der Domäne Benutzer läuft, auf ein Objekt der Domäne System zugegriffen werden. Bei dieser Sicherheitsverschärfung ist zu beachten, dass manche Software von Drittanbietern auf solche Zugriffe angewiesen ist.

Im Nachfolgenden sind einige Erläuterungen sowie Komponenten der verschiedenen Ebenen aufgeführt, die in späteren Kapiteln näher beleuchtet werden.

9.2
Seite 2

Sicherheit auf Systemebene

Auf Systemebene gibt es verschiedene Objekte, die verschiedene Ziele verfolgen:

Sicherheits-Funktionen	Vertraulichkeit	Integrität	Authentifizierung	Authorisierung	Protokollieren/ Auditieren
Benutzerprofil			X	X	X
Objektberechtigung				X	X
Objektsignierung		X			X
Systemwerte		X	X	X	X
Netzwerk-Attribute				X	
Digitale Zertifikate		X	X	X	
Security Auditjournal					X
Exit-Programme			X	X	X
Kerberos			X		X
DB2 Datenverschlüsselung	X				
Anwendungsverwaltung					
Viren-Scanner		X			

Benutzerprofile
Benutzerprofile beinhalten sicherheitsrelevante Informationen, die steuern, wie ein Anwender sich am System anmeldet, was er darf oder wie die Aktivitäten des Anwenders protokolliert (auditiert) werden.

Objektberechtigungen
Objektberechtigungen beinhalten Informationen darüber, wer diese wie verwenden darf. Die Objektberechtigung erteilt einem Anwender das Recht auf ein Objekt – z.B., dass ein Anwender ein Objekt benutzen, ein anderer das Objekt ändern und ein dritter überhaupt nicht auf das Objekt zugreifen darf.

Objektsignierung
Um ein Objekt vor Manipulation zu schützen, kann es signiert werden, um die Integrität zu gewährleisten. Digitale Zertifikate werden verwendet, um Objekte zu signieren. Außerdem enthalten sie einen öffentlichen Schlüssel (Public Key), um die Signatur zu prüfen.

Systemwerte
Systemwerte sind Teil der globalen Sicherheitseinstellung des Power i Systems. Die Systemwerte werden verwendet, um verschiedene Einstellungen des Systems anzupassen. Dazu gehören u.a. Systemwerte, um zu steuern, welche Informationen protokolliert (auditiert) werden müssen. Andere Systemwerte sind in der Lage zu entscheiden, nach welchen Kriterien ein Benutzer sein Kennwort vergeben kann. Systemwerte können mit dem Befehl WRKSYSVAL angezeigt und geändert werden.

Netzwerkattribute
Wie sich ein Power i System in ein Netzwerk integriert, wird zum Teil über die Netzwerkattribute gesteuert. Viele der Netzwerkattribute stammen noch aus der SNA- (System Network Architektur-) Zeit. Einige beinhalten TCP/IP-relevante Parameter (iSeries Access-Anforderungen – PCSACC – oder DDM/DRDA-Anforderungen – DDMACC). Mit den Befehlen DSPNETA bzw. CHGNETA können diese Einstellungen angezeigt bzw. geändert werden.

Digitale Zertifikate
Ein digitales Zertifikat ist wie ein Identifikationspapier, das verwendet wird, um die Identität bei elektronischen Transaktionen sicherzustellen. Immer mehr Anwendungen benutzen digitale Zertifikate für diesen Zweck. Auch das Power i System verwendet sie in den verschiedensten sicherheitsrelevanten Anwendungen.

Auditjournal
Ein sehr umfassendes Werkzeug zur Erhaltung der Transparenz in Power i Systemen ist das Auditjournal. Es wird verwendet, um alle Arten von Transaktionen im System aufzuzeichnen und zu protokollieren. Hauptsächlich wird das Auditjournal aus Sicherheitsgründen geführt, um jederzeit nachvollziehen zu können, wer im System welche Aktivitäten ausgeführt hat. Das Auditjournal auszulesen, ist nicht immer einfach, da jede Art von System-Aktivität ihren speziellen Aufbau im Auditjournal hat und somit entsprechend interpretiert werden muss. Der Befehl DSPJRN kann verwendet werden, um Daten aus dem Auditjournal (QAUDJRN) herauszulesen.

Exit-Programme
Exit-Programme sind Schnittstellen aus Anwendungen. Sie sind notwendig, damit diese Anwendungen miteinander oder mit Systemprozessen kommunizieren können. Mit Exit-Programmen können erweiterte Berechtigungsprüfungen in TCP/IP-Anwendungen durchgeführt werden, um Benutzer für bestimmte Objekte in TCP/IP-Anwendungen zu berechtigen oder kundenspezifische Protokollierungen durchzuführen.

Kerberos
Kerberos ist ein Sicherheitsmechanismus, der verwendet wird, um die Anwender in den Unternehmen systemübergreifend zu authentifizieren. Microsoft Windows verwendet beispielsweise Kerberos, um Benutzer in einer Windows-Domäne zu authentifizieren. Kerberos-Tickets werden verwendet, um Berechtigungen in einer Kerberos-Umgebung weiterzugeben. Damit entfällt die Notwendigkeit, dass Anwender, die auf verschiedene Systeme und Anwendungen zugreifen, sich immer wieder neu anmelden müssen. Systeme und Anwendungen sollten Kerberos unterstützen. Im IBM i System wird Enterprise Identity Mapping (EIM) verwendet, damit eine Verbindung zwischen Windows- und IBM i-Benutzern hergestellt wird.

DB2-Datenbankverschlüsselung
Eine Datenbankverschlüsselung wird benutzt, damit die Inhalte von Datenbanken verschlüsselt werden und damit sie einen noch höheren Schutz vor unberechtigtem Zugriff erhalten. Kryptographische Funktionen und APIs native im IBM i helfen diese Verschlüsselung zu realisieren.

Anwendungsverwaltung
Die Anwendungsverwaltung ermöglicht es den Administratoren festzulegen, welche Anwender auf welche Funktionen des Power i Systems zugreifen dürfen, wenn diese von Clients aus arbeiten.

Virenscan
Obwohl IBM i gegenüber Viren nicht anfällig ist (zumindest sind keinen Schäden bekannt), können im integrierten Dateisystem PC-Dokumente, die virenbefallen sind, gespeichert sein. Deshalb unterstützt IBM die Möglichkeit, Objekte über Drittanbieterprodukte gegen Virenbefall zu scannen und – im positiven Fall – eventuell die betreffenden Objekte über Betriebssystemfunktionen sperren zu lassen.

30. Ergänzung 3/2012

9.3
Seite 4

Sicherheit auf Netzwerkebene

Da Power i Systeme nahezu überall in Netzwerke integriert sind, hat IBM auch Funktionen implementiert, die den Schutz von Daten, die über Netzwerke gesandt werden, zu gewährleisten.

Sicherheits-Funktionen	Vertraulichkeit	Integrität	Authentifizierung	Authorisierung	Protokollieren Auditieren
IP Filterung und NAT			X	X	X
IDS		X			
VPN	X	X	X	X	X
L2TP			X	X	X
SSL/TLS	X	X	X	X	X
OpenSSH OpenSSL	X	X	X		
Kryptographie-Software	X	X	X		
Kryptographie-Hardware	X	X	X		

IP-Paketfilterung
IP-Paketfilterung erlaubt die Kontrolle des IP-Verkehrs in und aus dem Netzwerk heraus. Es schützt das Netzwerk durch die Filterung von Paketen – abhängig von den definierten Regeln.

NAT (Network Address Translation)
NAT übersetzt interne oder private IP-Adressen nach außen und trägt so zur Sicherheit bei, da interne Netzwerkadressen nach außen hin maskiert werden.

IDS (Intrusion Detection System)
IDS sammelt die Versuche von außen, die auf das System zuzugreifen. Dabei werden Protokollinformationen gesammelt und analysiert. Wenn diese eine kritische Masse übersteigen, wird von einer Netzwerkattacke ausgegangen, und es werden entsprechende Gegenmaßnahmen eingeleitet.

VPN (Virtual Private Network).
Ein VPM erlaubt die Verbindung aus einem geschützten Netzwerk heraus in ein anderes Netzwerk – über das öffentliche Internet hinweg. VPN kann über den iSeries Navigator installiert und konfiguriert werden. Die Daten in VPNs sind verschlüsselt und können von außen nicht mitgeschnitten werden.

L2TP (Layer 2 Tunneling Protocol)
L2TP ist ein Netzwerkprotokoll, das Frames von Protokollen der Sicherungsschicht (Layer 2) des OSI-Modells zwischen zwei Netzwerken über das Internet tunnelt, um ein virtuelles privates Netzwerk (VPN) herzustellen.

Secure Sockets Layer/Transport Layer Security
Transport Layer Security (TLS) – weitläufiger bekannt unter der Vorgängerbezeichnung Secure Sockets Layer (SSL) – ist ein hybrides Verschlüsselungsprotokoll zur sicheren Datenübertragung im Internet. Seit Version 3.0 wird das SSL-Protokoll unter dem neuen Namen TLS weiterentwickelt und standardisiert.

Portable Utilities für i5/OS
Portable Utilities für i5/OS ist ein Lizenzprogramm für IBM i und beinhaltet OpenSSH, Open SSL und Zip Open Source. Das Programm wurde auf IBM i – unter Verwendung von PASE (Portable Application Solutions Environment – die Unix-Unterstützung im i5/OS) – implementiert.

OpenSSL
OpenSSL ist eine Open Source Software, die eine vollständige SSL-Implementierung durch SSL V3 und TSL V1 bietet. OpenSSL wird für SSL-Anwendungen genutzt.

OpenSSH
OpenSSH ist ebenfalls eine Open Source Software, die SecureShell und Secure Tunneling Service anbietet, damit beispielsweise der Datenverkehr über unsichere Netzwerke – wie das Internet – geschützt wird.

Kryptographie Software Support
Benutzeranwendungen können kryptographische Dienste indirekt über IBM i Funktionen – wie z.B. SSL, VPN IPSEC und LDAP – oder direkt über APIs anwenden.

Kryptographie Hardware Support
IBM Power i Systeme bieten verschiedene kryptografische Hardwarekomponenten an. Dazu gehören u. a. die kryptographischen Koprozessoren.

Sicherheit auf Anwendungsebene
Die Anwendungsebene ermöglicht den Benutzeranwendungen – wie FTP, Telnet, HTTP und anderen Netzwerkservices – mit dem Netzwerk zu interagieren.

Allerdings ist eine Software, die den Dienst der Anwendungsebene nutzt, nicht als Teil dieser Ebene zu bewerten. Beispielsweise gehört ein Web-Browser nicht zur Anwendungsebene, weil sie die Services, die von HTTP angeboten werden, dafür nutzt, um die Informationen ans Netzwerk zu senden. Im Folgenden sehen Sie einige Sicherheitsfunktionen der Applikationsebene und ihre Ziele:

Sicherheits-funktionen	Vertrau-lichkeit	Integrität	Authenti-fizierung	Authori-sierung	Protokol-lieren Auditieren
Validation Lists			X	X	X
Digitale Zertifikate		X	X	X	
Exit-Programme			X	X	X
SSL	X	X	X	X	X
Port-Ein-schränkungen				X	
Kerberos			X		X
Kryptographie-Software	X	X	X		
Kryptpgraphie-Hardware	X	X	X		
Secure Socket APIs	X	X	X	X	X
OpenSSL	X	X	X	X	X

Validation Lists
Validation Lists sind Objekte, in denen Benutzernamen und Kennwörter oder die SSL-Zertifikate bei Zugriffskontrolle gespeichert sind. Beispielsweise können Validation Lists für Einschränkungen bei Zugriff auf den HTTP-Server verwendet werden.

Digitale Zertifikate
In der Anwendungsebene bieten digitale Zertifikate eine Verschlüsselung durch den Einsatz von öffentlichen und privaten Schlüsseln, um den Netzwerkverkehr zu schützen.

Exit-Programme
Exit-Programme werden oben in der Systemebene unter „Exit-Programme" beschrieben. Allerdings können sie auch Teil der Anwendungsebene sein, wenn sie verwendet werden, um die Funktionalität von IBM i-Funktionen oder -Anwendungen hinzuzufügen Beispielsweise bietet der FTP-Server keine Standard-Schnittstelle, um eine Protokollierung von FTP-Unterbefehlen eines angemeldeten Benutzers zu aktivieren. Mit Hilfe des Exit-Points ist es möglich, eigene Exit-Programme zu schreiben, um solche Befehle zu protokollieren.

SSL
SSL auf Netzwerkebene wird in „Secure Sockets Layer / Transport Layer Security" erklärt. Es wird verwendet, um den Netzwerkverkehr zu verschlüsseln, aber es existiert auch auf der Anwendungsebene, da die Anwendungen SSL unterstützen müssen.

Port-Beschränkungen
Port-Beschränkungen werden verwendet, um Ports, die von nicht autorisierten Anwendungen und Anwendern benutzt werden, einzuschränken. Standardmäßig erlaubt TCP/IP jedem Benutzer einen Zugang über jeden Port.

Kerberos
Kerberos wurde bereits beschrieben. Das Kerberos-Protokoll ermöglicht die Authentifizierung und Überprüfung der Identität.

Kryptographische Software
Kryptographische Software wurde auf Netzwerkebene beschrieben. Benutzer-Anwendungen können kryptographische Dienste indirekt über IBM i Funktionen – wie SSL, VPN oder IPSec – nutzen. Benutzer-Anwendungen können auch auf kryptographische Dienste direkt über APIs zugreifen.

Kryptographische Hardware
Kryptographische Hardware wurde bereits auf der Netzwerkebene beschrieben. Diese Adapter können verwendet werden, um die Performance bei SSL zu verbessern. Allerdings bieten die kryptographischen Koprozessoren viele weitere Funktionen – wie z.B. PIN-Verarbeitung, Erzeugung von Signaturen, sichere Schlüsselspeicher etc.

Secure Socket APIs
Secure Socket APIs ermöglichen es den Programmierern, Secure Socket Anwendungen auf dem System zu erstellen.

OpenSSL
OpenSSL wird verwendet, um die Umgebung zu schaffen, die benötigt wird, um SSL-fähige Anwendungen ausführen zu können.

Sicherheitsstufe – QSECURITY

QSECURITY ist der grundlegende Wert zur Festlegung der Sicherheitsstufe im IBM i System. Genau damit haben so manche Unternehmen Probleme. Die ursprünglich sehr niedrig eingestellte Sicherheitsstufe des Systems wurde über Jahre hinweg nicht beachtet – und somit ist bei vielen Firmen eine Sicherheitsstufe vorhanden, die nicht von IBM empfohlen wurde.

Diese oben erwähnte oberste Sicherheitsstufe wird im Systemwert QSECURITY festgelegt. Für diejenigen, die mit Systemwerten nichts anfangen können kann mithilfe des Befehls WRKSYSVAL auf eine Vielzahl von Systemeinstellungen zugegriffen werden. Die Systemwerte können – abhängig von der Berechtigungsstufe des Benutzers – angezeigt bzw. geändert werden.

In neueren Versionen des IBM i Systems werden selbst die Systemwerte – zumindest die sicherheitsrelevanten – extra behandelt. Über die System Service Tools (SST) können sie vor Veränderungen geschützt werden.

Mit dem Befehl DSPSYSVAL QSECURITY kann die Sicherheitsstufe des IBM i angezeigt werden. Mit WRKSYSVAL QSECURITY kann sie angezeigt und geändert werden.

```
                          Systemwert ändern

Systemwert . . . . . :   QSECURITY
Beschreibung . . . . . :   Sicherheitsstufe des Systems


Auswahl eingeben und Eingabetaste drücken.

  Sicherheitsstufe des Systems   40_   20=Nur Kennwortsicherheit
                                       30=Kennwort- und Objektsicherheit
                                       40=Integrität von Kennwort, Objekt und
                                         Betriebssystem
                                       50=Integrität von Kennwort, Objekt und
                                         erweitertem Betriebssystem
```

Systemwert QSECURITY

9.5
Seite 2

Welche Sicherheitsstufen gibt es im IBM i und was steckt dahinter?

Sicherheitsstufe	Beschreibung
10	Keine Sicherheit. Jeder Benutzer kann sich am System anmelden (wird von IBM nicht mehr unterstützt).
20	Für die Anmeldung am System ist ein Kennwort erforderlich. Benutzer haben Zugriff auf alle Systemressourcen.
30	Für die Anmeldung am System ist ein Kennwort erforderlich. Für den Zugriff auf Objekte und Systemressourcen benötigt der Benutzer die entsprechende Berechtigung.
40	Für die Anmeldung am System ist ein Kennwort erforderlich. Für den Zugriff auf Objekte und Systemressourcen benötigt der Benutzer die entsprechende Berechtigung. Die Ausführung von Programmen schlägt fehl, wenn versucht wird, über nicht unterstützte Schnittstellen auf Objekte zuzugreifen.
50	Für die Anmeldung am System ist ein Kennwort erforderlich. Für den Zugriff auf Objekte und Systemressourcen benötigt der Benutzer die entsprechende Berechtigung. Die Ausführung von Programmen schlägt fehl, wenn versucht wird, nicht unterstützte Parameterwerte an unterstützte Schnittstellen zu übergeben oder über nicht unterstützte Schnittstellen auf Objekte zuzugreifen.

Die Sicherheitsstufe 10 spielt also sowieso keine Rolle mehr. Dagegen ist die Sicherheitsstufe 50 das andere Extrem, dass ich in der Praxis auch noch nichts gesehen habe. Bleiben die Sicherheitsstufen 20 bis 40, mit denen die meisten Systeme arbeiten.

In Sicherheitsstufe 20 haben alle Benutzerprofile *ALLOBJ Berechtigung. Das bedeutet, dass die Objekt-Berechtigung sowie alle weiteren Objekte im Power i System uneingeschränkt verwendet werden können: Programme, Dateien, Benutzer, Ausgabewarteschlangen, Drucker, Bildschirme.

Jeder Benutzer – egal ob Geschäftsführer, Abteilungsleiter, Buchhalter, Verkäufer, Einkäufer, Lagerist etc. – hat bei Sicherheitsstufe 20 vollen Zugriff auf alles im System. Das ist nicht gut, meinen Sie vielleicht – und haben damit recht. Das bedeutet nämlich auch, dass jeder Benutzer Zugriff auf Buchhaltungsdaten, Kalkulationen, Konditionen und auf Lohndaten (sofern diese auf dem Power i System abgerechnet werden) hat.

Wie passt denn das zusammen? Sicheres System und Sicherheitsstufe 20? Hier sollte man die Arbeitsweise auf dem System in der Rückschau betrachten.

Ein wichtiges Merkmal bei der Verarbeitung von Anweisungen im IBM i sind Befehle. So ziemlich alle Aktionen im Betriebssystem sind mit Befehlen ausgestattet, die es einem Benutzer oder Administrator erlauben, bestimmte Funktionen aufzurufen. Um etwas zu arbeiten, ruft man WRKxxx auf, um etwas anzuzeigen DSPxxx und um etwas zu erstellen CRTxxx.

```
                                  Befehl auswählen

Auswahl eingeben und Eingabetaste drücken.
  1=Auswählen

Ausw  Befehl       Bibliothek  Text
 _    WRKACTJOB    QSYS        Mit aktiven Jobs arbeiten
 _    WRKALR       QSYS        Mit ALERTS arbeiten
 _    WRKALRD      QSYS        Mit ALERT-Beschreibungen arb.
 _    WRKALRTBL    QSYS        Mit ALERT-Tabelle arbeiten
 _    WRKAPPNSTS   QSYS        Mit APPN-Status arbeiten
 _    WRKARMJOB    QSYS        Work with ARM Jobs
 _    WRKASPJOB    QSYS        Mit ASP-Jobs arbeiten
 _    WRKAUT       QSYS        Mit Berechtigung arbeiten
 _    WRKAUTL      QSYS        Mit Berechtig.listen arbeiten
 _    WRKBNDDIR    QSYS        Mit Binderverzeichn. arbeiten
 _    WRKBNDDIRE   QSYS        Mit Binderverz.eintr. arbeiten
 _    WRKBPTBL     QSYS        Mit BOOTP-Tabelle arbeiten
```

Befehle WRK…

Ein Befehl funktioniert nur richtig, wenn man a) die zum Befehl dazugehörigen Parameter kennt und b), wenn man weiß, wie der Befehl korrekt heißt. Hierbei hilft das Betriebssystem IBM i weiter. Wenn man beispielsweise vorhat, sich nur etwas anzeigen zu lassen, dann gibt man DSP* ein und drückt die F4-Taste. Mit F4 werden neben dem Befehl auch noch die Werte angezeigt, mit denen die einzelnen Parameter gefüllt werden können.

```
                    Mit aktiven Jobs arbeiten (WRKACTJOB)

Auswahl eingeben und Eingabetaste drücken.

Ausgabe  . . . . . . . . . . . .   *_____        *, *PRINT

                 Zusätzliche Parameter

Statistik zurücksetzen . . . . .   *NO           *NO, *YES
Subsystem  . . . . . . . . . . .   *ALL_____     Name, *ALL
     + für weitere Werte           _________
CPU-Prozentgrenze  . . . . . . .   *NONE         .1-99.9, *NONE
Antwortzeitgrenze  . . . . . . .   *NONE         .1-999,9 Sek. *NONE
Folge  . . . . . . . . . . . . .   *SBS_____     *SBS, *AUXIO, *CPU...
Jobname  . . . . . . . . . . . .   *ALL_____     Name, generisch*, *ALL...
Intervall für autom. Aktualis.     *PRV          5-999 Sekunden, *PRV
```

WRKACTJOB mit allen Parametern

30. Ergänzung 3/2012

```
        Wert für Parameter angeben                SEQ

Auswahl eingeben und Eingabetaste drücken.


Folge  . . . . . . . . . . . . . .   *SBS

  *SBS                                   *THREADS
  *AUXIO                                 *TYPE
  *CPU                                   *USER
  *CPUPCT
  *CURUSR
  *FUNCTION
  *INT
  *JOB
  *NUMBER
  *POOL
  *PTY
  *RSP
  *STS
```

Werte für Parameter Folge mit F4 gepromptet

Doch zurück zum Wichtigen: Befehle werden oftmals nicht einfach nur aufgerufen. Genau das unterscheidet IBM i auch beispielsweise von einem Unix: Die Befehle sind in Menüs eingebunden. Es gibt Menüs für alle möglichen Betriebssystemfunktionen, aber auch für eigene Programmaufrufe aus Warenwirtschaft, Buchhaltung etc. Mit GOxxx wird ein Menü aufgerufen – beispielsweise zeigt GO MAIN das Hauptmenü des Betriebssystems und GO ASSIST das Operational Assistant Menü.

```
MAIN                              i5/OS-Hauptmenü

Auswahlmöglichkeiten:

     1. Benutzeraufgaben
     2. Büroaufgaben
     3. Allgemeine Systemaufgaben
     4. Dateien, Bibliotheken und Ordner
     5. Programmierung
     6. Datenfernverarbeitung
     7. System definieren oder ändern
     8. Problembehandlung
     9. Menü anzeigen
    10. Unterstützende Informationen - Auswahlmöglichkeiten
    11. iSeries Access-Aufgaben

    90. Abmelden
```

Hauptmenü des IBM i

Aber was hat das jetzt mit Sicherheit zu tun, fragen Sie sich vielleicht? Ganz einfach: Menüs und Benutzer gehören beim IBM i oft fest zusammen.

Jeder Anwender bekommt im IBM i ein Benutzerprofil, das neben seinem Anmeldenamen auch sein Kennwort und viele Parameter beinhaltet, um diesen Anwender – nach der Anmeldung im System – individuell zu behandeln. Zu den Benutzerprofilen später aber mehr.

Ein wichtiges Merkmal, um auch bei Sicherheitsstufe 20 eine brauchbare Sicherheit zu bekommen, ist die Zuordnung eines Menüs. Daran ist die Option gekoppelt, dass ein Benutzer lediglich nur eine Menüauswahl aufrufen kann und sonst nichts.

So genannte „Zwangsmenüs“ – so heißt das Zauberwort hier. Im Benutzerprofil gibt es einen Parameter, der die Möglichkeit, in der Befehlszeile einer Bildschirmsitzung etwas eingeben zu können, steuert.

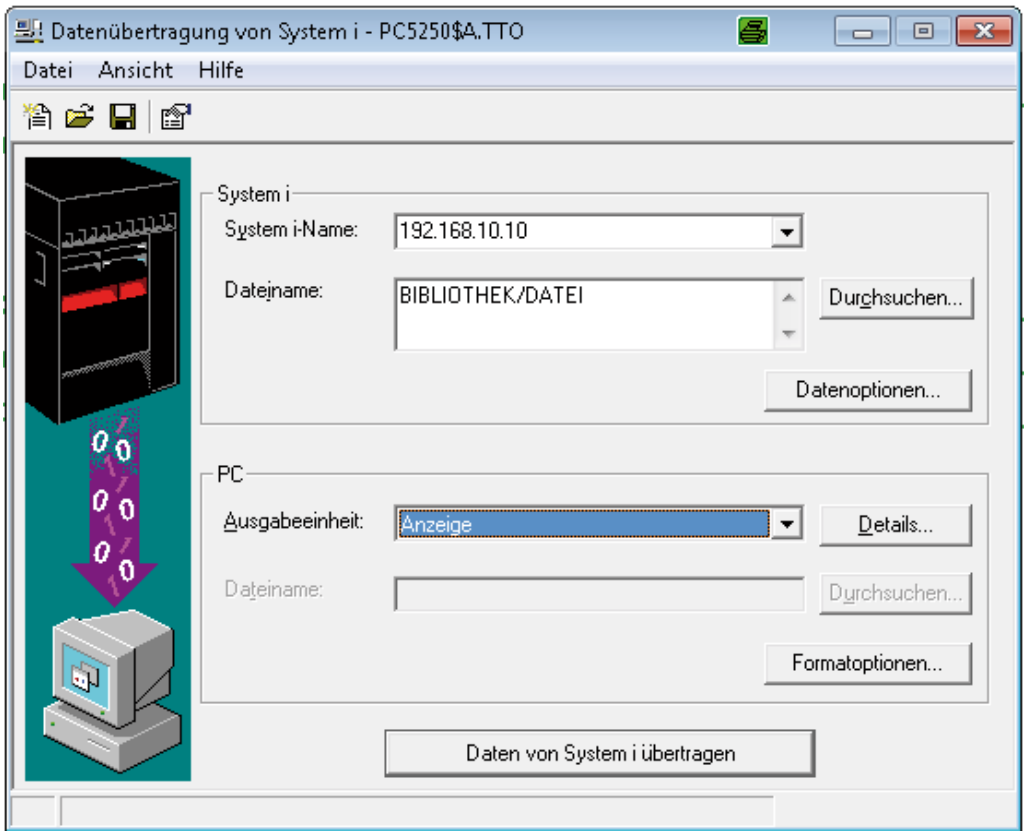
```
                    Benutzerprofil ändern (CHGUSRPRF)

Auswahl eingeben und Eingabetaste drücken.

Benutzerprofil . . . . . . . . . > RENGEL        Name
Benutzerkennwort . . . . . . . .   *SAME

Kennwort auf abgelaufen setzen     *NO           *SAME, *NO, *YES
Status . . . . . . . . . . . . .   *ENABLED      *SAME, *ENABLED, *DISABLED
Benutzerklasse . . . . . . . . .   *SECOFR       *SAME, *USER, *SYSOPR...
Unterstützungsstufe  . . . . . .   *SYSVAL       *SAME, *SYSVAL, *BASIC...
Aktuelle Bibliothek  . . . . . .   *CRTDFT       Name, *SAME, *CRTDFT
Aufzurufendes Startprogramm  . .   *NONE         Name, *SAME, *NONE
  Bibliothek . . . . . . . . . .                 Name, *LIBL, *CURLIB
Anfangsmenü  . . . . . . . . . .   MAIN          Name, *SAME, *SIGNOFF
  Bibliothek . . . . . . . . . .     *LIBL       Name, *LIBL, *CURLIB
Möglichkeiten einschränken . . .   *NO           *SAME, *NO, *PARTIAL, *YES
Text 'Beschreibung'  . . . . . .   'Robert Engel'
```

*Möglichkeiten einschränken – LMTCPB *YES/*NO*

Wenn der Parameter „Möglichkeiten einschränken“ auf *NO steht, kann der Benutzer – bei Anzeige eines Menüs – die Befehlszeile nutzen und einen Befehl aus dem Betriebssystem oder der Anwendung bzw. aus den Programmen aufzurufen. Steht diese Möglichkeit auf *YES, kann der Anwender nur die Optionen aus dem Menü aufrufen, die ihm die Programmierer zur Verfügung gestellt haben.

Prima – bei Sicherheitsstufe 20 gibt es Zwangsmenüs und alles ist in Ordnung? Jeder Anwender kann nur das aufrufen, was im Menü steht und das war es!? Schön und gut, aber das Ganze hat einen Haken: Seit vielen Jahren gibt es nicht nur die gleichermaßen geliebte und gehasste Anzeige der 5250-Emulation, sondern auch eine externe Zugriffsmöglichkeit, die durch die Einführung des TCP/IP-Protokolls – damals noch für die AS/400 – entstanden ist. Vorher war die Kommunikation mit dem SNA-Protokoll (Systems Network Architektur) weitestgehend auf Host-Systeme beschränkt, nun kamen plötzlich die PCs mit all ihren Möglichkeiten ins Spiel.

So konnten und können die Anwender immer noch über eine einfache IBM System i Access-Sitzung mit den entsprechend installierten Optionen, Daten aus den Power i Systemen mit wenigen Mausklicks auf die PCs übertragen. Oder sie können Auswertungsprogramme starten, die über SQL-Anweisungen Daten aus diesen Power i Datenbanken – z. B. in Excel – übertragen.



Datenübertragung vom System i

Was ist jetzt noch die Sicherheitsstufe 20 wert, wenn jeder Anwender auf alle Daten des Systems uneingeschränkt (*ALLOBJ-Berechtigung) zugreifen kann? Ok – es soll tatsächlich Firmen geben, in denen jeder Mitarbeiter alles wissen darf, aber normalerweise ist so eine Vorgehensweise securitytechnisch gesehen eine Katastrophe.

Dagegen muss etwas unternommen werden. Empfehlenswert ist die Sicherheitsstufe 40, da selbst bei dieser, erfahrene Programmierer die Sicherheitsprüfungen nicht umgehen können. Wenn ein neues Power i System in Betrieb genommen werden soll, dann empfehlen alle Sicherheitsexperten grundsätzlich mit der Sicherheitsstufe 40 zu beginnen.

Auch aus anderen Gründen ist die Menüsicherheit mit Sicherheitsstufe 20 nicht ausreichend. Oftmals haben Anwender hier aber auch den Zugriff auf eine Befehlszeile oder auf eine Auswertungssoftware zur Analyse von Unternehmensdaten. Benutzer auf Sicherheitsstufe 20 sollten jedoch keinerlei Zugriff auf diese Befehlszeile oder auf Query-Funktionalität erhalten, denn das Benutzerprofil mit der Sonderberechtigung *ALLOBJ berechtigt sie zum uneingeschränkten Zugriff auf alle Daten im System.

Sicherheitsstufe 30 bietet eine spürbare Verbesserung der gesamten Sicherheit. Auch Sicherheitsstufe 40 kostet nicht zusätzliche Systemleistung oder Overhead. Sicherheitsstufe 40 verhindert potentielle Integritäts- oder Sicherheitsprobleme, die von Programmen ausgehen, die in besonderen Fällen die Sicherheitsvorkehrungen umgehen können.

9.5
Seite 8

Sicherheitsstufe 40

Nachfolgend finden Sie eine Zusammenfassung des erhöhten Schutzes auf Sicherheitsstufe 40.

Interne Schnittstellen – Das System unterbindet Versuche, Systemprogramme, die nicht als Schnittstellen auf Aufrufebene definiert sind, direkt aufzurufen. Beispiel: Direkter Aufruf des befehlsverarbeitenden Programms für den Befehl SIGNOFF. Der Schutz der internen Schnittstellen auf Sicherheitsstufe 40 und höher beinhaltet beispielsweise die Verhinderung des Zugriffs auf interne Systemstrukturen über die Zeiger-Funktionalität von Sprachen – wie C, PASCAL oder MI.

Jobübergabe – Wenn in einer Jobbeschreibung ein Benutzerprofilnamen als Wert für das Feld USER verwendet wird, können alle Jobs, die mit der Jobbeschreibung übergeben werden, mit Attributen ausgeführt werden, die von diesem Benutzerprofil stammen. Ein unberechtigter Benutzer könnte einen Sicherheitsverstoß begehen, indem er einen Job für die Ausführung unter dem in der Jobbeschreibung angegebenem Benutzerprofil übergibt.

***USE-Berechtigung** – Auf Sicherheitsstufe 30 wird der Job unter der Voraussetzung ausgeführt, dass die übergebende Person *USE-Berechtigung für die Jobbeschreibung besitzt. Auf Sicherheitsstufe 40 und höher muss der Benutzer, der den Job übergibt, *USE-Berechtigung für die Jobbeschreibung und das in der Jobbeschreibung angegebene Benutzerprofil besitzen. Anderenfalls wird der Job fehlschlagen. Diese Eigenschaft ist für die meisten Power i Systeme der wichtigste Vorteil der Sicherheitsstufe 40 gegenüber der Sicherheitsstufe 30. Die Standardberechtigung für die Benutzerprofile ist *EXCLUDE. Daher müssen die Benutzer ausdrücklich die Berechtigung erhalten, um die Jobs als andere Benutzer ausführen zu können.

Anmelden ohne Kennwort – Auf Sicherheitsstufe 30 und niedriger ist es bei bestimmten Subsystembeschreibungen möglich, sich ohne Benutzerkennung und Kennwort anzumelden. Auf Sicherheitsstufe 40 und höher verhindert das System jeden Versuch der Anmeldung ohne Benutzerkennung und Kennwort.

Ich empfehle Ihnen, Ihr Power i System mit Sicherheitsstufe 40 in Betrieb zu nehmen, wenn Sie nicht Software von Drittanbietern installiert haben, die mit den Restriktionen der Sicherheitsstufe 40 nicht kompatibel ist.

Planen Sie jedoch nach Möglichkeit nicht von Sicherheitsstufe 20 direkt auf Sicherheitsstufe 40 zu wechseln, solange Ihr System aktiv im Produktionsbetrieb ist.

9.6
Seite 2

Die vorbereitenden Schritte, die für einen reibungslosen Übergang auf Sicherheitsstufe 40 erforderlich sind, werden in zwei Abschnitten beschrieben: Von Sicherheitsstufe 20 auf Sicherheitsstufe 30 und von Sicherheitsstufe 30 auf Sicherheitsstufe 40. Wenn Sie die beschriebenen Schritte ausführen, können Sie Geschäftsleitung, Benutzer und Auditoren von Ihrer neuen Power i Sicherheitsstufe überzeugen und beeindrucken.

Von Sicherheitsstufe 20 auf Sicherheitsstufe 30 wechseln

Der Wechsel von Sicherheitsstufe 20 auf Sicherheitsstufe 30 geschieht, indem Sie den Systemwert QSECURITY auf 30 ändern. Die Aktivierung von Sicherheitsstufe 30 wird mit dem nächsten IPL ausgeführt. Wenn Sie von Sicherheitsstufe 20 auf Sicherheitsstufe 30 wechseln, ändert das System beim nächsten IPL alle Benutzerprofile. Sonderberechtigungen werden den Benutzerprofilen hinzugefügt bzw. entfernt und somit an die Sonderberechtigung für die Benutzerklasse angepasst.

Leider ist die Umstellung auf Sicherheitsstufe 30 nicht ganz unproblematisch. Wenn beispielsweise Ihr System Produktionsanwendungen auf einer niedrigeren Sicherheitsstufe ausführt, sollten Sie die Sicherheit auf Objektebene konfigurieren und testen, bevor Sie zur Sicherheitsstufe 30 wechseln. Wenn Sie den Systemwert ohne ausreichende Planung ändern, ist es möglich, dass Ihr Telefon ständig von frustrierten Anwendern belegt wird, die sich darüber beklagen, dass Aufgaben, die früher problemlos ausgeführt werden konnten, mit Nachrichten wie „keine Berechtigung“ fehlschlagen.

Wenn Sie auf Sicherheitsstufe 30 wechseln, wird die Berechtigung *ALLOBJ aus allen Benutzerprofilen entfernt, bei denen es sich nicht um die Benutzerklasse *SECOFR handelt. Benutzer, die früher über *ALLOBJ-Zugriff auf alle Objekte im System verfügten, benötigen nun spezifische Berechtigung für den Zugriff auf alle Objekte. Der nachfolgend beschriebene schrittweise Prozess ermöglicht den reibungslosen Übergang von Sicherheitsstufe 20 auf Sicherheitsstufe 30.

1. Ändern Sie den Systemwert QSECURITY nicht sofort von 20 auf 30. Arbeiten Sie weiter mit Sicherheitsstufe 20, während Sie den Benutzern die Berechtigung für die Objekte erteilen.
2. Weisen Sie den Benutzern Gruppen zu.
   - Bestimmen Sie, ob Benutzergruppen mit ähnlichen Jobanforderungen vorhanden sind – wie z.B. Benutzer aus der gleichen Abteilung, die Zugriff auf die gleichen Informationen benötigen. Einige Benutzer gehören eventuell zu keiner derartigen Gruppe.
   - Spezifizieren Sie die Sonderberechtigung *NONE, wenn Sie ein Gruppenprofil für Benutzer mit ähnlichen Anforderungen erstellen. Das Kennwort für Gruppenprofile sollte *NONE sein, um so die Anmeldung als Gruppenprofil zu verhindern. Ich empfehle eine Namenskonvention für Gruppenprofile (wie z. B. GRPXXX), um Gruppenprofile

eindeutig zu kennzeichnen. Ein Beispiel für den Befehl zur Erstellung eines Gruppenprofils für die Vertriebsabteilung ist:

```
CRTUSRPRF USRPRF(GRPVTR) +
PASSWORD(*NONE) SPCAUT(*NONE) +
TEXT('Gruppenprofil Vertriebsabteilung')
```

- Verwenden Sie den Befehl CHGUSRPRF (Benutzerprofil ändern), um individuelle Benutzerprofile einer Gruppe zuzuweisen.

  Wenn Sie die Benutzer einem Gruppenprofil zuweisen, empfehle ich die Verwendung des Parameters OWNER(*GRPPRF). Damit kennzeichnen Sie das Gruppenprofil als Eigner der neuen Objekte, die von den Benutzern in der Gruppe erstellt werden. Der Befehl für die Zuweisung der Benutzer zum Gruppenprofil GRPCTR lautet:

```
CHGUSRPRF USRPRF(user_profile) +
GRPPRF(GRPVTR) OWNER(*GRPPRF)
```

3. Erstellen Sie ein Testbenutzerprofil. Sie können das Testbenutzerprofil verwenden, um fehlende Berechtigung zu entdecken und das Problem zu korrigieren, bevor es sich auf die Produktionsbenutzer auswirkt. Wenn Sie die Testbenutzerprofile erstellen, weisen Sie die Sonderberechtigung *NONE zu und nehmen das Benutzerprofil in die Gruppe auf. Der folgende Befehl beschreibt die Erstellung des Testprofils für die Vertriebsgruppe:

```
CRTUSRPRF USRPRF (TESTVTR) PASSWORD(X) +
USRCLS(*USER) INLPGM(–) INLMNU(–) +
SPCAUT(*NONE) GRPPRF(GRPVTR) + TEXT('Testprofil
für die Vertriebsgruppe')
```

Die Parameterwerte, die im oben beschriebenen Wert als „'—“ angegeben werden, sollten dieselben sein wie die Werte, die den Benutzern in dieser Gruppe aktuell zugewiesen worden sind. Eine einfache Methode für das Kopieren der Parameter aus einem vorhanden Benutzerprofil besteht darin, den Befehl WRKUSRPRF – (Mit Benutzerprofil arbeiten) zu verwenden und die Kopieroption (Option 3) auszuwählen. Mit dieser Aktion wird eine Bedienerführung für das neue Profil mit den aktuellen Werten ausgegeben. Sie müssen nur noch Profilnamen, Benutzerklasse, Sonderberechtigung, Gruppenprofil und Text ändern.

4. Suchen und korrigieren Sie mit dem Testprofil für die Gruppe Bedingungen, bei denen die Nachricht „Nicht berechtigt“ angezeigt wird.
   - Melden Sie sich mit dem Benutzerprofil TESTVTR an und führen Sie die Aufgaben eines Benutzers in der Vertriebsgruppe aus.
   - Wenn eine Aufgabe mit der Nachricht „Nicht berechtigt“ angezeigt wird, verwenden Sie den Befehl GRTOBJAUT (Objektberechtigung erteilen), um den erforderlichen Zugriff auf das Gruppenprofil zuzuweisen oder die Zugriffsberechtigung für das Objekt auf *PUBLIC zu setzen.
   - Eine Alternative für die Erteilung von Zugriffsberechtigung ist die Verwendung der Berechtigungsübernahme durch Programme.
   - Arbeiten Sie weiterhin mit dem Benutzerprofil TESTVTR, bis Sie alle Aufgaben eines Gruppenmitglieds ausführen können.
5. Entfernen Sie die Berechtigung *ALLOBJ eines einzelnen Produktionsbenutzers in der Gruppe.
   - Informieren Sie diesen Benutzer, dass Sie die Systemsicherheit verbessern möchten. Fordern Sie ihn dazu auf, sich an Sie zu wenden, sobald sicherheitsbezogene Anwendungsprobleme auftreten.
   - Entfernen Sie mit Hilfe des Befehls CHGUSRPRF die Berechtigung SPCAUT(*ALLOBJ) aus diesem individuellen Profil. Wenn der Produktionsbenutzer Nachrichten zur Nichtberechtigung erhält, erteilen Sie die erforderliche Berechtigung der Allgemeinheit (*PUBLIC), dem Gruppenprofil oder dem individuellen Profil. In den meisten Fällen sollten die Objekte über die Berechtigung *PUBLIC verfügen oder über Berechtigung für das Gruppenprofil, deshalb ist nicht zu erwarten, dass andere Mitglieder der Gruppe Nachrichten über eine Nichtberechtigung erhalten.
   - Setzen Sie Ihre Arbeit fort, bis der Produktionsbenutzer alle Aufgaben ausführen kann.
6. Entfernen Sie mit dem Befehl CHGUSRPRF den Wert *ALLOBJ aus allen anderen Profilen der Gruppenmitglieder.
7. Löschen Sie das Testbenutzerprofil TESTVTR, das in Schritt 3 erstellt wurde. Nicht verwendete Profile, die bei Testläufen übriggeblieben sind oder von Benutzern stammen, die versetzt wurden und nicht mehr im System arbeiten, sind eine in vielen Systemen häufig anzutreffende Sicherheitslücke. Das Problem kann am einfachsten gelöst werden, indem die nicht verwendeten Benutzerprofile sofort gelöscht werden.
8. Wiederholen Sie die Schritte 3 bis 7 für jede weitere Gruppe. Wählen Sie jeweils eine Gruppe aus und bearbeiten Sie diese vollständig, bevor Sie die nächste Gruppe konfigurieren.

9. Nun wenden Sie sich den Benutzern zu, die nicht Mitglieder eines Gruppenprofils sind. Nachdem Sie alle Produktionsbenutzer, die Mitglieder einer Gruppe sind, mit ausreichender Berechtigung ausgestattet haben, müssen Sie die *ALLOBJ-Zugriffsrechte den übrigen Benutzern entziehen, die nicht Mitglied einer Gruppe sind. Der Befehl DSPAUTUSR (Berechtigte Benutzer anzeigen) kann alle Benutzerprofile und ihre Gruppenmitgliedschaft anzeigen.

   DSPAUTUSR kann darüber hinaus Profile anzeigen, die zu keinem Gruppenprofil gehören. Ich empfehle Ihnen, die Einstellungen für Benutzer, die zu keiner Gruppe gehören, erst nach Beendigung der Arbeit mit allen Gruppen vorzunehmen. Auf diese Weise können Sie die Anzahl der Anrufe reduzieren, die Sie von Benutzern mit Berechtigungsproblemen erhalten.

10. Vergeben Sie an privilegierte Benutzer Berechtigungen für die Arbeitsstationen. Wenn der Systemwert QLMTSECOFR (Zugriff des Sicherheitsadministrators auf die Einheiten begrenzen) 1 (Ja) beträgt, müssen Benutzer mit spezieller *ALLOBJ – oder *SERVICE-Berechtigung – wie z. B. QSECOFR – spezifische Berechtigung erhalten, um auf Einheiten mit Sicherheitsstufe 30 oder höher zuzugreifen. Geben Sie diesen Benutzern für die Arbeitsstation, die sie verwenden, *CHANGE-Berechtigung.

11. Ändern Sie den Systemwert. Nachdem Sie aus allen Profilen – außer denen der Benutzer der Benutzerklasse *SECOFR – die Berechtigung *ALLOBJ entfernt haben, können Sie den Systemwert QSECURITY von 20 auf 30 ändern, ohne die Benutzer in Aufruhr zu versetzen. Wenn Sie den Systemwert von 20 auf 30 ändern und ein IPL ausführen, wird *ALLOBJ aus allen Benutzerprofilen entfernt. Wenn Sie die Schritte 1 bis 10 befolgen, muss das System keine Sonderberechtigung hinzufügen oder entfernen, um die Standardeinstellungen für Sicherheitsstufe 30 zu berücksichtigen. Nachdem der Systemwert geändert wurde und ein IPL erfolgt ist, können Sie sich gratulieren. Aber vergewissern Sie sich, dass Sie nicht den letzten, wichtigen Schritt vergessen haben.

    Nachdem Sie die Benutzerprofile und Benutzergruppen mit der korrekten Berechtigung versehen haben, müssen Sie SAVSECDTA (Sicherheitsdaten speichern) oder SAVSYS (System sichern) ausführen, um eine Sicherungskopie der Benutzerprofile und Berechtigungen zu erstellen.

Von Sicherheitsstufe 30 auf Sicherheitsstufe 40 wechseln

Die Prozedur für den Wechsel von Sicherheitsstufe 30 auf Sicherheitsstufe 40 ähnelt dem Wechsel von Sicherheitsstufe 20 auf Sicherheitsstufe 30, aber Ihr Augenmerk sollte dabei auf den Programmen liegen, die gegen die Einschränkungen der Sicherheitsstufe 40 verstoßen, die weiter oben in diesem Kapitel beschrieben wurden.

In der folgenden Übersicht ist zu sehen, was in den unterschiedlichen Sicherheitsstufen unterstützt wird:

Funktionsübersicht Sicherheitsstufen	20	30	40	50
Benutzername zum Anmelden erforderlich.	Ja	Ja	Ja	Ja
Kennwort zum Anmelden erforderlich.	Ja	Ja	Ja	Ja
Kennwortschutz aktiv.	Ja	Ja	Ja	Ja
Menü und Startprogramm aktiv.	Ja	Ja	Ja	Ja
Einschränkung der Eingaben aktiv.	Ja	Ja	Ja	Ja
Ressourcenschutz aktiv.	Nein	Ja	Ja	Ja
Zugriff zu allen Objekten.	Ja	Nein	Nein	Nein
Sicherheitsauditierung aktiv.	Ja	Ja	Ja	Ja
Programme mit nicht unterstützten Interfaces verwenden. Diese laufen auf Fehler.	Nein	Nein	Ja	Ja
Erweiterter Hardwareschutz unterstützten.	Nein	Nein	Ja	Ja
*USRSPC, *USRIDX, und *USRQ Objekte können nur in Bibliotheken, die im Systemwert QALWUSRDMN angegeben sind, erstellt werden.	Ja	Ja	Ja	Ja
Pointer in Parametern werden für Benutzerdomänenprogramme im System und User State geprüft.	Nein	Nein/ Ja	Ja	Ja
Nachrichtenbehandlungs-regeln zwischen System und User State gefordert.	Nein	Nein	Nein	Ja
Einem Programm zu gewiesener Speicher kann nicht direkt geändert werden.	Nein	Nein	Ja	Ja
Interne Control Blocks sind geschützt.	Nein	Nein	Ja	Ja

9.8
Seite 2

Neben den in Tabelle 1 genannten Funktionen ist es wichtig zu wissen, dass die Sicherheitsstufe die standardmäßig vergebenen Sonderberechtigungen bestimmt. Um zu verstehen, wie Sonderberechtigungen zu den Benutzerklassen gehören, wenn Sie ein Benutzerprofil erstellen, drücken Sie die Hilfe-Taste auf dem Parameter „Benutzerklasse“. Hier wird das Zusammenspiel erläutert.

Hilfe zur Benutzerklasse

Gibt die Art des Benutzers an, der diesem Benutzerprofil zugeordnet ist: Sicherheitsbeauftragter, Sicherheitsadministrator, Programmierer, Systembediener oder Benutzer. Die Benutzerklasse steuert die im Menü gezeigten Auswahlmöglichkeiten. Sonderberechtigungen werden nur erteilt, wenn *USRCLS für den Parameter „Sonderberechtigung“ (SPCAUT) angegeben wird. Wurde SPCAUT(*USRCLS) angegeben, variieren die Sonderberechtigungen abhängig vom QSECURITY-Wert.

***SAME**
Der Wert ändert sich nicht.

***USER**
Auf QSECURITY-Sicherheitsstufe 10 oder 20 verfügt der Benutzer über die Berechtigungen *ALLOBJ und *SAVSYS.

Auf QSECURITY-Sicherheitsstufe 30 oder höher hat der Benutzer keine Sonderberechtigungen.

***SECOFR**
Auf allen Sicherheitsstufen verfügt der Sicherheitsbeauftragte über die folgenden Sonderberechtigungen:

*ALLOBJ
*SAVSYS
*JOBCTL
*SERVICE
*SPLCTL
*SECADM
*AUDIT
*IOSYSCFG

***SECADM**
Auf QSECURITY-Sicherheitsstufe 10 oder 20 verfügt der Sicherheitsadministrator über die Sonderberechtigungen *ALLOBJ, *SAVSYS, *SECADM und *JOBCTL.

Auf QSECURITY-Sicherheitsstufe 30 oder höher verfügt der Benutzer über die Sonderberechtigungen *SECADM.

***PGMR**
Auf QSECURITY-Sicherheitsstufe 10 oder 20 verfügt der Programmierer über die Sonderberechtigungen *ALLOBJ, *SAVSYS und *JOBCTL.

Auf QSECURITY-Sicherheitsstufe 30 oder höher hat der Benutzer keine Sonderberechtigungen.

***SYSOPR**
Auf QSECURITY-Sicherheitsstufe 10 oder 20 verfügt der Systembediener über die Sonderberechtigungen *ALLOBJ, *SAVSYS und *JOBCTL.

Auf QSECURITY-Sicherheitsstufe 30 oder höher verfügt der Benutzer über die Sonderberechtigungen *SAVSYS und *JOBCTL.

Wenn Sie ein Benutzerprofil erstellen, können Sie Sonderberechtigungen auf der Grundlage der Benutzer-Klasse auswählen. Sonderberechtigungen werden zu den Benutzerprofilen ebenfalls hinzugefügt oder davon entfernt, wenn Sie die Sicherheitsstufe ändern.

Sicherheitsstufe 40 betrifft fast jedes, wenn nicht gar jedes Objekt auf dem System. Da es viel bei der Umstellung auf Level 40 zu beachten gibt, sollte man wissen, dass es einen „empfohlenen Weg“ gibt. Das Tool, das alle Experten empfehlen, ist das Auditjournal. Das Auditjournal muss erstellt und eingeschaltet werden. Der Inhalt sollte für eine Weile überwacht werden, um sicher zu sein, dass das System mit den Anwendungen die Sicherheitsstufe 40 oder höher unterstützt. Dies führt zur nächsten Frage: „Brauche ich die Journalisierung, damit die Sicherheitsstufe 40 laufen kann?“

Die einfache Antwort auf diese Frage ist „Ja“, aber dabei handelt es sich nicht exakt um das gleiche Konzept wie die Datenbank-Journalisierung. Sie brauchen zum Einrichten des Auditjournals ein Journalobjekt auf Ihrem System und müssen die Journalempfänger an dieses anhängen – ähnlich wie bei der Datenbank-Journalisierung. Dann müssen Sie Ihrem System über die Systemwerte QAUDLVL und QAUDLVL2 beibringen, welche Art von Sicherheitsprüfungen aufgezeichnet und gesucht werden sollen.

Sicherheitsprüfungen deshalb, weil sie bereits seit der Sicherheitsstufe 30 existieren. Nur sind sie in Sicherheitsstufe 30 noch nicht wirksam, doch es gibt sie. Mit anderen Worten – ab Sicherheitsstufe 40 weiß das System, dass Ihr Code die Integrität oder andere Regeln der Sicherheitsstufe 40 oder 50 Regeln verletzt. Nachdem das System den Fehler aufgezeichnet hat, was sich natürlich auf die Leistung auswirkt, wird bei Sicherheitsstufe 30 das Programm trotzdem funktionieren. In Sicherheitsstufe 40 und 50, erzeugt diese Ausführung Meldungen und der Job schlägt fehl.

Die gute Nachricht ist, dass der Sprung von Sicherheitsstufe 30 auf 40 keine Änderungen an den Profilen oder Berechtigungslisten mit sich bringt, so dass es sich leicht umstellen lässt. Darüber hinaus hat es keinen Einfluss auf die Programme, die Berechtigungen übernehmen (so genannte Adopted Authority), sofern diese Standard-Interfaces verwenden.

Nach Konfiguration der Auditjournalisierung zur Fehleraufzeichnung sollten Sie eine Zeit lang alle Geschäftsvorfälle, die vorkommen, beobachten. Ideal ist es zum Ende des Geschäftsjahres – mit allen jährlichen, vierteljährlichen, monatlichen und täglichen Abläufen. Ziel muss es sein, so viele Prozesse wie möglich durchzuspielen, um festzustellen, ob die Einträge im Auditjournal mit dem Typ AF (Authority Failure – Berechtigungsfehler) generiert werden.

Die Einträge im Auditjournal vom Typ AF werden nicht viele sein, selten gibt es Auditjournale mit mehr als 100 Seiten. Das ist in der Regel keine große Sache in Bezug auf die Speicherung; die Auswirkungen auf die Leistung gehen Richtung Null.

Sie werden erstaunt sein, was auf Ihren Auditprotokollen alles erscheint. Sie tun bestimmt nur, was notwendig ist, um schnell auf die Sicherheitsstufe 40 wechseln zu können. Aber es gibt eine Menge von anderen Optionen, die Ihnen helfen können, Ihr System transparenter zu machen. Sollten Sie mit der Auswertung der Auditjournale an ihre Grenzen stoßen, gibt es einige Produkte von Security-Anbietern, die Ihnen die Arbeit durch vordefinierte Regeln und Auswertungen leichter machen.

Bevor Sie mit der Auditjournal-Prozedur beginnen, sollten Sie eine Bestandsaufnahme aller Software-Pakete auf Ihrem System vornehmen und alle Anbieter kontaktieren, um zu sehen, ob diese mit der Sicherheitsstufe 40 konform sind. Sollten sie nicht kompatibel sein, können Sie die Sicherheitsstufe 40 nicht umsetzen. D.h., Sie sollten ein Update installieren, damit es verfügbar ist. Die meisten Anbieter sind bereits mit ihrer Software-Version kompatibel, aber trotzdem liegt es an Ihnen, die Entscheidung zu treffen. Deshalb: Egal, was Ihnen die Verkäufer sagen, lassen Sie trotzdem das Journal für eine Weile laufen, um sicherzustellen, dass alles sauber funktioniert.

Bevor wir uns der Einrichtung und Anhängens des Journals widmen, lassen Sie uns die Art der Einträge, anschauen, nach denen Sie im Journal schauen müssen. Die beste Quelle dafür ist die Tabelle des IBM Sicherheits-Referenz-Handbuch – nachstehend in vereinachter Form:

Szenarienbeschreibung	30	40	50
Ein Programm versucht über nicht unterstützte Schnittstellen auf ein Objekt zuzugreifen.	Journal-eintrag AF[1]	Journal-eintrag AF[1] Operation schlägt fehl.	Journal-eintrag AF[1] Operation schlägt fehl.
Ein Programm versucht eine eingeschränkte Anweisung auszuführen.	Journal-eintrag AF[1]	Journal-eintrag AF[1] Operation schlägt fehl.	Journal-eintrag AF[1] Operation schlägt fehl.
Der Anwender, der einen Job übergibt, hat keine *USE-Berechtigung auf das Benutzerprofil, das in der Jobbeschreibung angegeben ist.	Journal-eintrag AF[1]	Journal-eintrag AF[1] Job läuft nicht.	Journal-eintrag AF[1] Job läuft nicht.
Ein Anwender versucht eine Standardanmeldung ohne Benutzer-ID und Kennwort.	Journal-eintrag AF[1]	Journal-eintrag AF[1] Anmeldung ist erfolglos.	Journal-eintrag AF[1] Anmeldung ist erfolglos.
Ein *USER State-Programm versucht auf den Systembereich der Platte zu schreiben, die als schreibgeschützt oder nicht berechtigt definiert ist.	Versuch könnte Erfolg haben.	Journal-eintrag AF[1,2] Operation schlägt fehl[2].	Journal-eintrag AF[1,2] Operation schlägt fehl[2].
Es wird versucht ein Programm, das keinen gültigen Validierungswert hat zurückzuspeichern[3].	Keine Validierung wird ausgeführt. Programm muss vor Verwendung neu umgesetzt werden.	Keine Validierung wird ausgeführt. Programm muss vor Verwendung neu umgesetzt werden.	Keine Validierung wird ausgeführt. Programm muss vor Verwendung neu umgesetzt werden.
Es wird versucht, ein Programm das einen Validierungswert hat, zurückzuspeichern.	Programm – Validation wird durchgeführt.	Programm – Validation wird durchgeführt.	Programm – Validation wird durchgeführt.
Es wird versucht, den Speicher, der dem Programm gewiesenen wurde, zu ändern.	Versuch ist erfolgreich.	Journal-eintrag AF[1,2] Operation schlägt fehl[2].	Journal-eintrag AF[1,2] Operation schlägt fehl[2].

Szenarienbeschreibung	30	40	50
Es wird versucht, den Adressraum eines Jobs zu ändern	Versuch ist erfolgreich.	Journal-eintrag AF[1,2] Operation schlägt fehl[2].	Journal-eintrag AF[1,2] Operation schlägt fehl[2].
Ein User State-Programm versucht ein System-Domain Programm aufzurufen oder die Steuerung zu übertragen.	Versuch ist erfolgreich	Journal-eintrag AF[1,2] Operation schlägt fehl[2].	Journal-eintrag AF[1,2] Operation schlägt fehl[2].
Es wird versucht, ein User Domain Objekt vom Typ *USRSPC, *USRIDX oder *USRQ in einer Bibliothek aus dem Systemwert QALWUSRDMN zu erstellen.	Operation schlägt fehl.	Operation schlägt fehl.	Operation schlägt fehl.
Ein User State-Programm sendet eine Abbruchnachricht an ein System State-Programm, das nicht direkt darüber im Programmstapel ist.	Versuch ist erfolgreich.	Versuch ist erfolgreich.	Operation schlägt fehl.
Ein Parameter wird an ein User Domain-Programm, das im System-State läuft, übergeben.	Versuch ist erfolgreich.	Parameter-prüfung wird durchgeführt.	Parameter-prüfung wird durchgeführt.
Ein von IBM gelieferter Befehl wird über CHGCMD geändert, um ein anderes Programm aufzurufen. Der Befehl wird wieder zurück zum IBM-Programm geändert. Ein Anwender versucht den Befehl aufzurufen.	Versuch ist erfolgreich.	Journal-eintrag AF[1,2,4] Operation schlägt fehl[2,4].	Journal-eintrag AF[1,2,4] Operation schlägt fehl[2,4].

Ein Berechtigungsfehler (AF) wird in das Auditjournal geschrieben,

1 wenn die Auditierung aktiv ist.

2 wenn der Prozessor den erweiterten Hardware-Speicherschutz unterstützt.

3 wenn die Programme, die vor V1R1 erstellt wurden, keinen Validierungswert haben.

4 wenn Sie einen von IBM gelieferten Befehl ändern. Dann kann er nicht länger als System Domain Programm aufgerufen werden.

Wenn Sie die Auditierungs-Funktion in einer unteren Sicherheitsstufe – wie z.B. 30 – verwenden, meldet das System Journaleinträge für die meisten Aktionen, die in der obigen Tabelle dargestellt sind. Sie erhalten Warnungen in Form von Journaleinträgen für potenzielle Integritätsverletzungen. Wie Sie in der Tabelle sehen können, verursachen Integritätsverletzungen auf Sicherheitsstufe 40 und höher Fehlermeldungen – die ausgeführte Operation schlägt fehl.

Die andere Sache, die die Sicherheitsstufe 40 abfängt, ist die Verwendung von nicht unterstützten Schnittstellen. Bei Sicherheitsstufe 40 und höher, verhindert das System Versuche, System-Programme aufzurufen, die nicht als Call-Level-Interface dokumentiert sind. Zum Beispiel funktioniert ein direkter Aufruf des Befehlsverarbeitungsprogramms für den SIGNOFF-Befehl nicht. Das System nutzt seine eigenen Tricks, um hier einzugreifen. Ohne es detailliert erklären zu wollen, wird das Domain Attribut eines Objekts und das State-Attribut eines Programms geprüft, um diesen Schutz zu erzwingen. Jedes Objekt gehört entweder zur *SYSTEM Domäne oder zur *USER Domäne. Auf *SYSTEM Domänen-Objekte kann nur von *SYSTEM State-Programmen oder Programmen, die den *System State erben, in einem unterstützten Mode (*INHERIT) zugegriffen werden. Die Programme sind entweder als *SYSTEM State, *INHERIT State oder *USER State eingestuft. Die *USER State-Programme können direkt nur auf *USER Domain-Objekte zugreifen. Sollten *USER State-Programme direkt *SYSTEM Domain-Objekte verwenden, dann endet das in einem Fehler, wie in obiger Tabelle angegeben.

Eine Domänen – oder State-Verletzung bewirkt, dass die Operation an der Sicherheitsprüfung der Stufe 40 und höher fehlschlägt. Bei allen Sicherheitsstufen wird ein Eintrag vom Typ AF in das Auditjournal geschrieben, wenn die Auditierungs-Funktion aktiv ist. Wenn aber eine Domänen – oder State-Verletzung auftritt, die Auditierungs-Funktion aktiviert ist und der Systemwert QAUDLVL *PGMFAIL enthalten ist, dann wird ein Eintrag vom Typ D oder R in das Auditjournal geschrieben, sobald versucht wird, ein nicht unterstütztes Interface zu verwenden.

Bei Sicherheitsstufe 40 und höher, muss ein Benutzer, um einen Job zu übergeben, die *USE-Berechtigung haben – sowohl für die Jobbeschreibung als auch für das Benutzerprofil, das in der Jobbeschreibung angegeben ist. Andernfalls schlägt die Übergabe fehl. Bei Sicherheitsstufe 30 läuft der Job, sobald der Benutzer, der den Job übergibt, die *USE-Berechtigung nur auf der Jobbeschreibung hat. Wenn die Audit-Funktion aktiviert ist und der Systemwert QAUDLVL *AUTFAIL enthält, wird ein AF-Eintrag mit Subtyp J in das Auditjournal geschrieben.

Jeder, der sich ohne Benutzer-ID und Kennwort bei Sicherheitsstufe 30 durch das Drücken der Enter-Taste über bestimmte Subsysteme anmeldet, erhält Zugriff auf das System. Bei Sicherheitsstufe 40 und höher stoppt das System den Versuch, sich so anzumelden. Wenn Audit aktiv ist und der Systemwert QAUDLVL *AUTFAIL enthält, wird ein AF-Eintrag, Subtyp S im Auditjournal aufgezeichnet.

Dann gibt es noch die Enhanced Hardware Storage Protection. Diese Hardware-Einrichtung ermöglicht es, Blöcke von System-Informationen auf der Festplatte als read-write, read-only oder ohne Zugriff zu definieren. Bei Sicherheitsstufe 40 und höher steuert das System, wie die *USER State-Programme auf diese geschützten Blöcke zugreifen. Diese Unterstützung ist auf Systemen mit Sicherheitsstufen kleiner als 40 nicht verfügbar. Diese Fähigkeit ist bei den meisten, aktuellen Power i Systemen verfügbar. Wenn die Auditierung aktiviert wird und der Systemwert QAUDLVL *PGMFAIL enthält, wird ein AF-Eintrag, Subtype R in das Auditjournal geschrieben, sollten solche Verletzungen auftreten.

Während der Auditierung auf Sicherheitsstufe 30 vor der Migration auf 40 haben Sie die Möglichkeit, Ressourcen-Sicherheit für alle Ihre Anwendungen zu testen. Deshalb beginnen Sie den Audit-Prozess, bevor Sie diesen Schritt machen. Auf diese Weise und durch eine gründliche Analyse der Journaleinträge sind Sie in der Lage herauszubekommen, wann Sie keine Probleme beim Umstieg auf die Sicherheitsstufe 40 haben werden.

Für den QAUDLVL-Systemwert sind – wie oben erwähnt – die beiden kritischen und einzustellenden Werte: *PGMFAIL und *AUTFAIL. *PGMFAIL protokolliert die Journaleinträge für alle Zugriffsversuche, die bei der Sicherheitsstufe die Integrität verletzen. *AUTFAIL erfasst die ungültigen (leeren) Anmeldungen sowie die fehlenden Berechtigungen auf Jobbeschreibungen und Benutzerprofile.

Nachdem Sie die Audit-Umgebung eingerichtet und aktiviert haben, ist Ihre Aufgabe jetzt die Überwachung des Auditjournals auf *AUTFAIL und *PGMFAIL Einträge. Und zwar während des Betriebs all Ihrer Anwendungen, solange Sie noch auf der Sicherheitsstufe 30 sind. Die wichtigsten Subtypen innerhalb der AF-Einträge sind:

– B
  Ein Programm hat eine restriktive Maschineninterface-Anweisung aufgerufen.
– C
  Ein Programm ist bei Objekt-Validierung fehlgeschlagen.
– D
  Programm hat auf ein Objekt über ein nicht unterstütztes Interface zugegriffen.

– J
  Jobbeschreibungs – und Benutzerprofil-Berechtigungsfehler.
– R
  Versuch auf geschützten Bereich der Festplatte zuzugreifen.
– S
  Standard-Anmeldeversuch ohne Benutzer-ID/Kennwort.

Dies sind die Codes, die auf das Vorkommen von Integritätsverletzungen bei den Anwendungen hinweisen, wenn Sie Sicherheitsstufe 40 verwenden.

Noch eines zum Schluss, bevor Sie beginnen, die Objekte zu erstellen, die benötigt werden, um Audit-Journalisierung zu implementieren: Alle Programme, die vor OS/400 Version 1 Release 3 erstellt wurden (und das dürften relativ wenige sein), haben keine Validierungs-Codes. Dies ist kein wirkliches Problem, aber es kann zu einem Problem mit Sicherheitsstufe 40 werden, wenn Sie diese Objekte zurückspeichern. Die einfachste Möglichkeit, dem zu entgegnen, ist ein CHGPGM auf jedes dieser Programme anzuwenden und den FRCCRT-Parameter zur Erstellung der Validierungs-Codes für diese Programme einzusetzen.

Mit der Sicherheitsstufe 40, die auf Ihren Einstellungen der drei System-Werte basiert – Objekt beim Zurückspeichern prüfen (QVFYOBJRST), Umsetzung beim Zurückspeichern erzwingen (QFRCCVNRST) und Objektrückspeicherung zulassen (QALWOBJRST) – wird das System die Aktion, die Sie anfordern, verwenden. Bei der Wiederherstellung arbeiten diese Werte als eine Reihe von Filtern, um festzustellen, ob ein Programm ohne Änderung wiederhergestellt, ob es neu erstellt (konvertiert), wie es wiederhergestellt oder ob es überhaupt nicht wiederhergestellt wird.

Nachdem Sie alle diese Fragen gelöst haben, stellen Sie auf Sicherheitsstufe 40 um.

Erstellen des Auditjournals und Starten der Auditierung

Um die Auditierung einschalten zu können, benötigen Sie die Sonderberechtigung *AUDIT in Ihrem Benutzerprofil. Um diese Sonderberechtigung zu bekommen, können Sie Ihr Benutzerprofil am Besten unter Verwendung des QSECOFR-Profils ändern.

Es könnte sein, dass die Auditierung auf Ihrem System bereits von einem anderen Anwender aktiviert wurde. Schauen Sie deshalb zum einen erst mal nach, ob ein Audit bereits aktiv ist – und zum anderen, ob Auditjournal und Journalempfänger existieren. Dann erstellen Sie – falls noch nicht vorhanden – die Journalreceiver in einer Bibliothek Ihrer Wahl, indem Sie den Befehl „Journalreceiver erstellen“ (CRTJRNRCV) ausführen. In diesem Beispiel wird eine Bibliothek namens AUDITLIB für Journalempfänger verwendet.

1. Prüfen Sie, welche Journale und Journalempfänger auf Ihrem System vorhanden sind:

```
WRKOBJ OBJ (* ALL / * ALL) OBJTYPE (* JRN)
WRKOBJ OBJ (* ALL / * ALL) OBJTYPE (* JRNRCV)
```

   Wenn Sie in Ihrem Unternehmen Datenbank-Journalisierung verwenden, dann werden Sie überrascht sein, wie viele Journale und Journalempfänger es auf Ihrem System gibt.

   Das Auditjournal gibt es nur einmal im System mit dem Namen QAUDJRN. Darüber hinaus muss das QAUDJRN-Journal in der Bibliothek QSYS stehen. Da ein Journal eine zentrale Filter – und Kontrollfunktion für die Aufzeichnung von Journaleinträgen ist, kann und soll QAUDJRN in der Bibliothek QSYS belassen werden.

2. Wenn Ihnen die Liste der Journale und Journalempfänger zu umfangreich zum Durchsuchen ist, dann können den Befehl WRKJRN verwenden, um den Status des Auditjournals zu sehen. Beispielsweise können Sie damit auch sehen, wie viele Empfänger erstellt wurden.

```
WRKJRN QAUDJRN
```

   Wenn in Ihrer Firma bereits jemand das Auditjournal implementiert hat, dann können Sie in den meisten Fällen entweder:

   – die Audit-Journalisierung durch das Setzen des Systemwerts QAUDCTL auf *NONE ausschalten,

   – die gefundenen Journalreceiver löschen,

   – mit Punkt 3. fortfahren.

oder

- die Systemwerte QAUDLVL und QAUDLVL2 prüfen und sicherstellen, dass die Werte *PGMFAIL und *AUTFAIL angegeben wurden.
- Sollten diese Optionen nicht enthalten sein, fügen Sie sie hinzu.
- Sollten diese Werte existieren und die Einstellungen in Ordnung sein, dann fahren Sie mit Schritt 6 fort.

3. Erstellen Sie eine Bibliothek für die Audit-Journalisierung.

```
CRTLIB LIB (AUDITLIB) TEXT ('Bibliothek für
Auditjournal Receiver) AUT (* EXCLUDE)
```

4. Erstellen Sie die Journalempfänger für die Auditierung. Dieser wird später an das Journal angehängt.

```
CRTJRNRCV JRNRCV (AUDITLIB/AUDRCV0001) TEXT ('Journal Receiver
für Auditjournalisierung") AUT (* EXCLUDE)
```

Da hier eine neue Bibliothek erstellt wird, stellen Sie sicher, dass die Bibliothek AUDITLIB und ihre Journalempfänger in die Backup-Routinen mit aufgenommen werden, so dass die Audit-Daten regelmäßig gesichert werden. Sollten Sie die Backup-Routinen nicht ändern können, stellen Sie die Audit-Journalempfänger in eine Bibliothek, die standardmäßig mit gesichert wird – z.B. QGPL (obwohl das nicht empfehlenswert ist). Stellen Sie die Journalempfänger nicht in die Bibliothek QSYS.

Wählen Sie einen Journalempfängernamen, mit dem eine Namenskonvention für zukünftige Journalempfänger – wie AUDRCV0001 – erstellt werden kann. Sie können die *GEN Option verwenden, wenn Sie die Journalempfänger ändern und die Namenskonvention fortsetzen. Diese Art der Namensgebung ermöglicht es Ihnen, das System verwalten zu lassen, wenn Ihre Journalempfänger voll sind.

Geben Sie einen Schwellwert für die Größe des Empfängers im Treshold-Parameter ein. Die Größe, die Sie wählen, sollte auf die Anzahl der Transaktionen auf Ihrem System und die Anzahl der Aktionen, die Sie auswählen, angepasst sein. Wenn Sie die automatische Journalreceiver-Verwaltung – wie empfohlen – verwenden, muss die Journalreceiver-Schwelle mindestens 100.000 KB sein. Die Standardgröße, wie wir sie oben gewählt haben (durch Weglassen eines speziellen Wertes), ist als Schwellwert standardmäßig 1.500.000 KB. Geben Sie im Parameter AUT *EXCLUDE für die Einschränkung des Zugriffs auf die Informationen im Journal ein.

5. Erstellen Sie das QSYS-/ QAUDJRN-Journal mit dem Befehl „Create Journal“ (CRTJRN).

```
CRTJRN JRN (QSYS / QAUDJRN) JRNRCV (JRNLIB/AUDRCV0001)
MNGRCV (* SYSTEM) DLTRCV (* NO) AUT (* EXCLUDE)
TEXT ('Audit – Journal ")
```

Obwohl die Audit-Journalisierung eine Systemfunktion ist, wird das QAUDJRN-Journal nicht von IBM zur Verfügung gestellt. Sie müssen es selbst erstellen. Die Regeln für die Erstellung des Auditjournals lauten wie folgt:

– Der Name QSYS / QAUDJRN muss verwendet werden.
– Geben Sie den Namen des Journalempfängers, den Sie vorher erstellt haben, an.
– Geben Sie *EXCLUDE im AUT-Parameter an, um den Zugang auf die Informationen im Journal einzuschränken. Sie müssen die Berechtigung haben, um Objekte in die Bibliothek QSYS hinzuzufügen und um das Journal zu erstellen.
– Verwenden Sie den Parameter MNGRCV, damit das System die Journalempfänger ändern und neue hinzufügen kann – aber nur dann, wenn die angehängten Empfänger die Schwellwertgröße überschreiten. Wenn Sie diese Option wählen, müssen Sie den CHGJRN-Befehl nicht verwenden, um die Empfänger zu zu erstellen und neue manuell anzuhängen.
– Lassen Sie das System keine abgehängten Receiver löschen. Geben Sie DLTRCV (*NO) an, was auch die Standardeinstellung ist. Die QAUDJRN-Empfänger sind die Security Audit-Protokolle. Stellen Sie sicher, dass die entsprechend gesichert sind, bevor Sie sie löschen.

Einstellen der zusätzlichen Optionen für Auditjournal.

6. Stellen Sie die Audit-Ebene im (QAUDLVL) Systemwert ein. Zunächst werden Sie den Systemwert QAUDLVL2 nicht brauchen. Der ist nur erforderlich wenn der Systemwert QAUDLVL keinen Platz für weitere Einträge hat. Zum Verwalten der Prüfungen in den Systemwerten QAUDLVL und QAUDLVL2 verwenden Sie den Befehl WRKSYSVAL.

   Beginnen Sie mit folgender Eingabe:

```
WRKSYSVAL QAUD*
```

Drücken Sie die Eingabetaste. Sie sehen die verfügbaren Systemwerte im Audit-Bereich:

```
                          Mit Systemwerten arbeiten
                                                                System:   PWR
Listenanfang bei . . . .   __________   Anfangszeichen des Systemwerts
Teilauflistung nach Art    _______      F4=Liste

Auswahl eingeben und Eingabetaste drücken.
  2=Ändern   5=Anzeigen

Auswahl  Systemwert  Art      Beschreibung
   _      QAUDCTL     *SEC     Protokollierungssteuerung
   _      QAUDENDACN  *SEC     Aktion bei Protokollierungsende
   _      QAUDFRCLVL  *SEC     Protokollierungsdaten erzwingen
   _      QAUDLVL     *SEC     Sicherheitsprotokollebene
   _      QAUDLVL2    *SEC     Erweiterung der Sicherheitsprotokollierung
```

Audit-Systemwerte

Merken Sie sich die Existenz des Systemwertes QAUDLVL2. Dieser kann später eine Rolle spielen, nachdem Sie bereits zur Sicherheitsstufe 40 migriert sind und viele andere Security Auditierungs-Funktionen aktiviert haben. Nun rufen Sie mit der Auswahl 2 den Systemwert QAUDLVL zum Ändern auf. Stellen Sie sicher, dass das Auditjournal auf Ihrem System noch nicht aktiv ist (das sollte in Schritt 1 und 2 passiert sein). Ergänzen Sie die Auditierungsparameter:

```
*AUTFAIL
*PGMFAIL
```

Sobald Sie durch die Analyse des Auditjournals festgestellt haben, dass Ihr System auf Sicherheitsstufe 40 umgestellt werden kann, können Sie im QAUDLVL-Systemwert weitere nützliche Optionen aktivieren. Während Sie mit dem Systemwert arbeiten, verwenden Sie die F1-Hilfetaste und lassen sich die Optionen und deren Erklärungen anzeigen. Wenn sie mal Experte auf dem Gebiet sind und mehr Funktionen aktivieren können, als im QAUDLVL-Systemwert Platz haben, dann setzen Sie die Liste im Systemwert QAUDLVL2 fort.

7. Sobald Sie die QAUDLVL-Werte festgelegt haben, ist es an der Zeit die Auditierung einzuschalten. Dies erfolgt über den Systemwert QAUDCTL. Sie starten das Auditjournal indem Sie den QAUDCTL Systemwert auf einen anderen Wert als *NONE einstellen – nämlich:

```
* AUDLVL und
* NOQTEMP
```

Stellen Sie sicher, dass die zwei Optionen eingegeben wurden. Das ist alles, was nötig ist, um den Test für die Sicherheit der Sicherheitsstufe 40 oder 50 durchzuführen.

Bitte beachten Sie, dass das QSYS/QAUDJRN-Journal existieren muss, bevor Sie den Systemwert QAUTCTL auf einen anderen Wert als *NONE einstellen. Wenn die Auditierung startet, versucht das System einen Datensatz in das Auditjournal zu schreiben. Wenn das nicht erfolgreich ist, erhalten Sie eine Fehlermeldung und die Auditierung startet nicht. Ebenso müssen Journal und Journalreceiver passend eingestellt sein.

Sobald Sie die Journalisierung eingeschaltet haben, lassen Sie das System eine Weile laufen – zumindest so lange, bis die üblichen Geschäftsvorfälle abgearbeitet sind.

Andere Befehle und Optionen bei der Auditierung
Wenn Sie bereit sind, ein umfangreiches Audit mitzumachen, können Sie die folgenden Befehle verwenden, um weitere hilfreiche Aufzeichnungen für Aktivitäten zu starten:

```
CHGUSRAUD
```

Dieser Befehl startet die Auditierung für individuelle Anwender.

```
CHGOBJAUD
CHGDLOAUD
```

Diese beiden Befehle schalten die Objektauditierung für bestimmte Objekte ein. Der CHGUSRAUD-Befehl kann wiederum verwendet werden, um bestimmte Objekte bei bestimmten Anwendern zu auditieren.

Stellen Sie sicher, dass der Systemwert QAUDENDACN auf *NOTIFY anstelle auf *PWRDWNSYS eingestellt ist. *PWRDWNSYS schaltet Ihr System ab, wenn die Auditierung keine Journaldaten wegschreiben kann. *NOTIFY sendet in diesem Fall eine Nachricht an den QSYSOPR (Systemoperator). *NOTIFY ist die Standardoption für diesen Systemwert.

Mit dem Systemwert QAUDFRCLVL können Sie steuern, wie oft Audit-Journaleinträge aus dem Speicher auf die Platte weggeschrieben werden. Mit der Standardeinstellung *SYS wird das vom System aus gesteuert. Die Journaleinträge werden nur dann auf die Platte geschrieben, wenn das System feststellt, dass dies abhängig von der internen Systemverarbeitung erforderlich ist. Diese Angabe liefert zwar die besten Protokollierungsergebnisse, kann aber auch zum höchsten Datenverlust bei einem abnormalen Ende des Systems führen.

Sie können auch einen Wert von 1 bis 100 angeben. Der Wert gibt die Anzahl der ins Sicherheitsprotokoll-Journal geschriebenen Journaleinträge an, bevor die Protokollierungsdaten auf die Platte geschrieben werden. Je kleiner diese Anzahl angegeben wird, umso höher sind die Auswirkungen auf die Systemleistung. Aber die Wahrscheinlichkeit der Erfassung des letzten Fehler auf der Festplatte im Falle eines Crashs steigt.

Prüfung des Journals
Sobald Sie die Journalisierung eingeschaltet haben, können Sie die Inhalte anschauen.

Die beste Möglichkeit, die Audit-Daten zu analysieren, besteht darin ein für diese Zwecke spezialisiertes Programm zu erwerben. Diese Programme lesen die Audit-Journale aus und übertragen sie – je nach Audittyp – in lesbare Protokolleinträge. Das ist wesentlich einfacher und effektiver, als für jeden Audittyp ein Programm zu schreiben, dass die unterschiedlichen Audittypen aus den Journalen ausliest und dass für jeden Audityp mit Rücksicht auf die interne Struktur des jeweiligen Typs die Daten aufbereitet.

Die Verwendung des Befehls DSPAUDJRNE zur Anzeige der Auditjournal-Einträge ist eine rudimentäre Möglichkeit, die Daten anzuzeigen. Die Berichte aus diesem Befehl geben Auskunft über die Eintragstypen und die Benutzerprofile, welche die Ereignisse ausgelöst haben. Sie können die Berichte auf bestimmte Zeiträume einschränken und einzelne Journalempfänger durchsuchen. Eine Ausgabe auf den Bildschirm ist ebenso möglich wie die Ausgabe in eine Spooldatei. Für den Befehl DSPAUDJRNE müssen Sie über die Sonderberechtigungen *ALLOBJ und *AUDIT verfügen.

Um einen Bericht über alle Berechtigungsfehler für alle Benutzerprofile zu erzeugen, verwenden Sie den Befehl:

```
DSPAUDJRNE ENTTYP (AF) USRPRF (* ALL) OUTPUT (*)
```

Natürlich können Sie den Ausgabe-Parameter auf *PRINT ändern, um einen Bericht zu erzeugen. Dabei können im besten Fall einige Zeilen, im schlechtesten Fall viele Seiten ausgegeben werden:

	Verletzungs-art		Benutzer-profil	Objekt-name	Bibliotheks-name	Objekt-art
000001	AF	A	QUSER	*N	*N	*DIR
000002	AF	A	QUSER	*N	*N	*DIR
000003	AF	A	QUSER	*N	*N	*DIR
000004	AF	A	QUSER	*N	*N	*DIR
000005	AF	A	QUSER	*N	*N	*DIR
000006	AF	A	QUSER	*N	*N	*DIR
000007	AF	A	QUSER	*N	*N	*DIR
000008	AF	A	QUSER	*N	*N	*DIR
000009	AF	A	QUSER	*N	*N	*DIR
000010	AF	A	QUSER	*N	*N	*DIR

DSPAUDJRNE-Ausgabe am Bildschirm

Beachten Sie die AF-Einträge und das A neben dem Eintrag. Der Typ „A“ ist harmlos. Die wichtigeren AF-Einträge sind die mit B, C, D, J, R und S. Diese wurden weiter vorne im Text beschrieben. Wenn Sie den Audit-Wert *NOQTEMP verwenden, werden die Vorgänge in QTEMP nicht auditiert. Diese QTEMP-Bibliotheken sind die temporären Bibliotheken je User und brauchen hier auch nicht berücksichtigt werden.

Denken Sie daran, Ihr Ziel ist es, keine der o. g. fehlerhaften Einträge mehr zu finden. Sobald Sie das erreicht haben und die zugrunde liegenden Probleme gelöst wurden, können Sie über den Befehl

```
WRKSYSVAL QSECURITY
```

den Wert der Sicherheitsstufe 40 einstellen. Sollten trotzdem nach der Umstellung Probleme auftreten, können Sie jederzeit wieder zurück zur Sicherheitsstufe 30 wechseln.

9.10
Seite 8

Sicherheitsrelevante Systemwerte

Der Power i-Befehl WRKSYSVAL eröffnet den Zugriff auf viele übergeordnet gültige Werte, die sogenannten Systemwerte. Hier können an zentraler Stelle Systemeinstellungen vorgenommen werden, die eine Vielzahl von Einstellungen betreffen, u. a. auch solche, die direkt oder indirekt die Sicherheit des Systems bestimmen und beeinflussen. In diesem Kapitel gehe ich auf diese Systemwerte ein und versuche, Ihnen einen ausführlichen Überblick über sie zu geben.

WRKSYSVAL

Der Einstieg erfolgt – wie oben erwähnt – mit dem Befehl WRKSYSVAL. Mit diesem Befehl können Sie sich die Systemwerte des Power i-Systems anzeigen lassen, aber auch Änderungen vornehmen oder eine Liste der Systemwerte ausdrucken.

```
                    Mit Systemwert arbeiten (WRKSYSVAL)

Auswahl eingeben und Eingabetaste drücken.

Systemwert . . . . . . . . . . .   *ALL          Name, generisch*...
Ausgabe  . . . . . . . . . . . .   *             *, *PRINT
```

WRKSYSVAL + F4

Sie können die Systemwertanzeige einschränken, indem Sie einen exakten Systemwertnamen eingeben oder einen generischen Namen mit * am Ende verwenden. Mit der Ausgabeoption *PRINT können Sie eine Liste der ausgewählten Systemwerte ausgeben:

```
                Aktueller                    Gelieferter
Name            Wert                         Wert                         Beschreibung

QABNORMSW       0                            0                            Vorherige Systembeendigung
QACGLVL         *NONE                        *NONE                        Abrechnungsebene
QACTJOB      >  601                          200                          Anfängliche Anzahl aktiver Jobs
QADLACTJ     >  550                          30                           Zusätzliche Anzahl aktiver Jobs
QADLSPLA        2048                         2048                         Zusätzlicher Speicher für Spool-Steuerblock
QADLTOTJ     >  600                          30                           Zusätzliche Anzahl aller Jobs
QALWJOBITP      0                            0                            Unterbrechung von Jobs zulassen
QALWOBJRST      *ALL                         *ALL                         Objektrückspeicherung zulassen
QALWUSRDMN      *ALL                         *ALL                         Benutzerdomänenobjekte in Bibliotheken zulassen
QASTLVL      >  *INTERMED                    *BASIC                       Unterstützungsstufe für Benutzer
QATNPGM         *ASSIST                      *ASSIST                      Abrufprogramm
QAUDCTL      >  *AUDLVL                      *NONE                        Protokollierungssteuerung
                *OBJAUD                      ' '
```

Ausdruck der Systemwerte

9.11
Seite 2

Mit der Auswahl * werden die Systemwerte zur Bearbeitung am Bildschirm angezeigt:

```
Listenanfang bei . . . .                 Anfangszeichen des Systemwerts
Teilauflistung nach Art     *ALL         F4=Liste

Auswahl eingeben und Eingabetaste drücken.
  2=Ändern   5=Anzeigen

Auswahl   Systemwert   Art       Beschreibung
   _      QABNORMSW    *SYSCTL   Vorherige Systembeendigung
   _      QACGLVL      *MSG      Abrechnungsebene
   _      QACTJOB      *ALC      Anfängliche Anzahl aktiver Jobs
   _      QADLACTJ     *ALC      Zusätzliche Anzahl aktiver Jobs
   _      QADLSPLA     *ALC      Zusätzlicher Speicher für Spool-Steuerblock
   _      QADLTOTJ     *ALC      Zusätzliche Anzahl aller Jobs
   _      QALWJOBITP   *SYSCTL   Unterbrechung von Jobs zulassen
   _      QALWOBJRST   *SEC      Objektrückspeicherung zulassen
   _      QALWUSRDMN   *SEC      Benutzerdomänenobjekte in Bibliotheken zulassen
                                                                  Weitere ...
Befehl
===>
F3=Verlassen    F4=Bedienerführung    F5=Neuanzeige    F9=Auffinden
F11=Nur Namen anzeigen    F12=Abbrechen
```

Mit Systemwerten arbeiten

Hier können Sie auf die zu ändernden Systemwerte positionieren und sie mit der Auswahl 5 anzeigen bzw. über die Auswahl 2 ändern.

Der Befehl DSPSYSVAL dient der Anzeige von Systemwerten ohne Änderungsmöglichkeit. Hier müssen Sie allerdings den Systemwert kennen und qualifiziert eingeben.

Achtung:

Einige sicherheitsrelevante Systemwerte können Sie nicht ändern, wenn sie zuvor im Betriebssystem vor Veränderungen geschützt worden sind. In den Service-Tools (SST) – über STRSST, Auswahl 7 (Work with System Security), gibt es u. a. den Parameter „Allow system value security changes“. Ist dieser Parameter auf „2“ (= No) gesetzt, werden die folgenden Systemwerte vor Veränderung geschützt:

QALWJOBITP	QCRTOBJAUD	QPWDLMTCHR
QALWOBJRST	QDEVRCYACN	QPWDLMTREP
QALWUSRDMN	QDSCJOBITV	QPWDLVL
QAUDCTL	QDSPSGNINF	QPWDMAXLEN
QAUDENACN	QFRCCVNRST	QPWDMINLEN
QAUDFRCLVL	QINACTMSGQ	QPWDPOSDIF
QAUDLVL	QLMTDEVSSN	QPWDRQDDGT
QAUDLVL2	QLMTSECOFR	QPWDRQDDIF
QAUTOCFG	QMAXSGNACN	QPWDVLDPGM

QAUTORMT	QMAXSIGN	QRETSVRSEC
QAUTOVRT	QPWDEXPITV	QRMTSIGN
QCRTAUT	QPWDLMTAJC	QRMTSRVATR
QSCANFS	QSECURITY	QUSEADPAUT
QSCANFSCTL	QSHRMEMCTL	QVFYOBJRST

Nachfolgend finden Sie eine Liste der sicherheitsrelevanten Systemwerte mit deren Bedeutung, mit Einstellmöglichkeiten und Erläuterungen. Die Erläuterungen zu den einzelnen Systemwerten und zu deren Parametern stammen weitestgehend aus der IBM Dokumentation und zeigen Ihnen, wie kongruentes Arbeiten mit dem Betriebssystem möglich ist.

9.11
Seite 4

QALWJOBITP

Unterbrechung von Jobs zulassen
Dieser Systemwert erlaubt es, Jobs zu unterbrechen und zeigt an, wie das System reagieren wird, wenn ein Benutzer die Unterbrechung eines Jobs anfordert, damit ein benutzerdefiniertes Exitprogramm in diesem Job ausgeführt werden kann.

Der Unterbrechungsstatus eines aktiven Jobs kann jederzeit geändert werden; die Änderung wird jedoch erst wirksam, wenn der Wert QALWJOBITP die Unterbrechung des Jobs zulässt.

Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist 0.

Gültige Werte sind:
0 Das System erlaubt keine Jobunterbrechung zur Ausführung benutzerdefinierter Exitprogramme. Alle neuen Jobs, die aktiv werden, dürfen standardmäßig nicht unterbrochen werden.
1 Das System erlaubt eine Jobunterbrechung zur Ausführung benutzerdefinierter Exitprogramme. Alle neuen Jobs, die aktiv werden, dürfen standardmäßig nicht unterbrochen werden.

9.11.1
Seite 2

QALWOBJRST

Objektrückspeicherung zulassen
Mit diesem Systemwert können Sie die Rückspeicherung von Objekten erlauben oder verhindern. Er gibt an, ob Objekte mit sicherheitsrelevanten Attributen zurückgespeichert werden können. Dieser Wert besteht aus einer Liste mit Werten, die das zurückzuspeichernde Objekt steuern. Der Wert kann als *ALL, *NONE oder als Liste einzelner Werte angegeben werden.

Änderungen an diesem Systemwert werden mit der Ausführung des nächsten Rückspeicherungsvorgangs wirksam. Der Vorgabewert ist *ALL.

Wenn versucht wird, ein Objekt auf dem System zurückzuspeichern, arbeiten drei Systemwerte gemeinsam als Filter. Sie bestimmen, ob das Objekt zurückgespeichert werden darf oder ob es während der Rückspeicherung konvertiert wird. Der erste Filter ist der Systemwert **QVFYOBJRST** (Objekt beim Zurückspeichern prüfen). Dieser Systemwert steuert das Zurückspeichern einiger Objekte, die digital signiert werden können. Der zweite Filter ist der Systemwert **QFRCCVNRST** (Umsetzung beim Zurückspeichern erzwingen). Mit Hilfe dieses Systemwerts kann angegeben werden, ob während des Zurückspeicherns Programme, Serviceprogramme, SQL-Pakete und Modulobjekte konvertiert werden sollen oder nicht. Er kann auch verhindern, dass Objekte zurückgespeichert werden. Vom dritten Filter werden nur Objekte verarbeitet, die die ersten beiden Filter passiert haben. Dieser dritte Filter ist der Systemwert **QALWOBJRST** (Zurückspeichern des Objekts erlauben). Er gibt an, ob Objekte mit sicherheitsrelevanten Attributen zurückgespeichert werden können.

Gültige Werte sind:

*ALL	Alle Objekte können mit oder ohne sicherheitsrelevante Attribute zurückgespeichert werden.
*NONE	Objekte mit sicherheitsrelevanten Attributen können nicht zurückgespeichert werden.
*ALWSYSSTT	Alle Programme, Serviceprogramme und Module mit dem Attribut für Systemstatus oder übernommenen Status können zurückgespeichert werden. Wenn das Objekt aufgrund der Einstellung des Systemwerts QFRCCVNRST konvertiert wird, wird ihm das Attribut für Benutzerstatus zugeordnet.
*ALWPGMADP	Programme und Serviceprogramme mit dem Attribut für Berechtigungsübernahme können zurückgespeichert werden.

9.11.2
Seite 2

*ALWPTF	Systemstatusobjekte oder Objekte mit übernommenem Status, die Berechtigungen übernehmen, Objekte, für die das Attribut S_ISUID (Benutzer-ID setzen) aktiviert ist, und Objekte, für die das Attribut S_ISGID (Gruppen-ID setzen) aktiviert ist, werden bei einer PTF-Installation im System zurückgeschrieben.
*ALWSETUID	Dateien, bei denen das Attribut S_ISUID (Benutzer-ID setzen) aktiviert ist, werden zurückgespeichert.
*ALWSETGID	Dateien, bei denen das Attribut S_ISGID (Gruppen-ID setzen) aktiviert ist, werden zurückgespeichert.
*ALWVLDERR	Objekte mit Gültigkeitsfehlern oder mit denen möglicherweise manipuliert wurde, werden zurückgespeichert. Wenn das Objekt aufgrund der Einstellung des Systemwerts QFRCCVNRST konvertiert wird, werden alle möglicherweise vorhandenen Gültigkeitsfehler korrigiert.

QALWUSRDMN

Benutzerdomänenobjekte in Bibliotheken zulassen
Dieser Systemwert gibt an, welche Bibliotheken im System die Benutzerdomänenobjekte *USRSPC (Benutzeradressbereich), *USRIDX (Benutzerindex) und *USRQ (Benutzerwarteschlange) enthalten können.

Eine Änderung dieses Systemwerts wird sofort wirksam. Bei Angabe von *ALL können alle Bibliotheken im System Benutzerdomänenversionen von USRxxx-Objekten enthalten. Wird eine Liste mit Bibliotheksnamen angegeben, können bei Anwendungen, die derzeit mit Benutzerdomänenobjekten (*USRxxx) arbeiten, Fehler auftreten, wenn die Anwendungen Benutzerdomänenobjekte in Bibliotheken verwenden, die nicht vom Systemwert zugelassen werden. Bei Angabe einer Bibliotheksliste oder von *DIR muss QTEMP in der Liste enthalten sein.

Bei Angabe eines Bibliotheksnamens können alle Bibliotheken mit diesem Namen, die in separaten, unabhängigen Zusatzspeicherpools vorhanden sind, Benutzerdomänenobjekte enthalten. Der Vorgabewert ist *ALL.

Um potenzielle Sicherheitsrisiken zu minimieren, wird folgendes Vorgehen empfohlen: Bevor eine Bibliothek zu QALWUSRDMN hinzugefügt wird, sollte sie im Basis-ASP oder in allen unabhängigen ASPs erstellt werden und die allgemeine Berechtigung *EXCLUDE erhalten.

Gültige Werte sind:

*ALL	Alle Bibliotheken im System können Benutzerdomänenversionen von *USRxxx-Objekten enthalten. Bei Angabe von *ALL wird auch *DIR einbezogen.
*DIR	Die Verzeichnisse können Benutzerdomänenobjekte enthalten. Bei Angabe von *DIR muss QTEMP in der Liste enthalten sein.
Bibliotheken	Bestimmte Bibliotheken angeben, die Benutzerdomänenversionen von *USRxxx-Objekten enthalten können. Es können bis zu 50 Bibliotheken aufgelistet werden; die Bibliothek QTEMP muss in der Liste enthalten sein.

9.11.3
Seite 2

QAUDCTL

Protokollierungssteuerung
Dieser Systemwert enthält die Ein-/Ausschalter für die Objekt- und Benutzeraktionsprotokollierung. Er aktiviert die mit den Befehlen CHGOBJAUD (Objektprotokollierung ändern) und CHGUSRAUD (Benutzerprotokollierung ändern) sowie die mit den Systemwerten QAUDLVL und QAUDLVL2 ausgewählte Protokollierung.

- Zum Ändern dieses Systemwerts ist die Sonderberechtigung *AUDIT erforderlich.
- Der Benutzer muss entweder über die Sonderberechtigung *ALLOBJ oder über die Sonderberechtigung *AUDIT verfügen, um diesen Systemwert anzuzeigen.

Eine Änderung dieses Systemwerts wird für alle Jobs im System sofort wirksam.

Der Vorgabewert ist *NONE.

Um die Protokollierung zu aktivieren, muss entweder *OBJAUD oder *AUDLVL angegeben werden. Soll die Protokollierung inaktiviert werden, *NONE angeben.

Einen oder mehrere der folgenden Werte angeben. Bei Angabe von *NONE ist kein weiterer Wert mehr zulässig.

Gültige Werte sind:

*NOTAVL	Der Benutzer, der den Befehl ausführt, ist nicht berechtigt, den aktuellen Protokollierungswert anzuzeigen. Der Systemwert kann nicht in *NOTAVL (nicht verfügbar) geändert werden.
*NONE	Auf dem System findet keine Sicherheitsprotokollierung statt. Dies ist der Standardwert.
*OBJAUD	Es werden Aktionen für Objekte protokolliert, die einen anderen Objektprotokollierungswert als *NONE haben. Der Objektprotokollierungswert wird im Befehl CHGAUD (Protokollierung ändern) oder im Befehl CHGOBJAUD (Objektprotokollierung ändern) festgelegt.

9.11.4
Seite 2

*AUDLVL	Die in den Systemwerten QAUDLVL und QAUDLVL2 angegebenen Aktionen werden im Sicherheitsprotokolljournal protokolliert. Die von den Aktionsprotokollierungswerten eines Benutzerprofils angegebenen Aktionen werden ebenfalls protokolliert. Die Aktionsprotokollierungswerte eines Benutzerprofils werden im Parameter AUDLVL des Befehls CHGUSRAUD (Benutzerprotokollierung ändern) angegeben.
*NOQTEMP	Die meisten Objekte werden in QTEMP nicht protokolliert. *NOQTEMP muss entweder mit *OBJAUD oder *AUDLVL angegeben werden. *NOQTEMP darf nicht allein angegeben werden.

- Das Journal QAUDJRN muss in der Bibliothek QSYS vorhanden sein, damit dieser Systemwert in einen anderen Wert als *NONE geändert werden kann.
- Das Journal QAUDJRN kann erst dann gelöscht oder aus der Bibliothek QSYS in eine andere Bibliothek verschoben werden, wenn dieser Systemwert in *NONE geändert worden ist.

QAUDENDACN

Aktion bei Protokollierungsende
Dieser Systemwert gibt die Aktion an, die vom System ausgeführt wird, wenn keine Protokollsätze an das Protokolljournal gesendet werden können, da beim Senden des Journaleintrags Fehler aufgetreten sind.

Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist *NOTIFY.

- Zum Ändern dieses Systemwerts ist die Sonderberechtigung *AUDIT erforderlich.
- Der Benutzer muss entweder über die Sonderberechtigung *ALLOBJ oder über die Sonderberechtigung *AUDIT verfügen, um diesen Systemwert anzuzeigen.

Gültige Werte sind:

*NOTAVL	Der Benutzer, der den Befehl ausführt, ist nicht berechtigt, den aktuellen Protokollierungswert anzuzeigen. Der Systemwert kann nicht in *NOTAVL (nicht verfügbar) geändert werden.
*NOTIFY	Nachdem ein Fehlerhinweis auf das fehlgeschlagene Senden des Journaleintrags an das Sicherheitsprotokolljournal der Nachrichtenwarteschlangen QSYSOPR und QSYSMSG gesendet wurde, wird die Aktion, die den Protokollierungsversuch ausgelöst hat, fortgesetzt. Bei Angabe des Werts *NOTIFY wird der Systemwert QAUDCTL auf *NONE gesetzt, um die Protokollierung auszuschalten. Wenn der Systemprotokollierungscode den Systemwert QAUDCTL ausschaltet, wird stündlich ein entsprechender Hinweis an die Nachrichtenwarteschlangen QSYSOPR und QSYSMSG (falls vorhanden) gesendet. Das stündliche Hinweisen stoppt, sobald die Protokollierung wieder eingeschaltet wird.

*PWRDWNSYS Das System wird mit dem Systemreferenzcode B900 3D10 beendet, wenn ein Versuch, die Protokollierungsdaten an das Sicherheitsprotokolljournal zu senden, fehlschlägt. Wenn danach ein IPL durchgeführt wird, hat das System den Status des eingeschränkten Betriebs, weshalb für dieses IPL eine zugeordnete Konsole erforderlich ist. Bei Angabe des Werts *PWRDWNSYS wird der Systemwert QAUDCTL auf *NONE gesetzt, um die Protokollierung auszuschalten. Bei dem IPL, der auf das Abschalten des Systems folgt, muss sich ein Benutzer mit den Sonderberechtigungen *AUDIT und *ALLOBJ am System anmelden.

Der Systemwert QAUDENDACN gilt nur für Protokolleinträge, die vom Betriebssystem gesendet werden.

QAUDFRCLVL

Protokollierungsdaten erzwingen

Protokolljournal in Zusatzspeicher stellen. Dieser Systemwert gibt die Anzahl der Protokolljournaleinträge an, die in das Sicherheitsprotokolljournal geschrieben werden können, bevor die Journaleintragsdaten in den Zusatzspeicher gestellt werden. Der Wert gibt auch die Anzahl der Protokollierungsdaten an, die verloren gehen könnten, falls das System abnormal beendet wird. Wenn Protokolljournaleinträge häufig in den Zusatzspeicher gestellt werden, kann sich dies negativ auf die Systemleistung auswirken. Mit diesem Systemwert kann das Journal QUADJRN geändert werden, um anzugeben, wie oft Protokollierungsdaten in den Zusatzspeicher gestellt werden sollen. Eine Änderung dieses Systemwerts wird sofort wirksam.

Der Vorgabewert ist *SYS.

- Zum Ändern dieses Systemwerts ist die Sonderberechtigung *AUDIT erforderlich.
- Der Benutzer muss entweder über die Sonderberechtigung *ALLOBJ oder über die Sonderberechtigung *AUDIT verfügen, um diesen Systemwert anzuzeigen.

Gültige Werte sind:

*NOTAVL	Der Benutzer, der den Befehl ausführt, ist nicht berechtigt, den aktuellen Protokollierungswert anzuzeigen. Der Systemwert kann nicht in *NOTAVL (nicht verfügbar) geändert werden.
*SYS	Die Journaleinträge werden nur dann in den Zusatzspeicher geschrieben, wenn das System feststellt, dass dies abhängig von der internen Systemverarbeitung erforderlich ist. Diese Angabe liefert zwar die besten Protokollierungsergebnisse, kann aber auch zum höchsten Datenverlust bei einem abnormalen Ende des Systems führen.
1–100	Die Anzahl der ins Sicherheitsprotokolljournal geschriebenen Journaleinträge, bevor die Protokollierungsdaten in den Zusatzspeicher geschrieben werden. Je kleiner diese Anzahl angegeben wird, umso höher sind die Auswirkungen auf die Systemleistung.

Wird ein Journaleintrag, bei dem es sich nicht um eine Objektprotokollierung handelt, ins Sicherheitsprotokolljournal geschrieben (beispielsweise mit dem Befehl SNDJRNE – Journaleintrag senden) und ist diese Auswahl angegeben, werden neben diesem Journaleintrag auch alle anderen Protokollierungsdaten in den Zusatzspeicher geschrieben.

9.11.6
Seite 2

QAUDLVL

Sicherheitsprotokollebene
Steuert die Ebene der Sicherheitsprotokollierung im System.

Wenn der Systemwert QAUDLVL den Wert *AUDLVL2 enthält, werden auch die Werte im Systemwert QAUDLVL2 verwendet. Wenn der Systemwert QAUDLVL nicht den Wert *AUDLVL2 enthält, werden die Werte im Systemwert QAUDLVL2 ignoriert.

- Zum Ändern dieses Systemwerts ist die Sonderberechtigung *AUDIT erforderlich.
- Der Benutzer muss entweder über die Sonderberechtigung *ALLOBJ oder über die Sonderberechtigung *AUDIT verfügen, um diesen Systemwert anzuzeigen.

Eine Änderung dieses Systemwerts wird für alle Jobs im System sofort wirksam. Der Vorgabewert ist *NONE.

Entweder *NONE oder eine beliebige Kombination aus den anderen Werten angeben.

Gültige Werte sind:

*NONE	Auf dem System findet keine Sicherheitsprotokollierung statt. Dies ist der Standardwert.
*AUDLVL2	Die zu überwachenden Sicherheitsaktionen werden anhand der beiden Systemwerte QAUDLVL und QAUDLVL2 bestimmt. · Wenn ausschließlich der Systemwert QAUDLVL2 verwendet werden soll, für den Systemwert QAUDLVL den Wert *AUDLVL2 angeben und die Protokollierungswerte dem Systemwert QAUDLVL2 hinzufügen. · Wenn beide Systemwerte verwendet werden sollen, können die Werte im Systemwert QAUDLVL zusammen mit dem Wert *AUDLVL2 gesetzt werden; anschließend alle zusätzlichen Werte dem Systemwert QAUDLVL2 hinzufügen.

9.11.7 Seite 2

*ATNEVT	Abrufereignisse werden protokolliert. Abrufereignisse sind Bedingungen, die eine weitere Auswertung erfordern, damit die Bedeutung der Bedingung für die Sicherheit festgestellt werden kann. Beispiel: Überwachungsereignisse, die auf einen unbefugten Zugriff hinweisen, müssen näher untersucht werden, um festzustellen, ob es sich bei der Bedingung tatsächlich um einen unbefugten Zugriff oder einen falschen Alarm handelt.
*AUTFAIL	Berechtigungsfehler werden protokolliert. Beispiele: · Alle Zugriffsfehler (Anmeldung, Berechtigung, Jobübergabe) · Unzulässiges Kennwort oder unzulässige Benutzer-ID an einer Einheit eingegeben
*CREATE	Alle Objekterstellungen werden protokolliert. In der Bibliothek QTEMP erstellte Objekte werden nicht protokolliert. Beispiele: · Neu erstellte Objekte · Objekte, die erstellt wurden, um ein bestehendes Objekt zu ersetzen.
*DELETE	Alle Löschungen externer Objekte im System werden protokolliert. Objekte, die aus der Bibliothek QTEMP gelöscht werden, werden nicht protokolliert.
*JOBBAS	Jobbasisfunktionen werden protokolliert. Beispiele: · Start- und Stoppdaten eines Jobs · Anhalten, Freigeben, Stoppen, Fortsetzen, Ändern, Unterbrechen, Beenden, abnormal Beenden und Zuordnen der PSR (Programmstartanforderungen) zu vorab gestarteten Jobeinträgen
*JOBCHGUSR	Änderungen des aktiven Benutzerprofils für einen Thread oder seiner Gruppenprofile werden protokolliert.

*JOBDTA	Aktionen, die einen Job betreffen, werden protokolliert. Beispiele: · Start- und Stoppdaten eines Jobs · Anhalten, Freigeben, Stoppen, Fortsetzen, Ändern, Unterbrechen, Beenden, abnormal Beenden und Zuordnen der PSR (Programmstartanforderungen) zu vorab gestarteten Jobeinträgen · Ändern des aktiven Benutzerprofils für einen Thread oder Ändern von Gruppenprofilen · *JOBDTA setzt sich aus zwei Werten zusammen, damit die Protokollierung genauer angepasst werden kann. Wenn beide Werte angegeben werden, führt dies zur gleichen Art der Protokollierung wie bei Angabe von *JOBDTA. *JOBDTA setzt sich aus folgenden Werten zusammen: · *JOBBAS · *JOBCHGUSR
*NETBAS	Netzbasisfunktionen werden protokolliert. Beispiele: · IP-Regelaktionen · Sockets-Verbindungen · APPN-Verzeichnissuchfilter · APPN-Endpunktfilter
*NETCLU	Operationen für Cluster oder Cluster-Ressourcengruppen werden protokolliert. Beispiele: · Hinzufügen, erstellen und löschen · Verteilung · Beenden · Übernehmen · Listeninformationen · Entfernen · Starten · Umschalten · Attribute aktualisieren

9.11.7
Seite 4

*NETCMN	Netzbetriebs- und Übertragungsfunktionen werden protokolliert. Beispiele: · Netzbasisfunktionen (siehe *NETBAS) · Operationen für Cluster oder Cluster-Ressourcengruppen (siehe *NETCLU) · Netzfehler (siehe *NETFAIL) · Sockets-Funktionen (siehe *NETSCK) · *NETCMN setzt sich aus mehreren Werten zusammen, damit die Protokollierung genauer angepasst werden kann. Wenn alle Werte angegeben werden, führt dies zur gleichen Art der Protokollierung wie bei Angabe von *NETCMN. *NETCMN setzt sich aus folgenden Werten zusammen: · *NETBAS · *NETCLU · *NETFAIL · *NETSCK
*NETFAIL	Netzfehler werden protokolliert. Beispiele: · Socket-Port nicht verfügbar
*NETSCK	Sockets-Tasks werden protokolliert. Beispiele: · Accept · Connect · DHCP-Adresse zugeordnet · DHCP-Adresse nicht zugeordnet · Gefilterte Mail · Mail zurückweisen
*NOTAVL	Der Benutzer, der den Befehl ausführt, ist nicht berechtigt, den aktuellen Protokollierungswert anzuzeigen. Der Systemwert kann nicht in *NOTAVL (nicht verfügbar) geändert werden.
*OBJMGT	Generische Objekttasks werden protokolliert. Beispiele: · Übertragen von Objekten · Umbenennen von Objekten
*OFCSRV	Office Vision werden protokolliert. Beispiele: · Änderungen des Systemverteilerverzeichnisses · Aufgaben, die die elektronische Post betreffen

*OPTICAL	Alle optischen Funktionen werden protokolliert. Beispiele: · Hinzufügen oder Entfernen einer optischen Kassette · Ändern der Berechtigungsliste, die zum Schützen eines optischen Datenträgers verwendet wird · Öffnen einer optischen Datei oder eines optischen Verzeichnisses · Erstellen oder Löschen eines optischen Verzeichnisses · Ändern oder Abrufen der Attribute eines optischen Verzeichnisses · Kopieren, Versetzen oder Umbenennen einer optischen Datei · Kopieren eines optischen Verzeichnisses · Sichern eines optischen Datenträgers · Initialisieren oder Umbenennen eines optischen Datenträgers · Ändern eines optischen Sicherungsdatenträgers in einen primären Datenträger · Sichern oder Freigeben einer optischen Datei im Wartestatus · Absolutes Lesen eines optischen Datenträgers
*PGMADP	Die Übernahme der Programmeignerberechtigung wird protokolliert.
*PGMFAIL	Programmfehler werden protokolliert. Beispiele: · Geblockte Anweisung · Fehler bei Gültigkeitsprüfung · Domänenfehler
*PRTDTA	Druckfunktionen werden protokolliert. Beispiele: · Drucken einer Spool-Datei · Drucken mit dem Parameter SPOOL (*NO)

9.11.7
Seite 6

*SAVRST	Informationen, die das Sichern und Zurückspeichern betreffen, werden protokolliert. Beispiele: · wenn Programme zurückgespeichert werden, die das Profil des Programmeigners übernehmen; · wenn Jobbeschreibungen zurückgespeichert werden, die Benutzernamen enthalten; · wenn sich der Eigner oder Berechtigungen bei zurückgespeicherten Objekten ändern; · wenn die Berechtigung für Benutzerprofile zurückgespeichert wird; · wenn ein Systemstatusprogramm zurückgespeichert wird; · wenn ein Systembefehl zurückgespeichert wird; · wenn ein Objekt zurückgespeichert wird.
*SECCFG	Die Sicherheitskonfiguration wird protokolliert. Beispiele: · Erstellen, Ändern, Löschen und Zurückspeichern von Benutzerprofilen · Programmänderungen (CHGPGM), die nicht das Profil des Eigners übernehmen · Änderungen der Systemwerte, Umgebungsvariablen und Netzwerkattribute · Änderungen der Subsystemleitwege · Das Zurücksetzen des Kennworts QSECOFR von DST auf den Vorgabewert · Das Anfordern des Standardwerts für das Kennwort zur ID des Sicherheitsbeauftragten für die Serviceprogramme · Änderungen des Objektprotokollierungsattributs
*SECDIRSRV	Änderungen oder Updates bei der Ausführung von Verzeichnis-Servicefunktionen werden protokolliert. Beispiele: · Änderung des Protokolls · Erfolgreicher Bind · Änderung der Berechtigung · Änderung des Kennworts · Änderung des Eigentumsrechts · Erfolgreiches Auflösen (unbind)

*SECIPC	Änderungen der Interprozesskommunikation werden protokolliert. Beispiele: · Änderung des Eigentumsrechts oder der Berechtigung eines IPC-Objekts · Erstellen, Löschen oder Abrufen eines IPC-Objekts · Zuordnung des gemeinsam genutzten Speichers
*SECNAS	Aktionen des Netzwerkauthentifizierungsservice werden protokolliert. Beispiele: · Service-Ticket gültig · Service-Principals stimmen nicht überein · Client-Principals stimmen nicht überein · Diskrepanz bei Ticket-IP-Adresse · Entschlüsselung des Tickets fehlgeschlagen · Entschlüsselung des Authentifikators fehlgeschlagen · Realm befindet sich nicht innerhalb Client- und lokalem Realm · Ticket ist Replay-Versuch · Ticket ist noch nicht gültig · Diskrepanz bei lokaler IP-Adresse · Entschlüsselung von KRB_AP_PRIV- oder KRB_AP_SAFE-Kontrollsummenfehler · KRB_AP_PRIV-, KRB_AP_SAFE-Zeitmarkenfehler, Replay-Fehler, oder Reihenfolgefehler · GSS accept – abgelaufene Berechtigungsnachweise, Kontrollsummenfehler, Kanalbindungen · GSS unwrap oder GSS verify – abgelaufener Kontext, Entschlüsseln/Decodieren, Kontrollsummenfehler, Reihenfolgefehler
*SECRUN	Sicherheitslaufzeitfunktionen werden protokolliert. Beispiele: · Änderungen des Objekteignerrechts · Änderungen der Berechtigungsliste oder der Objektberechtigung · Änderungen der Primärgruppe eines Objekts
*SECSCKD	Socket-Deskriptoren werden protokolliert. Beispiele: · Ein Socket-Deskriptor wurde einem anderen Job zugeordnet · Deskriptor empfangen · Deskriptor kann nicht benutzt werden

9.11.7
Seite 8

*SECURITY

Alle sicherheitsrelevanten Funktionen werden protokolliert.
- Sicherheitskonfiguration (siehe *SECCFG)
- Änderungen oder Updates bei der Ausführung von Verzeichnisservicefunktionen (siehe *SECDIRSRV)
- Änderungen der Interprozesskommunikation (siehe *SECIPC)
- Aktionen des Netzwerkauthentifizierungsservice (siehe *SECNAS)
- Sicherheitslaufzeitfunktionen (siehe *SECRUN)
- Socket-Deskriptor (siehe *SECSCKD)
- Verwendung der Prüffunktionen (siehe *SECVFY)
- Änderungen von Prüflistenobjekten (siehe *SECVLDL)

*SECURITY setzt sich aus mehreren Werten zusammen, damit die Protokollierung genauer angepasst werden kann. Wenn alle Werte angegeben werden, führt dies zur gleichen Art der Protokollierung wie bei Angabe von *SECURITY. *SECURITY setzt sich aus folgenden Werten zusammen:
- *SECCFG
- *SECDIRSRV
- *SECIPC
- *SECNAS
- *SECRUN
- *SECSCKD
- *SECVFY
- *SECVLDL

*SECVFY

Die Verwendung der Prüffunktionen wird protokolliert. Beispiele:
- Ein Zielbenutzerprofil wurde während einer Pass-Through-Sitzung geändert
- Es wurde eine interne Profilkennung generiert
- Alle Profilmerkmale wurden inaktiviert
- Es wurde die maximale Anzahl an Profilmerkmalen generiert
- Es wurde ein Profilmerkmal generiert
- Alle Profilmerkmale für einen Benutzer wurden inaktiviert
- Benutzerprofil wurde authentifiziert
- Ein Office-Benutzer startete oder beendete die Verarbeitung im Namen eines anderen Benutzers

*SECVLDL	Änderungen von Prüflistenobjekten werden protokolliert. Beispiele: · Hinzufügen, Ändern, Entfernen eines Prüflisteneintrags · Suchen eines Prüflisteneintrags · Erfolgreiche oder nicht erfolgreiche Überprüfung eines Prüflisteneintrags
*SERVICE	Das System i Security-Referenzhandbuch, IBM Form SC41-5302 enthält eine Liste aller protokollierten Servicebefehle und API-Aufrufe.
*SPLFDTA	Spooldateifunktionen werden protokolliert. Beispiele: · Erstellen, Löschen, Anzeigen, Kopieren, Anhalten und Freigeben einer Spooldatei · Daten aus einer Spool-Datei abrufen (QSPGETSP) · Spool-Dateiattribute ändern (CHGSPLFA)
*SYSMGT	Systemverwaltungstasks werden protokolliert. Beispiele: · HFS-Registrierung · Änderungen von Funktionen für die Oberfläche von Anwendungen · Änderungen der Systemantwortliste · Änderungen am Verzeichnis der relationalen Datenbanken (DRDA) · Netzdateioperationen

9.11.7
Seite 10

QAUDLVL2

Erweiterung der Sicherheitsprotokollierung
Dieser Systemwert ist erforderlich, wenn mehr als 16 Protokollierungswerte benötigt werden. Wird *AUDLVL2 als einer der Werte im Systemwert QAUDLVL angegeben, sucht das System auch im Systemwert QAUDLVL2 nach Protokollierungswerten.

Wenn der Systemwert QAUDLVL den Wert *AUDLVL2 enthält, werden auch die Werte im Systemwert QAUDLVL2 verwendet. Wenn der Systemwert QAUDLVL nicht den Wert *AUDLVL2 enthält, werden die Werte im Systemwert QAUDLVL2 ignoriert.

- Zum Ändern dieses Systemwerts ist die Sonderberechtigung *AUDIT erforderlich.
- Der Benutzer muss entweder über die Sonderberechtigung *ALLOBJ oder über die Sonderberechtigung *AUDIT verfügen, um diesen Systemwert anzuzeigen.

Eine Änderung dieses Systemwerts wird für alle Jobs im System sofort wirksam. Der Vorgabewert ist *NONE.

9.11.8
Seite 2

QCRTAUT

Allgemeine Berechtigung für Objekterstellung
Gibt die standardmäßige allgemeine Berechtigung an, die beim Erstellen von Objekten in einer Bibliothek verwendet wird. Wird der Wert *LIBCRTAUT des Schlüsselworts AUT in einem Befehl zur Objekterstellung verwendet – etwa um die allgemeine Berechtigung für ein Objekt festzulegen – so gibt der Wert CRTAUT der Bibliothek, in der das Objekt erstellt wird, an, welche allgemeine Berechtigung für das Objekt verwendet wird. Ist der Wert CRTAUT der Bibliothek auf *SYSVAL gesetzt, so wird der im Systemwert QCRTAUT angegebene Wert verwendet, um die allgemeine Berechtigung für das Objekt, das gerade erstellt wird, festzulegen.

Wird der Systemwert QCRTAUT in den Wert *USE oder *EXCLUDE geändert, wird damit der Zugriff auf neu erstellte Objekte eingeschränkt. Der Eigner des Objekts oder der Sicherheitsbeauftragte müssen möglicherweise weitere Berechtigungen erteilen, bevor das Objekt verwendet werden kann. Ein Beispiel dafür ist die Anmeldung an einer neu erstellten Einheit. Wurde die Einheit mit dem Befehl CRTDEVDSP oder durch automatische Konfiguration erstellt, so wird die allgemeine Berechtigung auf *USE oder *EXCLUDE gesetzt. Da die Berechtigung *CHANGE für die Einheitenbeschreibung erforderlich ist, um sich am System anmelden zu können, ist in diesem Fall eine Anmeldung nicht möglich.

Wird der Systemwert QCRTAUT in *ALL geändert, können alle Benutzer des Systems, mit Ausnahme der Benutzer, denen eine stärker einschränkende Berechtigung als *ALL erteilt wurde, die neu erstellten Objekte vollständig steuern. Sie können diese Objekte lesen, ändern, löschen und deren Sicherheit verwalten.

Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist *CHANGE.

Gültige Werte sind:

*CHANGE	Die standardmäßige allgemeine Berechtigung lautet *CHANGE.
*ALL	Die standardmäßige allgemeine Berechtigung lautet *ALL.
*USE	Die standardmäßige allgemeine Berechtigung lautet *USE.
*EXCLUDE	Die standardmäßige allgemeine Berechtigung lautet *EXCLUDE.

9.11.9
Seite 2

QCRTOBJAUD

Objektprotokollierung
Dieser Systemwert gibt den Standardwert für die Protokollierung an, wenn Objekte in einer Bibliothek oder einem Verzeichnis erstellt werden. Lautet der CRTOBJAUD-Wert für die Bibliothek oder das Verzeichnis *SYSVAL, wird der Protokollierungswert für das zu erstellende Objekt durch den Wert festgelegt, der für den Systemwert QCRTOBJAUD angegeben ist.

Der Objektprotokollierungswert legt fest, ob beim Verwenden oder Ändern eines Objekts ein Protokolleintrag an das Systemprotokolljournal QAUDJRN in der Bibliothek QSYS gesendet wird. Der Protokolleintrag wird nur dann an das Systemprotokolljournal gesendet, wenn die Protokollierung im System aktiv ist. Um die Protokollierung zu starten, muss für den Systemwert QAUDCTL ein anderer Wert als *NONE angegeben werden.

Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist *NONE.

- Zum Ändern dieses Systemwerts ist die Sonderberechtigung *AUDIT erforderlich.
- Der Benutzer muss entweder über die Sonderberechtigung *ALLOBJ oder über die Sonderberechtigung *AUDIT verfügen, um diesen Systemwert anzuzeigen.

Gültige Werte sind:

*NOTAVL	Der Benutzer, der den Befehl ausführt, ist nicht berechtigt, den aktuellen Protokollierungswert anzuzeigen. Der Systemwert kann nicht in *NOTAVL (nicht verfügbar) geändert werden.
*NONE	Es werden keine Protokolleinträge beim Verwenden oder Ändern des Objekts gesendet.
*USRPRF	Es werden Protokolleinträge für das Objekt gesendet, das von einem Benutzer verwendet oder geändert wird, der derzeit protokolliert wird. Wird der Benutzer nicht protokolliert, werden auch keine Protokolleinträge gesendet. Um die Aktionen eines Benutzers zu protokollieren, muss sein Benutzerprofil mit dem Befehl CHGUSRAUD (Benutzerprotokollierung ändern) entsprechend geändert werden.
*CHANGE	Es werden Protokolleinträge für das Objekt gesendet, wenn es geändert wird.
*ALL	Es werden Protokolleinträge für das Objekt gesendet, wenn es verwendet oder geändert wird.

9.11.10

Seite 2

QDSPSGNINF

Steuerung der Anmeldeanzeigeinformationen
Dieser Systemwert steuert, ob der Benutzer eine Informationsnachricht angezeigt bekommt, die das Datum und die Uhrzeit der letzten Anmeldung sowie die Anzahl der ungültigen Anmeldeversuche angibt.

Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist 0.

Gültige Werte sind:

0	Die Anmeldeinformationen werden nicht angezeigt.
1	Die Anmeldeinformationen werden angezeigt.

9.11.11

Seite 2

QFRCCVNRST

Umsetzung beim Zurückspeichern erzwingen
Mit Hilfe dieses Systemwerts kann angegeben werden, ob die folgenden Objekttypen beim Zurückspeichern umgesetzt werden sollen oder nicht:

- Programme (*PGM)
- Serviceprogramme (*SRVPGM)
- SQL-Pakete (*SQLPKG)
- Module (*MODULE).

Dieser Systemwert kann auch verhindern, dass Objekte zurückgespeichert werden. Ein Objekt, das zwar vom System umgesetzt werden soll, dessen Umsetzung aber wegen unzureichender Erstellungsdaten nicht stattfinden kann, wird nicht zurückgespeichert.

Der *SYSVAL-Wert für den Parameter FRCOBJCVN der Rückspeicherungsbefehle (RST, RSTLIB, RSTOBJ, RSTLICPGM) übernimmt den Wert dieses Systemwerts. Durch Ändern des Systemwerts kann daher die Umsetzung für das gesamte System ein- oder ausgeschaltet werden. Der Parameter FRCOBJCVN kann den Systemwert jedoch in einigen Fällen überschreiben. Bei Angabe von *YES und *ALL für den Parameter FRCOBJCVN werden alle Einstellungen des Systemwerts überschrieben. Die Angabe von *YES und *RQD für den Parameter FRCOBJCVN entspricht der Angabe von ‚2' für diesen Systemwert und kann ihn überschreiben, wenn er ‚0' oder ‚1' lautet.

Wenn versucht wird, ein Objekt auf dem System zurückzuspeichern, arbeiten drei Systemwerte gemeinsam als Filter. Sie bestimmen, ob das Objekt zurückgespeichert werden darf oder ob es während der Rückspeicherung konvertiert wird. Der erste Filter ist der Systemwert QVFYOBJRST (Objekt beim Zurückspeichern prüfen). Dieser Systemwert steuert das Zurückspeichern einiger Objekte, die digital signiert werden können. Der zweite Filter ist der Systemwert QFRCCVNRST (Umsetzung beim Zurückspeichern erzwingen). Mit Hilfe dieses Systemwerts kann angegeben werden, ob während des Zurückspeicherns Programme, Serviceprogramme, SQL-Pakete und Modulobjekte konvertiert werden sollen oder nicht. Er kann auch verhindern, dass Objekte zurückgespeichert werden. Vom dritten Filter werden nur Objekte verarbeitet, die die ersten beiden Filter passiert haben. Der dritte Filter ist der Systemwert QALWOBJRST (Zurückspeichern des Objekts erlauben). Er gibt an, ob Objekte mit sicherheitsrelevanten Attributen zurückgespeichert werden können. Der Vorgabewert ist 1.

Bei allen Werten für QFRCCVNRST wird ein Objekt, das umgesetzt werden soll, dessen Umsetzung aber nicht erfolgen kann, nicht zurückgespeichert. Objekte, die über eine gültige Signatur einer vertrauenswürdigen Quelle verfügen, werden ohne Umsetzung der Werte dieses Systemwerts zurückgespeichert.

9.11.12
Seite 2

Gültige Werte sind:

0	Es erfolgt keine Umsetzung. Das Zurückspeichern wird durch nichts verhindert.
1	Objekte mit Gültigkeitsfehlern werden umgesetzt.
2	Objekte, die auf Grund der aktuellen Betriebssystemversion oder des aktuellen Systems umgesetzt werden müssen, werden umgesetzt. Objekte mit Gültigkeitsfehlern werden ebenfalls umgesetzt.
3	Objekte, die möglicherweise manipuliert wurden, Objekte mit Gültigkeitsfehlern und Objekte, die aufgrund der aktuellen Betriebssystemversion oder des aktuellen Systems umgesetzt werden müssen, werden umgesetzt.
4	Objekte, die genügend Erstellungsdaten für die Umsetzung enthalten, und nicht über die gültigen Signaturen verfügen, werden umgesetzt. Ein Objekt, das nicht genügend Erstellungsdaten enthält, wird ohne Umsetzung zurückgespeichert.

Anmerkung:

Objekte (mit oder ohne Signatur), die Gültigkeitsfehler enthalten, die möglicherweise manipuliert wurden oder die aufgrund der aktuellen Betriebssystemversion umgesetzt werden müssen, aber nicht umgesetzt werden können, werden nicht zurückgespeichert.

5	Objekte, die genügend Erstellungsdaten enthalten, werden umgesetzt. Ein Objekt, das nicht genügend Erstellungsdaten enthält, wird zurückgespeichert.

Anmerkung:

Objekte, die Gültigkeitsfehler enthalten, die möglicherweise manipuliert wurden oder die aufgrund der aktuellen Betriebssystemversion umgesetzt werden müssen, aber nicht umgesetzt werden können, werden nicht zurückgespeichert.

6	Alle Objekte ohne gültige digitale Signatur werden umgesetzt.

Anmerkung:

Ein Objekt mit einer gültigen digitalen Signatur, das außerdem Gültigkeitsfehler enthält, das möglicherweise manipuliert wurde oder das aufgrund der aktuellen Betriebssystemversion umgesetzt werden muss, aber nicht umgesetzt werden kann, wird nicht zurückgespeichert.

7 Jedes Objekt wird umgesetzt.

Wenn ein Objekt umgesetzt wird, wird seine digitale Signatur gelöscht. Das umgesetzte Objekt erhält den Benutzerstatus. Nach der Umsetzung haben Objekte einen fehlerlosen Gültigkeitswert und stehen nicht mehr im Verdacht, manipuliert worden zu sein.

9.11.12

Seite 4

QINACTITV

Zeitlimit für inaktiven Job
QINACTITV gibt an, wann das System bei nicht aktiven interaktiven Jobs eine Maßnahme ergreift. Der Systemwert QINACTMSGQ bestimmt die jeweils zu treffende Maßnahme. Weitere Informationen über das Erzwingen des Durchgriffs auf das Ziel und über TELNET-Sitzungen enthält das Handbuch Work Management. Lokale Jobs, die gegenwärtig an einem fernen System angemeldet sind, sind davon nicht betroffen.

Beispiel: Eine Datenstation ist direkt an System A angeschlossen, für das QINACTIV angegeben wurde. Wenn die Anmeldung an System B mit Datensichtgerätedurchgriff oder TELNET erfolgt, hat der in System A gesetzte Wert QINACTIV auf diese Datenstation keine Auswirkung. Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist *NONE.

Gültige Werte sind:

*NONE	Das System überprüft nicht, ob nicht aktive interaktive Jobs vorhanden sind.
5–300	Die Anzahl Minuten, die ein Job inaktiv sein kann, ehe eine entsprechende Maßnahme ergriffen wird.

9.11.13

Seite 2

QLMTDEVSSN

Einheitensitzungen begrenzen
Über diesen Systemwert wird die Anzahl der Einheitensitzungen gesteuert, bei denen sich ein Benutzer anmelden kann. Dabei wird nicht verhindert, dass der Benutzer an derselben Datenstation Gruppenjobs verwendet oder eine Systemanfrage durchführt (Taste für Systemanfrage). Der Vorgabewert ist 0.

Gültige Werte sind:

0	Benutzer sind nicht auf eine bestimmte Anzahl von Einheitensitzungen beschränkt.
1–9	Gibt die maximale Anzahl der gleichzeitig ablaufenden Einheitensitzungen an.

9.11.14

Seite 2

QLMTSECOFR

Einheitenzugriff des Sicherheitsbeauftragten begrenzen
Dieser Systemwert steuert, ob Benutzer mit den Sonderberechtigungen *ALLOBJ oder *SERVICE ausdrückliche Berechtigungen für bestimmte Datenstationen benötigen.

Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist 1.

Anmerkung:

Zum Ändern dieses Systemwerts sind die Sonderberechtigungen *ALLOBJ und *SECADM erforderlich.

Gültige Werte sind:

0	Benutzer mit der Sonderberechtigung *ALLOBJ oder *SERVICE können sich an jeder Datenstation anmelden.
1	Benutzer mit der Sonderberechtigung *ALLOBJ oder *SERVICE können sich nur an der Einheit anmelden, für die sie eine ausdrückliche Berechtigung besitzen.

9.11.15
Seite 2

QMAXSGNACN

Maßnahme bei fehlgeschlagenen Anmeldeversuchen
Wird bei Erreichen der maximal zulässigen Anzahl Anmeldeversuche ausgeführt. Gibt an, wie das System reagiert, wenn die maximal zulässige Anzahl aufeinanderfolgender ungültiger Anmeldeversuche (Systemwert QMAXSIGN) erreicht ist.

Eine Änderung dieses Systemwerts wird beim nächsten Anmeldeversuch eines Benutzers am System wirksam. Der Vorgabewert ist ‚3'.

Gültige Werte sind:

1	Bei Erreichen des Grenzwerts wird die Einheit abgehängt.
2	Bei Erreichen des Grenzwerts wird das Benutzerprofil inaktiviert.
3	Bei Erreichen des Grenzwerts wird die Einheit abgehängt und das Benutzerprofil inaktiviert.

9.11.16

Seite 2

QMAXSIGN

Maximal zulässige Anmeldeversuche
Ungültige Anmeldeversuche bei gesicherten Systemen (Sicherheitsstufe 20 oder höher, siehe Systemwert QSECURITY) können folgende Ursachen haben:

- Falsche Benutzer-ID
- Falsches Kennwort

Das Benutzerprofil hat keine Berechtigung für die Einheit, an der die Benutzer-ID eingegeben wurde.

Anmerkung:

Ein Anmeldeversuch wird nicht als ungültiger Versuch gewertet, wenn Kennwörter erforderlich sind und für das Benutzerprofil das Kennwort *NONE angegeben ist. Der Benutzer erhält die Nachricht, dass diesem Benutzerprofil kein Kennwort zugeordnet ist. Auch wenn der Programm- oder Menüname ungültig ist, wenn die Benutzer-ID in einem nicht gesicherten System ungültig ist, wenn die aktuelle Bibliothek nicht gefunden werden kann oder wenn das Benutzerprofil keine *AUDIT-Berechtigung enthält und der Versuch stattfindet, nachdem aufgrund eines Protokollierungsfehlers ein IPL für das System durchgeführt wurde, wird der Anmeldeversuch nicht als ungültig bewertet.

Ist der maximale Wert von QMAXSIGN erreicht, wird die durch den Systemwert QMAXSGNACN festgelegte Maßnahme ausgeführt. Die Datenstation wird abgehängt und/oder das Benutzerprofil inaktiviert. Eine Nachricht wird an die Nachrichtenwarteschlange QSYSMSG gesendet, vorausgesetzt, diese Warteschlange ist vorhanden; andernfalls wird die Nachricht an QSYSOPR gesendet. Wird ein Profil inaktiviert, muss es vor dem Anmelden eines Benutzers wieder aktiviert werden. Wird eine Einheit abgehängt, muss diese vor dem Anmelden eines Benutzers wieder angehängt werden. Befindet sich das Steuersubsystem im eingeschränkten Status (so dass nur eine Einheit in diesem System verwendet werden kann) und wird die Einheit abgehängt, wird das System beendet und die eingeschalteten Lampen an der Steuerkonsole zeigen an, dass ein IPL durchgeführt werden muss.

Liegt die Anzahl der ungültigen Anmeldeversuche um 1 niedriger als der Wert für QMAXSIGN, wird eine Nachricht angezeigt. Diese Nachricht warnt den Benutzer, dass bei einem weiteren ungültigen Anmeldeversuch die im Systemwert QMAXSGNACN angegebene Maßnahme ausgeführt wird.

9.11.17
Seite 2

Eine Änderung dieses Systemwerts wird wirksam, wenn sich der nächste Benutzer am System anmeldet. Der Vorgabewert ist 3.

Gültige Werte sind:

1–25	Maximale Anzahl zulässiger Anmeldeversuche
*NOMAX	Es gibt keine maximale Anzahl zulässiger Anmeldeversuche.

QPWDCHGBLK

Kennwortänderung blockieren
Gibt an, wie lange die Änderung eines Kennworts nach einer erfolgreichen Kennwortänderung blockiert ist. Kennwortänderungen, die mit dem Befehl CHGUSRPRF (Benutzerprofil ändern) erfolgen, werden durch diesen Systemwert nicht eingeschränkt.

Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist *NONE.

Gültige Werte sind:

*NONE	Der Benutzer kann das Kennwort beliebig oft ändern. Es gibt keine Einschränkung.
1-99	Zeigt an, wie viele Stunden der Benutzer nach einer erfolgreichen Kennwortänderung warten muss, bevor er das Kennwort erneut ändern darf.

9.11.18

Seite 2

QPWDEXPITV

Intervall für Kennwortablauf
Gibt die Anzahl der Tage an, die ein Kennwort gültig ist. Das ermöglicht Kennwortsicherheit, da die Benutzer gezwungen sind, ihr Kennwort nach einer bestimmten Anzahl Tage zu ändern. Wenn das Kennwort innerhalb dieser Frist nicht geändert wird, kann sich der Benutzer nicht anmelden.

Sieben Tage vor Ablauf des Kennworts erhält der Benutzer beim Anmelden eine entsprechende Warnmeldung, auch wenn die Anmeldeinformationen nicht angezeigt werden (Systemwert QDSPSGNINF).

Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist *NOMAX.

Gültige Werte sind:

*NOMAX	Ein Kennwort kann eine unbegrenzte Anzahl von Tagen verwendet werden.
1-366	Die Anzahl der Tage, bevor das Kennwort ungültig wird.

9.11.19

Seite 2

QPWDEXPWRN

Warnung vor Kennwortablauf
Steuert, wie viele Tage vor dem Kennwortablauf mit der Anzeige einer entsprechenden Warnung auf dem Anmeldebildschirm begonnen wird.

Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist 7.

Gültige Werte sind:

1-99	Gibt an, wie viele Tage vor Kennwortablauf mit der Anzeige einer entsprechenden Warnung auf dem Anmeldebildschirm begonnen wird.

9.11.20

Seite 2

QPWDLMTAJC

Zusammenhängende Ziffern in Kennwort begrenzen
Gibt an, ob in Kennwörtern aufeinanderfolgende Ziffern erlaubt sind. Wenn die Verwendung von Datumsangaben und Sozialversicherungsnummern unzulässig ist, ist es schwieriger, ein Kennwort zu erraten.

Eine Änderung dieses Systemwerts wird bei der nächsten Kennwortänderung wirksam. Der Vorgabewert ist 0.

Gültige Werte sind:

0	Aufeinanderfolgende Ziffern sind zulässig.
1	Aufeinanderfolgende Ziffern sind nicht zulässig.

9.11.21

Seite 2

QPWDLMTCHR

Zeichen in Kennwort begrenzen
Kennwortsicherheit kann erreicht werden, wenn bestimmte Zeichen (z. B. Großbuchstaben) in Kennwörtern verboten sind. Da keine allgemein gebräuchlichen Wörter oder Begriffe eingegeben werden können, ist es schwierig, das Kennwort zu erraten.

Eine Änderung dieses Systemwerts wird bei der nächsten Kennwortänderung wirksam. Der Vorgabewert ist *NONE.

Gültige Werte sind:

*NONE	Es gibt keine unzulässigen Zeichen.
eingeschränkte Zeichen	Es können bis zu zehn eingeschränkte Zeichen angegeben werden. Gültige Zeichen für das Kennwort sind: A–Z, 0–9 und die Sonderzeichen #, $, das Unterstreichungszeichen (_) sowie §.

Anmerkung:

0123456789 kann nicht angegeben werden, wenn in einem Kennwort Zahlen erforderlich sind (siehe Systemwert QPWDRQDDGT).

Dieser Systemwert wird ignoriert, wenn das System mit QPWDLVL 2 bzw. 3 arbeitet.

Wenn im Systemwert QPWDRULES ein anderer Wert als *PWDSYSVAL angegeben ist, wird dieser Systemwert bei der Prüfung neuer Kennwörter auf Formatfehler ignoriert.

9.11.22
Seite 2

QPWDLMTREP

Zeichenwiederholung in Kennwort begrenzen
Damit wird verhindert, dass in einem Kennwort dasselbe Zeichen mehrmals verwendet wird (beispielsweise AAAA).

Eine Änderung dieses Systemwerts wird bei der nächsten Kennwortänderung wirksam. Der Vorgabewert ist 0.

Gültige Werte sind:

0	Zeichen können mehrmals verwendet werden.
1	Zeichen können nur einmal verwendet werden.
2	Zeichen können nicht fortlaufend verwendet werden.

Anmerkung:

Wenn im Systemwert QPWDRULES ein anderer Wert als *PWDSYSVAL angegeben ist, wird dieser Systemwert bei der Prüfung neuer Kennwörter auf Formatfehler ignoriert.

9.11.23

Seite 2

QPWDLVL

Kennwortstufe
Gibt die Stufe für die Kennwortunterstützung im System an.

Die Kennwortstufe des Systems kann dahingehend festgelegt werden, dass Benutzerprofilkennwörter mit 1–10 Zeichen bzw. 1–128 Zeichen verwendet werden.

Die Kennwortstufe kann dahingehend festgelegt werden, dass eine ‚Kennphrase' als Kennwortwert verwendet wird. Der Begriff ‚Kennphrase' wird zuweilen in der Computerbranche verwendet, um einen Kennwortwert zu beschreiben, für den wenige bzw. keine Einschränkungen hinsichtlich der Zeichenlänge gelten. In einer Kennphrase können Leerzeichen zwischen einzelnen Zeichen verwendet werden. Das ermöglicht, dass ein Satz oder ein Satzteil als Kennwortwert verwendet werden kann.

Beim Ändern der Kennwortstufe des Systems von 1–10 Zeichen lange in 1–128 Zeichen lange Kennwörter ist Vorsicht geboten. Ist das System z. B. mit anderen Systemen in einem Netzwerk verbunden, müssen alle Systeme in der Lage sein, die längeren Kennwörter zu verarbeiten.

Im Handbuch System i Security Reference, IBM Form SC41-5302, werden weitere Überlegungen erläutert, die vor dem Ändern dieses Systemwerts angestellt werden sollten.

Eine Änderung dieses Systemwerts wird beim nächsten IPL wirksam. Um den aktuellen und den anstehenden Wert der Kennwortstufe anzuzeigen, wird der CL-Befehl DSPSECA (Sicherheitsattribute anzeigen) verwendet. Der Vorgabewert ist 0.

Gültige Werte sind:

0	Das System unterstützt Benutzerprofilkennwörter mit einer Länge von 1–10 Zeichen. Die zulässigen Zeichen sind A–Z, 0–9 und die Zeichen $, §, # sowie das Unterstreichungszeichen. QPWDLVL 0 sollte verwendet werden, wenn das System mit anderen IBM i-Systemen in einem Netzwerk verbunden ist und diese Systeme entweder mit dem QPWDLVL-Wert 0 oder mit einem Betriebssystem-Release vor V5R1M0 ausgeführt werden. QPWDLVL 0 sollte verwendet werden, wenn das System mit einem anderen System, das die Kennwortlänge auf 1–10 Zeichen beschränkt, Daten austauscht.

9.11.24
Seite 2

QPWDLVL 0 muss verwendet werden, wenn das System mit Windows 95/98/ME-Unterstützung für Windows Network Neighborhood (i5/OS NetServer) und mit anderen Systemen, die Kennwörter mit einer Länge von 1–10 Zeichen verwenden, Daten austauscht.

Ist der QPWDLVL-Wert des Systems 0, erstellt das Betriebssystem das verschlüsselte Kennwort für QPWDLVL 2 und 3. Der Wert des Kennworts ist dem Benutzerprofilkennwort gleich.

1 QPWDLVL 1 entspricht der Unterstützung von QPWDLVL 0 mit der folgenden Ausnahme: Auf dem System gespeicherte Kennwörter, die mit Windows 95/98/ME-Unterstützung für Windows Network Neighborhood (i5/OS NetServer) verwendet werden sollen, werden im System gelöscht. Wird die Client-Unterstützung für i5/OS NetServer verwendet, ist die Verwendung des QPWDLVL-Werts 1 nicht möglich.

QPWDLVL 1 verbessert die Sicherheit des Systems, da alle Kennwörter für i5/OS NetServer im System gelöscht werden.

2 Das System unterstützt Benutzerprofilkennwörter mit einer Länge von 1–128 Zeichen. Zeichen in Groß- und Kleinbuchstaben sind zulässig. Das Kennwort kann aus beliebigen Zeichen bestehen. Die Groß- und Kleinschreibung des Kennworts muss beachtet werden.

QPWDLVL 2 wird als Kompatibilitätsstufe betrachtet. Die Kennwörter werden zur Authentifizierung von Anmeldevorgängen und anderen Kennworttests verwendet. Diese Stufe ermöglicht auch eine Rückkehr zu QPWDLVL 0 oder 1, vorausgesetzt, dass das jeweilige Kennwort den Längen- und Syntaxvorgaben von QPWDLVL 0 bzw. 1 entspricht.

QPWDLVL 2 kann verwendet werden, wenn das System mit der Windows 95/98/ME-Unterstützung für Windows Network Neighborhood (i5/OS NetServer) Daten austauscht, vorausgesetzt, das Kennwort ist 1–14 Zeichen lang.

QPWDLVL 2 kann nicht verwendet werden, wenn das System mit anderen Systemen in einem Netzwerk verbunden ist und diese Systeme entweder mit dem QPWDLVL-Wert 0 bzw. 1 oder mit einem Betriebssystem-Release vor V5R1M0 ausgeführt werden.

QPWDLVL 2 kann nicht verwendet werden, wenn das System mit einem anderen System, das die Kennwortlänge auf 1–10 Zeichen beschränkt, Daten austauscht.

3 Das System unterstützt Benutzerprofilkennwörter mit einer Länge von 1–128 Zeichen. Zeichen in Groß- und Kleinbuchstaben sind zulässig. Das Kennwort kann aus beliebigen Zeichen bestehen. Die Groß- und Kleinschreibung des Kennworts muss beachtet werden.

In der Veröffentlichung System i Security Reference, IBM Form SC41-5302, werden weitere Überlegungen erläutert, die vor dem Ändern der Kennwortstufe auf 3 angestellt werden sollten. Es ist nicht zulässig, QPWDLVL 3 wieder in 0 zu ändern.

QPWDLVL 3 kann nicht verwendet werden, wenn das System mit anderen Systemen in einem Netzwerk verbunden ist und diese Systeme entweder mit dem QPWDLVL-Wert 0 bzw. 1 oder mit einem Betriebssystem-Release vor V5R1M0 ausgeführt werden.

QPWDLVL 3 kann nicht verwendet werden, wenn das System mit einem anderen System, das die Kennwortlänge auf 1–10 Zeichen beschränkt, Daten austauscht.

QQPWDLVL 3 kann nicht verwendet werden, wenn das System mit der Windows 95/98/ME-Unterstützung für Windows Network Neighborhood (i5/OS NetServer) Daten austauscht.

9.11.24

Seite 4

QPWDMAXLEN

Maximale Kennwortlänge
Gibt die maximale Anzahl Zeichen pro Kennwort an.

Eine Änderung dieses Systemwerts wird bei der nächsten Kennwortänderung wirksam. Der Vorgabewert ist 8.

Gültige Werte sind:

1–128	Die maximale Anzahl Zeichen, die für ein Kennwort zulässig ist.Wenn das System mit QPWDLVL 0 oder 1 arbeitet, sind 1–10 Zeichen zulässig. Wenn das System mit QPWDLVL 2 oder 3 arbeitet, sind 1–128 Zeichen zulässig.

Anmerkung:

Die maximale Kennwortgröße kann die im Systemwert QPWDMINLEN angegebene Mindestgröße nicht unterschreiten.

Anmerkung:

Wenn im Systemwert QPWDRULES ein anderer Wert als *PWDSYSVAL angegeben ist, wird dieser Systemwert bei der Prüfung neuer Kennwörter auf Formatfehler ignoriert.

9.11.25

Seite 2

QPWDMINLEN

Mindestkennwortlänge
Gibt die Mindestanzahl Zeichen für ein Kennwort an.

Eine Änderung dieses Systemwerts wird bei der nächsten Kennwortänderung wirksam. Der Vorgabewert ist 6.

Gültige Werte sind:

1–128	Die Mindestanzahl von Zeichen für ein Kennwort. Wenn das System mit QPWDLVL 0 oder 1 arbeitet, sind 1–10 Zeichen zulässig. Wenn das System mit QPWDLVL 2 oder 3 arbeitet, sind 1–128 Zeichen zulässig.

Anmerkung:

Die Mindestgröße des Kennworts kann die im Systemwert QPWDMAXLEN angegebene maximale Kennwortgröße nicht überschreiten. Wenn im Systemwert QPWDRULES ein anderer Wert als *PWDSYSVAL angegeben ist, wird dieser Systemwert bei der Prüfung neuer Kennwörter auf Formatfehler ignoriert.

9.11.26

Seite 2

QPWDPOSDIF

Zeichenpositionen in Kennwort begrenzen
Dieser Systemwert kontrolliert die Position der Zeichen in einem neuen Kennwort. Damit wird verhindert, dass der Benutzer ein Kennwort angibt, das größtenteils mit dem alten Kennwort übereinstimmt. Beispielsweise könnte das Kennwort DJS2 nicht verwendet werden, wenn das vorhergehende DJS1 gelautet hat (D, J und S stehen an derselben Stelle).

Eine Änderung dieses Systemwerts wird bei der nächsten Kennwortänderung wirksam. Der Vorgabewert ist 0.

Gültige Werte sind:

0	Es ist möglich, im neuen Kennwort das gleiche Zeichen an dieselbe Stelle zu setzen wie im vorhergehenden Kennwort.
1	Ein bestimmtes Zeichen kann nicht an dieselbe Stelle wie im vorhergehenden Kennwort gesetzt werden. Beispielsweise könnte das Kennwort DJS2 nicht verwendet werden, wenn das vorhergehende DJS1 gelautet hat (D, J und S stehen an derselben Stelle).

Anmerkung:

Wenn im Systemwert QPWDRULES ein anderer Wert als *PWDSYSVAL angegeben ist, wird dieser Systemwert bei der Prüfung neuer Kennwörter auf Formatfehler ignoriert.

9.11.27
Seite 2

QPWDRQDDGT

Ziffer in Kennwort erforderlich
Gibt an, ob in einem neuen Kennwort eine Ziffer erforderlich ist. Damit wird verhindert, dass der Benutzer nur Buchstaben verwendet.

Eine Änderung dieses Systemwerts wird bei der nächsten Kennwortänderung wirksam. Der Vorgabewert ist 0.

Gültige Werte sind:

0	Es sind keine Ziffern erforderlich.
1	Eine bzw. mehrere Ziffern sind erforderlich.

Anmerkung:

0123456789 ist als Angabe für den Systemwert QPWDLMTCHR nicht erlaubt, wenn in einem Kennwort Zahlen erforderlich sind.

Wenn im Systemwert QPWDRULES ein anderer Wert als *PWDSYSVAL angegeben ist, wird dieser Systemwert bei der Prüfung neuer Kennwörter auf Formatfehler ignoriert.

9.11.28

Seite 2

QPWDRQDDIF

Kontrolle auf doppelte Kennwörter
Dieser Systemwert begrenzt die Häufigkeit der Verwendung eines Kennworts.

Eine Änderung dieses Systemwerts wird bei der nächsten Kennwortänderung wirksam. Der Vorgabewert ist 0.

Gültige Werte sind:

0	Das Kennwort kann mit dem vorhergehenden identisch sein.
1	Ein Kennwort muss sich von den 32 vorhergehenden Kennwörtern unterscheiden.
2	Das Kennwort muss sich von den vorhergehenden 24 Kennwörtern unterscheiden.
3	Das Kennwort muss sich von den vorhergehenden 18 Kennwörtern unterscheiden.
4	Das Kennwort muss sich von den vorhergehenden zwölf Kennwörtern unterscheiden.
5	Das Kennwort muss sich von den vorhergehenden zehn Kennwörtern unterscheiden.
6	Das Kennwort muss sich von den vorhergehenden acht Kennwörtern unterscheiden.
7	Das Kennwort muss sich von den vorhergehenden sechs Kennwörtern unterscheiden.
8	Das Kennwort muss sich von den vorhergehenden vier Kennwörtern unterscheiden.

9.11.29

Seite 2

QPWDRULES

Kennwortregeln
Gibt die Regeln an, die zur Überprüfung eines Kennworts auf Formatfehler verwendet werden.

Änderungen dieses Systemwerts werden bei der nächsten Kennwortänderung wirksam. Der Vorgabewert ist *PWDSYSVAL.

Gültige Werte sind:

*PWDSYSVAL — Dieser Systemwert wird ignoriert, und alle anderen Kennwortsystemwerte werden zur Überprüfung eines Kennworts auf Formatfehler verwendet. Speziell die Systemwerte QPWDLMTAJC, QPWDLMTCHR, QPWDLMTREP, QPWDMAXLEN, QPWDMINLEN, QPWDPOSDIF und QPWDRQDDGT werden statt des Systemwerts QPWDRULES verwendet.

Anmerkung:

Wenn ein anderer Wert als *PWDSYSVAL für QPWDRULES angegeben wird, werden die Systemwerte QPWDLMTAJC, QPWDLMTCHR, QPWDLMTREP, QPWDMAXLEN, QPWDMINLEN, QPWDPOSDIF und QPWDRQDDGT ignoriert, wenn ein neues Kennwort auf Formatfehler überprüft wird.

*CHRLMTAJC — Das Kennwort darf keine identischen aufeinanderfolgenden Zeichen enthalten. Dieser Wert kann nicht angegeben werden, wenn der Wert *CHRLMTREP angegeben wurde.

*CHRLMTREP — Jedes Zeichen darf nur einmal im Kennwort enthalten sein. Dieser Wert kann nicht angegeben werden, wenn der Wert *CHRLMTAJC angegeben wurde.

*DGTLMTAJC — Das Kennwort darf keine identischen aufeinanderfolgenden Ziffern enthalten.

*DGTLMTFST — Als erstes Zeichen des Kennworts darf keine Ziffer verwendet werden. Dieser Wert kann nicht angegeben werden, wenn die Werte *LTRLMTFST und *SPCCHRLMTFST angegeben wurden.

*DGTLMTLST — Als letztes Zeichen des Kennworts darf keine Ziffer verwendet werden. Dieser Wert kann nicht angegeben werden, wenn die Werte *LTRLMTLST und *SPCCHRLMTLLST angegeben wurden.

9.11.30
Seite 2

*DGTMAXn	n steht für eine Zahl von 0 bis 9. Gibt die maximale Anzahl Ziffern an, die das Kennwort enthalten darf. Der Wert *DGTMAXn kann nur einmal angegeben werden. Wenn der Wert *DGTMINn ebenfalls angegeben wird, muss der für *DGTMAXn angegebene n-Wert größer oder gleich dem n-Wert sein, der für *DGTMINn angegeben wurde.
*DGTMINn	n steht für eine Zahl von 0 bis 9. Gibt die Mindestanzahl Ziffern an, die ein Kennwort enthalten muss. Der Wert *DGTMINn kann nur einmal angegeben werden. Wenn *DGTMAXn ebenfalls angegeben wird, muss der für *DGTMAXn angegebene n-Wert größer oder gleich dem n-Wert sein, der für *DGTMINn angegeben wurde.
*LMTSAMPOS	Ein bestimmtes Zeichen kann nicht an dieselbe Stelle wie im vorhergehenden Kennwort gesetzt werden.
*LMTPRFNAME	Das Kennwort in Großbuchstaben darf nicht den vollständigen Namen des Benutzerprofils (in aufeinanderfolgenden Positionen) enthalten.
*LTRLMTAJC	Das Kennwort darf keine identischen aufeinanderfolgenden Buchstaben enthalten.
*LTRLMTFST	Als erstes Zeichen des Kennworts darf kein Buchstabe verwendet werden. Dieser Wert kann nicht angegeben werden, wenn die Werte *DGTLMTFST und *SPCCHRLMTFST angegeben wurden. Wenn das System mit einem Wert von 0 oder 1 für QPWDLVL arbeitet, können die beiden Werte *LTRLMTFST und *SPCCHRLMTFST nicht zusammen angegeben werden.
*LTRLMTLST	Als letztes Zeichen des Kennworts darf kein Buchstabe verwendet werden. Dieser Wert kann nicht angegeben werden, wenn die Werte *DGTLMTLST und *SPCCHRLMTLST angegeben wurden.
*LTRMAXn	n steht für eine Zahl von 0 bis 9. Gibt die maximale Anzahl Buchstaben an, die das Kennwort enthalten darf. Der Wert *LTRMAXn kann nur einmal angegeben werden. Wenn der Wert *LTRMINn ebenfalls angegeben wird, muss der für *LTRMAXn angegebene n-Wert größer oder gleich dem n-Wert sein, der für *LTRMINn angegeben wurde.

*LTRMINn	n steht für eine Zahl von 0 bis 9. Gibt die Mindestanzahl Buchstaben an, die ein Kennwort enthalten muss. Der Wert *LTRMINn kann nur einmal angegeben werden. Wenn *LTRMAXn ebenfalls angegeben wird, muss der für *LTRMAXn angegebene n-Wert größer oder gleich dem n-Wert sein, der für *LTRMINn angegeben wurde.
*MAXLENnnn	nnn steht für eine Zahl von 1 bis 128 (ohne führende Nullen), die maximale Anzahl Zeichen pro Kennwort. Wenn das System mit QPWDLVL 0 oder 1 arbeitet, sind 1–10 Zeichen zulässig. Wenn das System mit QPWDLVL 2 oder 3 arbeitet, sind 1–128 Zeichen zulässig. Der angegebene nnn-Wert muss groß genug sein, um die Werte für *MIXCASEn, *DGTMAXn, *LTRMAXn, *SPCCHRMAXn aufzunehmen und die Einschränkungen bezüglich dem ersten und dem letzten Zeichen sowie die Anforderung bezüglich nicht aufeinanderfolgenden Zeichen zu erfüllen. Wenn *MINLENnnn ebenfalls angegeben wird, muss der für *MAXLENnnn angegebene nnn-Wert größer oder gleich dem nnn-Wert sein, der für *MINLENnnn angegeben wurde. Wurde der Wert *MAXLENnnn nicht angegeben, wird der Wert *MAXLEN10 angenommen, wenn das System mit einem Wert von 0 oder 1 für QPWDLVL arbeitet, oder *MAXLEN128, wenn das System mit einem Wert von 2 oder 3 für QPWDLVL arbeitet.
*MINLENnnn	nnn steht für eine Zahl von 1 bis 128 (ohne führende Nullen), die Mindestanzahl Zeichen für ein Kennwort. Wenn das System mit QPWDLVL 0 oder 1 arbeitet, sind 1–10 Zeichen zulässig. Wenn das System mit QPWDLVL 2 oder 3 arbeitet, sind 1–128 Zeichen zulässig. Wenn *MAXLENnnn ebenfalls angegeben wird, muss der für *MAXLENnnn angegebene nnn-Wert größer oder gleich dem nnn-Wert sein, der für *MINLENnnn angegeben wurde. Wird kein Wert für *MINLENnnn angegeben, wird der Wert 0 angenommen.

9.11.30
Seite 4

*MIXCASEn	n steht für eine Zahl von 0 bis 9. Das Kennwort muss mindestens n Großbuchstaben und n Kleinbuchstaben enthalten Dieser Wert wird zurückgewiesen, wenn das System mit einem Wert von 0 oder 1 für QPWDLVL arbeitet, da Kennwörter in Großbuchstaben angegeben werden müssen. Der Wert *MIXCASEn kann nur einmal angegeben werden. Wenn *LTRMAXn angegeben wird, muss der für *LTRMAXn angegebene n-Wert größer oder gleich dem zweifachen n-Wert sein, der für *MIXCASEn angegeben wurde.
*REQANY3	Das Kennwort muss Zeichen aus mindestens drei der folgenden vier Zeichenarten enthalten. · Großbuchstaben · Kleinbuchstaben · Ziffern · Sonderzeichen Wenn das System mit einem Wert von 0 oder 1 für QPWDLVL arbeitet, hat *REQANY3 dieselbe Auswirkung als ob *DGTMIN1, *LTRMIN1, und *SPCCHRMIN1 zusammen angegeben worden wären.
*SPCCHRLMTAJC	Das Kennwort darf keine identischen aufeinanderfolgenden Sonderzeichen enthalten.
*SPCCHRLMTFST	Als erstes Zeichen des Kennworts darf kein Sonderzeichen verwendet werden. Dieser Wert kann nicht angegeben werden, wenn die Werte *DGTLMTFST und *LTRLMTFST ebenfalls angegeben wurden. Wenn das System mit einem Wert von 0 oder 1 für QPWDLVL arbeitet, können die beiden Werte *LTRLMTFST und *SPCCHRLMTFST nicht zusammen angegeben werden.
*SPCCHRLMTLST	Als letztes Zeichen des Kennworts darf kein Sonderzeichen verwendet werden. Dieser Wert kann nicht angegeben werden, wenn die Werte *DGTLMTLST und *LTRLMTLST angegeben wurden.

*SPCCHRMAXn	n steht für eine Zahl von 0 bis 9. Gibt die maximale Anzahl Sonderzeichen an, die das Kennwort enthalten darf. Der Wert *SPCCHRMAXn kann nur einmal angegeben werden. Wenn *SPCCHRMINn ebenfalls angegeben wird, muss der für *SPCCHRMAXn angegebene n-Wert größer oder gleich dem n-Wert sein, der für *SPCCHRMINn angegeben wurde.
*SPCCHRMINn	n steht für eine Zahl von 0 bis 9 und gibt die Mindestanzahl Sonderzeichen an, die ein Kennwort enthalten muss. Der Wert *SPCCHRMINn kann nur einmal angegeben werden. Wenn *SPCCHRMAXn ebenfalls angegeben wird, muss der für *SPCCHRMAXn angegebene n-Wert größer oder gleich dem n-Wert sein, der für *SPCCHRMINn angegeben wurde.

9.11.30

Seite 6

QPWDVLDPGM

Kennwortprüfprogramm
Damit wird den vom Benutzer geschriebenen Programmen ermöglicht, zusätzliche Kennwortprüfungen durchzuführen. Das Programm muss im System-ASP (ASP = Zusatzspeicherpool) oder in einem Basis-Benutzer-ASP vorhanden sein.

Eine Änderung dieses Systemwerts wird bei der nächsten Kennwortänderung

wirksam. Der Vorgabewert ist *NONE.

Gültige Werte sind:

*NONE	Es wird kein Kennwortprüfprogramm verwendet.
*REGFAC	Das Kennwortprüfprogramm wird von der Registrierungsfunktion abgerufen.
Programm-spezifikation	Der Name des Kennwortprüfprogramms. Diese Auswahl ist nur gültig, wenn das System mit QPWDLVL 0 bzw. 1 arbeitet.

Gültige Bibliothekswerte sind:

*LIBL	Das Kennwortprüfprogramm wird in der Bibliotheksliste gesucht.
*CURLIB	Das Kennwortprüfprogramm wird in der aktuellen Jobbibliothek gesucht. Ist keine aktuelle Bibliothek für den Job angegeben, wird QGPL verwendet.
Bibliotheksname	Den Namen der Bibliothek angeben, in der sich das Kennwortprüfprogramm befindet.

Anmerkung:

Aktuelles und neues Kennwort werden unverschlüsselt an das Prüfprogramm übermittelt. Das Prüfprogramm könnte Kennwörter in einer Datenbankdatei speichern und die Sicherheit im System beeinträchtigen.

Die empfohlene Einstellung für diesen Systemwert ist *NONE.

9.11.31

Seite 2

QRETSVRSEC

Server-Sicherheitsdaten sichern
Gibt an, ob die von einem Server benötigten Sicherheitsdaten zur Authentifizierung eines Benutzers auf einem Zielsystem über Client-Server-Schnittstellen auf dem Host-System gesichert werden können.

Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist 0.

Der Systemwert QRETSVRSEC kann nur mit den Sonderberechtigungen *ALLOBJ und *SECADM geändert werden.

Gültige Werte sind:

0	Server-Sicherheitsdaten nicht sichern
1	Server-Sicherheitsdaten sichern

9.11.32

Seite 2

QRMTSIGN

Ferne Anmeldesteuerung
Dieser Systemwert gibt an, wie das System ferne Anmeldeanforderungen behandelt.

Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist *FRCSIGNON.

Alle Werte werden wie unten beschrieben für Datenstationsdurchgriff, System i Navigator Work Station Function (WSF) und andere 5250-Emulationsprogramme auf programmierbaren Datenstationen unterstützt. Weitere Informationen über die Unterstützungsebene für TELNET-Sitzungen enthält das Handbuch Work Management.

Gültige Werte sind:

*FRCSIGNON	Alle fernen Sitzungen müssen sich der normalen Anmeldeprozedur unterziehen.
*SAMEPRF	Wenn Quellen- und Zielname des Benutzerprofils identisch sind, wird die Anmeldung möglicherweise umgangen und ferne Anmeldeversuche werden bearbeitet.
*REJECT	Es ist keine ferne Anmeldung zulässig.
*VERIFY	Nachdem geprüft wurde, ob der Benutzer zugriffsberechtigt ist, erlaubt das System eine Umgehung der Anmeldeprozedur.

9.11.33

Seite 2

QSCANFS

Dateisysteme prüfen
Dieser Systemwert gibt das integrierte Dateisystem an, dessen Objekte überprüft werden, wenn Exitprogramme mit einem der prüfungsbezogenen Exitpunkte des integrierten Dateisystems registriert werden.

Die prüfungsbezogenen IFS-Exitpunkte sind:
- QIBM_QP0L_SCAN_OPEN – IFS-Prüfung beim Öffnen des Exitprogramms
- QIBM_QP0L_SCAN_CLOSE – IFS-Prüfung beim Schließen des Exitprogramms

Weitere Informationen über Exitpunkte sind im Handbuch API-Themensammlung in der Kategorie Programmierung im IBM i Information Center unter http://www.ibm.com/systems/i/infocenter/ zu finden.

Anmerkung:

Zum Ändern dieses Systemwerts sind die Sonderberechtigungen

*ALLOBJ und *SECADM erforderlich.

Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist *ROOTOPNUD. Um die Überprüfung auszuschalten, *NONE angeben.

Gültige Werte sind:

*NONE	Es werden keine Objekte des integrierten Dateisystems überprüft.
*ROOTOPNUD	Objekte vom Typ *STMF, die sich in *TYPE2-Verzeichnissen der Root(/) befinden, QOpenSys und benutzerdefinierte Dateisysteme werden überprüft.

9.11.34

Seite 2

QSCANFSCTL

Prüfung der Dateisysteme steuern
Dieser Systemwert steuert die Überprüfung des integrierten Dateisystems auf dem System, wenn Exitprogramme mit einem der prüfungsbezogenen Exitpunkte des integrierten Dateisystems registriert werden.

Die prüfungsbezogenen IFS-Exitpunkte sind:
- QIBM_QP0L_SCAN_OPEN – IFS-Prüfung beim Öffnen des Exitprogramms
- QIBM_QP0L_SCAN_CLOSE – IFS-Prüfung beim Schließen des Exitprogramms

Weitere Informationen über Exitpunkte sind im Handbuch API-Themensammlung in der Kategorie Programmierung im IBM i Information Center unter http://www.ibm.com/systems/i/infocenter/ zu finden.

Anmerkung:

Zum Ändern dieses Systemwerts sind die Sonderberechtigungen *ALLOBJ und *SECADM erforderlich.

Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist *NONE.

Einen oder mehrere der folgenden Werte angeben. Bei Angabe von *NONE ist kein weiterer Wert mehr zulässig.

Gültige Werte sind:

*NONE	Es werden keine Steuerelemente für die prüfungsbezogen Exitpunkte des integrierten Dateisystems angegeben.
*ERRFAIL	Treten beim Aufrufen des Exitprogramms Fehler auf (z. B.: „Programm nicht gefunden“ oder „Exitprogramm meldet Fehler“), bricht das System die Anforderung ab, die den Programmaufruf ausgelöst hat. Wenn diese Option nicht angegeben wird, übergeht das System das Exitprogramm und behandelt es so, als würde das Objekt nicht überprüft.
*FSVRONLY	Es werden nur Zugriffe über die Dateiserver überprüft. Es werden z. B. sowohl Zugriffe über das Network File System als auch andere Dateiservermethoden überprüft. Wenn diese Option nicht angegeben wird, werden alle Zugriffe überprüft.

9.11.35
Seite 2

*NOFAILCLO	Anforderungen zum Schließen werden vom System nicht mit einem Prüffehler abgebrochen; das gilt auch dann, wenn das Objekt eine Überprüfung abgebrochen hat, die als Bestandteil der Abschlussoperation durchgeführt wurde. Dieser Wert überschreibt außerdem die *ERRFAIL-Spezifikation für die Abschlussverarbeitung, nicht jedoch für alle anderen prüfungsbezogenen Exitpunkte.
*NOPOSTRST	Nachdem Objekte zurückgespeichert wurden, werden sie nicht allein aus dem Grund überprüft, weil sie zurückgespeichert wurden. Wenn das Objektattribut besagt, dass „das Objekt nicht überprüft wird“, findet zu keiner Zeit eine Überprüfung statt. Wenn das Objektattribut besagt, dass „das Objekt nur überprüft wird, wenn es seit der letzten Überprüfung geändert wurde“, findet eine Überprüfung nur statt, falls es nach dem Zurückspeichern geändert wurde. Wenn *NOPOSTRST nicht angegeben wird, werden die Objekte nach dem Zurückspeichern mindestens ein Mal überprüft. Wenn das Objektattribut besagt, dass „das Objekt nicht überprüft wird“, findet nach dem Zurückspeichern einmal eine Überprüfung statt. Wenn das Objektattribut besagt, dass „das Objekt nur überprüft wird, wenn es seit der letzten Überprüfung geändert wurde“, findet nach dem Zurückspeichern eine Überprüfung statt, da das Zurückspeichern als Änderung am Objekt gilt. Generell kann es gefährlich sein, Objekte zurückzuspeichern, ohne sie nicht mindestens ein Mal zu überprüfen. Diese Option sollte nur dann verwendet werden, wenn feststeht, dass die Objekte vor der Sicherung überprüft wurden oder dass sie aus einer vertrauenswürdigen Quelle stammen.
*NOWRTUPG	Das System versucht nicht, einen Upgrade des Zugriffs für den an das Exitprogramm übermittelten Prüfdeskriptor durchzuführen, indem es Schreibzugriff erteilt. Wird diese Option nicht angegeben, versucht das System den Upgrade auf den Schreibzugriff.

*USEOCOATR	Das System verwendet die Spezifikation des Attributs „nur Objektänderung“, um das Objekt nur dann zu überprüfen, wenn es geändert wurde (d. h., wenn die Prüfsoftware einen Update meldet, findet keine zusätzliche Überprüfung statt.) Wenn diese Option nicht angegeben wird, wird das Attribut „nur Objektänderung“ nicht verwendet, d. h., das Objekt wird sowohl überprüft nachdem es geändert wurde, als auch wenn die Prüfsoftware einen Update meldet.

9.11.35

Seite 4

QSECURITY

Sicherheitsstufe des Systems
Dieser Systemwert gibt die Sicherheitsstufe im System an.

Eine Änderung dieses Werts wird beim nächsten IPL wirksam. Der Vorgabewert ist 40.

Gültige Werte sind:

20	Für die Anmeldung am System ist ein Kennwort erforderlich. Benutzer haben Zugriff auf alle Systemressourcen.
30	Für die Anmeldung am System ist ein Kennwort erforderlich; für den Zugriff auf Objekte und Systemressourcen benötigt der Benutzer die entsprechende Berechtigung.
40	Für die Anmeldung am System ist ein Kennwort erforderlich; für den Zugriff auf Objekte und Systemressourcen benötigt der Benutzer die entsprechende Berechtigung. Die Ausführung von Programmen schlägt fehl, wenn diese über nicht unterstützte Schnittstellen auf Objekte zuzugreifen versuchen.
50	Für die Anmeldung am System ist ein Kennwort erforderlich; für den Zugriff auf Objekte und Systemressourcen benötigt der Benutzer die entsprechende Berechtigung. Die Ausführung von Programmen schlägt fehl, wenn diese versuchen, nicht unterstützte Parameterwerte an unterstützte Schnittstellen zu übergeben oder über nicht unterstützte Schnittstellen auf Objekte zuzugreifen.

9.11.36

Seite 2

QSHRMEMCTL

Steuerung des gemeinsam benutzten Speichers
Steuert die Berechtigung von Benutzern zur Verwendung des gemeinsam genutzten Speichers (Shared Memory) bzw. des adressierten Speichers (Mapped Memory) mit Schreibmöglichkeit.

Anmerkung:

Zum Ändern dieses Systemwerts sind die Sonderberechtigungen *ALLOBJ und *SECADM erforderlich.

Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist 1.

Gültige Werte sind:

0	Benutzer können keinen gemeinsam genutzten Speicher (Shared Memory) bzw. keinen adressierten Speicher (Mapped Memory) mit Schreibmöglichkeit verwenden. Dieser Wert bedeutet, dass Benutzer keine Shared-Memory-APIs (z. B. shmat() – Shared Memory Attach API) und keine adressierten Speicherobjekte mit Schreibmöglichkeit (z. B. mmap() – Memory Map a File API) verwenden können. Dieser Wert sollte in Umgebungen mit höheren Sicherheitsanforderungen verwendet werden.
1	Benutzer können einen gemeinsam genutzten Speicher oder einen adressierten Speicher mit Schreibmöglichkeit verwenden. Dieser Wert bedeutet, dass Benutzer Shared-Memory-APIs (z. B. shmat() – Shared Memory Attach API) und adressierte Speicherobjekte mit Schreibmöglichkeit (z. B. mmap() – Memory Map a File API) verwenden können. Dieser Wert sollte in Umgebungen verwendet werden, in denen Zeiger von Programmen in verschiedenen Jobs gemeinsam genutzt werden können.

9.11.37

Seite 2

QSSLCSL

SSL-Chiffrierspezifikationsliste
Gibt die Liste der Cipher Suites an, die vom Secure Sockets Layer (SSL) des Systems unterstützt werden. Die Werte sind schreibgeschützt, es sei denn, der Systemwert QSSLCSLCTL (SSL-Chiffriersteuerung) ist auf *USRDFN gesetzt.

Weitere Informationen zu System-SSL und SSL-Chiffrierverfahren enthält der Abschnitt zu SSL unter Security Reference im IBM i Information Center unter http://www.ibm.com/systems/i/infocenter/.

Eine Änderung dieses Systemwerts wird für alle folgenden SSL-Systemsitzungen direkt wirksam. Die Vorgabwerte sind:

*RSA_AES_128_CBC_SHA,

*RSA_RC4_128_SHA,

*RSA_RC4_128_MD5,

*RSA_AES_256_CBC_SHA,

*RSA_3DES_EDE_CBC_SHA,

*RSA_DES_CBC_SHA,

*RSA_EXPORT_RC4_40_MD5,

*RSA_EXPORT_RC2_CBC_40_MD5,

*RSA_NULL_SHA und

*RSA_NULL_MD5.

Anmerkung:

Zur Änderung dieses Systemwerts sind die Sonderberechtigungen *IOSYSCFG, *ALLOBJ und *SECADM erforderlich.

Anmerkung:

Die Standard-Chiffrierspezifikationsliste wird vom System-SSL anhand der Reihenfolge der Werte in QSSLCSL sortiert. Die Einträge in der Standard-Chiffrierspezifikationsliste werden vom System definiert und können sich bei Releasegrenzen ändern. Wird ein Standardchiffrierverfahren aus den QSSLCSL-Ergebnissen gelöscht, wird das Chiffrierverfahren aus der Standardliste entfernt. Das Standardchiffrierverfahren wird der Standard-Chiffrierspezifikationsliste beim Einfügen in QSSLCSL hinzugefügt. Es ist nicht möglich, der Standardliste andere Chiffrierverfahren, die über die vom System für das Release definierte Gruppe hinausgehen, hinzuzufügen.

9.11.38
Seite 2

Ein Chiffrierverfahren kann QSSLCSL nicht hinzugefügt werden, wenn der erforderliche SSL-Protokollwert für die Cipher Suite nicht auch für den Systemwert QSSLPCL (SSL-Protokoll-Liste) eingestellt wurde.

Zum Hinzufügen einer Cipher Suite die Folgenummer und die Cipher Suite eingeben. Zum Ändern einer Cipher Suite die vorhandene Folgenummer und den Namen der Cipher Suite überschreiben. Zum Entfernen einer Cipher Suite die vorhandene Folgenummer und den Namen der Cipher Suite mit Leerzeichen überschreiben.

Gültige Werte sind:

*RSA_AES_128_CBC_SHA
Für das AES-Chiffrierverfahren (Advanced Encryption Standard = AES) mit Verkettung von Chiffrierblöcken (CBC) und 128-Bit-Schlüsseln werden die RSA-Verschlüsselungsalgorithmen verwendet. Zur Generierung von Nachrichtenauthentifizierungscodes (Message Authentication Code = MAC) wird der sichere Hash-Algorithmus (Safe Hash Algorithm = SHA) verwendet.

*RSA_RC4_128_SHA
Für das RC4-Chiffrierverfahren und die 128-Bit-Schlüssel werden die RSA-Verschlüsselungsalgorithmen verwendet. Zur Generierung von Nachrichtenauthentifizierungscodes (Message Authentication Code = MAC) wird der sichere Hash-Algorithmus (Safe Hash Algorithm = SHA) verwendet.

*RSA_RC4_128_MD5
Für das RC4-Chiffrierverfahren und die 128-Bit-Schlüssel werden die RSA-Verschlüsselungsalgorithmen verwendet. Zur Generierung von Nachrichtenauthentifizierungscodes (Message Authentication Code = MAC) wird der Nachrichtenauszugsalgorithmus 5 (Message Digest Algorithm 5 = MD5) verwendet.

*RSA_AES_256_CBC_SHA
Für das AES-Chiffrierverfahren (Advanced Encryption Standard = AES) mit Verkettung von Chiffrierblöcken (CBC) und 256-Bit-Schlüsseln werden die RSA-Verschlüsselungsalgorithmen verwendet. Zur Generierung von MAC-Codes wird SHA verwendet.

*RSA_3DES_EDE_CBC_SHA
Für das Triple DES-Chiffrierverfahren mit Verschlüsselung/Entschlüsselung/ Verschlüsselung (EDE) und Verkettung von Chiffrierblöcken (CBC) sowie die 168-Bit-Schlüssel werden die RSA-Verschlüsselungsalgorithmen verwendet. Zur Generierung von Nachrichtenauthentifizierungscodes (Message Authentication Code = MAC) wird der sichere Hash-Algorithmus (Safe Hash Algorithm = SHA) verwendet.

*RSA_DES_CBC_SHA
Für das DES-Chiffrierverfahren (Data Encryption Standard = DES) mit Verkettung von Chiffrierblöcken (CBC) und 56-Bit-Schlüsseln werden die RSA-Verschlüsselungsalgorithmen verwendet. Zur Generierung von MAC-Codes wird SHA verwendet.

*RSA_EXPORT_RC2_CBC_40_MD5
Für das RC2-Chiffrierverfahren mit Verkettung von Chiffrierblöcken (CBC) und 40-Bit-Schlüsseln werden die RSA-Verschlüsselungsalgorithmen verwendet. Zur Generierung von MAC-Codes wird der MD5-Algorithmus verwendet.

*RSA_EXPORT_RC4_40_MD5
Für das RC4-Chiffrierverfahren und die 40-Bit-Schlüssel werden die RSA-Verschlüsselungsalgorithmen verwendet. Zur Generierung von MAC-Codes wird der MD5-Algorithmus 5 verwendet.

*RSA_NULL_SHA
Hierfür werden RSA-Verschlüsselungsalgorithmen, aber keine Chiffrierverfahren verwendet. Zur Generierung von MAC-Codes wird SHA verwendet.

*RSA_NULL_MD5
Hierfür werden RSA-Verschlüsselungsalgorithmen, aber keine Chiffrierverfahren verwendet. Zur Generierung von MAC-Codes wird der MD5-Algorithmus verwendet.

*RSA_RC2_CBC_128_MD5
Für das RC2-Chiffrierverfahren mit Verkettung von Chiffrierblöcken (CBC) und 128-Bit-Schlüssel werden die RSA-Verschlüsselungsalgorithmen verwendet. Zur Generierung von MAC-Codes wird der MD5-Algorithmus verwendet.

*RSA_3DES_EDE_CBC_MD5
Für das Triple DES-Chiffrierverfahren mit Verschlüsselung/Entschlüsselung/Verschlüsselung (EDE) und Verkettung von Chiffrierblöcken (CBC) sowie die 168-Bit-Schlüssel werden die RSA-Verschlüsselungsalgorithmen verwendet. Zur Generierung von Nachrichtenauthentifizierungscodes (Message Authentication Code = MAC) wird der Nachrichtenauszugsalgorithmus 5 (Message Digest Algorithm 5 = MD5) verwendet.

*RSA_DES_CBC_MD5
Für das DES-Chiffrierverfahren (Data Encryption Standard = DES) mit Verkettung von Chiffrierblöcken (CBC) und 56-Bit-Schlüsseln werden die RSA-Verschlüsselungsalgorithmen verwendet. Zur Generierung von MAC-Codes wird der MD5-Algorithmus verwendet.

9.11.38

Seite 4

QSSLCSLCTL

SSL-Chiffriersteuerung
Gibt an, ob der Systemwert QSSLCSL (SSL-Chiffrierspezifikationsliste) vom System oder vom Benutzer gesteuert wird.

Weitere Informationen zu System-SSL und SSL-Chiffrierverfahren enthält der Abschnitt zu SSL unter Security Reference im IBM i Information Center unter http://www.ibm.com/systems/i/infocenter/.

Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist *OPSYS.

Anmerkung

Zur Änderung dieses Systemwerts sind die Sonderberechtigungen *IOSYSCFG, *ALLOBJ und *SECADM erforderlich.

Anmerkung

Beim Upgrade auf ein höheres Release werden zusätzliche Cipher Suite-Funktionen nicht automatisch hinzugefügt. Der Benutzer muss festlegen, welche neuen Cipher Suites verfügbar sind und diese dem Systemwert QSSLCSL (SSL-Chiffrierspezifikationsliste) manuell hinzufügen.

Gültige Werte sind:

*OPSYS	Auf den Systemwert QSSLCSL (SSL-Chiffrierspezifikationsliste) besteht nur Lesezugriff. Die im Systemwert QSSLCSL (SSL-Chiffrierspezifikationsliste) enthaltenen Werte werden automatisch so geändert, dass sie die Liste der für das betreffende Betriebssystemrelease unterstützten Cipher Suites enthalten.

Anmerkung:

Wenn ein höheres Release mit neuen Cipher Suite-Funktionen installiert wird, erlaubt *OPSYS, dass die Werte automatisch mit neueren und besseren Chiffrierverfahren aktualisiert werden.

9.11.39
Seite 2

*USRDFN	Der Systemwert QSSLCSL (SSL-Chiffrierspezifikationsliste) ist änderbar.

Anmerkung:

Beim Upgrade auf ein höheres Release werden zusätzliche Cipher Suite-Funktionen nicht automatisch hinzugefügt. Der Benutzer muss festlegen, welche neuen Cipher Suites verfügbar sind und diese dem Systemwert QSSLCSL (SSL-Chiffrierspezifikationsliste) manuell hinzufügen.

QSSLPCL

SSL-Protokolle
Gibt die SSL-Protokollversionen an, die vom System-SSL unterstützt werden.

Weitere Informationen zu System-SSL und SSL-Protokollen enthält der Abschnitt zu SSL unter Security Reference im IBM i Information Center unter http://www.ibm.com/systems/i/infocenter/.

Eine Änderung dieses Systemwerts wird für alle folgenden SSL-Systemsitzungen direkt wirksam. Der Vorgabewert ist *OPSYS.

Anmerkung

Zur Änderung dieses Systemwerts sind die Sonderberechtigungen *IOSYSCFG, *ALLOBJ und *SECADM erforderlich.

Gültige Werte sind:

*OPSYS	Welche SSL-Protokolle unterstützt werden, wird vom System bestimmt. Auf jedem Betriebssystemrelease können andere Protokolle unterstützt werden. Weitere Informationen zu den von einem bestimmten Release unterstützten Systemwerten enthält der Abschnitt zu SSL unter Security Reference im IBM i Information Center unter http://www.ibm.com/systems/i/infocenter/
*TLSV1	Transport Layer Security Version 1.0 wird unterstützt. Dieser Wert kann nicht angegeben werden, wenn der Wert *OPSYS angegeben wurde.
*SSLV3	Secure Sockets Layer Version 3.0 wird unterstützt. Dieser Wert kann nicht angegeben werden, wenn der Wert *OPSYS angegeben wurde.
*SSLV2	Secure Sockets Layer Version 2.0 wird unterstützt. Dieser Wert kann nicht angegeben werden, wenn der Wert *OPSYS angegeben wurde.

9.11.40

Seite 2

QUSEADPAUT

Übernommene Berechtigung verwenden
Definiert, welche Benutzer Programme mit dem Attribut *USEADPAUT(*YES) (übernommene Berechtigung verwenden) erstellen können. Jeder Benutzer mit der erforderlichen Berechtigung kann Programme oder Serviceprogramme mit diesem Attribut erstellen, ändern oder aktualisieren.

Der Standardwert für QUSEADPAUT ist *NONE. Jeder Benutzer mit der erforderlichen Berechtigung kann Programme oder Serviceprogramme mit diesem Attribut erstellen, ändern oder aktualisieren.

QUSEADPAUT kann auch den Namen einer Berechtigungsliste enthalten. Die Berechtigung des Benutzers wird mit dieser Liste verglichen. Verfügt der Benutzer mindestens über die Berechtigung *USE für die genannte Berechtigungsliste, kann er Programme oder Serviceprogramme mit dem Attribut USEADPAUT(*YES) erstellen, ändern oder aktualisieren. Diese Berechtigung kann nicht von der übernommenen Berechtigung stammen.

Ist die im Systemwert genannte Berechtigungsliste nicht vorhanden, wird die gewünschte Funktion nicht abgeschlossen. Es wird eine entsprechend lautende Nachricht ausgegeben. Werden mehrere Funktionen über den Befehl oder das API angefordert und fehlt die Berechtigungsliste, wird die Funktion nicht ausgeführt. Wurde versucht, einen der Befehle CRTPASPGM (Pascal-Programm erstellen) oder CRTBASPGM (Basic-Programm erstellen) auszuführen, als die Berechtigungsliste nicht gefunden wurde, so kommt es zu einer Funktionsprüfung.

Die gültigen Werte für QUSEADPAUT sind *NONE oder der Name einer Berechtigungsliste.

Eine Änderung des Systemwerts tritt in Kraft, wenn die Programme des Benutzers erstellt werden. Der Vorgabewert ist *NONE.

9.11.41

Seite 2

QVFYOBJRST

Objekt beim Zurückspeichern prüfen
Dieser Systemwert gibt das Verfahren an, das für die Prüfung der Objektsignatur beim Zurückspeichern verwendet werden soll. Er bezieht sich auf folgende Objektarten: *CMD, *PGM, *SRVPGM, *SQLPKG und *MODULE. Außerdem bezieht sich der Wert auf *STMF-Objekte, die Java-Programme enthalten. Dieser Wert gibt auch die Richtlinie für PTFs an, die für das System angelegt werden, einschließlich vorläufiger Korrekturen des lizenzierten internen Codes.

Ist Digital Certificate Manager nicht auf dem System installiert, dann werden bei Prüfung des Systemwerts für die genannten Objekte alle Objekte als unsigniert behandelt.

Eine Änderung dieses Systemwerts wird sofort wirksam. Der Vorgabewert ist 3.

Programm-, Serviceprogramm- und Modulobjekte, die auf einem System mit einem Release vor V6R1 erstellt wurden, werden als nicht signiert betrachtet, wenn sie auf einem System mit V6R1 oder einem höheren Release zurückgespeichert werden. Umgekehrt werden Programm-, Serviceprogramm- und Modulobjekte, die in einem System mit V6R1 oder einem höheren Release erstellt oder umgesetzt wurden, als nicht signiert betrachtet, wenn sie auf einem System mit einem Release vor V6R1 zurückgespeichert werden.

Wenn versucht wird, ein Objekt auf dem System zurückzuspeichern, arbeiten drei Systemwerte gemeinsam als Filter. Sie bestimmen, ob das Objekt zurückgespeichert werden darf oder ob es während der Rückspeicherung konvertiert wird. Der erste Filter ist der Systemwert QVFYOBJRST (Objekt beim Zurückspeichern prüfen). Dieser Systemwert steuert das Zurückspeichern einiger Objekte, die digital signiert werden können. Der zweite Filter ist der Systemwert QFRCCVNRST (Umsetzung beim Zurückspeichern erzwingen). Mit Hilfe dieses Systemwerts kann angegeben werden, ob während des Zurückspeicherns Programme, Serviceprogramme, SQL-Pakete und Modulobjekte konvertiert werden sollen oder nicht. Er kann auch verhindern, dass Objekte zurückgespeichert werden. Vom dritten Filter werden nur Objekte verarbeitet, die die ersten beiden Filter passiert haben. Der dritte Filter ist der Systemwert QALWOBJRST (Zurückspeichern des Objekts erlauben). Er gibt an, ob Objekte mit sicherheitsrelevanten Attributen zurückgespeichert werden können.

9.11.42
Seite 2

Gültige Werte sind:

1 Signaturen beim Zurückspeichern nicht prüfen. Objekte im Benutzerstatus unabhängig von der Signatur zurückspeichern.

Dieser Wert sollte nur dann verwendet werden, wenn signierte Objekte, deren Signaturprüfung aus einem akzeptablen Grund fehlgeschlagen ist, zurückgespeichert werden sollen. Allgemein gilt jedoch, dass das Zurückspeichern von Objekten mit ungültigen Signaturen auf dem System riskant ist. Die unten aufgeführten Hinweise sind zu beachten, bevor dieser Wert auf 1 gesetzt wird.

2 Signaturen beim Zurückspeichern prüfen. Nicht signierte Befehle und Objekte im Benutzerstatus zurückspeichern. Signierte Befehle und Objekte im Benutzerstatus auch dann zurückspeichern, wenn ihre Signaturen nicht gültig sind. Dieser Wert sollte nur verwendet werden, wenn bestimmte Objekte mit ungültigen Signaturen zurückgespeichert werden sollen. Allgemein gilt jedoch, dass das Zurückspeichern von Objekten mit ungültigen Signaturen auf dem System riskant ist.

3 Signaturen beim Zurückspeichern prüfen. Nicht signierte Befehle und Objekte im Benutzerstatus zurückspeichern. Signierte Befehle und Objekte im Benutzerstatus nur dann zurückspeichern, wenn die Signaturen gültig sind. Dieser Wert kann für normale Operationen verwendet werden, wenn zwar erwartet wird, dass einige der zurückzuspeichernden Objekte unsigniert sind, jedoch sichergestellt werden soll, dass alle signierten Objekte gültige Signaturen besitzen. Dies ist der Standardwert.

4 Signaturen beim Zurückspeichern prüfen. Nicht signierte Befehle und Objekte im Benutzerstatus nicht zurückspeichern. Signierte Befehle und Objekte im Benutzerstatus auch dann zurückspeichern, wenn ihre Signaturen nicht gültig sind. Dieser Wert sollte nur verwendet werden, wenn bestimmte Objekte mit ungültigen Signaturen zurückgespeichert werden sollen, jedoch ausgeschlossen werden soll, dass unsignierte Objekte zurückgespeichert werden. Allgemein

gilt jedoch, dass das Zurückspeichern von Objekten mit ungültigen Signaturen auf dem System riskant ist.

5 Signaturen beim Zurückspeichern prüfen. Nicht signierte Befehle und Objekte im Benutzerstatus nicht zurückspeichern. Signierte Befehle und Objekte im Benutzerstatus nur dann zurückspeichern, wenn die Signaturen gültig sind.
Dieser Wert ist der restriktivste und sollte verwendet werden, wenn nur die Objekte, die durch gesicherte Quellen signiert wurden, zurückgespeichert werden sollen.

Objekte mit dem Attribut „System“ und Objekte mit dem Attribut „Übernommen“ müssen gültige Signaturen einer vertrauenswürdigen Quelle besitzen. Für Objekte in LIC-PTFs (LIC = Lizenzierter Interner Code) ist ebenfalls die gültige Signatur einer vertrauenswürdigen Quelle erforderlich. Wenn diese Objekte keine gültige Signatur haben, können sie, unabhängig vom Systemwert QVFYOBJRST, nicht zurückgespeichert werden.

Einige Befehle verwenden eine Signatur, die nicht alle Komponenten des Objekts abdeckt. Einige Komponenten der Befehle werden nicht signiert, andere wiederum nur, wenn sie einen anderen als den Standardwert enthalten. Bei dieser Art von Signatur können Änderungen am Befehl vorgenommen werden, die die Signatur nicht ungültig machen. Zu diesen Änderungen gehören:

- Befehlsvoreinstellungen ändern
- Einem Befehl ohne Programm zur Gültigkeitsprüfung ein solches Programm hinzufügen.
- Den Parameter ‚Ausführung wo zulässig‘ ändern
- Den Parameter ‚Benutzer begrenzt zulassen‘ ändern

Der Benutzer kann diesen Befehlen auch seine eigene Signatur hinzufügen, die diese Bereiche des Befehlsobjekts abdeckt.

9.11.42

Seite 4

Benutzer-ID – Ihr Schlüssel zum System i

Bestimmt haben Sie sich auch schon mal mit falscher Benutzer-ID und/oder falschem Kennwort an Ihrem System i angemeldet: „Wie war das noch mal mit dem Kennwort, das vom Facebook-Account oder doch das vom Electronic-Banking? Nein, die Richtlinien sehen ja vor, dass ein Kennwort alle 120 Tage geändert werden muss. Das war doch vor dem Urlaub. Keine Ahnung mehr, aber es gibt ja noch den Systemadministrator, der in solchen Fällen gerufen wird.“ Gut, dass wir immer noch ein „Hintertürchen“ haben, um doch noch auf das System zugreifen zu können. Weil ohne System i geht bei vielen von uns im Büro nichts.

Benutzer-ID, User-ID, Account, es gibt eine Vielzahl von Bezeichnungen für das gleiche Merkmal, nämlich den Zugang zu unseren Computersystemen. Die Benutzer-ID ist das Merkmal, mit dem Sie als Individuum im System registriert sind, über das ihre Zugriffsmöglichkeiten auf Programme, Menüs, Daten, kurzum auf so ziemlich alles im System i gewährt – oder eben verweigert wird. Umso wichtiger ist es, mit Benutzer-IDs sorgsam umzugehen – oder würden Sie Ihren Mitarbeitern auch raten, den Haustürschlüssel – oder noch schlimmer – den Autoschlüssel einfach so unter die Tastatur ihres Bildschirms zu legen?

Nein, könnte uns nicht passieren? Schauen Sie doch mal nach! Die unrechtmäßige Verwendung von Benutzer-IDs ist ein Problem, dem sehr schwer beizukommen ist. Weiß man doch in der Regel nicht, wer sich ihrer bemächtigt hat, was der Grund hierfür war und was damit angestellt wird.

9.12
Seite 2

Rund um die Benutzer-ID

Die Benutzer-ID ist nur ein Steuerungsmerkmal der Zugriffsberechtigung. Wir alle wissen, dass das System i eine sehr ausgeprägte Sicherheit bietet. Von Haus aus wurde das von IBM so implementiert. Deshalb spielen neben der Benutzer-ID noch weitere Stellschrauben entscheidende Rollen.

Im vorangegangenen Kapitel wurden die sicherheitsrelevanten Systemwerte im Detail vorgestellt. Wenn Sie sich erinnern, fällt Ihnen zu Systemwerten wie QSECURITY, QMAXSIGN, QPWD… bestimmt einiges ein. Eine Benutzer-ID ist also nur so sicher wie das Umfeld, in das sie eingebettet ist. So können Sie gerne für jeden Benutzer eine ID festlegen, auch ein sehr komplexes Kennwort vergeben, aber wenn Sie den Systemwert QSECURITY auf „20“ gesetzt haben, nützt Ihnen das recht wenig, wenn Sie Benutzer davon abhalten wollen, bestimmte Anwendungen auszuführen oder auf bestimmte Daten zuzugreifen.

Auch ist es nicht unbedingt ratsam, beim Anlegen eines neuen Benutzers das Profil QSECOFR als Grundlage für die Erstellung einer Kopie zu verwenden. Zur Sicherheit gehört mehr, als einfach ein Benutzerprofil zu erstellen und die Berechtigungsstufe auf *USER zu setzen. Um die i5/OS-Sicherheitsmöglichkeiten umfassend zu nutzen, müssen Sie auch Begriffe wie „Gruppenprofile“, „übernommene Berechtigung“, „Objektberechtigung“, „OS/400-Systemsicherheitswerte“, „Netzwerksicherheit“ und vieles mehr verstehen.

Es ist einfach, zu behaupten, Ihr System sei sicher. Aber wie können Sie Gewissheit erlangen? Wenn Sie eine physische Datei erstellen und in dieser Datei Datensätze speichern, ist diese Datei dann automatisch sicher? Vermutlich nicht. Um für diese Datei echte Sicherheit zu gewährleisten, können Sie die Zugriffsstufe auf *PUBLIC *NONE einstellen und anschließend die Berechtigung für die Arbeit mit der Datei bestimmten Benutzerprofilen oder Sicherheitsberechtigungslisten zuweisen.

„Gruppenprofile“, „*PUBLIC *NONE“, „übernommene Berechtigung“? Für einige von Ihnen sind das noch Fremdworte, die im Laufe dieses Kapitels jedoch erklärt und in Zusammenhänge gebracht werden sollen.

9.12.1
Seite 2

Mit Benutzerprofilen arbeiten

Es gibt für einen Systemadministrator wohl keinen Befehl der ähnlich oft verwendet wird wie derjenige zum Bearbeiten von Benutzerprofilen. Das liegt wohl daran, dass dieser Befehl nicht nur zum Erstellen von Benutzerprofilen verwendet wird, sondern die einfachste Möglichkeit darstellt, gesperrte Kennwörter wieder freizugeben, Kennwörter für Benutzer zurückzusetzen etc. Mit WRKUSRPRF und F4 gelangen Sie zu den Parametern des Befehls.

```
                    Mit Benutzerprofilen arbeiten (WRKUSRPRF)

Auswahl eingeben und Eingabetaste drücken.

Benutzerprofil . . . . . . . . .          Name, generisch*, *ALL

                 Zusätzliche Parameter

Unterstützungsstufe  . . . . . .   *PRV      *PRV, *BASIC, *INTERMED...
```

WRKUSRPRF + F4

Hier können Sie die anzuzeigenden Benutzerprofile sowie die Unterstützungsstufe einschränken, um nur das/die Benutzerprofile anzuzeigen, die bearbeitet werden sollen.

Über die Unterstützungsstufe steuern Sie, welche Informationen Ihnen bei der Bearbeitung angezeigt werden sollen. Die Auswahl *BASIC z. B. zeigt Ihnen die Informationen in einer vereinfachten Darstellung an, bei der nur bestimmte Funktionen zur Verfügung stehen.

```
                        Mit Benutzerprofilen arbeiten

Auswahl eingeben und Eingabetaste drücken.
  1=Erstellen   2=Ändern   3=Kopieren   4=Löschen   5=Anzeigen
  12=Mit Objekten eines Eigners arbeiten

Ausw  Benutzerprofil  Text

       RENGEL         Robert Engel

                                                                Ende
Parameter für Auswahlmöglichkeiten 1, 2, 3, 4 und 5 oder Befehl
===>
F3=Verlassen   F5=Aktualisieren   F12=Abbrechen   F16=Neuer Listenanfang
F17=Listenanfang bei   F21=Unterstützungsstufe auswählen   F24=Weitere
```

Mit Benutzerprofilen arbeiten

9.12.2
Seite 2

Sie können hier einzelne Benutzerprofile bearbeiten, neue Profile erstellen, kopieren, löschen und anzeigen sowie mit den Objekten arbeiten, deren Eigner das ausgewählte Benutzerprofil ist.

In dieser Anzeige haben Sie zudem die Möglichkeit, die Unterstützungsstufe generell zu ändern. Mit der Funktionstaste F21 setzen Sie die Unterstützungsstufe für Ihr Benutzerprofil nicht nur einmalig, sondern permanent.

```
                Unterstützungsstufe auswählen

  Aktuelle Unterstützungsstufe:   Erweitert


  Auswahl eingeben und Eingabetaste drücken.

    Unterstützungsstufe . . . .   2    1=Basis
                                       2=Erweitert




  F1=Hilfetext    F12=Abbrechen
```

Unterstützungsstufe einstellen

Der Wert für die Unterstützungsstufe ist hier nur exemplarisch zu sehen, gilt aber für eine Vielzahl von WRKxxx-Anzeigen gleichermaßen. Die Unterstützungsstufe können Sie an mehreren Stellen ändern, z. B.

- in einem WRKxxx-Befehl über das Schlüsselwort *ASTLVL (Assistance Level) nur für das einmalige Arbeiten mit dem Befehl;
- in einem WRKxxx-Befehl mit F21 zur permanenten Einstellung der Unterstützungsstufe nur für diesen Befehl;
- im Benutzerprofil über den Parameter „Unterstützungsstufe“ für die Anpassung der Unterstützungsstufe für alle Befehle, die dieser Benutzer aufruft;
- im Systemwert QASTLVL für die generelle Voreinstellung, wie neu angelegte Benutzerprofile ohne spezielle Einstellungen Befehlsanzeigen präsentiert bekommen.

Für die Unterstützungsstufe gibt es generell drei Auswahlmöglichkeiten:

***BASIC**
Die OA-Ebene der Systemanzeigen ist verfügbar. D.h. diese Anzeige ist vorwiegend für ungeübte Anwender vorgesehen, die eine Basisansicht der Parameter benötigten, ohne durch die Vielfalt der Werte verwirrt zu werden.

***INTERMED**
Eine Zwischenstufe der Systemanzeigen ist verfügbar. Hiermit sehen Sie die Parameter detaillierter und können die Schlüsselwörter einblenden.

***ADVANCED**
Eine erweiterte Stufe der Systemanzeigen ist verfügbar. Dies ist nicht für alle Befehle möglich, stellt aber die Spezialistenansicht von Befehlen dar.

Da der Befehl „Mit Benutzern arbeiten“ nicht alle drei Unterstützungsstufen anbietet (wie übrigens eine Menge anderer Befehle auch), zeige ich Ihnen im Folgenden exemplarisch die Darstellung des Befehls WRKSYSVAL (Anzeige Systemstatus) mit den unterschiedlichen Unterstützungsstufen:

```
                          Systemstatus anzeigen                  RAZLEE
                                                        16.02.13  09:01:01

Plattenspeicherplatz:
  Systemspeicher (in 1.000.000 Byte) . . . . . :     57219
  Belegter Systemspeicher  . . . . . :     76,87%

Benutzer:
  Angemeldet . . . . . :         2
  Vorübergehend abgemeldet . . . . . :         0
  Durch Systemabfrage oder Gruppenjobs ausgesetzt  . . . . :         0
  Abgemeldet mit noch nicht gedruckter Druckausgabe  . . . :        41

Stapeljobs:
  Wartet auf Nachrichten . . . . . :         0
  Wird ausgeführt  . . . . . :        60
  Bei Ausführung angehalten  . . . . . :         0
  Wird beendet . . . . . :         0
                                                                Weitere ...
Eingabetaste --> Weiter

F1=Hilfetext     F3=Verlassen    F5=Aktualisieren    F9=Befehlszeile
F12=Abbrechen    F21=Unterstützungsstufe auswählen
```

*WRKSYSSTS – Unterstützungsstufe *BASIC*

9.12.2
Seite 4

```
                         Mit Systemstatus arbeiten                    RAZLEE
                                                          16.02.13  09:02:56
% CPU benutzt  . . . . . :        3,1    Zusatzspeicher:
Abgelaufene Zeit . . . . :   15:56:52      System-ASP . . . . . . :    57,21 G
Jobs im System . . . . . :       1231      % System-ASP benutzt . :    76,8724
% perm. Adressen . . . . :      0,007      Gesamtsumme  . . . . . :    57,21 G
% temp. Adressen . . . . :      0,016      Unges. Platz akt.
                                             belegt . . . . . . . :     3006 M
                                           Max. ungeschützt . . . :     3111 M
Änderungen (falls zulässig) eingeben und die Eingabetaste drücken.


System    Pool-    Reserv.    Max.  -DB-Seiten--  --Nicht-DB--
 Pool   Größe(M)  Größe (M)  Aktiv  fehl.  geles  fehl.  geles
   1     291,03    143,74   +++++    0,0    0,0    0,2    4,1
   2     683,03     24,52      42    0,1    3,9   13,9   47,6
   3    1058,67      8,63      52    0,0    0,0    0,0    0,0
   4      15,25      <.01       5    0,0    0,0    0,0    0,0
                                                                  Ende
Befehl
===>
F3=Verlassen    F4=Bedienerführung   F5=Neuanzeige   F9=Auffinden
F12=Abbrechen   F19=Erweiterter Systemstatus         F24=Weitere Tasten
```

*WRKSYSSTS – Unterstützungsstufe *INTERMED*

```
                         Mit Systemstatus arbeiten                    RAZLEE
                                                          16.02.13  09:03:23
% CPU benutzt  . . . . . :        3,1    System-ASP . . . . . . . :    57,21 G
Abgelaufene Zeit . . . . :   15:57:18    % System-ASP benutzt . . :    76,8725
Jobs im System . . . . . :       1231    Gesamtzusatzspeicher . . :    57,21 G
% perm. Adressen . . . . :      0,007    Unges. Platz akt. belegt :     3006 M
% temp. Adressen . . . . :      0,016    Max. ungeschützt . . . . :     3111 M


Sys-    Pool-    Reserv.   Max.  -DB-Seiten-  -Nicht-DB--  Aktiv  Wart.  Aktiv
Pool   Größe M   Größe M    akt  fehl. geles  fehl. geles  Wart.  n.wäh  n.wäh
  1     291,03    143,74  +++++    0,0   0,0    0,2   4,1   18,7    0,0    0,0
  2     683,03     24,54     42    0,1   3,9   13,9  47,6  393,0    0,0    0,0
  3    1058,67      8,63     52    0,0   0,0    0,0   0,0    0,1    0,0    0,0
  4      15,25      <.01      5    0,0   0,0    0,0   0,0    0,0    0,0    0,0




                                                                  Ende
===>
F21=Unterstützungsstufe auswählen
```

*WRKSYSSTS – Unterstützungsstufe *ADVANCED*

Sie sehen hier sehr schön, wie die Anzeige mit zunehmender Unterstützungsstufe ins Detail geht.

Doch dies nur am Rande, zurück zu den Benutzerprofilen.

Die Funktionen zum Erstellen, Ändern und Anzeigen von Benutzerprofilen fasse ich in der Folge unter „Benutzerprofile bearbeiten“ zusammen. Zum Kopieren lässt sich wenig sagen, das ist mehr oder weniger selbsterklärend.

Doch bereits beim Löschen von Benutzerprofilen ist Vorsicht geboten. Löschen Sie bitte nicht einfach ein Benutzerprofil eines Anwenders, der z. B. aus dem Unternehmen ausgeschieden ist, sondern schauen Sie sich die Optionen beim Löschen des Profils genau an. Mehr dazu später.

Die Anzeige der Objekte, deren Eigner (Owner) das ausgewählte Benutzerprofil ist, zeigt Ihnen etwas mehr über den Benutzer an, weil Sie alle Objekte sehen, die dieser Benutzer erstellt oder übernommen hat. Auch dazu später mehr.

9.12.2
Seite 6

Benutzer erstellen/ändern

Mit Auswahl 1 bzw. 2 können Sie ein Benutzerprofil erstellen bzw. ändern, mit Auswahl 5 zeigen Sie das Profil an.

Was wird am häufigsten gemacht, wenn ein neuer Mitarbeiter ins Unternehmen kommt und ein Benutzerprofil für den Zugriff auf das Power i System braucht? Richtig, man sucht sich das Profil eines Kollegen raus, vorzugsweise das des Vorgängers in der gleichen Position, übernimmt bzw. kopiert dessen Berechtigungen und vergibt einen neuen Namen, ein neues Kennwort und allenfalls noch eine neue Bezeichnung.

Das ist grundsätzlich in Ordnung, allerdings sollte man sich nach dem Kopieren das neue Benutzerprofil genau anschauen, damit unerwünschte Nebeneffekte – wie zusätzliche Berechtigungen, die der Basisbenutzer vielleicht im Lauf der Zeit über Gruppenprofile (was das ist, wird später erklärt) erhalten hat – nicht ungeprüft für seinen Nachfolger übernommen werden. Auch Sonderwerte des Benutzerprofils würden so vielleicht unerwünschter Weise kopiert.

Um mehr Sachbezug zu bekommen, verwende ich bei der Erklärung der Parameter nicht die Funktion zum Erstellen eines Benutzerprofils, sondern die Anzeige zum Ändern.

Vorher ging es schon mal um die Unterstützungsebene. Nachfolgend eine Übersicht der Funktionen zum Ändern eines Benutzerprofils mit unterschiedlichen Unterstützungsstufen. (Ich arbeite im Weiteren mit der Unterstützungsstufe *INTERMED).

```
                              Benutzer ändern

Benutzer  . . . . . :   RENGEL

Auswahl eingeben und Eingabetaste drücken.

Benutzerbeschreibung  . .   Robert Engel
Kennwort  . . . . . .   *SAME
Benutzerart . . . . . .   *SECOFR          Art, F4=Liste
Benutzergruppe  . . . . .   RAZUSR          Name, F4=Liste

Befehlszeilenverwendung
  einschränken  . . . . .   N               J=Ja,N=Nein

Standardbibliothek  . . .                   Name
Standarddrucker . . . . .   *WRKSTN         Name, *WRKSTN, F4=Liste
In Programm anmelden  . .   *NONE           Name, *NONE
  Bibliothek  . . . . . .                   Name

Erstes Menü . . . . . .   MAIN              Name
  Bibliothek  . . . . . .     *LIBL         Name
                                                                  Weitere ...
F1=Hilfetext    F3=Verlassen    F5=Aktualisieren    F12=Abbrechen
```

*Benutzerprofil mit Unterstützungsstufe *BASIC*

9.12.3
Seite 2

Die am häufigsten verwendete Unterstützungsstufe *INTERMED zeigt Ihnen alle Details eines Benutzerprofils an und ermöglicht auch die Anzeige der Schlüselwörter für die einzelnenen Parameter des Profils.

```
                    Benutzerprofil ändern (CHGUSRPRF)

Auswahl eingeben und Eingabetaste drücken.

Benutzerprofil . . . . . > RENGEL        Name
Benutzerkennwort . . . . . *SAME         Zeichenwert, *SAME, *NONE
Kennwort auf abgelaufen setzen *NO       *SAME, *NO, *YES
Status . . . . . *ENABLED    *SAME, *ENABLED, *DISABLED
Benutzerklasse . . . . . *SECOFR         *SAME, *USER, *SYSOPR...
Unterstützungsstufe  . . . . . *SYSVAL   *SAME, *SYSVAL, *BASIC...
Aktuelle Bibliothek  . . . . . *CRTDFT   Name, *SAME, *CRTDFT
Aufzurufendes Startprogramm  . . *NONE   Name, *SAME, *NONE
  Bibliothek . . . . . .                 Name, *LIBL, *CURLIB
Anfangsmenü  . . . . . MAIN              Name, *SAME, *SIGNOFF
  Bibliothek . . . . . .   *LIBL        Name, *LIBL, *CURLIB
Möglichkeiten einschränken . . . *NO     *SAME, *NO, *PARTIAL, *YES
Text 'Beschreibung'  . . . . . 'Robert Engel'


                                                                    Ende
F3=Verlassen    F4=Bedienerf.    F5=Aktualisieren    F10=Zusätzl. Parameter
F12=Abbrechen   F13=Verwendung der Anzeige    F24=Weitere Tasten
```

*Benutzerprofil mit Unterstützungsstufe *INTERMED*

Mit der Funktionstaste F11 können Sie sich die Schlüsselwörter der Parameter anzeigen lassen, die dazu dienen, später auch mal eine Änderung eines Benutzerprofils ohne den Weg über das Benutzerinterface vornehmen zu können.

```
                    Benutzerprofil ändern (CHGUSRPRF)

Auswahl eingeben und Eingabetaste drücken.

Benutzerprofil . . . . . USRPRF     >   RENGEL
Benutzerkennwort . . . . . PASSWORD      *SAME
Kennwort auf abgelaufen setzen   PWDEXP  *NO
Status . . . . . STATUS      *ENABLED
Benutzerklasse . . . . . USRCLS      *SECOFR
Unterstützungsstufe  . . . . . ASTLVL    *SYSVAL
Aktuelle Bibliothek  . . . . . CURLIB    *CRTDFT
Aufzurufendes Startprogramm  . . INLPGM  *NONE
  Bibliothek . . . . . .
Anfangsmenü  . . . . . INLMNU            MAIN
  Bibliothek . . . . . .                   *LIBL
Möglichkeiten einschränken . . . LMTCPB  *NO
Text 'Beschreibung'  . . . . . TEXT      'Robert Engel'


                                                                    Ende
F3=Verlassen    F4=Bedienerf.    F5=Aktualisieren    F10=Zusätzl. Parameter
F12=Abbrechen   F13=Verwendung der Anzeige    F24=Weitere Tasten
```

*Benutzerprofil mit Unterstützungsstufe *INTERMED und Anzeige der Schlüsselwörter*

Wenn Sie ein Benutzerprofil aufrufen, um es zu ändern, erhalten Sie zunächst eine Seite angezeigt (wie oben beim Benutzerprofil mit *INTERMED). Das ist aber beileibe nicht alles. Nur ein kleiner Teil wird auf der ersten Seite angezeigt. Auf die Vielfalt der Möglichkeiten können Sie mit Hilfe der Funktionstasten F9 und F10 zugreifen. F10 blendet dabei zusätzliche Parameter ein, F9 zeigt alle Parameter – auch diejenigen, die abhängig von Eingaben vorheriger Parameter sind – und eignet sich für eine Übersicht über alle Möglichkeiten, die eingegeben werden können.

So ein Benutzerprofil bietet eine Menge Einstellungsmöglichkeiten, mit denen unterschiedliche Bereiche der Darstellung und Sicherheit angepasst werden können. Schauen wir uns das mal im Detail an.

```
                    Benutzerprofil ändern (CHGUSRPRF)

Auswahl eingeben und Eingabetaste drücken.

Benutzerprofil . . . . . > RENGEL          Name
Benutzerkennwort . . . . . *SAME           Zeichenwert, *SAME, *NONE
Kennwort auf abgelaufen setzen *NO         *SAME, *NO, *YES
Status . . . . . *ENABLED      *SAME, *ENABLED, *DISABLED
Benutzerklasse . . . . . *SECOFR       *SAME, *USER, *SYSOPR...
Unterstützungsstufe  . . . . . *SYSVAL     *SAME, *SYSVAL, *BASIC...
Aktuelle Bibliothek  . . . . . *CRTDFT     Name, *SAME, *CRTDFT
Aufzurufendes Startprogramm  . . *NONE     Name, *SAME, *NONE
  Bibliothek . . . . . .                   Name, *LIBL, *CURLIB
Anfangsmenü  . . . . . MAIN           Name, *SAME, *SIGNOFF
  Bibliothek . . . . . .   *LIBL          Name, *LIBL, *CURLIB
Möglichkeiten einschränken . . *NO         *SAME, *NO, *PARTIAL, *YES
Text 'Beschreibung'  . . . . . 'Robert Engel'
```

Benutzerprofil – Seite 1

Benutzerprofil

Dieser Parameter gibt das Benutzerprofil an, dessen Werte geändert werden sollen. Es kann ein numerischer Benutzerprofilname angegeben werden. Ein numerischer Benutzerprofilname muss mit dem Buchstaben Q beginnen.

Es können verschiedene Namenskonventionen realisiert werden, die von unternehmensinternen Regeln abhängig sein können. So vergeben manche kleinere Unternehmen Benutzerprofilnamen, bei denen an erster Stelle Vornamen und Nachnamen stehen, weil hierbei die Wahrscheinlichkeit doppelter Namen geringer ist, als wenn nur der Nachname oder nur der Vorname für das Benutzerprofil verwendet wird. Andere Unternehmen verwenden Nummerierungen (A15, B27 etc.) oder Benutzernamen, die Niederlassungsbezeichnungen mit beinhalten (MUCAMEIER, HAMBBECK etc.), die eine Zuordnung zur Filiale über den Benutzernamen ermöglichen und bei generischen Regeln die Verwaltung vereinfachen.

Grundsätzlich sind Sie in der Vergabe der Benutzerprofilnamen relativ frei (siehe unten), sollten aber – wenn möglich – Benutzernamen und Kennworte für iSeries-Systeme denen aus der Windows- oder Unix-Welt anpassen, um einen einfacheren Zugriff über Systemgrenzen hinweg zu ermöglichen.

Die folgenden von IBM gelieferten Objekte sind für diesen Parameter nicht gültig, weil es sich um reservierte Namen handelt:
QANZAGENT, QAUTPROF, QCLUMGT, QCLUSTER, QCOLSRV, QDBSHR, QDBSHRDO, QDFTOWN, QDIRSRV, QDLFM, QDOC, QDSNX, QFNC, QGATE, QIBMHELP, QIPP, QLPAUTO, QLPINSTALL, QMGTC, QMSF, QNTP, QPEX, QPM400, QSNADS, QSPL, QSPLJOB, QSRVAGT, QSYS, QTCP, QTFTP, QTSTRQS, QYCMCIMOM, QYPSJSVR.

Benutzerkennwort

Dieser Parameter gibt das Kennwort an, mit dem sich ein Benutzer am System anmeldet. Das Kennwort ist einem Benutzerprofil zugeordnet und wird zur Darstellung des Benutzers im System verwendet. Das Kennwort sollte nur dem jeweiligen Benutzer bekannt sein. Es kann ein numerisches Kennwort angegeben werden.

Arbeitet das System mit Kennwortstufe 0 oder 1 und ist das Kennwort numerisch, so muss das Kennwort mit Q beginnen, z. B. Q1234. 1234 ist das Kennwort für die Anmeldung am System.

Die Kennwortstufe wird mit dem Systemwert QPWDLVL (Kennwortstufe) gesteuert.

Das neue Kennwort wird nicht anhand der Kennwortgültigkeitsregeln überprüft. Die Kennwortgültigkeitsregeln werden durch die Systemwerte (IBM i) definiert. Eine Beschreibung der Kennwortgültigkeitsregeln ist im Handbuch „System i Security Reference“, IBM Form SC41-5302, enthalten.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***NONE**
Diesem Benutzerprofil ist kein Kennwort zugeordnet. Benutzer können sich nicht mit einem Profil, für das PASSWORD(*NONE) angegeben ist, an einem System anmelden.

Wenn das System auf Kennwortstufe 0 oder 1 läuft, muss eine alphanumerische Zeichenfolge von maximal 10 Zeichen angegeben werden. Das erste Zeichen muss ein Buchstabe sein, die anderen Zeichen müssen alphanumerisch sein.

Wenn das System auf Kennwortstufe 2 oder 3 läuft, muss eine alphanumerische Zeichenfolge von maximal 128 Zeichen angegeben werden. Auf Kennwortstufe 2 oder 3 muss für das Kennwort Groß-/Kleinschreibung beachtet werden. Ist für die lokale Kennwortverwaltung (LCLPWDMGT) *NO angegeben, wird das lokale IBM i-Kennwort auf *NONE gesetzt, so dass für den Benutzer dieselben Einschränkungen gelten wie bei Angabe von *NONE für das Kennwort. Der angegebene Kennwortwert wird an andere IBM-Produkte oder -Lösungen gesendet, die eine Kennwortsynchronisation durchführen (siehe z. B. Integration von IBM i mit BladeCenter und System x unter http://www.ibm.com/systems/i/bladecenter/). Weitere Informationen zur Verwaltung der Kennwörter für das Produkt – wenn LCLPWDMGT(*NO) für das Benutzerprofil angegeben ist – sind in der entsprechenden Dokumentation zu finden.

Kennwort auf „abgelaufen" setzen (PWDEXP)

Dieser Parameter gibt an, ob das Kennwort eines Benutzers auf „abgelaufen" gesetzt ist. Ist das der Fall, muss der Benutzer das Kennwort ändern, um sich am System anmelden zu können. Beim Anmelden am System erscheint die Anzeige mit den Anmeldeinformationen, die auch eine Auswahl zum Ändern des Kennworts enthält.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***NO**
Das Kennwort ist nicht auf „abgelaufen" gesetzt.

***YES**
Das Kennwort ist auf „abgelaufen" gesetzt.

Status (STATUS)

Dieser Parameter gibt den Status eines Benutzerprofils an. Das System deaktiviert ein Benutzerprofil, wenn Folgendes eintritt:

Das Benutzerprofil erreicht die maximal zulässige Anzahl an Kennwortüberprüfungen. Die maximale Anzahl von Kennwortüberprüfungen wurde erreicht, wenn die Anzahl der erfolglosen Kennwortüberprüfungsversuche die im Systemwert QMAXSIGN angegebene Grenze erreicht und im Systemwert QMAXSGNACN Auswahl 2 oder 3 angegeben wurde.

Ablaufdatum für Benutzerprofil wurde erreicht. Mit dem Befehl DSPUSRPRF (Benutzerprofil anzeigen) kann das Ablaufdatum für ein Benutzerprofil angezeigt werden.

Der Parameter STATUS ist im Befehl CRTUSRPRF (Benutzerprofil erstellen) oder im Befehl CHGUSRPRF (Benutzerprofil ändern) auf *DISABLED gesetzt.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***ENABLED**
Das Benutzerprofil berechtigt zur Anmeldung.

***DISABLED**
Das Benutzerprofil berechtigt erst dann zur Anmeldung, wenn es von einem berechtigten Benutzer erneut aktiviert wird. Stapeljobs können auch unter einem gesperrten Benutzerprofil übergeben werden.

Benutzerklasse (USRCLS)

Der Parameter „Benutzerklasse“ gibt die Art des Benutzers an, der einem Benutzerprofil zugeordnet ist: Sicherheitsbeauftragter, Sicherheitsadministrator, Programmierer, Systembediener oder Benutzer. Die Benutzerklasse steuert die im Menü gezeigten Auswahlmöglichkeiten. Sonderberechtigungen werden nur erteilt, wenn *USRCLS für den Parameter „Sonderberechtigung“ (SPCAUT) angegeben wird. Wurde SPCAUT(*USRCLS) angegeben, variieren die Sonderberechtigungen abhängig vom QSECURITY-Wert.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***USER**
Auf QSECURITY-Stufe 10 oder 20 verfügt der Benutzer über die Berechtigungen *ALLOBJ und *SAVSYS.

Auf QSECURITY-Stufe 30 oder höher hat der Benutzer keine Sonderberechtigungen.

***SECOFR**
Auf allen Sicherheitsstufen verfügt der Sicherheitsbeauftragte über die folgenden Sonderberechtigungen:

*ALLOBJ

*SAVSYS

*JOBCTL

*SERVICE

*SPLCTL

*SECADM

*AUDIT

*IOSYSCFG

***SECADM**
Auf QSECURITY-Stufe 10 oder 20 verfügt der Sicherheitsadministrator über die Sonderberechtigungen *ALLOBJ, *SAVSYS, *SECADM und *JOBCTL.

Auf QSECURITY-Stufe 30 oder höher verfügt der Benutzer über die Sonderberechtigungen *SECADM.

***PGMR**
Auf QSECURITY-Stufe 10 oder 20 verfügt der Programmierer über die Sonderberechtigungen *ALLOBJ, *SAVSYS und *JOBCTL.

Auf QSECURITY-Stufe 30 oder höher hat der Benutzer keine Sonderberechtigungen.

***SYSOPR**
Auf QSECURITY-Stufe 10 oder 20 verfügt der Systembediener über die Sonderberechtigungen *ALLOBJ, *SAVSYS und *JOBCTL.

Auf QSECURITY-Stufe 30 oder höher verfügt der Benutzer über die Sonderberechtigungen *SAVSYS und *JOBCTL.

Unterstützungsstufe (ASTLVL)

Dieser Parameter gibt an, welche Unterstützungsstufe verwendet werden soll.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***SYSVAL**
Die im Systemwert QASTLVL definierte Unterstützungsstufe wird verwendet.

***BASIC**
Die einfache Schnittstelle für die Oberfläche von Anwendungen wird verwendet.

***INTERMED**
Die Systemschnittstelle wird benutzt.

***ADVANCED**
Die Systemschnittstelle für Experten wird verwendet. Damit mehr Einträge in der Liste angezeigt werden können, werden Auswahl- und Funktionstasten nicht angezeigt. Verfügt ein Befehl nicht über eine Schnittstelle für Experten (*ADVANCED), wird die normale Schnittstelle (*INTERMED) verwendet.

Aktuelle Bibliothek (CURLIB)

Dieser Parameter gibt den Namen der aktuellen Bibliothek an, die dem gerade durchgeführten Job zugeordnet ist. Beziehungsweise gibt dieser Parameter den Namen der Bibliothek an, die als aktuelle Bibliothek für einen Benutzer verwendet werden soll. Wenn *PARTIAL oder *YES für den Parameter „Möglichkeiten einschränken“ (LMTCPB) des Befehls CRTUSRPRF (Benutzerprofil erstellen) oder CHGUSRPRF (Benutzerprofil ändern) angegeben ist, kann der Benutzer die aktuelle Bibliothek weder beim Anmelden noch mit dem Befehl CHGPRF (Profil ändern) ändern.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***CRTDFT**
Der Benutzer hat keine aktuelle Bibliothek. Die Bibliothek QGPL wird standardmäßig als aktuelle Bibliothek benutzt.

Name
Den Namen der Bibliothek angeben, die als aktuelle Bibliothek für den Benutzer verwendet werden soll.

Aufzurufendes Startprogramm (INLPGM)

Dieser Parameter gibt bei interaktiven Jobs das Programm an, das aufgerufen wird, wenn ein neuer Leitwegschritt mit QCMD als Verarbeitungsprogramm der Anforderung gestartet wird. Wenn *PARTIAL oder *YES für den Parameter „Möglichkeiten einschränken“ (LMTCPB) angegeben ist, kann der Programmwert weder beim Anmelden noch mit dem Befehl CHGPRF (Profil ändern) geändert werden. Es können keine Parameter an das Programm übergeben werden.

Der Name einer IBM System /36-Umgebungsprozedur kann als Startprogramm angegeben werden, wenn die Prozedur eine Teildatei der Datei QS36PRC (in der Bibliotheksliste oder der angegebenen Bibliothek) ist und eine der folgenden Bedingungen wahr ist:

– *S36 wurde im Parameter SPCENV angegeben.

– Im Parameter SPCENV wurde *SYSVAL angegeben und der Systemwert QSPCENV lautet *S36.

Gültige Einzelwerte sind

***SAME**
Der Wert ändert sich nicht.

***NONE**
Nach Anmeldung des Benutzers wird kein Programm aufgerufen. Wurde für den Parameter Anfangsmenü (INLMNU) ein Menüname angegeben, wird dieses Menü angezeigt.

Qualifikationsmerkmal 1: Aufzurufendes Startprogramm

Name
Den Namen des Programms angeben, das nach der Anmeldung des Benutzers aufgerufen wird.

Qualifikationsmerkmal 2: Bibliothek

***LIBL**
Alle Bibliotheken in der Bibliotheksliste des aktuellen Threads werden durchsucht, bis die erste Übereinstimmung gefunden wird.

***CURLIB**
Das Programm wird in der aktuellen Bibliothek für den Job gesucht. Wenn keine Bibliothek als aktuelle Jobbibliothek angegeben ist, wird QGPL verwendet.

Name
Den Namen der Bibliothek angeben, in der sich das Startprogramm befindet.

Anfangsmenü (INLMNU)

Dieser Parameter gibt das Anfangsmenü an, das angezeigt wird, wenn sich ein Benutzer am System anmeldet und das Leitwegprogramm des Benutzers der Befehlsprozessor QCMD ist. Wenn *YES für den Parameter „Möglichkeiten einschränken“ (LMTCPB) angegeben ist, kann der Benutzer das Menü weder beim Anmelden noch mit dem Befehl CHGPRF (Profil ändern) ändern.

Ein Menü der IBM System /36-Umgebung kann als Anfangsmenü angegeben werden, wenn eine der folgenden Bedingungen wahr ist:

– *S36 ist für den Parameter „Sonderumgebung“ (SPCENV) angegeben.

– Im Parameter SPCENV wurde *SYSVAL angegeben und der Systemwert QSPCENV lautet *S36.

Gültige Einzelwerte sind

***SAME**
Der Wert ändert sich nicht.

***SIGNOFF**
Das System meldet den Benutzer nach Beendigung des Programms ab. Dieser Wert ist für Benutzer vorgesehen, die nur zur Ausführung des Programms berechtigt sind.

Qualifikationsmerkmal 1: Anfangsmenü

Name
Den Namen des Anfangsmenüs angeben, das nach der Anmeldung des Benutzers am System aufgerufen wird.

Qualifikationsmerkmal 2: Bibliothek

***LIBL**
Alle Bibliotheken in der Bibliotheksliste des aktuellen Threads werden durchsucht, bis die erste Übereinstimmung gefunden wird.

***CURLIB**
Zur Lokalisierung des Menüs wird die aktuelle Bibliothek des Jobs verwendet. Wenn keine Bibliothek als aktuelle Jobbibliothek angegeben ist, wird QGPL verwendet.

Name
Den Namen der Bibliothek angeben, in der sich das Anfangsmenü befindet.

Möglichkeiten einschränken (LMTCPB)

Dieser Parameter gibt an, bis zu welcher Grenze ein Benutzer das Programm, das Menü, die aktuelle Bibliothek und die Werte des Programms für die Abruftaste steuern kann. Es wird außerdem festgelegt, ob der Benutzer Befehle über die Befehlszeile ausführen kann. Auf Sicherheitsstufe 10 wird der Parameter ignoriert.

Beim Erstellen oder Ändern der Benutzerprofile anderer Benutzer können in diesem Parameter keine Werte angegeben werden, die anderen Benutzern umfangreichere Möglichkeiten einräumen als im eigenen Benutzerprofil definiert sind. Ist zum Beispiel im eigenen Benutzerprofil *PARTIAL beim Parameter „Möglichkeiten einschränken“ (LMTCPB) angegeben, kann für andere Benutzer *PARTIAL oder *YES angegeben werden. Die Angabe *NO für einen anderen Benutzer ist nicht zulässig.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***NO**
Die Werte für Programm, Menü und aktuelle Bibliothek können bei der Anmeldung eines Benutzers im System geändert werden. Der Benutzer kann mit Hilfe des Befehls CHGPRF (Profil ändern) das Startprogramm, das Anfangsmenü, die aktuelle Bibliothek bzw. das Programm für die Abruftaste in seinem eigenen Benutzerprofil ändern. Befehle können über die Befehlszeile ausgeführt werden.

***PARTIAL**
In der Anmeldeanzeige können die Werte für Programm und aktuelle Bibliothek nicht geändert werden. Das Menü kann geändert werden und Befehle können über die Befehlszeile ausgeführt werden. Das Menü kann mit dem Befehl CHGPRF (Profil ändern) geändert werden. Das Programm, die aktuelle Bibliothek und das Programm für die Abruftaste können mit dem Befehl CHGPRF nicht geändert werden.

***YES**
In der Anmeldeanzeige können die Werte für Programm, Menü und aktuelle Bibliothek nicht geändert werden. Es können keine Befehle über die Befehlszeile oder durch Auswahl in einem Befehlsgruppenmenü, wie z. B. CMDADD, ausgeführt werden, Befehle können aber nach wie vor über die Befehlseingabeanzeige ausgeführt werden. Der Benutzer kann das Programm, das Menü, die aktuelle Bibliothek oder das Programm für die Abruftaste mit dem Befehl CHGPRF (Profil ändern) nicht ändern.

Text ‚Beschreibung' (TEXT)

Dieser Parameter betrifft den Text, mit dem das Objekt kurz beschrieben wird.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***BLANK**
Es wird kein Text angegeben.

Beschreibung
Maximal 50 Zeichen Text in Hochkommas angeben.

```
                    Benutzerprofil ändern (CHGUSRPRF)

Auswahl eingeben und Eingabetaste drücken.


                Zusätzliche Parameter

Sonderberechtigung . . . . . .   ALLOBJ       *SAME, *USRCLS, *NONE...
                                 *AUDIT
                                 *IOSYSCFG
                                 *JOBCTL
                                 *SAVSYS
                                 *SECADM
                                 *SERVICE
                                 *SPLCTL
Sonderumgebung . . . . . .       *SYSVAL       *SAME, *SYSVAL, *NONE, *S36
Anmeldeinformationen anzeigen .  *SYSVAL       *SAME, *NO, *YES, *SYSVAL
Intervall für Kennwortablauf . .  *SYSVAL      1-366, *SAME, *SYSVAL, *NOMAX
Kennwortänderung blockieren  . .  *SYSVAL      1-99, *SAME, *SYSVAL, *NONE
Lokale Kennwortverwaltung  . . .  *YES         *SAME, *YES, *NO
```

Benutzerprofil – Seite 2

9.12.3
Seite 14

Sonderberechtigung (SPCAUT)

Dieser Parameter gibt die Sonderberechtigungen eines Benutzers an. Sonderberechtigungen sind zum Ausführen bestimmter Funktionen im System erforderlich. Sie können vielen der vom System bereitgestellten Benutzerprofile, einschließlich QSECOFR und QSYS, nicht entzogen werden.

Normalerweise werden folgende Sonderberechtigungen erteilt:

- Die Sonderberechtigung zur Systemsicherung (*SAVSYS) für Benutzer, die für die Systembedienung zuständig sind.
- Die Sonderberechtigung zur E/A-Systemkonfiguration (*IOSYSCFG) für Benutzer, die für das Ändern der Konfiguration von E/A-Einheiten des Systems zuständig sind.
- Der Benutzer erhält die Sonderberechtigung zur Jobsteuerung (*JOBCTL). Mit dieser Berechtigung kann der Benutzer sämtliche Jobs, die im System ausgeführt werden oder die sich in einer Jobwarteschlange oder Ausgabewarteschlange befinden, für die OPRCTL (*YES) angegeben ist, ändern, anzeigen, anhalten, freigeben, abbrechen oder löschen. Der Benutzer verfügt außerdem über die Berechtigung zum Laden des Systems, zum Starten von Ausgabeprogrammen und zum Stoppen aktiver Subsysteme.
- Die Sonderberechtigung des Sicherheitsadministrators (*SECADM) zur Erstellung, Änderung oder Löschung von Benutzerprofilen.
- Die Sonderberechtigung für alle Objekte (*ALLOBJ) für Benutzer, die mit Systemressourcen arbeiten.
- Die Servicesonderberechtigung (*SERVICE) für Benutzer, die Servicefunktionen ausführen.
- Die Sonderberechtigung zur Spool-Steuerung (*SPLCTL) für Benutzer, die alle den Spool-Betrieb betreffenden Funktionen ausführen.
- Die Sonderberechtigung *AUDIT für Benutzer, die Protokollierungsfunktionen ausführen.

Einschränkungen:

- Das Benutzerprofil, das ein anderes Benutzerprofil erstellt oder ändert, muss über alle zu vergebenden Sonderberechtigungen verfügen. Ein Benutzer muss über alle Sonderberechtigungen verfügen, damit er einem anderen Benutzerprofil alle Sonderberechtigungen erteilen kann.
- Ein Benutzer muss bei Verwendung des Befehls CRTUSRPRF über die Sonderberechtigungen *ALLOBJ und *SECADM verfügen, um einem anderen Benutzer die Sonderberechtigung *SECADM erteilen zu können.
- Ein Benutzer muss bei Verwendung des Befehls CHGUSRPRF über die Sonderberechtigungen *ALLOBJ, *SECADM und *AUDIT verfügen, um einem anderen Benutzer die Sonderberechtigung *AUDIT erteilen zu können.

Gültige Einzelwerte sind:

***SAME**
Der Wert ändert sich nicht.

***USRCLS**
Einem Benutzer werden abhängig von dem für den Parameter „Benutzerklasse“ (USRCLS) angegebenen Wert Sonderberechtigungen erteilt.

***NONE**
Einem Benutzer werden keine Sonderberechtigungen erteilt.

Weitere Werte

***ALLOBJ**
Einem Benutzer wird die Berechtigung für alle Objekte erteilt. Er kann auf alle Systemressourcen zugreifen, unabhängig davon, ob er über persönliche Berechtigungen verfügt oder nicht.

***AUDIT**
Einem Benutzer wird die Protokollierungsberechtigung erteilt, mit der Protokollierungsfunktionen ausgeführt werden können. Protokollierungsfunktionen schließen das Ein- und Ausschalten der Protokollierung für das System sowie die Steuerung der Protokollierungsstufe für ein Objekt oder einen Benutzer ein.

***JOBCTL**
Einem Benutzer wird die Berechtigung zur Jobsteuerung erteilt. Mit dieser Berechtigung kann der Benutzer sämtliche Jobs, die im System ausgeführt werden oder die sich in einer Job- oder Ausgabewarteschlange befinden, für die OPRCTL (*YES) angegeben ist, ändern, anzeigen, anhalten, freigeben, abbrechen oder löschen. Der Benutzer hat auch die Berechtigung, Ausgabeprogramme zu starten und aktive Subsysteme anzuhalten.

***SAVSYS**
Einem Benutzerprofil wird die Berechtigung zur Systemsicherung erteilt. Diese Berechtigung erlaubt dem Benutzer, alle Objekte im System zu sichern, zurückzuspeichern und den von diesen Objekten belegten Speicherplatz freizugeben, unabhängig davon, ob er über die Objektverwaltungsberechtigung verfügt oder nicht.

***IOSYSCFG**
Einem Benutzer wird die Berechtigung für E/A-Systemkonfigurationen erteilt. Damit erhält er die Berechtigung zum Ändern der Konfiguration von E/A-Einheiten des Systems.

***SECADM**
Einem Benutzer wird die Berechtigung des Sicherheitsadministrators erteilt. Er kann Benutzerprofile erstellen, ändern oder löschen, wenn er für die Befehle CRTUSRPRF (Benutzerprofil erstellen), CHGUSRPRF (Benutzerprofil ändern) und DLTUSRPRF (Benutzerprofil löschen) sowie das Benutzerprofil berechtigt ist. Diese Berechtigung befugt nicht zur Erteilung von Sonderberechtigungen, über die dieses Benutzerprofil nicht selbst verfügt. Um einem anderen Benutzer die Sonderberechtigung *SECADM erteilen zu können, muss ein Benutzer die Sonderberechtigungen *ALLOBJ und *SECADM haben.

***SERVICE**
Einem Benutzer wird die Serviceberechtigung erteilt. Er darf Servicefunktionen ausführen.

***SPLCTL**
Einem Benutzer wird die Berechtigung zur Spool-Steuerung erteilt. Er darf alle Spool-Funktionen ausführen.

Sonderumgebung (SPCENV)

Dieser Parameter gibt die Sonderumgebung an, in welcher der Benutzer nach der Anmeldung arbeitet.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***SYSVAL**
Der Systemwert QSPCENV wird verwendet, um die Systemumgebung nach der Anmeldung des Benutzers im System festzulegen.

***NONE**
Der Benutzer arbeitet nach der Anmeldung am System in der IBM i-Systemumgebung.

***S36**
Der Benutzer arbeitet nach der Anmeldung im System in der Umgebung IBM System /36.

Anmeldeinformationen anzeigen (DSPSGNINF)

Dieser Parameter gibt an, ob die Anzeige „Anmeldungsinformationen“ angezeigt wird.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***SYSVAL**
Mit dem Systemwert QDSPSGNINF kann festgelegt werden, ob die Anzeige „Anmeldeinformationen“ aufgerufen wird.

***NO**
Die Anzeige mit den Anmeldeinformationen wird nicht aufgerufen.

***YES**
Die Anzeige mit den Anmeldeinformationen wird aufgerufen.

Intervall für Kennwortablauf (PWDEXPITV)

Dieser Parameter gibt an, in wie viel Tagen das Kennwort verfällt.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***SYSVAL**
Mit dem Systemwert QPWDEXPITV kann der Zeitraum festgelegt werden, nach dem das Kennwort verfällt.

***NOMAX**
Das Kennwort wird nicht ungültig.

1-366
Die Anzahl der Tage seit der letzten Kennwortänderung bis zum Verfall des aktuellen Kennworts angeben. Gültig sind Werte von 1 bis 366.

Kennwortänderung blockieren (PWDCHGBLK)

Dieser Parameter gibt den Zeitraum an, in dem ein Kennwort nach einer erfolgreichen Kennwortänderung blockiert ist, d. h., nicht geändert werden darf. Mithilfe dieses Werts kann verhindert werden, dass Benutzer dasselbe abgelaufene Kennwort erneut benutzen, indem sie ihr Kennwort wiederholt ändern, um wieder den abgelaufenen Kennwortwert zu erhalten (dies würde der Absicht des Systemwerts QPWDRQDDIF widersprechen). Der Parameter verhindert jedoch nicht, dass der Systemadministrator einen Befehl wie CHGUSRPRF (Benutzerprofil ändern) zum Ändern eines Kennworts verwenden kann.

Außerdem verhindert dieser Parameter nicht, dass ein Benutzer sein Profilkennwort ändern kann, wenn der Wert für PWDEXP *YES ist. Dies ermöglicht dem Sicherheitsadministrator, ein Benutzerprofil mit einem abgelaufenen Kennwort zu erstellen und dem Benutzer zu erlauben, sich einmal anzumelden und das Kennwort zu ändern, ohne dass er durch den Wert zum Blockieren der Kennwortänderung daran gehindert würde.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***SYSVAL**
Mit dem Systemwert QPWDCHGBLK kann der Wert für „Kennwortänderung blockieren“ festgelegt werden.

***NONE**
Das Kennwort kann jederzeit geändert werden.

1-99
Gibt die Anzahl Stunden an, die ein Benutzer nach einer erfolgreichen Kennwortänderung warten muss, bevor er das Kennwort erneut ändern kann.

Lokale Kennwortverwaltung (LCLPWDMGT)

Dieser Parameter gibt an, ob das Kennwort eines Benutzerprofils lokal verwaltet werden soll.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***YES**
Das Kennwort wird auf dem lokalen System verwaltet.

***NO**
Das Kennwort wird nicht auf dem lokalen System verwaltet. Bei Angabe dieses Werts wird das lokale IBM i-Kennwort auf *NONE gesetzt. Der im Kennwortparameter angegebene Kennwortwert wird an andere IBM-Produkte oder -Lösungen gesendet, die eine Kennwortsynchronisation durchführen (siehe z. B. Integration von IBM i mit BladeCenter und System x unter http://www.ibm.com/systems/i/bladecenter/).

Benutzer können ihr eigenes Kennwort in diesem Fall mit dem Befehl CHGPWD (Kennwort ändern) nicht ändern. Sie können sich außerdem nicht direkt am System anmelden.

Die Angabe dieses Werts wirkt sich auf andere IBM-Produkte oder -Lösungen aus, die eine Kennwortsynchronisation durchführen (siehe z. B. Integration von IBM i mit BladeCenter und System x unter http://www.ibm.com/systems/i/bladecenter/). Einzelheiten sind in der Dokumentation zu dem Produkt oder der Lösung zu finden.

Dieser Wert sollte verwendet werden, wenn ein Benutzer über eine andere Plattform, wie z. B. Windows, auf das System zugreifen muss.

```
                    Benutzerprofil ändern (CHGUSRPRF)

Auswahl eingeben und Eingabetaste drücken.

Einheitensitzungen begrenzen . .   *SYSVAL        *SAME, *SYSVAL, *YES, *NO...
Tastaturpufferung  . . . . . .     *SYSVAL        *SAME, *SYSVAL, *NO...
Maximal zulässiger Speicher  . .   *NOMAX         Kilobyte, *SAME, *NOMAX
Höchste Planungspriorität  . . .   3             0-9, *SAME
Jobbeschreibung  . . . . . .       QDFTJOBD       Name, *SAME
  Bibliothek . . . . . .             QGPL         Name, *LIBL, *CURLIB
Gruppenprofil  . . . . . .         RAZUSR         Name, *SAME, *NONE
Eigner . . . . . .                 *USRPRF        *SAME, *USRPRF, *GRPPRF
Gruppenberechtigung  . . . . .     *NONE          *SAME, *NONE, *ALL...
Art der Gruppenberechtigung  . .   *PRIVATE       *PRIVATE, *PGP, *SAME
Zusätzliche Gruppenprofile . . .   *NONE          Name, *SAME, *NONE
     + für weitere Werte
Berechnungscode  . . . . . .       *BLANK
Dokumentkennwort . . . . . .       *SAME          Name, *SAME, *NONE
Nachricht- warteschl.  . . . . .   RENGEL         Name, *SAME, *USRPRF
  Bibliothek . . . . . .             QUSRSYS      Name, *LIBL, *CURLIB
```

Benutzerprofil – Seite 3

9.12.3
Seite 20

Einheitensitzungen begrenzen (LMTDEVSSN)

Dieser Parameter gibt an, ob die Anzahl der für einen Benutzer zulässigen Einheitensitzungen begrenzt werden soll. Davon sind jedoch nicht SYSREQ oder ein zweites Anmelden betroffen.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***SYSVAL**
Mit dem Systemwert QLMTDEVSSN wird festgelegt, ob ein Benutzer nur eine begrenzte Anzahl Einheitensitzungen anmelden kann.

***NO**
Der Benutzer ist nicht auf eine bestimmte Anzahl von Einheitensitzungen begrenzt.

***YES**
Der Benutzer kann nur eine Einheitensitzung anmelden.

0
Der Benutzer ist nicht auf eine bestimmte Anzahl von Einheitensitzungen begrenzt. Dieser Wert hat dieselbe Bedeutung wie *NO.

1
Der Benutzer kann nur eine Einheitensitzung anmelden. Dieser Wert hat dieselbe Bedeutung wie *YES.

2-9
Der Benutzer wird auf die angegebene Anzahl von Einheitensitzungen begrenzt.

Tastaturpufferung (KBDBUF)

Dieser Parameter gibt den Tastaturpufferungswert an, der verwendet wird, wenn für ein Benutzerprofil ein Job initialisiert wird. Wenn die Eingabepufferung aktiviert ist, können die Tastenanschläge gepuffert werden. Wenn die Abruftastenpufferung aktiviert ist, wird die Abruftaste wie jede andere Taste gepuffert. Ist sie nicht aktiviert, wird die Abruftaste nicht gepuffert und selbst dann an das System gesendet, wenn bei dem Datensichtgerät die Eingabe gesperrt ist. Dieser Wert kann auch durch eine Benutzeranwendung gesetzt werden. Weitere Informationen enthält das Handbuch „API-Themensammlung“ in der Kategorie „Programmierung“ im IBM i Information Center unter http://www.ibm.com/systems/i/infocenter/.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***SYSVAL**
Der Systemwert QKBDBUF wird verwendet, um den Tastaturpufferungswert zu bestimmen.

***NO**
Die Eingabepufferung und die Abruftastenpufferung werden nicht aktiviert.

***TYPEAHEAD**
Die Eingabepufferung wird aktiviert, die Abruftastenpufferung jedoch nicht.

***YES**
Die Eingabepufferung und die Abruftastenpufferung werden aktiviert.

Maximal zulässiger Speicher (MAXSTG)

Dieser Parameter gibt die maximale Größe des Zusatzspeichers (in Kilobyte) an, der einem Benutzerprofil zur Speicherung von permanenten Objekten zugeordnet wird. Wird das Maximum überschritten, wenn ein interaktiver Benutzer versucht, ein Objekt zu erstellen, wird eine Fehlernachricht angezeigt und das Objekt wird nicht erstellt. Wird das Maximum bei der Erstellung eines Objekts in einem Stapeljob überschritten, wird eine Fehlernachricht an das Jobprotokoll gesendet (abhängig von der Protokollierungsstufe des Jobs) und das Objekt wird nicht erstellt.

Der Speicher wird in Segmenten von jeweils 4 KB zugeordnet. Wird z. B. MAXSTG(9) angegeben, werden dem Profil 12 KB Speicher zugeordnet.

Bei der Planung des maximalen Speicherplatzes für Benutzerprofile müssen folgende Systemaktionen berücksichtigt werden:

Bei einer Wiederherstellungsoperation wird dem Benutzer, der die Operation durchführt, Speicherplatz zugeordnet und dann das Objekt an den Eigner übertragen. Bei einer Wiederherstellung mit großem Datenvolumen muss MAXSTG(*NOMAX) angegeben werden.

Einem Benutzerprofil, das einen Journalempfänger erstellt, wird der erforderliche Speicherplatz entsprechend dem größer werdenden Empfänger zugeordnet. Werden neue Empfänger mit JRNRCV(*GEN) erstellt, wird der Speicherplatz weiterhin demjenigen Benutzerprofil zugeordnet, dem der aktive Journalempfänger gehört. Handelt es sich um einen sehr aktiven Journalempfänger, muss MAXSTG(*NOMAX) angegeben werden.

Benutzerprofile, die erstellte Objekte an ihr Gruppenprofil übertragen, müssen über entsprechend großen Speicherplatz für die Aufnahme der Objekte verfügen, bevor diese an das Gruppenprofil übertragen werden.

Dem Eigner der Bibliothek wird Speicherplatz für die Beschreibungen der Objekte zugeordnet, die in der Bibliothek gespeichert sind, selbst wenn die Objekte einem anderen Benutzerprofil gehören. Beispiele für solche Objekte sind Text- und Programmreferenzen.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***NOMAX**
Diesem Benutzerprofil wird der erforderliche Speicherplatz zugewiesen.

Größe
Den maximalen Speicherplatz für den Benutzer in Kilobyte angeben.

Höchste Planungspriorität (PTYLMT)

Dieser Parameter gibt die höchste Planungspriorität an, die ein Benutzer bei jedem dem System zu übergebenden Job haben darf. Dieser Wert steuert die Jobverarbeitungspriorität sowie die Ausgabepriorität für jeden Job, der unter diesem Benutzerprofil ausgeführt wird. Das bedeutet, dass die für die Parameter JOBPTY und OUTPTY in einem beliebigen Jobbefehl angegebenen Werte den PTYLMT-Wert des Benutzerprofils, unter dem der Job ausgeführt wird, nicht überschreiten können. Der Wert der Planungspriorität liegt zwischen 0 und 9, wobei 0 die höchste und 9 die niedrigste Priorität bezeichnet.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

0-9
Einen Wert zwischen 0 und 9 für die höchste Planungspriorität angeben, die dem Benutzer erlaubt ist.

Jobbeschreibung (JOBD)

Dieser Parameter gibt die Jobbeschreibung an, die für diejenigen Jobs verwendet wird, die über Subsystemdatenstationseinträge gestartet werden. Falls die Jobbeschreibung beim Erstellen oder Ändern eines Benutzerprofils noch nicht vorhanden ist, muss ein Bibliotheksqualifikationsmerkmal angegeben werden, da der Jobbeschreibungsname im Benutzerprofil gespeichert ist.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

Qualifikationsmerkmal 1: Jobbeschreibung

Name
Den Namen der Jobbeschreibung angeben, die für die Datenstationseinträge verwendet wird, deren Jobbeschreibungsparameterwerte auf den Benutzer JOBD(*USRPRF) hinweisen.

Qualifikationsmerkmal 2: Bibliothek

***LIBL**
Alle Bibliotheken in der Bibliotheksliste des aktuellen Threads werden durchsucht, bis die erste Übereinstimmung gefunden wird.

***CURLIB**
Die aktuelle Bibliothek für den Thread wird durchsucht. Ist keine aktuelle Bibliothek für den Thread angegeben, wird QGPL verwendet.

Name
Den Namen der Bibliothek angeben, die durchsucht werden soll.

9.12.3
Seite 24

Gruppenprofil (GRPPRF)

Dieser Parameter gibt den Namen des Gruppenprofils eines Benutzers an, dessen Berechtigung verwendet wird, wenn dem Benutzer keine spezielle Berechtigung erteilt wurde. Der aktuelle Benutzer dieses Befehls muss über die Objektverwaltungsberechtigung (*OBJMGT) sowie die Änderungsberechtigung (*CHANGE) für das Profil verfügen, das für den Parameter Gruppenprofil (GRPPRF) angegeben ist. Die Berechtigung *OBJMGT kann nicht durch eine Programmübernahmeoperation erworben werden.

Wenn ein Gruppenprofil angegeben wird, erhält der Benutzer automatisch die Berechtigungen *CHANGE und *OBJMGT für das Gruppenprofil.

Die folgenden von IBM gelieferten Objekte sind für diesen Parameter nicht gültig.

QANZAGENT, QAUTPROF, QCLUMGT, QCLUSTER, QCOLSRV, QDBSHR, QDBSHRDO, QDFTOWN, QDIRSRV, QDLFM, QDOC, QDSNX, QEJB, QFNC, QGATE, QIBMHELP, QIPP, QLPAUTO, QLPINSTALL, QLWISVR, QMGTC, QMSF, QNETSPLF, QNFSANON, QNTP, QPEX, QPM400, QRJE, QSNADS, QSPL, QSPLJOB, QSRV, QSRVAGT, QSRVBAS, QSYS, QTCM, QTCP, QTFTP, QTSTRQS, QWEBADMIN, QWSERVICE, QYCMCIMOM, QYPSJSVR.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***NONE**
Das Benutzerprofil verfügt nicht über ein Gruppenprofil.

Name
Den Namen des mit einem Benutzerprofil zu verwendenden Gruppenprofils angeben.

Eigner (OWNER)

Dieser Parameter gibt an, welches Benutzerprofil der Eigner der Objekte werden soll, die von einem Benutzer erstellt werden.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***USRPRF**
Das zum Job gehörige Benutzerprofil ist Eigner des Objekts.

***GRPPRF**
Das Gruppenprofil wird zum Eigner der neu erstellten Objekte und erhält alle Berechtigungen für das Objekt. Das dem Job zugeordnete Benutzerprofil hat keine spezielle Berechtigung für das Objekt. Bei Angabe von *GRPPRF muss für den Parameter Gruppenprofil (GRPPRF) der Name eines Benutzerprofils angegeben werden und der Parameter Gruppenberechtigung (GRPAUT) kann nicht angegeben werden.

Gruppenberechtigung (GRPAUT)

Dieser Parameter gibt die spezielle Berechtigung an, die einem Gruppenprofil für neu erstellte Objekte erteilt wird. Bei Angabe von *GRPPRF für den Parameter „Eigner“ (OWNER) ist eine Angabe dieses Parameters nicht zulässig.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***NONE**
Es wird keine Gruppenberechtigung erteilt.

***ALL**
Der Benutzer kann alle Operationen ausführen. Ausgenommen sind auf den Eigner beschränkte Operationen oder Operationen, die durch die Berechtigung zum Verwalten der Berechtigungsliste gesteuert werden. Der Benutzer kann die Objektexistenz steuern, die Sicherheit für das Objekt angeben, das Objekt ändern und allgemeine Funktionen für das Objekt durchführen. Der Benutzer kann außerdem das Eigentumsrecht für das Objekt ändern.

***CHANGE**
Der Benutzer kann alle Operationen ausführen, mit Ausnahme derjenigen, die auf den Eigner beschränkt sind oder von der Objektexistenzberechtigung (*OBJEXIST) sowie der Objektverwaltungsberechtigung (*OBJMGT) gesteuert werden. Der Benutzer kann das Objekt ändern und allgemeine Funktionen für das Objekt ausführen. Die Änderungsberechtigung (*CHANGE) schließt die Objektverwendungsberechtigung (*OBJOPR) sowie alle Datenberechtigungen ein. Falls das Objekt eine Berechtigungsliste ist, kann der Benutzer weder andere Benutzer hinzufügen noch Benutzer ändern oder entfernen.

***USE**
Der Benutzer kann grundlegende Operationen für das Objekt vornehmen, also z. B. ein Programm ausführen oder eine Datei lesen. Der Benutzer kann das Objekt nicht ändern. Die Berechtigung *USE stellt die Objektverwendungsberechtigung *OBJOPR, die Leseberechtigung *READ sowie die Ausführungsberechtigung *EXECUTE bereit.

***EXCLUDE**
Der Benutzer kann nicht auf das Objekt zugreifen.

Art der Gruppenberechtigung (GRPAUTTYP)

Dieser Parameter gibt die Art der Berechtigung an, die einem Gruppenprofil für neu erstellte Objekte erteilt werden soll. Bei Angabe von *NONE für den Parameter Gruppenberechtigung (GRPAUT) wird dieser Parameter ignoriert.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***PRIVATE**
Das Gruppenprofil erhält persönliche Berechtigung für neu erstellte Objekte, wobei der Berechtigungswert durch den Parameter GRPAUT bestimmt wird. Lautet der im Parameter GRPAUT angegebene Berechtigungswert *NONE, wird dieser Wert ignoriert.

***PGP**
Das Gruppenprofil wird zur Primärgruppe für neu erstellte Objekte, wobei der Berechtigungswert durch den Parameter GRPAUT bestimmt wird. Lautet der im Parameter GRPAUT angegebene Berechtigungswert *NONE, wird dieser Wert ignoriert.

Zusätzliche Gruppenprofile (SUPGRPPRF)

Dieser Parameter gibt die zusätzlichen Gruppenprofile eines Benutzers an. Diese Profile werden zusammen mit dem für den Parameter Gruppenprofil (GRPPRF) angegebenen Gruppenprofil verwendet, um die Berechtigung eines Benutzers festzulegen, wenn keine spezielle Benutzerberechtigung für den Job besteht. Wenn Profile für diesen Parameter angegeben werden, muss im Parameter GRPPRF ein Gruppenprofilname für das Benutzerprofil vorhanden sein, der entweder in diesem Befehl oder einem früheren Befehl CRTUSRPRF (Benutzerprofil erstellen) oder im Befehl CHGUSRPRF (Benutzerprofil ändern) angegeben wurde. Der aktuelle Benutzer des Befehls muss über die Objektverwaltungsberechtigung (*OBJMGT) sowie die Änderungsberechtigung (*CHANGE) für die Profile verfügen, die für diesen Parameter angegeben sind. Die Berechtigung *OBJMGT kann nicht durch eine Programmübernahmeoperation erworben werden.

Wenn ein Gruppenprofil angegeben wird, erhält der Benutzer automatisch die Berechtigungen *CHANGE und *OBJMGT für das Gruppenprofil.

Die folgenden von IBM gelieferten Benutzerprofile sind für diesen Parameter nicht gültig:

QANZAGENT, QAUTPROF, QCLUMGT, QCLUSTER, QCOLSRV, QDBSHR, QDBSHRDO, QDFTOWN, QDIRSRV, QDLFM, QDOC, QDSNX, QEJB, QFNC, QGATE, QIBMHELP, QIPP, QLPAUTO, QLPINSTALL, QLWISVR, QMGTC, QMSF, QNETSPLF, QNFSANON, QNTP, QPEX, QPM400, QRJE, QSNADS, QSPL, QSPLJOB, QSRV, QSRVAGT, QSRVBAS, QSYS, QTCM, QTCP, QTFTP, QTSTRQS, QWEBADMIN, QWSERVICE, QYCMCIMOM, QYPSJSVR.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***NONE**
Mit diesem Benutzerprofil werden keine zusätzlichen Gruppenprofile verwendet.

Name
Maximal 15 Gruppenprofilnamen angeben, die zusammen mit dem Benutzerprofil sowie dem im Parameter GRPPRF angegebenen Gruppenprofil bestimmen, ob ein Job Zugriff auf bestehende Objekte sowie Sonderberechtigung erhält.

9.12.3
Seite 28

Berechnungscode (ACGCDE)

Dieser Parameter gibt den diesem Benutzerprofil zugeordneten Berechnungscode an.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***BLANK**
Dem Benutzerprofil wird ein aus 15 Leerzeichen bestehender Berechnungscode zugeordnet.

Zeichenwert
Den aus 15 Zeichen bestehenden Berechnungscode für Jobs angeben, die den Berechnungscode des Benutzerprofils erhalten. Werden weniger als 15 Zeichen angegeben, wird die Zeichenfolge nach rechts mit Leerzeichen aufgefüllt.

Dokumentkennwort (DOCPWD)

Dieser Parameter gibt das Dokumentkennwort an, das Benutzern der DIA-Dokumentverteilungsservices ermöglicht, persönliche Verteilungen vor dem Zugriff von Personen zu schützen, die in ihrem Auftrag arbeiten.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***NONE**
Der Benutzer verwendet kein Dokumentkennwort.

Name
Das Dokumentkennwort angeben, das dem Benutzer zugeordnet werden soll. Das Kennwort muss 1 bis 8 alphanumerische Zeichen (Buchstaben A bis Z und Ziffern 0 bis 9) lang sein. Das erste Zeichen des Dokumentkennworts muss alphabetisch sein, die übrigen Zeichen können alphanumerisch sein. Eingebettete und führende Leerzeichen sowie Sonderzeichen sind unzulässig.

Nachrichtenwarteschlange (MSGQ)

Dieser Parameter gibt die Nachrichtenwarteschlange an, an die Nachrichten gesendet werden.

Anmerkung:

Die Nachrichtenwarteschlange wird erstellt, falls sie noch nicht besteht. Das für den Parameter Benutzerprofil (USRPRF) angegebene Benutzerprofil ist der Eigner der Nachrichtenwarteschlange.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***USRPRF**
Eine Nachrichtenwarteschlange mit demselben Namen wie die für den Parameter USRPRF angegebene wird als Nachrichtenwarteschlange für den Benutzer verwendet. Die Nachrichtenwarteschlange befindet sich in der Bibliothek QUSRSYS.

Qualifikationsmerkmal 1: Nachrichtenwarteschlange

Name
Den Namen der Nachrichtenwarteschlange angeben, die mit dem Profil verwendet werden soll.

Qualifikationsmerkmal 2: Bibliothek

***LIBL**
Alle Bibliotheken in der Bibliotheksliste des aktuellen Threads werden durchsucht, bis die erste Übereinstimmung gefunden wird.

***CURLIB**
Die aktuelle Bibliothek für den Thread wird durchsucht. Ist keine aktuelle Bibliothek für den Thread angegeben, wird QGPL verwendet.

Name
Den Namen der Bibliothek angeben, die durchsucht werden soll.

9.12.3
Seite 30

```
                    Benutzerprofil ändern (CHGUSRPRF)

Auswahl eingeben und Eingabetaste drücken.

Zustellung . . . . . .   *NOTIFY       *SAME, *NOTIFY, *BREAK...
Bewertungscodefilter . . . . .   0           0-99, *SAME
Druckeinheit . . . . . .   *WRKSTN       Name, *SAME, *WRKSTN, *SYSVAL
Ausgabewarteschlange . . . . .   *WRKSTN       Name, *SAME, *WRKSTN, *DEV
  Bibliothek . . . . . .                 Name, *LIBL, *CURLIB
Abrufprogramm  . . . . . .   *SYSVAL       Name, *SAME, *SYSVAL...
  Bibliothek . . . . . .                 Name, *LIBL, *CURLIB
Sortierfolge . . . . . .   *SYSVAL       Name, *SAME, *SYSVAL, *HEX...
  Bibliothek . . . . . .                 Name, *LIBL, *CURLIB
Sprachen-ID  . . . . . .   *SYSVAL       *SAME, *SYSVAL...
Landes- oder Regions-ID  . . . .   *SYSVAL       *SAME, *SYSVAL...
Zeichensatz-ID . . . . . .   *SYSVAL       *SAME, *SYSVAL, *HEX...
Steuerung für Zeichen-ID . . . .   *SYSVAL       *SAME, *SYSVAL, *DEVD...
Jobattr. länderspez. Angaben . .   *SYSVAL       *SAME, *SYSVAL, *NONE...
     + für weitere Werte
```

Benutzerprofil – Seite 4

Zustellung (DLVRY)

Dieser Parameter gibt an, wie die für einen Benutzer an die Nachrichtenwarteschlange zu sendenden Nachrichten zugestellt werden sollen.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***NOTIFY**
Der Job, dem die Nachrichtenwarteschlange zugeordnet ist, wird verständigt, wenn eine Nachricht in der Nachrichtenwarteschlange eintrifft. Bei interaktiven Jobs an einer Datenstation ertönt der akustische Alarm (sofern aktiviert), und der Anzeiger für eine anstehende Nachricht leuchtet auf. Wenn die Nachrichtenwarteschlange auch von anderen Jobs benutzt wird, kann der Zustellungsmodus nicht in *NOTIFY geändert werden.

***HOLD**
Die Nachrichten bleiben in der Nachrichtenwarteschlange, bis sie vom Benutzer oder einem Programm angefordert werden.

***BREAK**
Der Job, dem die Nachrichtenwarteschlange zugeordnet ist, wird unterbrochen, sobald eine Nachricht in der Nachrichtenwarteschlange eintrifft. Bei interaktiven Jobs ertönt der akustische Alarm (sofern aktiviert). Wird die Nachrichtenwarteschlange auch von anderen Jobs benutzt, kann der Zustellungsmodus nicht in *BREAK geändert werden.

***DFT**
Dieser Standardwert wird auf die Anfragenachricht gesendet. Ist in der Nachrichtenbeschreibung der Anfragenachricht keine Standardantwort angegeben, wird die Systemstandardantwort *N verwendet.

Bewertungscodefilter (SEV)

Dieser Parameter gibt den niedrigsten Bewertungscode an, mit dem Nachrichten im Durchbruch- oder Hinweismodus an einen Benutzer gesendet werden. Bei Nachrichten, die mit einem niedrigeren Bewertungscode als dem für diesen Parameter angegebenen Bewertungscode in die Nachrichtenwarteschlange gestellt werden, wird weder der Job unterbrochen, noch ertönt ein Signalton, noch leuchtet der Anzeiger für anstehende Nachrichten. Die Nachrichten verbleiben in der Warteschlange, bis sie mit dem Befehl DSPMSG (Nachricht anzeigen) angezeigt werden. Wenn *BREAK oder *NOTIFY beim Parameter „Zustellung" (DLVRY) angegeben wurde und bei Erhalt der Nachricht in der Nachrichtenwarteschlange aktiv ist, wird die Nachricht zugestellt, wenn der der Nachricht zugeordnete Bewertungscode gleich oder größer dem hier angegebenen Wert ist. Andernfalls wird die Nachricht in der Warteschlange zurückgehalten bis sie angefordert wird.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

0-99
Einen Bewertungscode von 00 bis 99 angeben.

Druckeinheit (PRTDEV)

Dieser Parameter gibt die Standarddruckereinheit für einen Benutzer an. Ist in der Druckerdatei zum Erstellen der Ausgabe angegeben, dass die Daten gespoolt werden sollen, so wird die Spool-Datei in die Ausgabewarteschlange derjenigen Einheit gestellt, die denselben Namen wie die Einheit hat.

Es wird davon ausgegangen, dass im Parameter „Ausgabewarteschlange" (OUTQ) die Standardwerte für Druckerdatei, Jobbeschreibung, Benutzerprofil und Datenstation angegeben sind.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***WRKSTN**
Der der Datenstation des Benutzers zugeordnete Drucker wird verwendet.

***SYSVAL**
Der im Systemwert QPRTDEV angegebene Wert wird verwendet.

Name
Den Namen eines Druckers angeben, der verwendet werden soll, um die Ausgabe eines Benutzers auszudrucken.

Ausgabewarteschlange (OUTQ)

Dieser Parameter gibt die Ausgabewarteschlange an, die von einem Benutzerprofil verwendet werden soll. Die Ausgabewarteschlange muss bereits bestehen, wenn dieser Befehl ausgeführt wird.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***WRKSTN**
Die der Arbeitsstation eines Benutzers zugeordnete Ausgabewarteschlange wird verwendet.

***DEV**
Die Ausgabewarteschlange, die dem für den Parameter Druckeinheit (PRTDEV) angegebenen Drucker zugeordnet ist, wird verwendet. Die Ausgabewarteschlange hat denselben Namen wie der Drucker. (Der Parameter DEV der Druckerdatei wird durch den Befehl CRTPRTF, CHGPRTF oder OVRPRTF bestimmt.)

Es wird davon ausgegangen, dass im Parameter „Ausgabewarteschlange" (OUTQ) die Standardwerte für Druckerdatei, Jobbeschreibung, Benutzerprofil und Datenstation angegeben sind.

Qualifikationsmerkmal 1: Ausgabewarteschlange

Name
Den Namen der Ausgabewarteschlange angeben, die von einem Benutzerprofil verwendet werden soll.

Qualifikationsmerkmal 2: Bibliothek

***LIBL**
Alle Bibliotheken in der Bibliotheksliste des aktuellen Threads werden durchsucht, bis die erste Übereinstimmung gefunden wird.

***CURLIB**
Die aktuelle Bibliothek für den Thread wird durchsucht. Ist keine aktuelle Bibliothek für den Thread angegeben, wird QGPL verwendet.

Name
Den Namen der Bibliothek angeben, die durchsucht werden soll.

Abrufprogramm (ATNPGM)

Dieser Parameter gibt das Programm an, das für die Abruftaste eines Benutzers verwendet werden soll. Es wird aufgerufen, wenn in einem interaktiven Job die Abruftaste gedrückt wird. Das Programm ist nur aktiv, wenn eine Weiterleitung an den vom System bereitgestellten Befehlsprozessor QCMD besteht. Das Programm für die Abruftaste wird aktiviert, bevor das Startprogramm (sofern vorhanden) aufgerufen wird, und ist sowohl für das Programm als auch für das Menü aktiv. Ändert das Programm das ATNPGM (mit dem Befehl SETATNPGM), bleibt das neue Programm nur für die Dauer seiner Laufzeit aktiv. Wenn die Steuerung zurückgegeben wird und QCMD das Menü aufruft, wird wieder das ursprüngliche Programm für die Abruftaste aktiv. Wird der Befehl SETATNPGM aus den Menüs ausgeführt oder wird eine Anwendung aus den Menüs aufgerufen, überschreibt das neue Programm für die Abruftaste das ursprünglich angegebene. Bei Angabe von *YES oder *PARTIAL für den Parameter „Möglichkeiten einschränken“ (LMTCPB) im Befehl CRTUSRPRF (Benutzerprofil erstellen) oder im Befehl CHGUSRPRF (Benutzerprofil ändern), kann das Programm für die Abruftaste nicht geändert werden.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***SYSVAL**
Der Systemwert QATNPGM wird verwendet.

***NONE**
Der Benutzer verwendet kein Programm für die Abruftaste.

***ASSIST**
Das Programm für die Abruftaste der einfachen Oberfläche von Anwendungen, QEZMAIN, wird verwendet.

Qualifikationsmerkmal 1: Abrufprogramm

Name
Den Namen des Programms für die Abruftaste angeben, das für ein Benutzerprofil verwendet werden soll.

Qualifikationsmerkmal 2: Bibliothek

***LIBL**
Alle Bibliotheken in der Bibliotheksliste des aktuellen Threads werden durchsucht, bis die erste Übereinstimmung gefunden wird.

9.12.3
Seite 34

***CURLIB**
Die aktuelle Bibliothek für den Thread wird durchsucht. Ist keine aktuelle Bibliothek für den Thread angegeben, wird QGPL verwendet.

Name
Den Namen der Bibliothek angeben, die durchsucht werden soll.

Sortierfolge (SRTSEQ)

Dieser Parameter gibt die Sortierfolgetabelle an, die zum Vergleich von Zeichenfolgen für ein Profil verwendet werden soll.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***SYSVAL**
Der Systemwert QSRTSEQ wird verwendet.

***HEX**
Es wird keine Sortierfolgetabelle verwendet. Die Hexadezimalwerte der Zeichen werden zur Festlegung der Sortierfolge verwendet.

***LANGIDUNQ**
Es wird eine Sortiertabelle mit Zeichen unterschiedlicher Wertigkeit verwendet.

***LANGIDSHR**
Es wird eine Sortiertabelle mit Zeichen gleicher Wertigkeit verwendet.

Qualifikationsmerkmal 1: Sortierfolge

Name
Den Namen der mit diesem Profil zu verwendenden Sortierfolgetabelle angeben.

Qualifikationsmerkmal 2: Bibliothek

***LIBL**
Alle Bibliotheken in der Bibliotheksliste des aktuellen Threads werden durchsucht, bis die erste Übereinstimmung gefunden wird.

***CURLIB**
Die aktuelle Bibliothek für den Thread wird durchsucht. Ist keine aktuelle Bibliothek für den Thread angegeben, wird QGPL verwendet.

Name
Den Namen der Bibliothek angeben, die durchsucht werden soll.

Sprachen-ID (LANGID)

Dieser Parameter gibt die für einen Benutzer zu verwendende Sprachen-ID an.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***SYSVAL**
Der Systemwert QLANGID wird verwendet.

Sprachen-ID
Die zu verwendende Sprachen-ID angeben. Weitere Informationen über gültige Sprachen-IDs enthält das Handbuch „Themensammlung zu i5/OS Globalization“ im IBM i Information Center unter http://www.ibm.com/systems/i/infocenter/.

Landes- oder Regions-ID (CNTRYID)

Dieser Parameter gibt die für einen Benutzer zu verwendende Landes- oder Regions-ID an.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***SYSVAL**
Der Systemwert QCNTRYID wird verwendet.

Zeichenwert
Eine Landes- oder Regions-ID angeben. Zum Anzeigen einer vollständigen Liste der IDs – wenn die Bedienerführung für diesen Befehl angezeigt wird – den Cursor auf das Feld für diesen Parameter stellen und F4 (Bedienerführung) drücken.

9.12.3
Seite 36

Zeichensatz-ID (CCSID)

Dieser Parameter gibt die ID des codierten Zeichensatzes (CCSID) an, die für einen Benutzer verwendet werden soll.

Eine CCSID ist eine aus 16 Bits bestehende Zahl, die eine bestimmte Menge von Kennungen für Codeumsetzungsschemata, Zeichensatzkennungen, Codepagekennungen und weitere codierungsbezogene Informationen angibt, welche eine codierte grafische Darstellung eindeutig kennzeichnen.

Wird der CCSID-Wert geändert, hat die Änderung keine Auswirkung auf die gerade ausgeführten Jobs.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***SYSVAL**
Der Systemwert QCCSID wird verwendet.

***HEX**
Die CCSID 65535 wird verwendet.

Kennung
Die CCSID angeben, die für dieses Benutzerprofil verwendet werden soll. Weitere Informationen über gültige CCSIDs enthält das Handbuch „Informationen zu Globalization“ im iSeries Information Center unter http://www.ibm.com/eserver/iseries/infocenter.

Steuerung für Zeichen-ID (CHRIDCTL)

Dieser Parameter gibt die Steuerung der Zeichen-ID (CHRIDCTL) für einen Job an. Er steuert die Art der CCSID-Umsetzung, die für Bildschirmdateien, Druckerdateien und Anzeigengruppen durchgeführt wird. Bevor der Parameter verwendet werden kann, muss der Sonderwert *CHRIDCTL (CHRID) in den Befehlen zum Erstellen, Ändern und Überschreiben von Bildschirmdateien, Druckerdateien und Anzeigengruppen angegeben werden.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***SYSVAL**
Der Systemwert QCHRIDCTL wird verwendet.

***DEVD**
Der Sonderwert *DEVD führt dieselbe Funktion aus wie im Befehlsparameter CHRID für Bildschirmdateien, Druckerdateien und Anzeigengruppen.

***JOBCCSID**
Der Sonderwert *JOBCCSID führt dieselbe Funktion aus wie im Befehlsparameter CHRID für Bildschirmdateien, Druckerdateien und Anzeigengruppen.

Jobattribut länderspezifische Angaben (SETJOBATR)

Dieser Parameter gibt an, welche Jobattribute aus den länderspezifischen Angaben (LOCALE) übernommen werden sollen, wenn der Job eingeleitet wird.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***SYSVAL**
Der Systemwert QSETJOBATR bestimmt, welche Jobattribute aus den länderspezifischen Angaben übernommen werden.

***NONE**
Es werden keine Jobattribute aus den länderspezifischen Angaben übernommen.

***CCSID**
Die ID des codierten Zeichensatzes (CCSID) aus den länderspezifischen Angaben wird verwendet. Der CCSID-Wert aus den länderspezifischen Angaben überschreibt die CCSID im Benutzerprofil.

9.12.3
Seite 38

***DATFMT**
Das Datumsformat aus den länderspezifischen Angaben wird verwendet.

***DATSEP**
Das Datumstrennzeichen aus den länderspezifischen Angaben wird verwendet.

***DECFMT**
Das Dezimalformat aus den länderspezifischen Angaben wird verwendet.

***SRTSEQ**
Die Sortierfolge aus den länderspezifischen Angaben wird verwendet. Die Sortierfolge aus den länderspezifischen Angaben überschreibt die im Benutzerprofil angegebene Sortierfolge.

***TIMSEP**
Das Zeittrennzeichen aus den länderspezifischen Angaben wird verwendet.

```
                    Benutzerprofil ändern (CHGUSRPRF)

Auswahl eingeben und Eingabetaste drücken.

Länderspezifische Angaben  . . .   *SAME

Benutzerangaben  . . . . . .  . .  *NONE          *SAME, *NONE, *CLKWD...
     + für weitere Werte
Benutzernummer . . . . . .  . . .  114           1-4294967294, *SAME
Gruppennummer  . . . . . .  . . .  *NONE          1-4294967294, *SAME, *GEN...
Benutzerverzeichnis  . . . . . .   *SAME

EIM-Zuordnung:
  EIM-Kennung  . . . . . .  . . .  *NOCHG
  Zuordnungstyp  . . . . . .  . .                 *TARGET, *SOURCE, *TGTSRC...
  Zuordnungsaktion . . . . . .  .                 *REPLACE, *ADD, *REMOVE
  EIM-Kennung erstellen  . . . .                  *NOCRTEIMID, *CRTEIMID
Ablaufdatum für Benutzerprofil     *NONE          Datum, *NONE, *USREXPITV...
Ablaufintervall f. Benutzerpr.                    1-366
```

Benutzerprofil – Seite 5

Länderspezifische Angaben (LOCALE)

9.12.3
Seite 39

Dieser Parameter gibt den Pfadnamen für die länderspezifischen Angaben an, die der Umgebungsvariablen LANG für diesen Benutzer zugeordnet sind.

Dieser Parameter ist unicodefähig. Weitere Informationen sind im Abschnitt „Unicode support in CL“ unter „CL-Themensammlung“ in der Kategorie „Programmierung“ im IBM i Information Center unter http://www.ibm.com/systems/i/infocenter/ zu finden.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***SYSVAL**
Der Systemwert QLOCALE bestimmt den Pfadnamen für die länderspezifischen Angaben, der diesem Benutzer zugeordnet werden soll.

***NONE**
Dem Benutzer wird kein Pfadname für länderspezifische Angaben zugeordnet.

***C**
Dem Benutzer wird der Pfadname der länderspezifischen Angaben für die Programmiersprache C zugeordnet.

***POSIX**
Dem Benutzer wird der Pfadname der länderspezifischen Angaben für POSIX zugeordnet.

‘Pfadname’
Den Pfadnamen für die länderspezifischen Angaben angeben, der einem Benutzer zugeordnet werden soll.

9.12.3
Seite 40

Benutzerangaben (USROPT)

Gibt die Detailstufe des Hilfetexts an, die angezeigt werden soll, sowie die Standardbelegung der Tasten zum Vor- und Zurückblättern. Das System ruft mehrere Anzeigen auf, die für einen weniger erfahrenen Benutzer geeignet sind. Erfahrene Benutzer müssen noch eine zusätzliche Aktion durchführen, um detaillierte Informationen angezeigt zu bekommen. Werden Werte für diesen Parameter angegeben, zeigt das System detaillierte Informationen an, ohne dass der erfahrene Benutzer eine weitere Aktion durchführen muss.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***NONE**
Es werden keine Detailinformationen angezeigt.

***CLKWD**
Wenn die Bedienerführung für einen CL-Befehl angezeigt wird, werden statt der gültigen Parameterwerte die Parameterschlüsselwörter angezeigt.

***EXPERT**
Detailliertere Informationen werden angezeigt, wenn der Benutzer Anzeige- und Editieroperationen zum Definieren oder Ändern des Systems durchführt (z. B. Objektberechtigung editieren oder anzeigen).

***ROLLKEY**
Die Aktionen der Tasten zum Vor- und Zurückblättern werden umgekehrt.

***NOSTSMSG**
An den Benutzer gesendete Statusnachrichten werden nicht angezeigt.

***STSMSG**
An den Benutzer gesendete Statusnachrichten werden angezeigt.

***HLPFULL**
Der Hilfetext wird in einer Gesamtanzeige gezeigt, nicht in einem Fenster.

***PRTMSG**
Es wird eine Nachricht an die Nachrichtenwarteschlange eines Benutzers gesendet, wenn eine Spool-Datei für den Benutzer gedruckt oder vom Druckausgabeprogramm angehalten wird.

Benutzernummer (UID)

Dieser Parameter gibt die Benutzernummer (UID-Nummer) für ein Benutzerprofil an. Die UID-Nummer dient zur Kennzeichnung des Benutzers, wenn er mit dem Verzeichnisdateisystem arbeitet. Die UID-Nummer für einen Benutzer kann nicht geändert werden, wenn ein oder mehrere aktive Jobs für den Benutzer vorhanden sind.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

Nummer
Die dem Benutzerprofil zuzuordnende UID-Nummer angeben. Es kann ein Wert von 1 bis 4294967294 eingegeben werden. Die angegebene UID-Nummer darf nicht bereits einem anderen Benutzerprofil zugeordnet sein.

Gruppennummer (GID)

Dieser Parameter gibt die Gruppennummer (GID-Nummer) für ein Benutzerprofil an. Die GID-Nummer dient zur Kennzeichnung des Gruppenprofils, wenn ein Mitglied der Gruppe mit dem Verzeichnisdateisystem arbeitet. Die GID-Nummer für einen Benutzer kann in den folgenden Fällen nicht geändert werden:

Das Benutzerprofil ist die Primärgruppe eines Objekts in einem Verzeichnis.

Für den Benutzer sind ein oder mehrere aktive Jobs vorhanden.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***NONE**
Der Benutzer verfügt nicht über eine GID-Nummer oder eine bestehende GID-Nummer wurde entfernt.

Dieser Wert kann nicht angegeben werden, wenn es sich bei dem Benutzer um ein Gruppenprofil oder die Primärgruppe eines Objekts handelt.

***GEN**
Die GID-Nummer wird für den Benutzer generiert. Das System generiert eine GID-Nummer, die nicht bereits einem anderen Benutzer zugeordnet ist. Die generierte GID-Nummer ist größer als 100.

Nummer
Die GID-Nummer angeben, die dem Benutzerprofil zugeordnet werden soll. Es kann ein Wert von 1 bis 4294967294 eingegeben werden. Die angegebene GID-Nummer darf nicht bereits einem anderen Benutzerprofil zugeordnet sein.

Benutzerverzeichnis (HOMEDIR)

Dieser Parameter gibt den Pfadnamen des Benutzerverzeichnisses für ein Benutzerprofil an. Das Benutzerverzeichnis ist das ursprüngliche Arbeitsverzeichnis des Benutzers. Das einem Prozess zugeordnete Arbeitsverzeichnis wird in einem Verzeichnisdateisystem zur Auflösung von Pfadnamen benutzt, die nicht mit einem Schrägstrich (/) beginnen. Existiert das angegebene Benutzerverzeichnis beim Anmelden eines Benutzers nicht, wird das Stammverzeichnis (/) zum ursprünglichen Arbeitsverzeichnis für den Benutzer.

Dieser Parameter ist unicodefähig. Weitere Informationen sind im Abschnitt „Unicode support in CL“ unter „CL-Themensammlung“ in der Kategorie „Programmierung“ im IBM i Information Center unter http://www.ibm.com/systems/i/infocenter/ zu finden.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***USRPRF**
Das dem Benutzer zugeordnete Benutzerverzeichnis lautet ‚/home/USRPRF‘, wobei ‚USRPRF‘ der Name des Benutzerprofils ist.

‘Pfadname’
Den Pfadnamen des Ausgangsverzeichnisses angeben, das einem Benutzer zugeordnet werden soll.

Weitere Informationen über das Angeben von Pfadnamen enthält der Abschnitt „Object naming rules“ des Handbuchs „CL-Themensammlung“ in der Kategorie „Programmierung“ im IBM i Information Center unter http://www.ibm.com/systems/i/infocenter/.

EIM-Zuordnung (EIMASSOC)

Dieser Parameter gibt an, ob für einen Benutzer eine EIM-Zuordnung (Enterprise Identity Mapping) zu einer EIM-Kennung verarbeitet werden soll.

- Die Informationen werden nicht im Benutzerprofil gespeichert und nicht mit gesichert oder zurückgespeichert.
- Wenn dieses System nicht für EIM konfiguriert ist, findet keine Verarbeitung statt. Wenn keine EIM-Operationen ausgeführt werden können, bedeutet das nicht, das der Befehl fehlschlägt.

Gültige Werte sind:

***NOCHG**
Die EIM-Zuordnungsinformation ändert sich nicht.

Element 1: EIM-Kennung
Gibt die EIM-Kennung für eine Zuordnung an.

***USRPRF**
Der Name der EIM-Kennung ist mit dem Namen des Benutzerprofils identisch.

Zeichenwert
Den Namen der EIM-Kennung angeben.

Element 2: Zuordnungstyp
Gibt den Zuordnungstyp an. Es wird empfohlen, für einen IBM i-Benutzer eine Zielzuordnung hinzuzufügen.

Zielzuordnungen dienen hauptsächlich zum Schutz vorhandener Daten. Sie werden bei Abgleichsuchoperationen gefunden (eimGetTargetFromSource()), können aber nicht als Quellenidentität für eine Abgleichsuchoperation verwendet werden.

Quellenzuordnungen dienen hauptsächlich für Authentifizierungszwecke. Sie können als Quellenidentität einer Abgleichsuchoperation benutzt werden, werden aber nicht als Ziel einer Abgleichsuchoperation gefunden. Administrative Zuordnungen werden benutzt, um anzuzeigen, dass einer EIM-Kennung eine Identität zugeordnet ist. Sie können aber nicht als Quelle einer Abgleichsuchoperation verwendet werden und werden nicht als Ziel bei einer solchen Suchoperation gefunden.

***TARGET**
Eine Zielzuordnung verarbeiten.

***SOURCE**
Eine Quellenzuordnung verarbeiten.

9.12.3
Seite 44

***TGTSRC**
Sowohl eine Ziel- als auch eine Quellenzuordnung verarbeiten.

***ADMIN**
Eine administrative Zuordnung verarbeiten.

***ALL**
Alle Zuordnungstypen verarbeiten.

Element 3: Zuordnungsaktion

***REPLACE**
Zuordnungen des angegebenen Typs werden aus allen EIM-Kennungen entfernt, die eine Zuordnung für ein Benutzerprofil und das lokale EIM-Register enthalten. Der angegebenen EIM-Kennung wird eine neue Zuordnung hinzugefügt.

***ADD**
Eine Zuordnung hinzufügen.

***REMOVE**
Eine Zuordnung entfernen.

Element 4: EIM-Kennung erstellen
Gibt an, ob die EIM-Kennung erstellt werden soll, wenn sie noch nicht besteht.

***NOCRTEIMID**
Die EIM-Kennung wird nicht erstellt.

***CRTEIMID**
Die EIM-Kennung wird erstellt, wenn sie noch nicht besteht.

Ablaufdatum für Benutzerprofil (USREXPDATE)

Dieser Parameter gibt das Datum an, zu dem ein Benutzerprofil abläuft und automatisch deaktiviert wird. Mit dem Befehl DSPEXPSCD (Verfallsplan anzeigen) kann eine Liste aller auf ein Ablaufdatum gesetzten Benutzerprofile angezeigt werden.

Wenn ein Benutzerprofil auf ein Ablaufdatum gesetzt wurde, wird eine nächtliche Ausführung des Jobs QSECEXP1 geplant.

Für die folgenden von IBM bereitgestellten Benutzerprofile kann kein Ablaufdatum für „Benutzerprofil" angegeben werden:

QANZAGENT, QAUTPROF, QCLUMGT, QCLUSTER, QCOLSRV, QDBSHR, QDBSHRDO, QDFTOWN, QDIRSRV, QDLFM, QDOC, QDSNX, QEJB, QEJBSVR, QFNC, QGATE, QIBMHELP, QIPP, QLPAUTO, QLPINSTALL, QLWISVR, QMGTC, QMSF, QNETSPLF, QNFSANON, QNTP, QPEX, QPM400, QSECOFR, QSNADS, QSPL, QSPLJOB, QSRVAGT, QSYS, QTCM, QTCP, QTFTP, QTMHHTTP, QTMHHTP1, QTSTRQS, QWEBADMIN, QWSERVICE, QYCMCIMOM, QYPSJSVR.

Gültige Werte sind:

***SAME**
Der Wert ändert sich nicht.

***NONE**
Das Benutzerprofil hat kein Ablaufdatum.

***USREXPITV**
Das Ablaufdatum für ein Benutzerprofil wird anhand des Werts errechnet, der für das Ablaufintervall des Benutzerprofils (Parameter USREXPITV) angegeben wurde.

Datum
Gibt das Datum an, zu dem ein Benutzerprofil abläuft. Das Datum muss im Jobdatumsformat angegeben werden.

Ablaufintervall für Benutzerprofil (USREXPITV)

Dieser Parameter gibt das Ablaufintervall (in Tagen) an, bevor es automatisch deaktiviert wird. Mit dem Befehl DSPUSRPRF (Benutzerprofil anzeigen) kann das Ablaufdatum für ein Benutzerprofil angezeigt werden. Mit dem Befehl DSPEXPSCD (Verfallsplan anzeigen) kann eine Liste aller auf ein Ablaufdatum gesetzten Benutzerprofile angezeigt werden.

Für diesen Parameter muss ein Wert angegeben werden, wenn für den Parameter USREXPDATE der Wert *USREXPITV angegeben wurde. Wenn für den Parameter USREXPDATE ein anderer Wert als *USREXPITV angegeben wurde, ist kein Wert für diesen Parameter zulässig.

1-366
Wenn für ein Benutzerprofil kein Ablaufdatum angegeben wurde oder das Benutzerprofil abgelaufen ist und der Parameter „Status“ auf *ENABLED gesetzt wurde, gibt dieser Wert die Anzahl der Tage zwischen dem aktuellen Datum und dem neuen Ablaufdatum des Benutzerprofils an. Wenn das Benutzerprofil noch nicht abgelaufen ist, wird das Ablaufintervall für das Benutzerprofil geändert, aber nicht das bestehende Ablaufdatum des Benutzerprofils. Gültig sind Werte von 1 bis 366.

Benutzerprofil löschen

Um ein Benutzerprofil aus dem System zu löschen, verwenden Sie den Befehl DLTUSRPRF. Sie müssen über die Sonderberechtigung *SECADM verfügen, um das durchführen zu können.

Das Löschen von Benutzerprofilen ist die sauberste Art, einen ausgeschiedenen Mitarbeiter aus einem System zu entfernen. Oftmals werden die Benutzerprofile bei Ausscheiden nur deaktiviert (Status *DISABLED). Dies hat jedoch zur Folge, dass ein Benutzerprofil auch mal wieder aktiviert werden kann und so ungewollte Zugriffe auf das System erfolgen können. Auch wird durch deaktivierte Benutzerprofile die Übersicht über die Benutzer komplizierter.

Das Löschen eines Benutzerprofils ist jedoch nicht ganz ungefährlich, weil je nach Tätigkeit des Mitarbeiters mit diesem Profil viele Objekte verbunden sein können. Wenn Sie beim Löschen also nicht aufpassen, so kann das zu unvorhersehbaren Ergebnissen führen, die manchmal nicht sofort, sondern erst später zu Tage treten.

Der Benutzer, der ein Profil löschen will, muss über die Benutzungsberechtigung (*USE) sowie die Objektexistenzberechtigung (*OBJEXIST) für das Benutzerprofil verfügen.

Für das Löschen einer Nachrichtenwarteschlange, die einem Benutzerprofil zugeordnet ist und dessen Eigner das Benutzerprofil ist, muss der Benutzer über die Berechtigungen *OBJEXIST, *USE und *DLT für die entsprechende Nachrichtenwarteschlange verfügen.

Das Löschen eines Benutzerprofils ist nicht möglich, während ein Benutzer unter diesem Profil aktiv ist oder wenn das Benutzerprofil ein Objekteigner ist und OWNOBJOPT(*NODLT) angegeben wurde. Alle Objekte des betroffenen Benutzerprofils müssen entweder erst mit dem Befehl CHGOBJOWN (Objekteigner ändern) auf andere Eigner übertragen oder aus dem System gelöscht werden. Die Objekte können auch mit der Angabe OWNOBJOPT(*DLT) gelöscht und das Eigentumsrecht kann mit der Angabe OWNOBJOPT(*CHGOWN Benutzerprofilname) geändert werden. Berechtigungen, die dem Benutzer erteilt wurden, müssen nicht gesondert mit dem Befehl RVKOBJAUT (Objektberechtigung entziehen) entzogen werden, sie werden automatisch beim Löschen des Benutzerprofils entzogen.

Zum Löschen jedes Objekts ist die Berechtigung *OBJEXIST für das Objekt erforderlich.

9.12.4
Seite 2

Ein Benutzerprofil kann nicht gelöscht werden, wenn es die Primärgruppe für ein Objekt ist. Alle Objekte, für die ein Benutzer die Primärgruppe ist, müssen entweder mit dem Befehl CHGOBJPGP (Primärgruppe des Objekts ändern) übertragen oder aus dem System gelöscht werden. Die Übertragung kann durchgeführt werden, indem PGPOPT(*CHGPGP Benutzerprofilname) zum Ändern der Primärgruppe angegeben wird.

Mit diesem Befehl wird die Unterstützungsfunktion des Systemverteilerverzeichnisses aufgerufen, um einen Benutzer aus dem Verzeichnis und, falls erforderlich, aus dem Verteiler zu löschen. Von der Unterstützungsfunktion wird für die Dateien des Systemverteilerverzeichnisses (QUSURSYS/QAOS*) Journalaufzeichnung und COMMIT-Steuerung verwendet. Wenn diese Funktion angefordert wird, muss die COMMIT-Steuerung inaktiv sein. Wenn die COMMIT-Steuerung beim Anfordern dieser Funktion aktiv ist, muss das Journal QUSRSYS/QAOSDIAJRN sein.

Weitere Hinweise zum Löschen von Benutzerprofilen:

1. Die Objekte der Art *USRPRF, *RCT und *AUTHLR, deren Eigner ein Benutzerprofil ist, werden nicht aus dem System gelöscht. Das Eigentumsrecht an diesen Objekten wird an das Benutzerprofil QDFTOWN übertragen.
2. Die Objekte der Art *PRDDFN, deren Eigner ein Benutzerprofil ist, werden nicht gelöscht. Das Eigentumsrecht wird an das Benutzerprofil QSYS übertragen.
3. Neben den aufgeführten Einschränkungen gelten alle den Befehl DLTLIB betreffenden Einschränkungen auch für die Angabe von OWNOBJOPT(*DLT). Z. B. kann ein Objekt nicht gelöscht werden, während es verwendet wird. Und auch eine physische Datei mit einer zugeordneten logischen Datei – wobei die logische Datei einen anderen Eigner hat – kann nicht gelöscht werden.
4. Das Löschen von Objekten der Art *LIB aus dem System, deren Eigner ein Benutzerprofil ist, ist nicht möglich, wenn Objekte in der Bibliothek ein anderes Benutzerprofil als Eigner haben. Das Eigentumsrecht an der Bibliothek wird an das Systembenutzerprofil QDFTOWN übertragen.
5. Das Löschen von Objekten der Art *DIR aus dem System, deren Eigner ein Benutzerprofil ist, ist nicht möglich, wenn Objekte in dem Verzeichnis ein anderes Benutzerprofil als Eigner haben. Das Eigentumsrecht am Verzeichnis wird an das Systembenutzerprofil QDFTOWN übertragen.
6. Das Löschen von Objekten der Art *BLKSF aus dem System, deren Eigner ein Benutzerprofil ist, ist nicht möglich, wenn Objekte im benutzerdefinierten Dateisystem (dargestellt durch *BLKSF) ein anderes Benutzerprofil als Eigner haben. Das Eigentumsrecht an dem benutzerdefinierten Dateisystem wird an das Systembenutzerprofil QDFTOWN übertragen.

7. Ein Benutzerprofil kann nicht gleichzeitig der Eigner des Objekts und die Primärgruppe des Objekts sein. Wenn der neue Eigner schon die Primärgruppe eines Objekts ist, dessen Eigner der aktuelle Benutzer ist, schlägt das Übertragen des Eigentumsrechts fehl. Wenn die neue Primärgruppe schon Eigner eines Objekts ist, für das der aktuelle Eigner die Primärgruppe ist, schlägt das Übertragen der Primärgruppe fehl.

```
                    Benutzerprofil löschen (DLTUSRPRF)

Auswahl eingeben und Eingabetaste drücken.

Benutzerprofil . . . . . > RENGEL        Name
Auswahl eigener Objekte:
  Wert eigener Objekte . . . . .  *NODLT        *NODLT, *DLT, *CHGOWN
  Ben.profilname, wenn *CHGOWN    _____         Name
Angabe für Primärgruppe:
  Primärgruppenwert  . . . . . .  *NOCHG       *NOCHG, *CHGPGP
  Neue Primärgruppe  . . . . . .  _____         Name, *NONE
  Berecht. für neue Primärgruppe  _____         *OLDPGP, *PRIVATE, *ALL...
EIM-Zuordnung  . . . . . .  *DLT        *DLT, *NODLT

                                                                      Ende
F3=Verlassen    F4=Bedienerf.    F5=Aktualisieren    F12=Abbrechen
F13=Verwendung der Anzeige       F24=Weitere Tasten
```

DLTUSRPRF + F4

Benutzerprofil (USRPRF)

Dieser Parameter gibt ein zu löschendes Benutzerprofil an.

Die folgenden von IBM gelieferten Benutzerprofile sind für diesen Parameter nicht gültig:

QANZAGENT, QAUTPROF, QCLUMGT, QCLUSTER, QCOLSRV, QDBSHR, QDBSHRDO, QDFTOWN, QDIRSRV, QDLFM, QDOC, QDSNX, QEJB, QEJBSVR, QFNC, QGATE, QIBMHELP, QIPP, QLPAUTO, QLPINSTALL, QLWISVR, QMGTC, QMSF, QNETSPLF, QNFSANON, QNTP, QPEX, QPGMR, QPM400, QSECOFR, QSNADS, QSPL, QSPLJOB, QSRV, QSRVAGT, QSRVBAS, QSYS, QSYSOPR, QTCM, QTCP, QTFTP, QTMHHTTP, QTMHHTP1, QTSTRQS, QUSER, QWEBADMIN, QWSERVICE, QYCMCIMOM, QYPSJSVR.

Name
Den Namen eines zu löschenden Benutzerprofils angeben.

9.12.4
Seite 4

Auswahl eigener Objekte (OWNOBJOPT)

Dieser Parameter gibt die Art der Maßnahmen an, die bei eigenen Objekten des Benutzerprofils durchzuführen sind.

Gültige Werte sind:

***NODLT**
Die eigenen Objekte des Benutzerprofils werden nicht geändert und das Benutzerprofil wird nicht gelöscht, wenn der Benutzer eigene Objekte hat.

***DLT**
Die eigenen Objekte des Benutzerprofils werden gelöscht. Das Benutzerprofil wird gelöscht, wenn alle eigenen Objekte erfolgreich gelöscht werden konnten.

Element 1: Wert eigener Objekte

***CHGOWN**
Die eigenen Objekte des Benutzerprofils gehen in das Eigentum des angegebenen Benutzerprofils über. Das Benutzerprofil wird gelöscht, wenn alle eigenen Objekte erfolgreich übertragen werden konnten.

Bei Angabe von *CHGOWN muss für das neue Benutzerprofil ein Benutzerprofilname angegeben werden. Das neue Benutzerprofil wird zum Eigner aller Objekte, deren Eigner das für den Parameter „Benutzerprofil“ (USRPRF) angegebene Benutzerprofil war.

Element 2: Benutzerprofilname, wenn *CHGOWN

Name
Den Namen des Benutzerprofils angeben, das der neue Eigner sein soll.

Angabe für Primärgruppe (PGPOPT)

Dieser Parameter gibt die Art der Maßnahmen an, die bei Objekten durchgeführt werden, für die das zu löschende Benutzerprofil die Primärgruppe ist.

Gültige Werte sind:

***NOCHG**
Die Objekte, für die das Benutzerprofil die Primärgruppe ist, ändern sich nicht, und das Benutzerprofil wird nicht gelöscht, wenn der Benutzer die Primärgruppe für eines der Objekte ist.

Element 1: Primärgruppenwert

***CHGPGP**
Die Objekte, für die das Benutzerprofil die Primärgruppe ist, werden in das angegebene Benutzerprofil übertragen. Das Benutzerprofil wird gelöscht, nachdem alle Objekte übertragen worden sind.

Wenn *CHGPGP angegeben wird, muss ein Benutzerprofilname oder *NONE angegeben werden. Wird ein Benutzerprofilname angegeben, ist dieser Benutzer die Primärgruppe für alle Objekte, für die das im Parameter USRPRF angegebene Benutzerprofil die Primärgruppe ist. Wird *NONE angegeben, ist allen Objekten, für die das im Parameter USRPRF angegebene Benutzerprofil die Primärgruppe ist, keine Primärgruppe mehr zugeordnet.

Element 2: Neue Primärgruppe

***NONE**
Den Objekten wird keine Primärgruppe zugeordnet.

Name
Den Namen des Benutzerprofils angeben, das die neue Primärgruppe sein soll. Das angegebene Benutzerprofil muss über eine Gruppennummer (gid) verfügen.

Element 3: Berechtigung für neue Primärgruppe

***OLDPGP**
Die neue Primärgruppe erhält dieselbe Berechtigung für ein Objekt wie die alte Primärgruppe.

***PRIVATE**
Verfügt die neue Primärgruppe über eine persönliche Berechtigung für ein Objekt, wird sie zur Primärgruppe für das Objekt und erhält als Primärgruppenberechtigung die bisherige persönliche Berechtigung. Verfügt die neue Primärgruppe über keine persönliche Berechtigung für ein Objekt, wird sie die neue Primärgruppe ohne Berechtigung für das Objekt.

***ALL**
Die neue Primärgruppe erhält die Berechtigung *ALL für ein Objekt.

***CHANGE**
Die neue Primärgruppe erhält die Berechtigung *CHANGE für ein Objekt.

***USE**
Die neue Primärgruppe erhält die Berechtigung *USE für ein Objekt.

***EXCLUDE**
Die neue Primärgruppe erhält die Berechtigung *EXCLUDE für ein Objekt.

EIM-Zuordnung (EIMASSOC)

Dieser Parameter gibt an, ob Enterprise Identity Mapping (EIM)-Zuordnungen für einen Benutzer im lokalen Register gelöscht werden sollen. Es werden alle Arten von Zuordnungen für den Benutzer im lokalen Register gelöscht, einschließlich Ziel- und Quellenzuordnungen, administrative Zuordnungen und Richtlinien. Wenn das System nicht für EIM konfiguriert ist, findet keine Verarbeitung statt. Wenn das System für EIM konfiguriert ist, aber keine Verbindung zu EIM hergestellt werden kann (z. B. weil der LDAP-Server, für den EIM konfiguriert ist, nicht aktiv ist), wird ein QSYEIM-Job übergeben, der für die Dauer von einer Stunde versucht, eine Verbindung zu EIM herzustellen. Falls die EIM-Zuordnungen nicht gelöscht werden können, bedeutet dies nicht, dass das Profil nicht gelöscht werden kann.

Nicht gelöschte Zuordnungen werden von einem neu erstellten gleichnamigen Profil verwendet.

***DLT**
EIM-Zuordnungen werden gelöscht.

***NODLT**
EIM-Zuordnungen werden nicht gelöscht.

Objekte eines Eigners anzeigen

Die Anzeige "Mit Objekten eines Eigners arbeiten" (WRKOBJOWN) zeigt eine Liste aller Objekte an, deren Eignern das ausgewählte Benutzerprofil ist. Außerdem werden die Bibliothek, in der sich die Objekte befinden, die jeweilige Objektart, die Attribute der Objekte sowie der jeweils zugehörige Text angezeigt.

Diese Auflistung zeigt Ihnen, bei welchen Objekten ein Benutzer als Eigner eingetragen ist. In den meisten Fällen sind das Objekte, die der entsprechende Benutzer erstellt hat, also z. B. Dateien, die durch Ausgaben aus Query entstanden sind, oder Programme, die durch Kompilieren aus Quellen entstanden sind etc.

```
                    Mit Objekten eines Eigners arbeiten

Benutzerprofil . . . . . :   RENGEL

Auswahl eingeben und Eingabetaste drücken.
  2=Berechtigung editieren   4=Löschen     5=Berechtigung anzeigen
  7=Umbenennen   8=Beschreibung anzeigen   9=Eigner ändern

Ausw  Objekt                 Bibliothek    Art       Attribut    ASP-Einh.
      /home/liste.pdf/QP >                 *STMF                 *SYSBAS
  _   /tmp/AUQSHF_output                   *STMF                 *SYSBAS
  _   /tmp/AVQSHF_output                   *STMF                 *SYSBAS
  _   /SMZVDTA/quarantin >                 *STMF                 *SYSBAS
  _   /tmp/EXCQSH_output                   *STMF                 *SYSBAS
  _   /tmp/IsLocked.sh                     *STMF                 *SYSBAS
  _   /tmp/JAQSHF_output                   *STMF                 *SYSBAS
  _   /tmp/ODQSHF_output                   *STMF                 *SYSBAS
  _   /tmp/RPT000001_All >                 *STMF                 *SYSBAS
                                                                   Weitere ...
Parameter oder Befehl
===>
F3=Verlassen     F5=Aktual.    F9=Finden    F11=Beschreibungen anzeigen
F12=Abbrechen    F17=Anfang    F18=Ende     F24=Weitere Tasten
```

Mit Objekten eines Eigners arbeiten

Benutzerprofil

Dieser Parameter gibt den Namen eines Benutzerprofils an.

9.12.4
Seite 8

Auswahl

2=Berechtigung editieren
Die Berechtigung eines Benutzers für ein Objekt hinzufügen, ändern oder löschen.

4=Löschen
Löscht ein Objekt.

5=Berechtigung anzeigen
Zeigt eine Liste der Benutzer an, die zur Arbeit mit den angegebenen Objekten berechtigt sind.

7=Umbenennen
Ändert den Namen eines Objekts.

8=Beschreibung anzeigen
Zeigt die Namen und Attribute ausgewählter Objekte an. Diese Option wird für verzeichnisbasierte Objekte nicht unterstützt.

9=Eigner ändern
Ermöglicht es, das Eigentumsrecht für ein Objekt zu ändern.

Objekt
Der Name eines Objekts.

Bibliothek
Name der Bibliothek, die ein Objekt enthält, oder Name des Ordners, der ein Dokument enthält.

Art
Die Art eines Objekts, wie z. B. *PGM, *FILE oder *LIB.

Attribute
Da es mehrere Objektarten gibt, wird das Attribut zur näheren Beschreibung eines Objekts verwendet.

ASP-Einheit
Der Name der ASP-Einheit (ASP = Zusatzspeicherpool), in der sich ein Objekt befindet. *SYSBAS bedeutet, dass sich das Objekt im System-ASP oder in einem Basis-Benutzer-ASP befindet.

Objekte und Berechtigungen

In den vorangegangenen Kapiteln haben wir uns mit den übergeordneten Systemeinstellungen, sicherheitsrelevanten Systemwerten, Sicherheitsstufen und Benutzerprofilen sowie deren Einstellungen beschäftigt.

Objekte
Etwas Entscheidendes fehlt in dieser Aufstellung allerdings noch – die Objekte. Auf unserem IBM Power i-System beschäftigen wir uns ja ständig mit Objekten. Datenbanken sind Objekte, Programme sind Objekte, Datenwarteschlangen, Jobwarteschlangen, Spool-Dateien ebenso. Jedes Objekt besitzt Eigenschaften und u. a. auch Berechtigungen, mit denen gesteuert wird, welche Benutzer oder Benutzergruppen mit diesen Objekten arbeiten dürfen.

Objektberechtigungen
Objektberechtigungen beinhalten Informationen darüber, wer Objekte wie verwenden darf, also z. B., dass ein Anwender ein Objekt benutzen, ein zweiter Anwender das Objekt ändern und ein dritter überhaupt nicht auf das Objekt zugreifen darf.

In diesem Teil des Werks geht es also um Objekte und deren Berechtigungen. Den Einstieg dazu finden wir, indem wir mit Objekten arbeiten.

36. Ergänzung 9/2013

9.13
Seite 2

Objekte

Welche Objekttypen es im IBM i-Betriebssystem gibt, sehen wir u. a., wenn wir den Befehl WRKOBJ aufrufen:

```
                        Mit Objekten arbeiten (WRKOBJ)

Auswahl eingeben und Eingabetaste drücken.

Objekt . . . . . . . . . . . . .                Name, generisch*, *ALL
  Bibliothek . . . . . . . . . .    *LIBL       Name, *LIBL, *CURLIB...
Objektart  . . . . . . . . . . .  *ALL          *ALL, *ALRTBL, *AUTL...
```

WRKOBJ + F4

Der Befehl selbst hat nur wenige Parameter. Neben dem Objektnamen und der Bibliothek versteckt sich hinter dem unscheinbaren Parameter Objektart die Vielfalt der unterschiedlichen Objekttypen. Diese Vielfalt gibt es nicht schon immer. Mit jeder Version des Betriebssystems kommen neue Objekttypen hinzu, andere verlieren an Bedeutung.

Das ist auch eine Besonderheit unseres Systems. Es wird nichts weggeworfen, alles kann noch verwendet werden, so alt es auch sein mag. Nicht zuletzt hat die AS/400/iSeries/IBM Power i deshalb so lange durchgehalten, weil Innovation und Historie so eng verbunden sind. Noch immer laufen Teile von Anwendungen, die aus der Urzeit der AS/400 oder von noch älteren Systemen stammen. Daher ist es schwierig, etwas wegzulassen.

Wir betrachten in der Folge einmal ein aktuelles System mit Betriebssystemversion V7R1. Haben Sie eine ältere Version im Einsatz, finden Sie ggf. neuere Objekttypen noch nicht. Eine grobe Unterscheidung findet sich darin, ob ein Objekttyp in einer Bibliothek oder im integrierten Dateisystem IFS gespeichert ist.

36. Ergänzung 9/2013

9.13.1
Seite 2

Objekte in Bibliotheken

***ALRTBL**
Das Objekt ALERT-Tabelle (*ALRTBL) enthält ALERT-Beschreibungen, die den Inhalt einer ALERT-Meldung der Systemnetzwerkarchitektur (SNA) für bestimmte Fehlerbedingungen definieren. Der IBM i Alert-Manager verwendet zur Erstellung einer Alert-Meldung Alert-Beschreibungen aus einer Alert-Tabelle.

***AUTL**
Das Objekt Berechtigungsliste (*AUTL) enthält eine Liste von Benutzern und deren Berechtigungen zum Schützen bzw. Benutzen anderer Objekte.

***BNDDIR**
Das Objekt Binderverzeichnis (*BNDDIR) enthält die Namen von Objekten, die Bibliothek, in der sie sich befinden, sowie die Objektart (*MODULE oder *SRVPGM). Anhand des Binderverzeichnisobjekts können die Namen der *MODULE- und *SRVPGM-Objekte im Befehl CRTPGM (Programm erstellen) oder im Befehl CRTSRVPGM (Serviceprogramm erstellen) zusammengefasst werden.

***CFGL**
Das Objekt Konfigurationsliste (*CFGL) enthält eine Liste von DFV-Einträgen, die von asynchroner Übertragung, APPN-Übertragung und der Übertragung für den Durchgriff auf Einzelhandelseinrichtungen zum Definieren ferner und lokaler Standorte und Netzwerkadressen verwendet werden.

***CHTFMT**
Das Objekt Diagrammformat (*CHTFMT) enthält Definitionen für die Darstellung von Diagrammen mit der Präsentationsgrafik (BGU). Mit dem Diagrammformat können Balkendiagramme oder andere Diagrammarten angegeben werden.

***CLD**
Das Objekt Beschreibung der länderspezifischen Angaben für die Programmiersprache ILE C (*CLD) enthält Formatierungsangaben für numerische Einheiten, Währungseinheiten, Sortierfolgen und Codepunktzuordnungen.

***CLS**
Das Objekt Klasse (*CLS) enthält die Umgebungsattribute zur Ausführung eines Jobs. Die von einem Leitwegschritt verwendete Klasse wird im Leitwegeintrag der Subsystembeschreibung angegeben.

***CMD**
Ein Objekt-Befehl (*CMD) fordert eine Funktion des Systems an.

9.13.1.1
Seite 2

***CNNL**
Das Objekt Verbindungsliste (*CNNL) enthält Konfigurationsinformationen, die vom System benötigt werden, um das Senden und Empfangen von Anrufen mit einem dienstintegrierenden digitalen Fernmeldenetz (ISDN) zu verwalten.

***COSD**
Das Objekt Serviceklassenbeschreibung (*COSD) enthält Bereiche von Verbindungs- und Knotenkenndaten, die für die Leitwegwahl einer APPN-Sitzung durch ein APPC-Netzwerk verwendet werden.

***CRG**
Das Objekt Clusterressourcengruppe (*CRG) definiert die Beziehung zwischen den Knoten (Systemen), die einer Gruppe an flexiblen Ressourcen zugeordnet sind. Das Objekt definiert auch die Aktionen, die bei einem Wechsel des Zugriffpunkts der flexiblen Ressourcen durchzuführen sind.

***CRQD**
Das Objekt Änderungsanforderungsbeschreibung (*CRQD) definiert eine Änderung, die in einem Systemnetzwerk vorzunehmen ist. Die Änderungsanforderungsbeschreibung umfasst eine oder mehrere Aktivitäten, mit denen die Änderung ausgeführt wird.

***CSI**
Das Objekt DFV-Nebeninformation (*CSI) enthält DFV-Informationen, die den Leitweg zum fernen System definieren, sowie Initialisierungsinformationen für eine DFV-Sitzung.

***CSPMAP**
Das Objekt Produktliste zwischen Systemen (*CSPMAP) enthält die Organisierung und die Merkmale von Informationen, die auf einer Anzeigen- oder Druckerseite bei der Ausführung eines Anwendungsprogramms angezeigt werden.

***CSPTBL**
Das Objekt Produkttabellen zwischen Systemen (*CSPTBL) enthält eine CSP/AE-Tabelle. Diese Tabelle ist eine Sammlung zugehöriger Datenelemente für die Gültigkeitsprüfung der Eingaben.

***CTLD**
Das Objekt Steuereinheitenbeschreibung (*CTLD) enthält die Merkmale einer Steuereinheit, die entweder direkt an das System oder an eine DFV-Leitung angeschlossen ist. Die Steuereinheitenbeschreibung wird für die Konfiguration verwendet.

***DEVD**
Das Objekt Einheitenbeschreibung (*DEVD) enthält eine Beschreibung einer an das System angeschlossenen Einheit und beschreibt alle Merkmale der Einheit. Die Einheitenbeschreibung (*DEVD) wird für die Konfiguration verwendet.

***DOC**
Das Objekt Dokument (*DOC) enthält den Text eines Dokuments sowie in der Dokumentbibliothek gespeicherte Informationen zur Beschreibung des Dokuments.

***DTAARA**
Das Objekt Datenbereich (*DTAARA) enthält einen Datenwert, der von verschiedenen Jobs verwendet und geändert werden kann.

***DTADCT**
Das Objekt Datenverzeichnis (*DTADCT) enthält Datei-, Format- und Felddefinitionen zur Beschreibung von Datenbankdateien.

***DTAQ**
Das Objekt Datenwarteschlange (*DTAQ) dient zum Übertragen und Speichern von Daten, die von verschiedenen Programmen innerhalb eines Jobs oder zwischen verschiedenen Jobs verwendet werden. Mehrere Jobs können Daten von einer einzigen Datenwarteschlange senden und empfangen.

***EDTD**
Das Objekt Editierbeschreibung (*EDTD) definiert Zahlenformate für Bildschirm- und Druckerdateien.

***EXITRG**
Das Objekt Repository des Exitregistrierungsservice (*EXITRG) enthält Exitpunkte und Benutzerexitprogramme.

***FCT**
Das Objekt Formularsteuertabelle (*FCT) enthält die Verarbeitungserfordernisse für die von einem Host-System empfangenen Daten zwecks Jobferneingabe (RJE).

***FILE**
Das Objekt Datei (*FILE) definiert eine Datenbankdatei, eine Einheitendatei oder zusammengehörige Datensätze innerhalb einer Datei.

***FLR**
Das Objekt Ordner (*FLR) dient als Verzeichnis für Dokumente und andere Ordner.

***FNTRSC**
Das Objekt Schriftartressource (*FNTRSC) enthält eine Darstellung dazu, wie Zeichen auf einer Seite abgebildet werden.

***FNTTBL**
Eine Schriftartentabelle (*FNTTBL) ist eine von PSF benutzte Tabelle, in der druckerresidente Schriftarten und Codepages den entsprechenden hostresidenten Schriftarten und Codepages zugeordnet sind.

***FORMDF**
Das Objekt Formulardefinition (*FORMDF) ist ein externes Bibliotheksressourcenobjekt in Rasterpunktarchitekturunterstützung, das die charakteristischen Merkmale des Formulars definiert.

***FTR**
Das Objekt Filter (*FTR) enthält Auswahleinträge, die Einheiten in Gruppen zusammenfassen. Das Objekt enthält ferner Aktionseinträge, die die für die Gruppe durchgeführte Aktion definieren.

***GSS**
Das Objekt Grafiksymbol (*GSS) enthält Zeichensätze, die von einem Grafikanwendungsprogramm verwendet werden können, um andere Zeichen als die Zeichen des mit dem Betriebssystem gelieferten Standardzeichensatzes anzuzeigen.

***IGCDCT**
Das Objekt Doppelbyte-Zeichensatzverzeichnis (*IGCDCT) wird von Character Generator Utility (CGU) zur Auswahl von Zeichen und Symbolen verwendet, die auf dem System benutzt werden.

***IGCSRT**
Das Objekt Doppelbyte-Zeichensatzsortiertabelle (*IGCSRT) wird von Character Generator Utility (CGU) zur Auswahl von Zeichen und Symbolen verwendet, die auf dem System benutzt werden.

***IGCTBL**
Das Objekt Doppelbyte-Zeichensatztabelle (*IGCTBL) enthält die Abbilder für die auf dem System verwendeten DBCS-Erweiterungszeichen. Das System verwendet diese Abbilder zum Anzeigen und Drucken von Erweiterungszeichen.

***IMGCLG**
Das Objekt Imagekatalog (*IMGCLG) enthält Informationen über Imagekatalogeinträge. Jeder Imagekatalogeintrag ist ein optisches Image, das als Datenstromdatei (*STMF) in einem vom Benutzer angegebenen Integrated File System-Verzeichnis vorhanden ist. Jeder Imagekatalog ist einem einzigen Integrated File System-Verzeichnis zugeordnet.

***IPXD**
Das Objekt Internetwork Packet Exchange Description (*IPXD) enthält Werte, die für die Konfiguration von IPX- und SPX-Kommunikationsprotokollen verwendet werden (SPX = Sequenced Packet Exchange). IPX- und SPX-Protokolle werden von IBM i für die Unterstützung der Novell NetWare** verwendet. Dazu gehören die Protokolle RIP (Router Information Protocol), SAP (Service Advertising Protocol), NLSP (Netware Link Services Protocol), SPX, IPX und andere NetWare-Funktionen. Eine IPX-Beschreibung wird vom Netzwerk-Server (NWS) für die NetWare-FSIOP-Verarbeitung sowie von der nativen IBM i NetWare-Unterstützung verwendet.

***JOBD**
Das Objekt Jobbeschreibung (*JOBD) enthält eine bestimmte Gruppe von jobbezogenen Attributen, die von einem bzw. mehreren Jobs verwendet werden können.

***JOBQ**
Das Objekt Jobwarteschlange (*JOBQ) enthält Einträge für zum Ablauf an das System übergebene Stapeljobs. Diese Stapeljobs werden vom Betriebssystem für den Ablauf aus der Jobwarteschlange ausgewählt.

***JOBSCD**
Das Objekt Jobplanungseintrag (*JOBSCD) enthält Einträge (Angaben zu Datum und Uhrzeit) für den Zeitpunkt, zu dem Stapeljobs an eine Jobwarteschlange übergeben werden.

***JRN**
Das Objekt Journal (*JRN) enthält Informationen zu Journal-Datenbankdateien und dient zu Datenbank- bzw. Pfadänderungen. Das System verwendet das Journal, um Daten über die Journalempfänger und Datenbankdateien bzw. den zugehörigen Pfad zu protokollieren.

***JRNRCV**
Das Objekt Journalempfänger (*JRNRCV) enthält Journaleinträge, die bei der Änderung von Datenbankdateien erstellt werden.

***LIB**
Das Objekt Bibliothek (*LIB) enthält Dateien, Programme und andere Objekte, die als Verzeichnis für andere Objekte dienen.

***LIND**
Das Objekt Leitungsbeschreibung (*LIND) definiert eine DFV-Leitung und beschreibt dem System ihre Zusatzeinrichtungen; dies umfasst die Beschreibung des an die Leitung angeschlossenen Modems (und seiner Zusatzeinrichtungen).

***LOCALE**
Das Objekt für länderspezifische Angaben (*LOCALE) enthält die Kombination aus Sprache, länderspezifischen Angaben und Zeichensatz, mit der bestimmte sprachliche Konventionen gekennzeichnet werden.

***M36**
Ein M36-Objekt (*M36) repräsentiert eine Advanced 36-Maschine. Auf einem IBM i-System können sich mehrere *M36-Objekte befinden. Jedes *M36-Objekt repräsentiert eine einzelne Advanced 36-Maschine.

***M36CFG**
Ein Objekt M36CFG (*M36CFG) definiert die Konfiguration, die beim IPL (einleitendes Programmladen) oder Starten für ein *M36-Objekt verwendet werden soll. Wurde das vorhandene System ausgehend von einer i5/OS Advanced 36, Modell 236, aufgerüstet, entspricht ein i5/OS Advanced 36-Maschinenkonfigurationsobjekt einer Konfigurationsteildatei auf einem IBM System /36. Das *M36CFG-Objekt hat den gleichen Namen und wird in der gleichen Bibliothek erstellt wie das *M36-Objekt.

***MEDDFN**
Ein Objekt Datenträgerdefinition (*MEDDFN) definiert die Datenträger und Einheiten, die für eine parallel durchgeführte Sicherung oder Rückspeicherung verwendet werden sollen.

***MENU**
Das Objekt Menü (*MENU) enthält Objekte, die zum Anzeigen in einem Menü verwendet werden.

***MGTCOL**
Ein Verwaltungserfassungsobjekt (*MGTCOL) wird von verschiedenen Systemfunktionen (z. B. Erfassungsservices, Management Central-Überwachungsfunktionen und Performance Explorer) zum Speichern von erfassten Leistungsdaten verwendet.

***MODD**
Das Objekt Modusbeschreibung (*MODD) enthält eine logische Gruppe von Sitzungen mit gemeinsamen Merkmalen, die von APPN- oder APPC-Protokollen verwendet werden.

***MODULE**
Ein Objekt der Art Modul (*MODULE) enthält eine Reihe von Anweisungen und Informationen, die erforderlich sind, wenn das Objekt zum Erstellen eines Binderprogramms verwendet wird. Modulobjekte werden als Ausgabe der CRTxxxMOD-Befehle erstellt.

***MSGF**
Das Objekt Nachrichtendatei (*MSGF) enthält Nachrichtenbeschreibungen, die zum Beschreiben der Status- und Fehlerbedingungen verwendet werden.

***MSGQ**
Das Objekt Nachrichtenwarteschlange (*MSGQ) enthält Nachrichten, die an eine Person, an ein Programm oder an das System gesendet oder von diesen abgerufen werden.

***NODGRP**
Ein Knotengruppenobjekt (*NODGRP) dient zur Identifizierung einer Gruppe von Systemen, über die eine verteilte Datei erstellt werden kann. Eine Knotengruppe enthält außerdem Informationen darüber, wie Daten über die verschiedenen Systeme (Knoten) verteilt werden. Das Knotengruppenobjekt kann beim Erstellen einer physischen Datenbankdatei mit dem Befehl CRTPF (Physische Datei erstellen) oder bei Verwendung der Anweisung CREATE TABLE in SQL (Structured Query Language) angegeben werden.

***NODL**
Das Objekt Knotenliste (*NODL) enthält eine Liste der SNA-Knoten. Der Name eines SNA-Knotens besteht aus einer Netzwerk-ID und einem Kontrollpunktnamen.

***NTBD**
Eine NetBIOS (Network Basic Input/Output System) -Beschreibung (*NTBD) enthält Werte, die als Verfahrensmerkmale für NetBIOS-Sitzungen dienen. NetBIOS-Sitzungen werden von FSIOP-Hardware- und Softwarekomponenten verwendet. Die NetBIOS-Beschreibung wird für die Konfiguration benutzt.

***NWID**
Das Objekt Netzwerkschnittstellenbeschreibung (*NWID) enthält Informationen zur Konfiguration einer Schnittstelle. Informationen, die vom System benötigt werden, um über die Schnittstelle mit einem dienstintegrierenden digitalen Fernmeldenetz (ISDN) kommunizieren zu können.

***NWSCFG**
Das NWS-Konfigurationsobjekt (*NWSCFG) definiert Attribute, die von einer NWS-Beschreibung (NWSD) mit iSCSI-Anschluss dazu verwendet werden, Verbindungssicherheit, Hardware und Konfiguration eines fernen Systems sowie Serviceprozessorattribute zu beschreiben.

***NWSD**
Das Objekt Netzwerk-Server-Beschreibung (*NWSD) enthält Informationen zur Konfiguration eines Netzwerk-Servers, Informationen, die vom System benötigt werden, um über die Systemschnittstelle kommunizieren und die Systemschnittstelle für einen FSIOP (File Serving Input/Output Processor) beschreiben zu können.

***OUTQ**
Das Objekt Ausgabewarteschlange (*OUTQ) enthält eine Liste von Spool-Dateien, die über ein Ausgabeprogramm an eine Ausgabeeinheit geschrieben werden.

***OVL**
Das Objekt Schablone (*OVL) enthält elektronische Versionen vordefinierter Informationen, z. B. von Briefköpfen etc.

***PAGDFN**
Das Objekt Seitendefinition (*PAGDFN) enthält Formatsteuerzeichen für Zeilendaten, Zeilenanzahl, Schriftartdruckrichtung und die Anordnung der Felder auf einer Seite.

***PAGSEG**
Das Objekt Seitensegment (*PAGSEG) enthält eine Darstellung der Standardteile einer Seite (z. B. Standardabsatz- und Standardunterschriftsinformationen).

***PDFMAP**
Eine PDF-Maske (*PDFMAP) ist ein von PSF verwendetes Objekt, das Spool-Dateiattributen Aktionen zuordnet, die beim Konvertieren einer Spool-Datei in eine PDF-Datei ausgeführt werden. Dazu gehört das Versenden als E-Mail, das Speichern der PDF-Datei im IFS oder das erneute Spoolen der PDF-Datei.

***PDG**
Das Objekt Druckdeskriptorgruppe (*PDG) enthält Druckdeskriptoren, die Benutzerauswahlmöglichkeiten zum Drucken enthalten.

***PGM**
Das Objekt Programm (*PGM) enthält eine Reihe von Anweisungen, die einem Computer mitteilen, wo sich eine Eingabe befindet, wie diese verarbeitet werden soll und wohin die Ergebnisse gestellt werden sollen. Programme können in verschiedenen Sprachen abgefasst sein (z. B. C, CL oder RPG).

***PNLGRP**
Das Objekt Anzeigengruppe (*PNLGRP) enthält Anzeigeninformationen oder Onlinehilfetexte, die an einem Bildschirm dargestellt werden.

***PRDAVL**
Das Produktverfügbarkeitsobjekt (*PRDAVL) enthält Datensätze aller bekannten IBM Software- und Kundenprodukte, die für das System zumindest definiert sind.

***PRDDFN**
Das Objekt Produktdefinition (*PRDDFN) enthält Produktinformationen, die geladen werden, wenn ein Lizenzprogramm auf dem System installiert wird. Zu diesen Informationen gehören Sonderumgebungen, Programmiercode und Produktsteuerinformationen, z. B. Programmname, Produkt-ID, Copyright und Modifikationsstufe.

***PRDLOD**
Das Objekt Produktlademodul (*PRDLOD) enthält Produktinformationen, die geladen werden, wenn ein Lizenzprogramm auf dem System installiert wird. Diese Informationen umfassen detaillierte Angaben u. a. zu Standort und Inhalt.

***PSFCFG**
Das Objekt PSF-Konfiguration (*PSFCFG) enthält zusätzliche Einheitenkonfigurationsdaten für einen AFP-Drucker. Diese Daten werden zusammen mit den Daten verwendet, die bei der mit dem Befehl CRTDEVPRT (Einheitenbeschreibung erstellen – Drucker) erstellten Druckereinheitenkonfiguration angegeben wurden.

***QMFORM**
Das Objekt Abfrageverwaltungsformular (*QMFORM) enthält das Format einer Abfrage oder eines Berichts, die/der vom Benutzer des Lizenzprogramms IBM Query für IBM i und der Systems Application Architecture (SAA) definiert wurde.

***QMQRY**
Das Objekt Abfrageverwaltungsabfrage (*QMQRY) enthält die Daten, die von einer Abfrage oder einem Bericht zurückgegeben werden sollen, die/der von einem Benutzer des Lizenzprogramms IBM Query für IBM i und der Systems Application Architecture (SAA) definiert wurde.

***QRYDFN**
Das Objekt Abfragedefinition (*QRYDFN) enthält Informationen über eine Abfrage oder einen Bericht, die/der von einem Benutzer des Lizenzprogramms IBM Query für IBM i definiert wurde.

***RCT**
Das Objekt Referenzcodeumsetztabelle (*RCT) enthält Informationen zur Unterstützung bei der Problemanalyse von Hardwarefehlern.

***SBSD**
Das Objekt Subsystembeschreibung (*SBSD) enthält eine Definition eines Subsystems und seiner Betriebsumgebung im System.

9.13.1.1
Seite 10

***SCHIDX**
Das Objekt Informationssuchindex (*SCHIDX) enthält Informationen, die von der Suchfunktion (STRSCHIDX) verwendet werden.

***SPADCT**
Das Objekt Benutzerdefiniertes Wörterverzeichnis (*SPADCT) enthält eine Liste von Wörtern, Synonymen und Silbentrennungsangaben.

***SQLPKG**
Das Objekt SQL-Paket (*SQLPKG) enthält Informationen, die von SQL zum Durchführen einer Anforderung eines Programms oder eines fernen Systems benötigt werden.

***SQLUDT**
Ein SQL-Objekt der Art UDT (User Defined Type) (*SQLUDT) enthält Informationen zu einer bestimmten Datenart, die in SQL mit der Anweisung CREATE DISTINCT TYPE erstellt wird. Diese Informationen werden von SQL verwendet, um Operationen unter Verwendung dieser speziellen Datenart durchzuführen.

***SQLXSR**
Ein *SQLXSR-Objekt (SQL Extensible Markup Language Schema Repository) enthält Informationen zur Verwendung einer XML-Spalte in einer Tabelle. Die Informationen im Objekt SQLXSR enthalten die Definition zur Strukturierung der Daten in der XML-Spalte der Tabelle. Beim Einfügen von Daten in die XML-Spalte werden sie anhand der Definition im Objekt SQLXSR validiert.

***SRVPGM**
Ein Serviceprogrammobjekt (*SRVPGM) enthält Anweisungen, die einem Computer mitteilen, wo sich die Eingabe befindet, wie diese verarbeitet werden soll und wohin die Ergebnisse gestellt werden sollen. Ein Serviceprogramm wird mit dem Befehl CRTSRVPGM (Serviceprogramm erstellen) erstellt. Es kann nicht auf dieselbe Weise wie ein Objekt *PGM aufgerufen werden.

***SSND**
Das Objekt Sitzungsbeschreibung (*SSND) enthält für die Funktion Jobferneingabe (RJE) alle Objekte und Einheiten für die RJE-Betriebsumgebung.

***SVRSTG**
Das Objekt Server-Speicherbereich (*SVRSTG) enthält Systemdateien und Daten, die von einer Netzwerk-Server-Beschreibung benutzt werden.

***S36**
Ein Objekt IBM System /36 (*S36) enthält Konfigurationsdaten, die von der System /36-Umgebung zum Ausführen von System /36-Jobs auf einem IBM i-System verwendet werden.

***TBL**
Das Objekt Umsetztabelle (*TBL) enthält eine Gruppe von Werten für Byte-to-Byte-Daten, oder zum Definieren einer Sortierfolge.

***TIMZON**
Das Objekt Zeitzonenbeschreibung (*TIMZON) enthält Informationen zur Berechnung der Ortszeit.

***USRIDX**
Das Objekt Benutzerindex (*USRIDX) enthält Suchfunktionen und ordnet Daten automatisch gemäß ihrem Wert an.

***USRPRF**
Das Objekt Benutzerprofil (*USRPRF) enthält Angaben, die einen Benutzer gegenüber dem System kennzeichnen, und steuert, welche Systemfunktionen und Anzeigen dem Benutzer zur Verfügung stehen.

***USRQ**
Das Objekt Benutzerwarteschlange (*USRQ) enthält eine Liste der Nachrichten für andere Anwendungsprogramme.

***USRSPC**
Das Objekt Benutzerbereich (*USRSPC) enthält eine große Menge Daten, die vom Benutzer geändert und verwendet werden können.

***VLDL**
Das Objekt Prüfliste (*VLDL) enthält Daten, die eine ID beinhalten, Daten, die beim Speichern verschlüsselt werden, und Daten im freien Format.

***WSCST**
Das Objekt Datenstationsbenutzeranpassung (*WSCST) enthält angepasste Tabellen mit Informationen über lokale und ferne Datenstationssteuereinheiten zur Unterstützung weiterer Bildschirme und Drucker oder neudefinierter Tastaturen.

9.13.1.1
Seite 12

Objekte im integrierten Dateisystem

***BLKSF**
Eine blockorientierte Datei (*BLKSF) repräsentiert ein benutzerdefiniertes Dateisystem, das sich in einem bestimmten Zusatzspeicherpool (ASP) befindet. Diese Art von Datei ermöglicht es, das benutzerdefinierte Dateisystem als Einheit anzuzeigen und zu verwalten. Die Benutzer erhalten Zugriff auf ein benutzerdefiniertes Dateisystem, indem die blockorientierte Datei an den Namensbereich des integrierten Dateisystems (Integrated File System – IFS) angehängt wird.

***CHRSF**
Ein zeichenorientiertes Gerätedateiobjekt (*CHRSF) ist einer Einheit oder Ressource eines Computersystems zugeordnet. Es hat Pfadnamen, die in Verzeichnissen erscheinen und verwendet denselben Zugriffsschutz wie reguläre Dateien.

***DDIR**
Das Objekt Verteiltes Verzeichnis (*DDIR) wird im IFS verwendet. Ein *DDIR-Objekt kann auf einem System vorhanden sein, und die Systeme im Netzwerk können darauf zugreifen.

***DIR**
Das Objekt Verzeichnis (*DIR) wird im IFS verwendet.

***DSTMF**
Das Objekt Verteilte Datenstromdatei (*DSTMF) wird im IFS verwendet. Ein *DSTMF-Objekt kann auf einem System vorhanden sein, und die Systeme im Netzwerk können darauf zugreifen. Eine Datenstromdatei enthält einen kontinuierlichen Datenstrom. Beispiele für Datenstromdateien:
– Dokumente, die in IBM i-Ordnern gespeichert sind
– PC-Dateien
– UNIX-Dateien

***FIFO**
Ein First In/First Out-Objekt (*FIFO) ist ein von UNIX definierter Dateityp. Dabei handelt es sich um eine Datei, die für Lese- und Schreibzugriffe geöffnet werden kann und die die Eigenschaft hat, dass alle Daten nach dem FIFO-Verfahren gelesen werden. Dabei werden die Daten, die am längsten vorhanden sind, zuerst gelesen. Sind die Daten gelesen, werden sie aus der Datei entfernt. Alle übrigen Daten werden gelöscht, sobald alle geöffneten Instanzen geschlossen sind. Sind keine Daten verfügbar oder ist die FIFO-Datei zu einem bestimmten Zeitpunkt voll, können bei Lese- und Schreibzugriffen Blockierungen auftreten.

9.13.1.2
Seite 2

***SOCKET**
Das Objekt Lokaler Socket (*SOCKET) ermöglicht die gleichzeitige Ausführung zweier Jobs auf dem gleichen System, um eine Kommunikationsverbindung zueinander herzustellen. Das Objekt wird zugeordnet, wenn einer der Jobs die Socket-Funktion bind() verwendet.

Wenn die Verbindung hergestellt ist, können die Jobs Daten untereinander austauschen, indem sie Socket-Funktionen wie sendto() und recvmsg() oder die Socket-Funktion connect() gefolgt von Funktionen des IFS, wie write() und read(), verwenden.

Ist die Kommunikation der Jobs beendet, verwendet jeder Job die Funktion close(), um die Socket-Verbindung zu beenden. Der lokale Socket verbleibt so lange im System, bis er mit der Funktion unlink() oder dem Befehl RMVLNK (Verbindung entfernen) entfernt wird.

Ein lokaler Socket kann nicht gesichert werden.

***STMF**
Das Objekt Datenstromdatei (*STMF) wird im integrierten Dateisystem verwendet. Eine Datenstromdatei enthält einen kontinuierlichen Datenstrom. Beispiele für Datenstromdateien:
– Dokumente, die in IBM i-Ordnern gespeichert sind
– PC-Dateien
– UNIX-Dateien

***SYMLNK**
Eine symbolische Verbindung (*SYMLNK) enthält den Pfadnamen eines IFS, der auf ein anderes Objekt verweist. Das andere Objekt kann vorhanden sein oder nicht. Wenn über die symbolische Verbindung auf ein Objekt verwiesen wird, verwendet das System den in der symbolischen Verbindung angegebenen Pfadnamen, um das Objekt zu lokalisieren.

Sie sehen, es gibt eine schier unüberschaubare Anzahl verschiedener Objekttypen, die man mehr oder weniger oft benötigt. Ehrlich gesagt, viele dieser Objekttypen habe ich noch nie benutzt bzw. benötigt.

Allen Objekttypen ist gemeinsam, dass man Benutzern, Gruppen oder Berechtigungslisten Rechte auf sie zuweisen kann.

Objektberechtigungen

Um Rechte für ein Objekt zu vergeben, kann man verschiedene Möglichkeiten nutzen, die teilweise abhängig vom Typ des Objekts sind. So unterscheidet man grundsätzlich die Berechtigungsvergaben für Objekte, die in Bibliotheken gespeichert und z. B. mit **WRKOBJ** bearbeitet werden, von Rechten für IFS-Objekte, die z. B. über **WRKLNK** vergeben werden.

9.13.2
Seite 2

Berechtigungen für Bibliotheksobjekte

Wollen Sie sich die Berechtigungen für Objekte anzeigen lassen, verwenden Sie den Befehl DSPOBJAUT (Objektberechtigung anzeigen).

```
                    Objektberechtigung anzeigen (DSPOBJAUT)

Auswahl eingeben und Eingabetaste drücken.

Objekt . . . . . > SMZ4          Name
  Bibliothek . . . . . *LIBL     Name, *LIBL, *CURLIB
Objektart . . . . . > *LIB       *ALRTBL, *AUTL, *BNDDIR...
ASP-Einheit . . . . . *          Name, *, *SYSBAS
Ausgabe . . . . . *              *, *PRINT, *OUTFILE
```

Auswahl bei Objektberechtigungen anzeigen

Nach Eingabe der Auswahlparameter erhalten Sie die Anzeige der Berechtigungen.

```
                        Objektberechtigung anzeigen

Objekt . . . . . :   SMZ4          Eigner . . . . . :   SECURITY2P
  Bibliothek . . . . :   QSYS      Primärgruppe . . . . :   *NONE
Objektart . . . . . :   *LIB       ASP-Einheit . . . . :   *SYSBAS

Objekt durch Berechtigungsliste geschützt . . . . . :   *NONE

                       Objektbe-
Benutzer    Gruppe     rechtigung
*PUBLIC                 *USE
SECURITY2P              *ALL




                                                              Ende
Eingabetaste --> Weiter

F3=Verlassen    F11=Objektberechtigungen detailliert anzeigen    F12=Abbrechen
F17=Anfang      F18=Ende
```

Objektberechtigungen

Die Bedeutung der Werte entnehmen Sie der nachstehenden Erläuterung des Befehls „Objektberechtigungen editieren".

Die Anzeige „Objektberechtigung editieren", die mit Auswahl 2 aus **WRKOBJ** oder über den Befehl **EDTOBJAUT** aufgerufen werden kann, zeigt eine Liste der Benutzer an, die momentan objektberechtigt sind, sowie die Berechtigungsebenen, die ihnen jeweils zugewiesen wurden. Benutzer können der Liste hinzugefügt oder aus ihr entfernt werden, und die Berechtigungsebenen der Benutzer können geändert werden.

9.13.2.1
Seite 2

Bei einer Änderung der allgemeinen Berechtigung für von IBM gelieferte Objekte können Probleme auftreten. Wird beispielsweise die allgemeine Berechtigung der Nachrichtenwarteschlange QSYSOPR in eine Berechtigung geändert, die weniger Rechte umfasst als die Berechtigung *CHANGE, so können Fehler bei Systemprogrammen auftreten. Die Berechtigung der Systemprogramme reicht nämlich nicht aus, um z. B. Nachrichten an die Nachrichtenwarteschlange QSYSOPR zu senden.

Berechtigungsebenen können auf zwei Arten angegeben werden:

1. Einen der vordefinierten Systemwerte für Objektberechtigungsebenen zuordnen (*USE, *ALL, *CHANGE oder *EXCLUDE).
2. Benutzerdefinierte Berechtigungsebenen zuordnen. Benutzerdefinierte Berechtigungsebenen werden erstellt, indem die Spalten, die nach dem Drücken der Funktionstaste 11 beim Anzeigen von Details erscheinen, mit X gefüllt bzw. die X daraus entfernt werden.

Bei allen Verfahren bestimmt das System die Objektberechtigungswerte bzw. die speziellen Berechtigungen aufgrund der eingegebenen Informationen. Wenn für einen Benutzer sowohl die Objektberechtigungswerte als auch die speziellen Berechtigungen eingegeben werden, werden stets die speziellen Berechtigungen verwendet.

Mit der Taste F11 kann durch drei verschiedene Versionen der Objektberechtigungsanzeigen geblättert werden:

- Anzeige „Keine Details anzeigen“. Diese Anzeige enthält systemdefinierte Berechtigungsuntergruppen und USER DEF:

Benutzer	Gruppe	Objekt-berechtg.
*PUBLIC		*USE
QSYS		*ALL

- Anzeige „Objektberechtigung”:

			Objekt-	----------Objekt----------				
Benutzer	Gruppe		berechtg.	Opr	Verw.	Exist.	Ändern	Referenz
*PUBLIC			*USE	X				
QSYS			*ALL	X	X	X	X	X

- Anzeige „Datenberechtigung”:

		Objekt-	----------Daten----------				
Benutzer	Gruppe	berechtg.	Lesen	Hinzuf	Aktual	Lösch.	Ausf.
*PUBLIC		*USE	X				X
QSYS		*ALL	X	X	X	X	X

Um die Berechtigungsebene eines Benutzers zu ändern, werden die entsprechenden Spalten der Benutzerzeile überschrieben. Außer bei der Spalte „Objektberechtigung" wird dem Benutzer durch Eingabe eines „X" in die Spalte die genannte Berechtigung erteilt bzw. durch Eingabe einer Leerstelle in die Spalte die Berechtigung entzogen. Um die Spalte „Objektberechtigung" zu ändern, überschreibt man einfach die alte Berechtigung durch die neue.

Um der Liste Benutzer hinzuzufügen, drücken Sie die Funktionstaste 6 (Neue Benutzer hinzufügen). Die daraufhin erscheinende Anzeige erlaubt die Eingabe neuer Benutzer und Berechtigungsebenen.

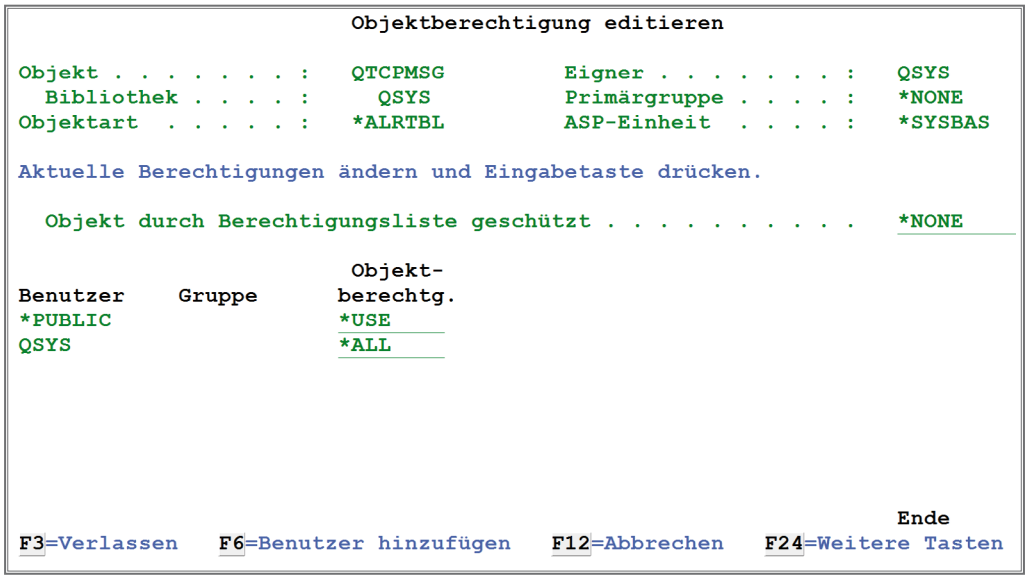
```
                          Neue Benutzer hinzufügen

Objekt . . . . . :   #CGULIB          Eigner . . . . . :   QSYS
  Bibliothek . . . :     QSYS         Primärgruppe . . . :   *NONE
Objektart  . . . . :   *LIB           ASP-Einheit  . . . :   *SYSBAS

Neue Benutzer eingeben und Eingabetaste drücken.

              Objekt-
Benutzer      berechtg.
              _________
              _________
              _________
              _________
              _________
              _________
              _________
              _________
              _________
              _________
                                                          Weitere ...
F3=Verlassen    F11=Objektberechtigungen detailliert anzeigen    F12=Abbrechen
F17=Anfang      F18=Ende
```

Neue Benutzer hinzufügen

Um Benutzer aus der Liste zu entfernen, werden in alle angezeigten Berechtigungsspalten Leerstellen eingegeben (auch in die Spalte „Objektberechtigung"). Erscheint „Weitere..." rechts unten in der Anzeige, sind weitere Informationen vorhanden. Taste „Bild auf" (Vorblättern) drücken, um zum Ende der angezeigten Informationen zu gelangen. Taste „Bild ab" (Zurückblättern) drücken, um zum Anfang zurückzukehren.

Das System ändert die Berechtigung nur, nachdem die Eingabetaste gedrückt wurde. Wenn bei der Anzeige einer Liste nach Eingabe von Änderungen die Eingabetaste erneut gedrückt wird, so wird die Liste mit den entsprechenden Änderungen angezeigt. Nach erneutem Drücken der Eingabetaste – ohne Verändern der angezeigten Liste durch Eingabe – wird an den Ausgangspunkt zurückgekehrt, von dem aus die Editierfunktion angefordert wurde.

9.13.2.1
Seite 4



Objektberechtigung editieren

Objekt
Name des Objekts, für das Informationen angezeigt werden.

Bibliothek
Der Name der Bibliothek, die dieses Objekt enthält.

Objektart
Die Objektart. Sie beschreibt die Art des angezeigten Objekts.

Eigner
Der Eigner des genannten Objekts.

Primärgruppe
Der Name des Benutzerprofils, das die Primärgruppe des Objekts ist. Die Primärgruppe kann mit dem Befehl CHGOBJPGP (Objektprimärgruppe ändern) geändert werden.

ASP-Einheit
Der Name der ASP-Einheit (ASP = Zusatzspeicherpool), in der sich das Objekt befindet. *SYSBAS bedeutet, dass sich das Objekt im System-ASP oder einem Basis-Benutzer-ASP befindet.

Berechtigungsliste
Der Name der Berechtigungsliste, mit der das genannte Objekt gesichert wird. Der Wert *NONE zeigt an, dass bei der Bestimmung der Berechtigung für das Objekt keine Berechtigungsliste verwendet wird.

Den gezeigten Wert überschreiben, um die Berechtigungsliste zu ändern, die zur Sicherung des Objekts verwendet wird. *NONE oder Leerstellen eingeben, falls keine Liste verwendet werden soll.

Benutzer
Die Benutzerprofilnamen der Benutzer, die über spezielle Berechtigungen für dieses Objekt verfügen. Der Wert *PUBLIC legt die Berechtigungen aller Benutzer fest, die nicht speziell genannt werden und keine Mitglieder der Berechtigungsliste sind, die zur Sicherung dieses Objekts verwendet wird. Der Wert *GROUP bestimmt die Berechtigungen für das Gruppenprofil des aktuellen Benutzers, das in der Spalte „Gruppe" erscheint. *GROUP kann mehrmals erscheinen, wenn der Benutzer die Berechtigung für das Objekt aus mehreren Gruppen erhält.

Der Wert *ADOPT definiert alle Berechtigungen, die das Objekt vom aktuellen Job übernimmt. Wird *ADOPT angezeigt, führt dieser Job ein Programm aus, welches das Benutzerprofil des Programmeigners angenommen hat und dessen Eigner für das Objekt berechtigt ist.

Gruppe
Der Name des Gruppenprofils, aus dem der Benutzer die Berechtigung erhält. Der Name eines Gruppenprofils wird angezeigt, wenn der Wert der Benutzerspalte *GROUP lautet.

Objektberechtigung
Die Berechtigungen, die ein Benutzer für ein Objekt besitzt. Benutzern können mehrere verschiedene Objektberechtigungsebenen zugeordnet werden. Dabei handelt es sich um folgende Ebenen:

***ALL**
Erlaubt alle Operationen am Objekt bis auf diejenigen, die auf den Eigner begrenzt sind oder die durch die Berechtigung zur Verwaltung der Berechtigungsliste kontrolliert werden.

***CHANGE**
Erlaubt alle Operationen am Objekt bis auf diejenigen, die auf den Eigner begrenzt sind oder die durch die Objektexistenzberechtigung, die Objektänderungsberechtigung, die Objektreferenzberechtigung und die Objektverwaltungsberechtigung gesteuert werden.

***EXCLUDE**
Alle Operationen am Objekt sind verboten.

***USE**
Erlaubt den Zugriff auf die Objektattribute sowie die Benutzung des Objekts. Der Benutzer kann das Objekt nicht ändern.

USER DEF
Wird vom System angezeigt, wenn die Sonderobjektberechtigungen und Sonderdatenberechtigungen keiner der vordefinierten oben angegebenen Objektberechtigungsebenen entsprechen. Die Sonderberechtigungen werden angezeigt, wenn die Funktionstaste „Details anzeigen" gedrückt wird.

Der Wert *AUTL ist auch gültig, wenn allgemeine Berechtigungen definiert werden. Er zeigt an, dass die in der von diesem Objekt verwendeten Berechtigungsliste angegebenen Angaben zur allgemeinen Berechtigung verwendet werden sollen, um die Berechtigung festzulegen.

Die Objektberechtigung kann geändert werden, indem der aktuelle Wert mit einem neuen Wert überschrieben wird.

Berechtigungen auf das Objekt
Die speziellen Berechtigungen, über die der Benutzer für das Objekt verfügt. Ein „X" in der Spalte zeigt an, dass der Benutzer die angegebene Berechtigung für das aufgeführte Objekt besitzt.

Bei bestimmten Objekten mit der Angabe *FILE wird möglicherweise ein Schrägstrich (/) in der Spalte angezeigt. Er besagt, dass der Benutzer über die angegebene Berechtigung für einige, aber nicht alle Felder in der Datei verfügt.

Die speziellen Datenberechtigungen sind:

Opr
Objektverwendungsberechtigung (gelegentlich abgekürzt als Objopr) bietet die Berechtigung, sich die Objektattribute anzeigen zu lassen und das Objekt so zu verwenden, wie es durch die Datenberechtigung des Benutzers für das Objekt angegeben ist.

Verwaltung
Objektverwaltungsberechtigung umfasst die Berechtigung, die Sicherheitsstufe anzugeben, das Objekt zu verschieben oder es umzubenennen sowie dem Objekt Teildateien hinzuzufügen, wenn es sich um eine Datenbankdatei handelt.

Existenz
Objektexistenzberechtigung umfasst die Berechtigung, die Objektexistenz und das Eigentumsrecht zu kontrollieren.

Änderung
Objektänderungsberechtigung umfasst die Berechtigung zum Ändern der Attribute eines Objekts, wie z. B. Auslöser für eine Datenbankdatei hinzufügen oder entfernen und einer Datenbankdatei Teildateien hinzufügen.

Referenz
Objektreferenzberechtigung umfasst die Berechtigung, das Objekt als erste Ebene in einer referenziellen Integritätsbedingung anzugeben.

Durch Eingabe von „X" bzw. einer Leerstelle können spezielle Objektberechtigungen erteilt oder entfernt werden.

Berechtigung auf Daten
Die speziellen Berechtigungen eines Benutzers für die in einem Objekt enthaltenen Daten. Ein „X“ in der Spalte zeigt an, dass der Benutzer die angegebene Berechtigung besitzt.

Bei bestimmten Objekten mit der Angabe *FILE wird möglicherweise ein Schrägstrich (/) in der Spalte angezeigt. Ein Schrägstrich zeigt an, dass der Benutzer über die angegebene Berechtigung für einige, aber nicht alle Felder in der Datei verfügt.

Die speziellen Datenberechtigungen sind:

Lesen
Die Leseberechtigung erlaubt den Zugriff auf den Inhalt eines Objekts.

Hinzufügen
Die Hinzufügeberechtigung erlaubt es, einem Objekt Einträge hinzuzufügen.

Aktualisieren
Die Aktualisierungsberechtigung erlaubt es, den Inhalt der vorhandenen Einträge eines Objekts zu ändern.

Löschen
Die Löschberechtigung erlaubt es, Einträge aus einem Objekt zu entfernen.

Ausführen
Die Ausführungsberechtigung umfasst die Berechtigung zur Ausführung eines Programms oder zur Suche in einer Bibliothek oder in einem Verzeichnis.

Durch Eingabe von „X“ bzw. einer Leerstelle können Datenberechtigungen erteilt oder entfernt werden.

Objektberechtigungen können jedoch nicht nur interaktiv vergeben werden, sondern werden oftmals auch via Command zugewiesen. Dafür gibt es den Befehl **GRTOBJAUT**.

9.13.2.1
Seite 8

```
                    Objektberechtigung erteilen (GRTOBJAUT)

Auswahl eingeben und Eingabetaste drücken.

Objekt . . . . . . . . . . . . . .                 Name, generisch*, *ALL
  Bibliothek . . . . . . . . . . .    *LIBL         Name, *LIBL, *CURLIB, *ALL...
Objektart  . . . . . . . . . . . .                 *ALL, *ALRTBL, *BNDDIR...
ASP-Einheit  . . . . . . . . . . .  *              Name, *, *SYSBAS
Benutzer . . . . . . . . . . . . .                 Name, *PUBLIC
     + für weitere Werte
Berechtigung . . . . . . . . . . .  *CHANGE        *CHANGE, *ALL, *USE...
     + für weitere Werte
Berechtigungsliste . . . . . . . .                 Name, *NONE
Bezugsobjekt . . . . . . . . . . .                 Name
  Bibliothek . . . . . . . . . . .    *LIBL         Name, *LIBL, *CURLIB
Bezugsobjektart  . . . . . . . . .  *OBJTYPE       *OBJTYPE, *ALRTBL, *BNDDIR...
Referenz-ASP-Einheit . . . . . . .  *              Name, *, *SYSBAS
Berechtigung ersetzen  . . . . . .  *NO            *NO, *YES


                                                                  Ende
F3=Verlassen   F4=Bedienerf.   F5=Aktualisieren   F12=Abbrechen
F13=Verwendung der Anzeige     F24=Weitere Tasten
```

Objektberechtigungen erteilen

Hier können Sie die oben beschriebenen Parameterwerte einem Objekt direkt über die Befehlsschnittstelle zuweisen.

Berechtigungen werden erteilt für:

- Benannte Benutzer
- Benutzer (*PUBLIC), die keine besonderen Berechtigungen für ein Objekt oder eine Berechtigungsliste haben.
- Benutzer eines Objekts, auf das durch die Parameter Bezugsobjekt (REFOBJ) und Bezugsobjektart (REFOBJTYPE) verwiesen wird.
- Berechtigungslisten

Bei Angabe von AUT(*AUTL) wird die allgemeine Berechtigung für ein Objekt aus der allgemeinen Berechtigung der Berechtigungsliste übernommen, die das Objekt schützt.

Mit dem Parameter AUTL wird ein Objekt durch eine Berechtigungsliste geschützt oder die Berechtigungsliste für ein Objekt entfernt.

Benutzerprofile können nicht durch eine Berechtigungsliste (*AUTL) geschützt werden.

Dieser Befehl kann von einem Objekteigner oder von einem Benutzer mit Objektverwaltungsberechtigung für ein angegebenes Objekt verwendet werden. Ein Benutzer mit Objektverwaltungsberechtigung kann anderen Benutzern jede Berechtigung erteilen, über die er selbst verfügt, mit Ausnahme der Objektverwaltungsberechtigung. Nur der Eigner eines Objekts oder ein Benutzer mit der Sonderberechtigung für alle Objekte (*ALLOBJ) kann einem Benutzer die Objektverwaltungsberechtigung erteilen.

Ein Benutzer mit der Berechtigung *ALL kann eine neue Berechtigungsliste zuordnen.

Beim Erteilen von Benutzerberechtigungen wird mit dem Parameter REPLACE festgelegt, ob die angegebenen Berechtigungen die Berechtigungen, über die der Benutzer verfügt, ersetzen sollen. Mit dem Standardwert REPLACE(*NO) wird die angegebene Berechtigung erteilt, aber keine Berechtigung entzogen, die die angegebene überschreitet, es sei denn, die Berechtigung *EXCLUDE wird erteilt.

Mit REPLACE(*YES) werden dem Benutzer seine aktuellen Berechtigungen entzogen und dann die angegebene Berechtigung erteilt. Bei der Berechtigungsvergabe über ein Bezugsobjekt wird mit diesem Befehl die angegebene Berechtigung erteilt, aber keine Berechtigung, die die angegebene überschreitet, entzogen, es sei denn, die Berechtigung *EXCLUDE wird erteilt.

Mit diesem Befehl wird die angegebene Berechtigung erteilt, aber es werden keine Berechtigungen entzogen, die mehr Rechte umfassen als die angegebenen Berechtigungen. Eine Ausnahme besteht beim Erteilen der Berechtigung *EXCLUDE oder der Angabe REPLACE(*YES).

Einschränkungen:

- Dieser Befehl **GRTOBJAUT** muss eine exklusive Sperre für eine Datenbankdatei erhalten, bevor einem Benutzer die Lese- oder Objektverwendungsberechtigung erteilt werden kann.
- Fordert ein Benutzer die Berechtigung für einen anderen angegebenen Benutzer für eine Einheit an, die momentan von einem weiteren berechtigten Benutzer verwendet wird, so wird keine Berechtigung für die Einheit erteilt.
- Die Objektart *AUTL kann nicht angegeben werden.
- AUT(*AUTL) ist nur gültig in Verbindung mit USER(*PUBLIC).
- Ein Benutzer muss entweder der Eigner des Objekts sein oder über die Berechtigung *ALL verfügen, um den Parameter AUTL verwenden zu können.
- Der Benutzer muss über die Objektverwaltungsberechtigung für das Objekt verfügen.
- Handelt es sich bei dem Objekt um eine Datei, muss der Benutzer über die Objektverwendungs- und die Objektverwaltungsberechtigung verfügen.

9.13.2.1
Seite 10

- Für Datensichtgeräte oder Nachrichtenwarteschlangen von Datenstationen, die einem Datensichtgerät zugeordnet sind, gilt außerdem, dass diesem Befehl der Befehl ALCOBJ (Objekt zuordnen) vorangehen und der Befehl DLCOBJ (Objekt freigeben) folgen muss, wenn dieser Befehl GRTOBJAUT nicht an der Einheit eingegeben wird, für die die Berechtigungen erteilt werden.
- Für die ASP-Einheit (ASP = Zusatzspeicherpool) ist die Berechtigung *USE erforderlich, sofern eine solche Einheit angegeben ist.

Es wird empfohlen, die allgemeine Berechtigung für von IBM gelieferte Objekte nur nach Prüfung aller möglichen Folgen zu ändern. Wird die allgemeine Berechtigung für die Nachrichtenschlange QSYSOPR in eine einschränkendere Berechtigung als *CHANGE geändert, können einige Systemprogramme beispielsweise nicht mehr ausgeführt werden. Die Systemprogramme verfügen dann nicht mehr über die erforderliche Berechtigung zum Senden von Nachrichten an die Nachrichtenwarteschlange QSYSOPR. Weitere Informationen enthält das Handbuch System i Security Reference, IBM Form SC41-5302.

Berechtigungen für IFS-Objekte

Wollen Sie sich die Berechtigungen für IFS-Objekte anzeigen lassen, verwenden Sie den Befehl DSPAUT (Objektberechtigung anzeigen).

```
                        Berechtigung anzeigen (DSPAUT)

 Auswahl eingeben und Eingabetaste drücken.

 Objekt . . . . . . . . . . . . .   '/temp'
_
 Symbolische Verbindung . . . . .   *NO           *NO, *YES
 Ausgabe  . . . . . . . . . . . .   *             *, *PRINT
```

Auswahl bei IFS-Objektberechtigungen anzeigen

Nach Eingabe der Auswahlparameter erhalten Sie die Anzeige der Berechtigungen.

```
                            Berechtigung anzeigen

Objekt . . . . . . . . . . . . . :   /temp
Art  . . . . . . . . . . . . . . :   DIR
Eigner . . . . . . . . . . . . . :   QSECOFR
Primärgruppe . . . . . . . . . . :   *NONE
Berechtigungsliste . . . . . . . :   *NONE

            Daten-     ----Objektberechtigungen----
Benutzer    berecht.   Exist  Verw  Änder  Referenz
*PUBLIC     *RWX         X     X      X       X
QSECOFR     *RWX         X     X      X       X









                                                                        Ende
Eingabetaste --> Weiter

F3=Verlassen    F11=Datenberechtigungen detailliert anzeigen    F12=Abbrechen
F17=Anfang      F18=Ende
```

IFS-Objektberechtigungen

Die Bedeutung der Werte entnehmen Sie der nachstehenden Erläuterung des Befehls „IFS-Objektberechtigungen editieren".

Die Anzeige „Mit Berechtigungen arbeiten", die mit Auswahl 9 aus **WRKLNK** vor einem Objekt aufgerufen werden kann, enthält eine Liste der Benutzer, die über die Berechtigung für das angeforderte Objekt verfügen, sowie die speziellen Berechtigungen dieser Benutzer. Die gleiche Anzeige kann über den Befehl **WRKAUT** erreicht werden, bei dem allerdings der Objektname als Parameterwert mitgegeben werden muss.

9.13.2.2
Seite 2

Die allgemeine Berechtigung, die Eignerberechtigung und die Berechtigung der Primärgruppe werden ebenfalls angezeigt.

Es können auch Parameter in die Befehlszeile eingegeben und die Eingabetaste gedrückt werden, um den Befehl sofort auszuführen. Sollen weitere Parameterwerte angegeben werden, die Funktionstaste 4 (Bedienerführung) drücken. Wenn Parameter eingegeben werden und mehr als eine Auswahl getroffen wird, gelten die Parameter für alle eingegebenen Auswahlmöglichkeiten.

Um einen Befehl unmittelbar auszuführen, den Befehl eingeben und die Eingabetaste drücken. Für Hilfe bei der Befehlseingabe: den Befehl eingeben und Funktionstaste 4 (Bedienerführung) drücken.

Mit der Taste F11 kann durch zwei verschiedene Versionen der Berechtigungsanzeigen geblättert werden:

- Die Anzeige „Objektberechtigung anzeigen“, in der die systemdefinierten Datenberechtigungsgruppen und die Objektberechtigungen aufgelistet werden.
- Die Anzeige „Datenberechtigungen detailliert anzeigen“.

Ist diese Anzeige abgeschlossen, kehrt der Benutzer durch Drücken der Eingabetaste wieder in die Anzeige zurück, von der aus diese Anzeige aufgerufen wurde.

```
                          Mit Berechtigung arbeiten

Objekt . . . . . :   /alex
Art  . . . . . :   DIR
Eigner . . . . . :   RAZLEEIL
Primärgruppe . . . . . :   *NONE
Berechtigungsliste . . . . . :   *NONE

Auswahl eingeben und Eingabetaste drücken.
  1=Benutzer hinzufügen   2=Benutzerberechtigung ändern
  4=Benutzer entfernen

                  Daten-    -------------Datenberechtigungen-------------
Ausw  Benutzer    berecht.  Objopr  Lesen  Hinzu  Aktual  Lösch.  Ausführ.
 _    *PUBLIC     *RWX        X       X      X      X       X        X
 _    RAZLEEIL    *RWX        X       X      X      X       X        X
 _    QDIRSRV     *X          X                                       X
 _                                                                   Ende
Parameter oder Befehl
===>
F3=Verlassen    F4=Bedienerführung    F5=Aktualisieren    F9=Auffinden
F11=Objektberechtigungen anzeigen     F12=Abbrechen       F24=Weitere Tasten
```

Objektberechtigung für IFS-Objekte

Objekt
Der Pfadname des Objekts, für das Informationen angezeigt werden sollen.

Art
Die Art des Objekts.

Häufig vorkommende Objektarten, die wahrscheinlich in dieser Anzeige enthalten sind:

AUTL	Berechtigungsliste
BLKSF	Blockorientierte Einheitendatei
CFGL	Konfigurationsliste
CHRSF	Zeichenorientierte Einheitendatei
CLS	Klasse
CMD	Befehl
CTLD	Steuereinheitenbeschreibung
DDIR	Verteiltes Verzeichnis
DEVD	Einheitenbeschreibung
DIR	Verzeichnis
DOC	Dokument
DSTMF	Verteilte Datenstromdatei
FIFO	First-In/First-Out-Einheitendatei
FILE	Datenbank- oder Einheitendatei
FLR	Ordner
JOBD	Jobbeschreibung
JOBQ	Jobwarteschlange
LIB	Bibliothek
LIND	Leitungsbeschreibung
MSGQ	Nachrichtenwarteschlange
OUTQ	Ausgabewarteschlange
PGM	Programm
SBSD	Subsystembeschreibung
STMF	Datenstromdatei
SYMLNK	Symbolische Verbindung
USRPRF	Benutzerprofil

9.13.2.2
Seite 4

Datenbankteildateien werden ebenfalls als Objektart MBR angesehen. Einzelheiten über alle Objektarten enthält das Handbuch CL-Themensammlung in der Kategorie Programmierung im IBM i Information Center unter http://www.ibm.com/systems/i/infocenter/.

Eigner
Der Eigner des genannten Objekts.

Primärgruppe
Der Name des Benutzerprofils, das die Primärgruppe des Objekts ist. Die Primärgruppe kann mit dem Befehl CHGPGP (Primärgruppe ändern) geändert werden.

Berechtigungsliste
Der Name der Berechtigungsliste, mit der das genannte Objekt gesichert wird. Der Wert *NONE zeigt an, dass bei der Bestimmung der Berechtigung für das Objekt keine Berechtigungsliste verwendet wird.

Benutzer
Die Namen der Benutzer, die zur Verwendung des Objekts berechtigt sind. Der Wert *PUBLIC zeigt die Berechtigungen der Benutzer an, die nicht speziell genannt werden und sich nicht auf der Berechtigungsliste des Objekts befinden. Der Wert *NOUSRRPRP zeigt an, dass das Benutzerprofil, das eine Berechtigung für dieses Objekt besitzt, auf diesem System nicht vorhanden ist.

Berechtigung auf Daten
Die Datenberechtigung, über die der Benutzer für ein Objekt verfügt. Benutzern können verschiedene systemdefinierte Datenberechtigungsebenen zugeordnet werden. Dabei handelt es sich um folgende Ebenen:

***RWX**
Erlaubt die Ausführung sämtlicher Operationen für das Objekt, mit Ausnahme der Operationen, die auf den Eigner beschränkt sind oder durch die Objektrechte gesteuert werden.

***RX**
Erlaubt Zugriff auf die Objektattribute sowie die Benutzung des Objekts. Der Benutzer kann das Objekt nicht ändern.

***RW**
Erlaubt Zugriff auf die Objektattribute sowie die Änderung des Objekts. Der Benutzer hat keine Berechtigung zum Verwenden des Objekts.

***WX**
Erlaubt die Verwendung und die Änderung des Objekts. Der Benutzer hat keinen Zugriff auf die Objektattribute.

***R**
Erlaubt Zugriff auf die Objektattribute.

***W**
Erlaubt die Änderung des Objekts.

***X**
Erlaubt die Verwendung des Objekts.

***EXCLUDE**
Alle Operationen am Objekt sind verboten.

***NONE**
Wird vom System angezeigt, wenn der Benutzer keine Datenberechtigungen hat.

USER DEF
Wird vom System angezeigt, wenn die speziellen Datenberechtigungen nicht mit den zuvor aufgeführten, vordefinierten Datenberechtigungsebenen übereinstimmen. Die speziellen Datenberechtigungen können mit der Funktionstaste zum detaillierten Anzeigen der Berechtigungen aufgelistet werden.

Der Wert *AUTL ist auch gültig, wenn allgemeine Berechtigungen definiert werden. Er zeigt an, dass die Angaben in der von diesem Objekt verwendeten Berechtigungsliste zur allgemeinen Berechtigung verwendet werden sollen, um diese Berechtigungen festzulegen.

Berechtigungen auf Objekte
Die Objektberechtigungen, über die der Benutzer für das Objekt verfügt. Ein „X“ in der Spalte bedeutet, dass der Benutzer über die angegebene Objektberechtigung für das aufgeführte Objekt verfügt. Die speziellen Objektberechtigungen sind:

Existenz
Objektexistenzberechtigung umfasst die Berechtigung, die Objektexistenz und das Eigentumsrecht zu kontrollieren.

Verwaltung
Objektverwaltungsberechtigung umfasst die Berechtigung, die Sicherheitsstufe anzugeben, das Objekt zu verschieben oder es umzubenennen, sowie dem Objekt Teildateien hinzuzufügen, wenn es sich um eine Datenbankdatei handelt.

Änderung
Objektänderungsberechtigung umfasst die Berechtigung zum Ändern der Attribute eines Objekts, wie z. B. Auslöser für eine Datenbankdatei hinzufügen oder entfernen und einer Datenbankdatei Teildateien hinzufügen.

Referenz
Objektreferenzberechtigung umfasst die Berechtigung, das Objekt als erste Ebene in einer referenziellen Integritätsbedingung anzugeben.

9.13.2.2
Seite 6

Objekteigner

Vorstehend haben wir schon mehrfach den Begriff „Objekteigner“ gelesen. Was hat es damit auf sich?

Wie in vielen Betriebssystemen gibt es auch im IBM i einen Besitzer eines Objekts. Das ist in der Regel der Benutzer, der das Objekt erstellt hat oder der dessen Eignerschaft später übernommen hat.

Der Eigner eines Objekts verfügt immer über alle für das Objekt zutreffenden Berechtigungen, wenn diese nicht ausdrücklich entzogen wurden. Der Objekteigner hat die Berechtigung, allen Benutzern beliebige Berechtigungen für das Objekt zu erteilen. Eigner können außerdem sich selbst Berechtigungen erteilen, die zuvor entzogen wurden. Sie können z. B. als Vorsichtsmaßnahme einige ihrer speziellen Berechtigungen aufheben und dann im Bedarfsfall wieder erteilen.

Ein Benutzer mit der Sonderberechtigung *ALLOBJ verfügt über die vollständige Berechtigung für alle Objekte und kann das Eigentumsrecht an jedem Objekt übertragen. Alle Benutzer verfügen über die Hinzufüge- und Löschberechtigung für ihre eigenen Benutzerprofile, d. h., die Benutzer können durch Übertragen des Objekteigentumsrechts Objekte in ihren Benutzerprofilen hinzufügen oder (selbst erstellte) Objekte aus ihren Benutzerprofilen löschen.

Einschränkungen:

- Um das Eigentumsrecht übertragen zu können, muss jeder Benutzer (einschließlich des derzeitigen Objekteigners) über folgende Berechtigungen verfügen:
  - Objektexistenzberechtigung (*OBJEXIST) für das Objekt (außer für die Berechtigungsliste)
  - Objektverwendungsberechtigung (*OBJOPR) und Objektexistenzberechtigung (*OBJEXIST), wenn es sich bei dem Objekt um eine Datei, eine Bibliothek oder eine Subsystembeschreibung handelt.
  - Die Sonderberechtigung für alle Objekte (*ALLOBJ) oder das Eigentumsrecht, wenn das Objekt eine Berechtigungsliste ist.
  - Hinzufügeberechtigung (*ADD) für das Benutzerprofil des neuen Eigners
  - Löschberechtigung (*DLT) für das Benutzerprofil des gegenwärtigen Eigners

– Die Sonderberechtigung für alle Objekte (*ALLOBJ) oder die Sonderberechtigung des Sicherheitsadministrators (*SECADM), um den Objekteigner eines Programms oder eines SQL-Pakets, das die Berechtigung des Eigners übernimmt, ändern zu können.

– Benutzungsberechtigung (*USE) für die ASP-Einheit (ASP = Zusatzspeicherpool), sofern eine angegeben ist.

- Wird bei Datensichtgeräten dieser Befehl nicht an der Einheit eingegeben, dessen Eigentumsrecht oder von dessen Warteschlange das Eigentumsrecht geändert werden soll, dann sollten vor diesem Befehl zuerst der Befehl ALCOBJ (Objekt zuordnen) und dann der Befehl DLCOBJ (Objekt freigeben) ausgeführt werden.
- Die Objektarten *DOC und *FLR können nicht angegeben werden, der Benutzer muss DLO-Unterstützung (Dokumentaustausch) verwenden.
- Eine Änderung des Eigentumsrechts für ein Objekt, dem ein Berechtigungsobjekt zugeordnet ist, bewirkt auch eine Änderung im Eigentumsrecht des Berechtigungsobjekts.

Mit dem Befehl CHGOBJOWN kann die Eignerschaft eines Objekts an ein anderes Benutzerprofil übertragen werden. Die Berechtigungen anderer Benutzer für das Objekt werden davon nicht berührt:

```
                         Objekteigner ändern (CHGOBJOWN)

Auswahl eingeben und Eingabetaste drücken.

Objekt . . . . . . . . . . . . . .   __________   Name
  Bibliothek . . . . . . . . . . .     *LIBL      Name, *LIBL, *CURLIB
Objektart  . . . . . . . . . . . .   _______      *ALRTBL, *AUTL, *BNDDIR...
ASP-Einheit  . . . . . . . . . . .   *__________   Name, *, *SYSBAS
Neuer Eigner . . . . . . . . . . .   __________   Name
Aktuelle Eignerberechtigung  . .     *REVOKE      *REVOKE, *SAME
```

Objekteigner ändern

Neuer Eigner (NEWOWN)
Gibt das Benutzerprofil oder den neuen Eigner eines Objekts an. Das Benutzerprofil muss bereits vorhanden sein, wenn dieser Befehl ausgeführt wird.

Aktuelle Eignerberechtigung (CUROWNAUT)
Gibt an, ob die Berechtigungen für den aktuellen Eigner entzogen werden, wenn die Angabe für den Eigner geändert wird.

***REVOKE**
Dem aktuellen Eigner wird die Berechtigung entzogen, wenn das Objekteignerrecht einem neuen Eigner übertragen wird.

***SAME**
Die Berechtigung des aktuellen Eigners bleibt als persönliche Berechtigung für das Objekt bestehen.

Beispiel für CHGOBJOWN:

```
CHGOBJOWNOBJ(USERLIB/PROGRAM1) OBJTYPE(*PGM) NEWOWN(ANN)
```

Mit diesem Befehl wird dem Benutzer ANN das Eigentumsrecht an dem Programm PROGRAM1, das sich in der Benutzerbibliothek USERLIB befindet, zugeordnet. Dem aktuellen Eigner wird die Berechtigung entzogen.

9.13.2.3

Seite 4

Primärgruppe

Ein weiterer Begriff, der erwähnt wurde, ist die Primärgruppe. Was ist das?

Sie können eine Primärgruppe für ein Objekt festlegen. Der Name der Primärgruppe und die Primärgruppenberechtigungen werden zusammen mit dem Objekt gespeichert. Die Verwendung von Primärgruppen resultiert in einer besseren Performance als die Verwendung der Gruppenberechtigung, wenn Rechte auf ein Objekt geprüft werden.

Ein Profil muss ein Gruppenprofil sein (eine GID – Parameter im Benutzerprofil – haben), um als Primärgruppe zu einem Objekt zugewiesen werden zu können. Das gleiche Profi kann nicht zugleich der Eigner des Objekts und die Primärgruppe sein.

Wenn ein Benutzer ein neues Objekt erstellt, steuern die Parameter im Benutzerprofil, ob die Gruppe des Benutzers die Berechtigung an das Objekt weitergibt und welcher Typ von Berechtigungen vergeben wird.

Der Parameter „Gruppenberechtigungstyp“ (GRPAUTTYP) im Benutzerprofil kann verwendet werden, um die Benutzergruppe zur Primärgruppe für ein Objekt zu machen. Die Zuweisung von Berechtigung und Eignerschaft zu neuen Objekten zeigt, wie Berechtigung zugeordnet wird, wenn neue Objekte erstellt werden. Für ein verzeichnisbasiertes Objekt in einigen Dateisystemen erbt das Objekt die Primärgruppe des übergeordneten Verzeichnisses. Wenn z. B. das übergeordnete Verzeichnis als Primärgruppe RUD hat, wird RUD Probleme bekommen, wenn er versucht irgendetwas im übergeordneten Verzeichnis zu erstellen. Dies passiert deshalb, weil das gleiche Profil nicht gleichzeitig beides sein kann, Eigner und Primärgruppe für das gleiche Objekt.

Sie können die Primärgruppe für ein bibliotheks- oder verzeichnisbasiertes Objekt mit Hilfe der folgenden Befehle ändern:

- Primärgruppe des Objekts ändern (CHGOBJPGP)
- Primärgruppe ändern (CHGPGP)
- Option 9 aus „Mit Objekten über Primärgruppen arbeiten“ (WRKOBJPGP)

Sie können die Berechtigung der Primärgruppe mit dem Befehl „Objektberechtigungen bearbeiten“ (**EDTOBJAUT**) oder den Grant- und Revoke-Befehlen ändern. Sie können die Primärgruppenberechtigung für ein bibliotheks- oder verzeichnisbasiertes Objekt mit dem Befehl „Berechtigung ändern“ (**CHGAUT**) oder mit dem Befehl „Mit Berechtigung arbeiten“ (**WRKAUT**) ändern.

9.13.2.4

Seite 2

Berechtigungslisten

Ebenfalls erwähnt wurde der Begriff der Berechtigungslisten. Was sind Berechtigungslisten?

Eine Berechtigungsliste besteht aus einer Liste von Benutzern oder Gruppen, dem Typ der Berechtigung (verwenden, ändern und ausschließen) für jeden Benutzer oder jede Gruppe sowie einer Liste von Objekten, auf die diese Liste Zugriff bietet.

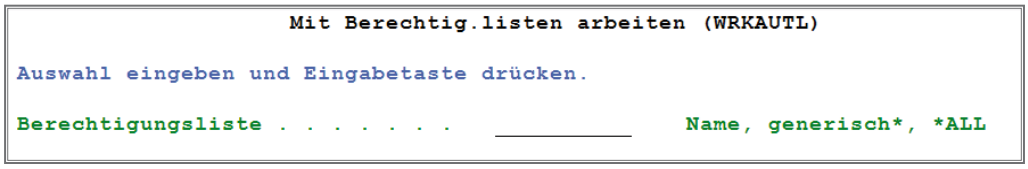
Wenn jeder Benutzer separat Zugriff auf jedes für die Arbeit erforderliche Objekt erhalten würde, hätte dies möglicherweise eine Unmenge an doppelter Arbeit zur Folge, da viele Benutzer auf dieselbe Gruppe von Objekten zugreifen müssen. Eine viel einfachere Methode für die Erteilung dieses Zugriffs ist die Erstellung von Berechtigungslisten. Benutzer und Gruppen erhalten dann die Berechtigung für diese Liste und damit die Zugriffsberechtigung auf alles, was in der Liste aufgeführt ist.

Sie können zum Beispiel eine Berechtigungsliste erstellen, in der eine Liste mit Objekten enthalten ist, die für eine Bestandsdatenbank benötigt werden. Ein Benutzer, der für die Bestellung neuer Lagerteile zuständig ist, kann die Berechtigung erhalten, den Inhalt der Datenbankobjekte anzuzeigen. Außerdem müsste die Benutzergruppe, die für den Transport und die Annahme zuständig ist, diese Datenbank aktualisieren, falls Teile im Lager eintreffen oder das Lager verlassen. Diese Gruppe kann die Berechtigung erhalten, den Inhalt der Objekte zu ändern.

Mit dem Befehl **WRKAUTL** (Mit Berechtigungslisten arbeiten) kann eine Liste von Berechtigungslisten angezeigt werden, die geändert werden können.

Einschränkungen:
- Es werden nur die Berechtigungslisten angezeigt, für die der Benutzer berechtigt ist.
- Um Operationen an Berechtigungslisten durchzuführen, muss der Benutzer über die Benutzungsberechtigung (*USE) für den von der Operation verwendeten Befehl verfügen sowie über die entsprechende Berechtigung für die Berechtigungsliste, für die die Operation durchgeführt werden soll.

9.13.2.5
Seite 2



Mit Berechtigungslisten arbeiten

Berechtigungsliste (AUTL)

Gibt die Berechtigungslisten an, die angezeigt werden sollen.

***ALL**
Es werden alle Berechtigungslisten angezeigt, deren Eigner der Benutzer ist und für die er die Berechtigung zum Anzeigen besitzt.

Generischer Name
Den generischen Namen der Berechtigungslisten angeben, die angezeigt werden sollen. Ein generischer Name besteht aus einem oder mehreren Zeichen, gefolgt von einem Stern (*). Wird ein generischer Name angegeben, werden alle Berechtigungslisten angezeigt, deren Namen dasselbe Präfix haben wie der generische Name.

Name
Den Namen der Berechtigungsliste angeben, die angezeigt werden soll.

Beispiele für WRKAUTL:

```
WRKAUTLAUTL(FR*)
```

Mit diesem Befehl kann mit einer Liste aller Berechtigungslisten gearbeitet werden, deren Namen mit ‚FR' beginnen und für die die Berechtigung zum Anzeigen besteht.